

Monocular 3D Lane Detection via Structure Uncertainty-Aware Network with Curve-Point Queries

Ruixin Liu, and Zejian Yuan, *Member, IEEE*

Abstract—Monocular 3D lane detection is challenged by aleatoric uncertainty arising from inherent observation noise. Existing methods rely on simplified geometric assumptions, such as independent point predictions or global planar modeling, failing to capture structural variations and aleatoric uncertainty in real-world scenarios. In this paper, we propose MonoUnc, a bird’s-eye view (BEV)-free 3D lane detector that explicitly models aleatoric uncertainty informed by local lane structures. Specifically, 3D lanes are projected onto the front-view (FV) space and approximated by parametric curves. Guided by curve predictions, curve-point query embeddings are dynamically generated for lane point predictions in 3D space. Each segment formed by two adjacent points is modeled as a 3D Gaussian, parameterized by the local structure and uncertainty estimations. Accordingly, a novel 3D Gaussian matching loss is designed to constrain these parameters jointly. Experiments on the ONCE-3DLanes and OpenLane datasets demonstrate that MonoUnc outperforms previous state-of-the-art (SoTA) methods across all benchmarks under stricter evaluation criteria. Additionally, we propose two comprehensive evaluation metrics for ONCE-3DLanes, calculating the average and maximum bidirectional Chamfer distances to quantify global and local errors. Codes are released at <https://github.com/lrx02/MonoUnc>.

Index Terms—Monocular 3D lane detection, local-structure-aware uncertainty, curve-point queries, 3D gaussian matching loss.

I. INTRODUCTION

3D lane detection is a fundamental task in autonomous driving, identifying and localizing lanes in 3D space to support downstream tasks such as high-definition (HD) map construction [1]–[3] and trajectory prediction [4]–[6]. Monocular cameras are widely adopted for this task due to their low cost, rich visual cues, and long-range perception capabilities.

However, monocular 3D lane detection faces aleatoric uncertainty [7], limiting the accuracy of recovering 3D lane structures from a single FV image. Specifically, aleatoric uncertainty refers to inherent observation noise and randomness that cannot be reduced by collecting additional data. Recent monocular 3D lane datasets, such as ONCE-3DLanes [8] and OpenLane [9], generate 3D lane labels based on 2D annotations, LiDAR point clouds, and heuristic techniques, which inevitably introduce noise at multiple stages of the labeling process. Fig. 1 (a) shows 2D annotation errors from human or automated labeling, which commonly occur under extreme lighting, weather, and occlusions. Fig. 1 (b) illustrates the amplification of 2D annotation errors during 3D projection, caused by sensor calibration errors, perspective compression, and depth ambiguity.

Despite the unavoidable aleatoric uncertainty, previous methods [8]–[17] typically produce predictions without explicitly modeling it. Motivated by these challenges, we pro-

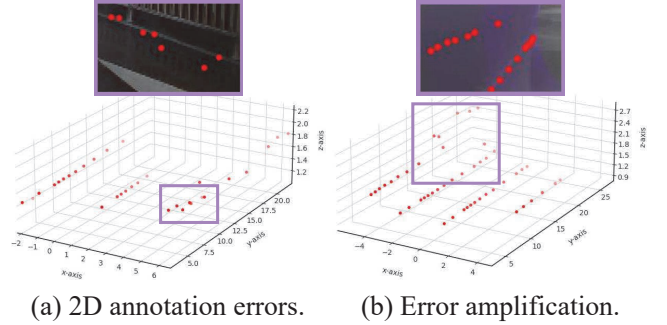


Fig. 1. Aleatoric uncertainty sources in ONCE-3DLanes. (a) 2D annotation errors. (b) Error amplification.

pose MonoUnc, an uncertainty-aware monocular lane detector, based on two key insights: (1) *parametric curve-based representations to preserve global geometric consistency*; and (2) *local-structure-aware aleatoric uncertainty modeling to capture spatially varying noise*.

MonoUnc detects lane curves in FV space and utilizes curve-point query embeddings to localize lane points in 3D space. Specifically, lanes are hierarchically represented: polynomial parameters approximate 2D curves, while a 3D point set captures precise details. Curve-point query embeddings, including curve queries extracted from FV features along predicted curves and learnable point queries, are fed into a point decoder to predict 3D lanes.

To model aleatoric uncertainty in local lane structures, we model segments formed by two adjacent points as 3D Gaussians and design a 3D Gaussian matching loss to jointly constrain their differences. Extensive experiments are conducted on the ONCE-3DLanes and OpenLane datasets, with thorough validation of each component. It is worth noting that existing evaluation metrics for ONCE-3DLanes rely on unilateral Chamfer distance, which fails to capture full-shape discrepancies between predictions and ground truths. To address this, we introduce two new metrics that compute the average and maximum bidirectional Chamfer distance to quantify both global and local errors.

The main contributions of our work are threefold:

- We propose MonoUnc, an end-to-end framework that predicts parametric curves to dynamically guide curve-point query generation for monocular 3D lane detection.
- Local lane structures with aleatoric uncertainty are modeled as 3D Gaussians, with a 3D Gaussian matching loss designed to capture spatially varying noise.
- Two evaluation metrics are introduced for the ONCE-3DLanes dataset, enabling a comprehensive quantification

of global and local errors.

II. RELATED WORK

A. Different Lane Representations

Current CNN-based lane detection methods can be broadly categorized into non-parametric and parametric paradigms, depending on how lanes are represented.

Non-parametric paradigms represent lanes as pixels or discrete points. Segmentation-based methods [8], [18], [19] formulate lane detection as a pixel-level segmentation problem, requiring dense predictions and heuristic post-processing for structural modeling. Grid-based methods [13], [20]–[22] divide the representation space into grids and represent lanes as a set of points, combining sparse grid-level segmentation with point regression, where clustering techniques are typically used to separate lane instances. In contrast to the aforementioned bottom-up methods, anchor-based methods [9]–[11], [15], [16], [23], [24] adopt a top-down framework, defining lines as anchors and regressing point offsets relative to them. These methods are widely used in 3D lane detection but suffer from limited adaptability due to data-specific anchor shapes.

Parametric paradigms represent lanes using polynomials, Bézier curves, or B-splines, which provide smooth and continuous representations without the need for post-processing. Polynomial-based methods [12], [25]–[27] regress polynomial coefficients via sparse curve queries. While achieving higher speed, they often compromise precision due to the limited flexibility of low-degree polynomials in adapting to high-curvature lanes. Alternatively, BézierLaneNet [28] estimates control points to improve curve fitting. However, adjusting any control point leads to undesirable global shape shifts due to the strong coupling among all control points. Unlike polynomials and Bézier curves that emphasize global adjustments, B-Splines [29]–[31] offer local control over curve segments, enabling flexible position refinements but exhibiting reduced robustness under severe occlusions. Our MonoUnc introduces hierarchical lane representations to balance global consistency and local flexibility, utilizing parametric curves to capture global structures while accurately localizing lane points.

B. Idealized Assumptions and Uncertainty Modeling

Idealized assumptions are common in monocular 3D lane detection but often unrealistic for real-world traffic scenarios. 3D-LaneNet [10] and CLGO [12] rely on the flat-ground assumption, predicting camera poses and employing inverse perspective mapping (IPM) to transform FV images or features into BEV space. Persformer [9] unifies 2D and 3D lane detection by learning robust BEV features via deformable attention based on IPM. However, these approaches struggle with non-flat roads. LATR [14] instead models the ground as a dynamic plane that approximates global terrain and constrains lanes in FV space without relying on IPM. Nevertheless, it fails to capture local lane structures, especially under complex terrain variations. These methods produce predictions based on idealized assumptions while neglecting aleatoric uncertainty, which limit the training stability and overall performance.

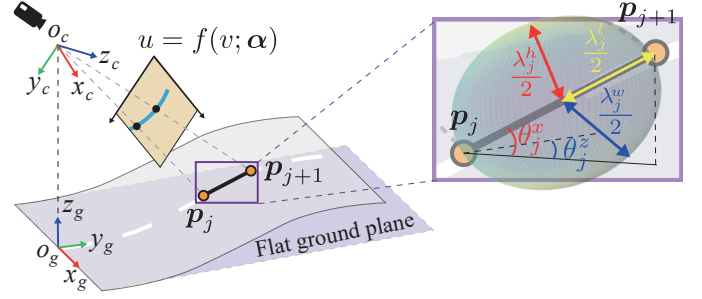


Fig. 2. Illustration of coordinate systems and hierarchical lane representations. The orange lane points lie on the ground and can be projected onto the FV space, approximated by the blue curve. The black segment formed by two adjacent points (p_j, p_{j+1}) is represented by its length λ_j^l , uncertainties (λ_j^w, λ_j^h), a pitch angle θ_j^x , and a yaw angle θ_j^z .

Uncertainty modeling can be divided into epistemic and aleatoric uncertainties [7], with the former reflecting model uncertainty and the latter capturing observation noise inherent in data. Aleatoric uncertainty has been well studied in monocular 3D object detection [32]–[35]. MonoRun [33] independently models the reprojected 2D coordinates as univariate Gaussians with a robust KL loss. In contrast, GUPNet [34] explicitly models the dependency between 3D height and depth uncertainties using Laplace distributions, propagating height uncertainty to depth through projection and supervising with a negative log-likelihood loss. These methods focus on object-level uncertainty while ignoring spatially varying uncertainty within local structures. Our MonoUnc models local lane structures as 3D Gaussians to capture lateral and vertical aleatoric uncertainties relative to the local ground plane. A 3D Gaussian matching loss is designed to jointly constrain these parameters for coupled adjustments.

III. METHODOLOGY

Coordinate systems relevant to lanes are illustrated in Fig. 2. The camera coordinate system is defined with origin o_c at the optical center, where the x_c -axis points right, the z_c -axis points forward, and the y_c -axis points downward (in meters). The ground coordinate system has its origin o_g at the vertical projection of o_c onto the ground, with x_g, y_g , and z_g axes oriented right, forward, and upward, respectively. Each 3D lane (white) is located on the ground and can be projected onto the FV space as a 2D lane (blue curve).

A. Hierarchical Lane Representations

To simultaneously capture global structures and local details of lanes, hierarchical representations are designed to consistently model parametric curves in FV space while flexibly localizing non-parametric points in 3D space.

Motivated by strong shape priors, parametric representations provide a natural and compact approximation of global lane trends. We adopt polynomial coefficients α , as proposed by LSTR [25], to represent lane curves:

$$u = f(v; \alpha), \quad (1)$$

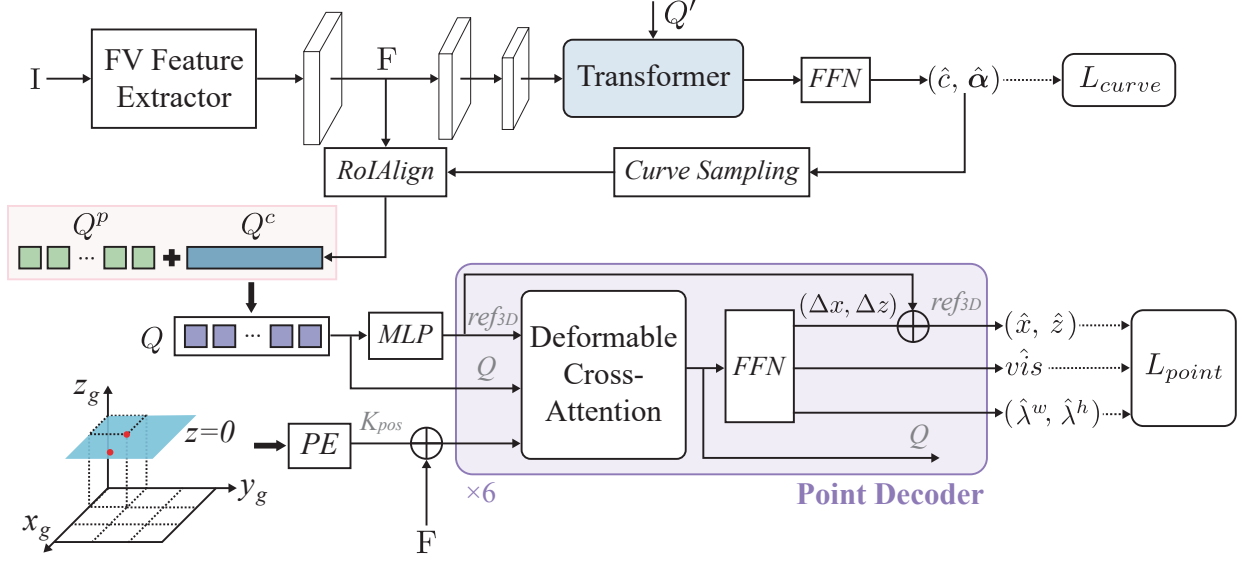


Fig. 3. Overall architecture. FV feature extractor produces multi-scale features from image I . Curve parameters are predicted by a Transformer and used to generate curve-point queries Q . Point decoder outputs 3D lane points with aleatoric uncertainties.

where α comprises shared curvature parameters $\{\rho^t\}_{t=1}^4$, individual biases $\{\beta', \beta''\}$ and boundaries $\{v^{low}, v^{up}\}$.

Alongside the curve α , a point set $\{\mathbf{p}_j\}_{j=1}^J$ of size J is to capture local details, where the j -th point $\mathbf{p}_j = (x_j, y_j, z_j, vis_j)$ consists of a ground coordinate (x_j, y_j, z_j) and a visibility indicator vis_j . These points are distributed along predefined longitudinal y -coordinates $Y = \{y_j\}_{j=1}^J$.

To model local structures, we define a segment formed by two adjacent points $(\mathbf{p}_j, \mathbf{p}_{j+1})$ as an intermediate representation bridging curves and points, as shown in Fig. 2. Given that lane markings are thin coatings on the road surface, we assume a zero roll angle. Each segment is characterized by its length λ_j^l , aleatoric uncertainties $(\lambda_j^w, \lambda_j^h)$, and pitch θ_j^x and yaw θ_j^z angles. Here, λ_j^w denotes the lateral uncertainty on the local ground plane, while λ_j^h is the vertical uncertainty orthogonal to it.

B. Network Architecture

The overall architecture of MonoUnc, illustrated in Fig. 3, consists of an FV feature extractor, a curve-point query generation module, and a point decoder for 3D lane predictions.

FV Feature Extractor. Given a monocular image $I \in \mathbb{R}^{3 \times H \times W}$ as input, a shared backbone with a 4-layer feature pyramid network [36] (FPN) extracts multi-scale FV features, which are then aggregated into $F \in \mathbb{R}^{C \times H' \times W'}$ to enhance spatial context representations.

Curve-Point Query Generation. To model intra-lane and inter-lane relationships, we design a curve-point query generation module that decouples lane queries into curve and point components. Curve queries are dynamically derived from curve predictions, with embeddings extracted from F . Specifically, learnable queries $Q' \in \mathbb{R}^{K \times d'}$ are fed into a Transformer to predict K sets of confidence scores and polynomial parameters, denoted as $\{(\hat{c}_k, \hat{\alpha}_k)\}_{k=1}^K$.

(1) *Curve Sampling.* Each curve is sampled at J' points with uniformly spaced v -coordinates $\{v_j\}_{j=1}^{J'}$ along the im-

age height. Based on the k -th curve prediction, lane points $\{(\hat{u}_{j,k}, v_j)\}_{j=1}^{J'}$ are computed via Equation 1. For each point, a validity indicator $\hat{m}_{j,k} \in \{0, 1\}$ determines whether the j -th point lies within the image and lane boundaries:

$$\hat{m}_{j,k} = \begin{cases} 1, & \text{if } v_j \in [v_k^{low}, v_k^{up}] \wedge \hat{u}_{j,k} \in [0, W] \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

(2) *RoIAlign.* Using RoIAlign [37], features of valid points are sampled from the feature map F and organized as a sequence $\{\hat{f}_{j,k}\} \in \mathbb{R}^{K \times (J' \times C)}$, where:

$$\hat{f}_{j,k} = \begin{cases} \text{RoIAlign}((\hat{u}_{j,k}, v_j), F), & \hat{m}_{j,k} = 1 \\ \text{constants}, & \hat{m}_{j,k} = 0 \end{cases} \quad (3)$$

A shared fully connected (FC) layer followed by a ReLU activation function aggregates these sequences into corresponding curve queries $Q^c \in \mathbb{R}^{K \times C}$:

$$Q_k^c = \text{ReLU}(\text{FC}(\{\hat{f}_{j,k}\}_{j=1}^{J'})). \quad (4)$$

Meanwhile, point-level information is encoded by learnable point queries $Q^p \in \mathbb{R}^{J' \times C}$, which are shared across all curves and indexed with respect to the predefined y -coordinates Y . The curve-point queries $Q \in \mathbb{R}^{K \times J' \times C}$ are obtained by summing:

$$Q_{j,k} = Q_{j,k}^c + Q_{j,k}^p. \quad (5)$$

Point Decoder. A point decoder with L layers is designed to iteratively refine the curve-point queries Q , employing deformable attention [38] to improve computational efficiency. Specifically, Q is first fed into an MLP to predict 3D lane locations $(\hat{x}_{j,k}, y_j, \hat{z}_{j,k})$, where y_j is indexed from the predefined Y -coordinates. The resulting predictions initialize the 3D reference points ref_{3D} , which are then projected into the FV space to serve as 2D reference points for interaction with the FV feature map F . They are iteratively updated by the predicted 3D offsets of $\Delta x_{j,k}$ and $\Delta z_{j,k}$. For the l -th decoder

layer, the deformable attention process exchanges messages as follows:

$$\begin{aligned} Q^{(l)} &= \text{DeformAttn}(Q^{(l-1)}, F + K_{pos}, \text{ref}_{3D}^{(l-1)}), \\ \text{ref}_{3D}^{(l)} &= \text{ref}_{3D}^{(l-1)} + [\Delta x, 0, \Delta z]^T. \end{aligned} \quad (6)$$

Here, $K_{pos} \in \mathbb{R}^{C \times H' \times W'}$ denotes the 3D ground positional embeddings associated with the FV image, similar to LATR [14]. These embeddings are generated by a positional encoding (PE) module, where the ground is discretized into a set of 3D points corresponding to BEV grid locations at a height of $z = 0$. These points are projected into the FV space and fed into an MLP to generate K_{pos} .

3D Lane Predictions. For each lane point, 3D location offsets $(\Delta x_{j,k}, \Delta z_{j,k})$ are predicted relative to the reference points from the last decoder layer, accompanied by a visibility indicator $\hat{vis}_{j,k}$ to determine whether the projected point is valid in the FV space. For each segment formed by two adjacent points, two aleatoric uncertainties are predicted, consisting of a lateral uncertainty $\hat{\lambda}_{j,k}^w$ and a vertical uncertainty $\hat{\lambda}_{j,k}^h$. The final 3D lane predictions are expressed as:

$$\begin{aligned} \{(\hat{x}_{j,k}, y_j, \hat{z}_{j,k}, \hat{vis}_{j,k}) | 1 \leq j \leq J, 1 \leq k \leq K\}, \\ \{(\hat{\lambda}_{j,k}^w, \hat{\lambda}_{j,k}^h) | 1 \leq j \leq J-1, 1 \leq k \leq K\}. \end{aligned} \quad (7)$$

C. Training with 3D Gaussian Matching Loss

MonoUnc is trained with point-level and curve-level constraints. The total loss is defined as:

$$L_{Total} = L_{point} + L_{curve}. \quad (8)$$

The point-level loss L_{point} supervises visibility and locations of individual points, as well as aleatoric uncertainties of segments, as shown in Fig. 4 and defined as follows:

$$L_{point} = \gamma_1 L_{unc} + L_{vis} + L_{loc}, \quad (9)$$

where L_{vis} is the binary cross-entropy loss for visibility, and L_{loc} is the L1 loss applied to the visible x and z coordinates:

$$\begin{aligned} L_{loc} &= \sum_{k=1}^K \mathbb{I}(c_k \neq 0) \sum_{j=1}^J \mathbb{I}(vis_{j,k} \neq 0) \cdot \\ &(\gamma_2 |\hat{x}_{j,\hat{e}(k)} - x_{j,k}| + \gamma_3 |\hat{z}_{j,\hat{e}(k)} - z_{j,k}|). \end{aligned} \quad (10)$$

Here, $\mathbb{I}(\cdot)$ is the indicator function. Bipartite matching correspondences $\hat{e}$ are computed at the curve level using Hungarian algorithm [39]. γ_1 , γ_2 , and γ_3 are weight terms.

To capture the uncertainty of local structures, we model each segment formed by two adjacent points as a 3D Gaussian $\mathcal{N}(\mu, \Sigma)$, with mean μ and covariance Σ defined as:

$$\mu = (x_c, y_c, z_c)^T, \Sigma^{1/2} = R\Lambda R^T, \quad (11)$$

where $(x_c, y_c, z_c) = (\frac{x_j+x_{j+1}}{2}, \frac{y_j+y_{j+1}}{2}, \frac{z_j+z_{j+1}}{2})$ is the 3D center of the segment, R represents the rotation matrix, and Λ denotes the diagonal matrix of eigenvalues.

In BEV lane detection, R and Λ are 2D Gaussian parameters defined by segment length, line width, and heading

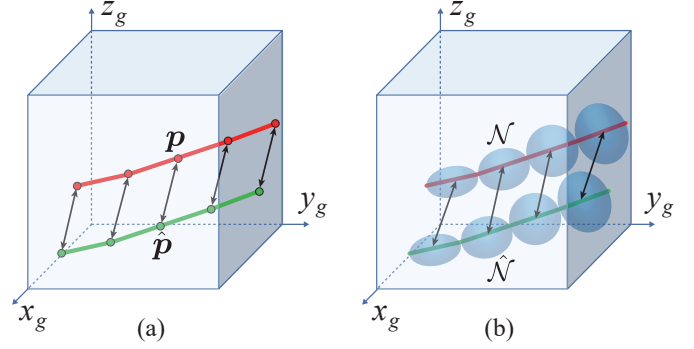


Fig. 4. Components of the point-level constraint. (a) 3D location. (b) Aleatoric uncertainty.

angle [22], [40]. In contrast, 3D lane segments are modeled as 3D Gaussians (shown in Fig. 2 and Fig. 4 (b)), characterized by segment length, uncertainty, and Euler angles. The rotation matrix R is defined by the pitch angle $\theta^x = \arctan(\frac{z_{j+1}-z_j}{\sqrt{(x_{j+1}-x_j)^2 + (y_{j+1}-y_j)^2}})$ and the yaw angle $\theta^z = \arctan(\frac{y_{j+1}-y_j}{x_{j+1}-x_j})$, written as:

$$R = \begin{pmatrix} \cos \theta^z & -\sin \theta^z \cos \theta^x & \sin \theta^x \sin \theta^z \\ \sin \theta^z & \cos \theta^z \cos \theta^x & -\cos \theta^x \sin \theta^z \\ 0 & \sin \theta^x & \cos \theta^x \end{pmatrix}. \quad (12)$$

And the diagonal matrix of eigenvalues Λ is formulated as:

$$\Lambda = \begin{pmatrix} \frac{\lambda^l}{2} & 0 & 0 \\ 0 & \frac{\lambda^w}{2} & 0 \\ 0 & 0 & \frac{\lambda^h}{2} \end{pmatrix}, \quad (13)$$

where λ^l denotes the segment length, computed as $\lambda_j^l = \sqrt{(x_{j+1}-x_j)^2 + (y_{j+1}-y_j)^2 + (z_{j+1}-z_j)^2}$, while λ^w and λ^h are the lateral and vertical uncertainty estimations shared by the prediction and its corresponding ground truth.

Considering that uncertainty and local lane structure are inherently coupled rather than independent variables, we design a 3D Gaussian matching loss based on symmetric Kullback-Leibler divergence (KLD) for joint optimization:

$$\begin{aligned} L_{unc} &= \frac{1}{2} \sum_{k=1}^K \mathbb{I}(c_k \neq 0) \sum_{j=1}^{J-1} \mathbb{I}(vis_{j,k} \neq 0 \wedge vis_{j+1,k} \neq 0) \cdot \\ &(\text{KLD}(\hat{\mathcal{N}}_{j,\hat{e}(k)}, \mathcal{N}_{j,k}) + \text{KLD}(\mathcal{N}_{j,k}, \hat{\mathcal{N}}_{j,\hat{e}(k)})), \end{aligned} \quad (14)$$

where $\hat{\mathcal{N}}_{j,\hat{e}(k)}$ and $\mathcal{N}_{j,k}$ denote the predicted and ground-truth Gaussians, respectively.

The curve-level loss L_{curve} is composed of a classification loss implemented by the cross-entropy loss L_{ce} and a curve fitting loss L_f in the FV space, expressed as:

$$L_{curve} = \sum_{k=1}^K (\gamma_4 L_{ce}(c_k, \hat{c}_{\hat{e}(k)}) + \mathbb{I}(c_k \neq 0) L_f(\hat{\alpha}_{\hat{e}(k)})). \quad (15)$$

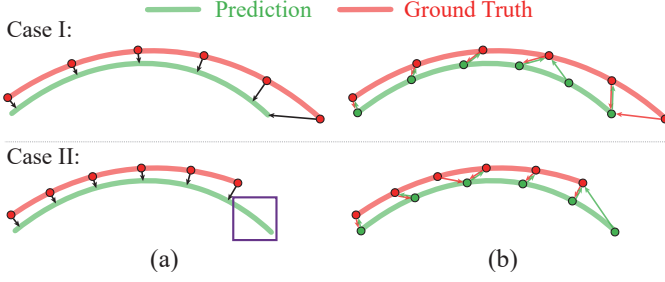


Fig. 5. Illustration of unilateral and bidirectional CDs. (a) Unilateral CD. (b) Bidirectional CD.

Here, the curve fitting loss L_f is formulated by:

$$L_f(\hat{\alpha}_{\hat{\epsilon}(k)}) = \gamma_5 \sum_{j=1}^{J'} |\hat{u}_{j,\hat{\epsilon}(k)} - u_{j,k}| + \gamma_6 (|\hat{v}_{\hat{\epsilon}(k)}^{low} - v_k^{low}| + |\hat{v}_{\hat{\epsilon}(k)}^{up} - v_k^{up}|). \quad (16)$$

The weight coefficients γ_4 , γ_5 , and γ_6 follow the settings in LSTR [25].

IV. EXPERIMENTS

A. Datasets and Evaluation Metrics

ONCE-3DLanes [8] is a large-scale real-world dataset built upon the ONCE dataset [41]. It contains 211K images collected by a front-facing camera, covering diverse traffic scenarios, with accurate 3D annotations and camera intrinsic parameters provided.

OpenLane [9] is another large-scale 3D lane detection dataset derived from the Waymo dataset [42]. It includes 220K frames and 880K annotated lanes with rich geometric and semantic information. OpenLane provides both camera intrinsic and extrinsic parameters.

ONCE-3DLanes evaluation metrics. The official protocol computes the matching degree by calculating the Intersection over Union (IoU) [43] between the BEV areas of ground-truth and predicted lanes within a predefined lane width. A true positive is recognized if its IoU exceeds the IoU threshold and the unilateral Chamfer distance (CD) is below the distance threshold τ_{CD} . Precision (P), recall (R), F1 scores (F1), and CD errors (CDE) are reported.

However, unilateral CD only accounts for errors from ground truths to predictions, failing to fully capture shape differences. As shown in Fig. 5 (a), it penalizes the entire predicted lane curve when the ground truth is longer than the prediction (Case I), but ignores extra false predictions (Case II, marked in purple), which leads to overly optimistic evaluations for methods producing elongated predictions.

To comprehensively assess both global and local errors, two alternative evaluation metrics based on the bidirectional Chamfer distance are introduced to quantify errors in 3D lane predictions, including the average bidirectional Chamfer distance and the maximum bidirectional Chamfer distance. Here, the average bidirectional Chamfer distance captures the global geometric errors by jointly considering deviations in both directions between predictions and ground truths. Based

Algorithm 1 True Positives and False Positives Selection Based on Bidirectional Chamfer Distance

Input: Ground truths $\mathcal{G} = \{G_1, \dots, G_{N_g}\}$, Predicted lanes $\mathcal{P} = \{P_1, \dots, P_{N_p}\}$, Threshold τ_{BCD} , Interpolation size N

Output: TP flags $\mathbf{tp} \in \{0, 1\}^{N_p}$, FP flags $\mathbf{fp} \in \{0, 1\}^{N_p}$

Step 1: Dense Interpolation

1: Interpolate each lane to N points: $\hat{G}_i, \hat{P}_j$.

Step 2: Bidirectional Distance Calculation

2: **for** each pair $(\hat{P}_j, \hat{G}_i)$ **do**

3: Compute **prediction-to-truth distance**:

$$d_{P \rightarrow G} = \frac{1}{N} \sum_{p \in \hat{P}_j} \min_{q \in \hat{G}_i} \|p - q\|_2$$

4: Compute **truth-to-prediction distance**:

$$d_{G \rightarrow P} = \frac{1}{N} \sum_{q \in \hat{G}_i} \min_{p \in \hat{P}_j} \|q - p\|_2$$

5: **Bidirectional distance**:

$$D_{ij} = \frac{d_{P \rightarrow G} + d_{G \rightarrow P}}{2}$$

6: **end for**

Step 3: TP and FP Calculation through Optimal Matching

7: Initialize $\mathbf{tp} \leftarrow [0]^{N_p}$, $\mathbf{fp} \leftarrow [0]^{N_p}$, $\mathbf{covered} \leftarrow [\text{False}]^{N_g}$

8: **for** each prediction P_j **do**

9: Find best-matched ground truth:

$$i^* = \arg \min_i D_{ij}$$

10: **if** $D_{i^*j} \leq \tau_{BCD}$ **and** $\mathbf{covered}[i^*] = \text{False}$ **then**

11: $\mathbf{tp}[j] \leftarrow 1$, $\mathbf{covered}[i^*] \leftarrow \text{True}$

12: **else**

13: $\mathbf{fp}[j] \leftarrow 1$

14: **end if**

15: **end for**

Edge Cases

16: **if** $N_g = 0$ **then**

17: $\mathbf{fp} \leftarrow [1]^{N_p}$

18: **end if**

19: **if** $N_p = 0$ **then**

20: **Return** $\mathbf{tp} = \emptyset$, $\mathbf{fp} = \emptyset$

21: **else**

22: **Return** $\mathbf{tp}$, $\mathbf{fp}$

23: **end if**

on this metric, precision P_B , recall R_B , and F1 score $F1_B$ are defined as follows:

$$\begin{aligned} P_B &= \frac{TP}{TP + FP}, \\ R_B &= \frac{TP}{TP + FN}, \\ F1_B &= \frac{2 \times P_B \times R_B}{P_B + R_B}. \end{aligned} \quad (17)$$

Here, true positives (TP) and false positives (FP) are aggregated over all validation samples. For each sample, TP and

TABLE I

QUANTITATIVE COMPARISON ON ONCE-3DLANES. “P”, “R”, “F1”, AND “CDE” ARE COMPUTED USING THE UNILATERAL CD UNDER $\tau_{CD} = 0.3$ METERS. “P_B”, “R_B”, “F1_B”, AND “MBD” ARE COMPUTED USING THE BIDIRECTIONAL CD UNDER $\tau_{BCD} = 0.3$ METERS, WHERE “MBD” INDICATES THE MAXIMUM BIDIRECTIONAL CD. ALL UNOFFICIAL RESULTS ARE REPORTED BASED ON THEIR BEST-PERFORMING CONFIGURATIONS.

Method	F1 (%) $\uparrow$	P (%) $\uparrow$	R (%) $\uparrow$	CDE (m) $\downarrow$	F1 _B (%) $\uparrow$	P _B (%) $\uparrow$	R _B (%) $\uparrow$	MBD (m) $\downarrow$
Persformer [9]	74.33	80.30	69.18	0.074	26.67	28.94	24.73	27.268
Anchor3DLane [15]	74.87	80.85	69.71	0.060	33.58	44.61	26.92	2.305
LATR [14]	80.59	86.12	75.73	0.052	41.62	44.58	39.03	2.432
MonoUnc (Ours)	84.29	86.35	82.32	0.049	45.29	51.10	40.66	2.179
<i>Improvement</i>	$\uparrow 3.70$	$\uparrow 0.23$	$\uparrow 6.59$	$\downarrow 0.003$	$\uparrow 3.67$	$\uparrow 6.49$	$\uparrow 1.63$	$\downarrow 0.126$

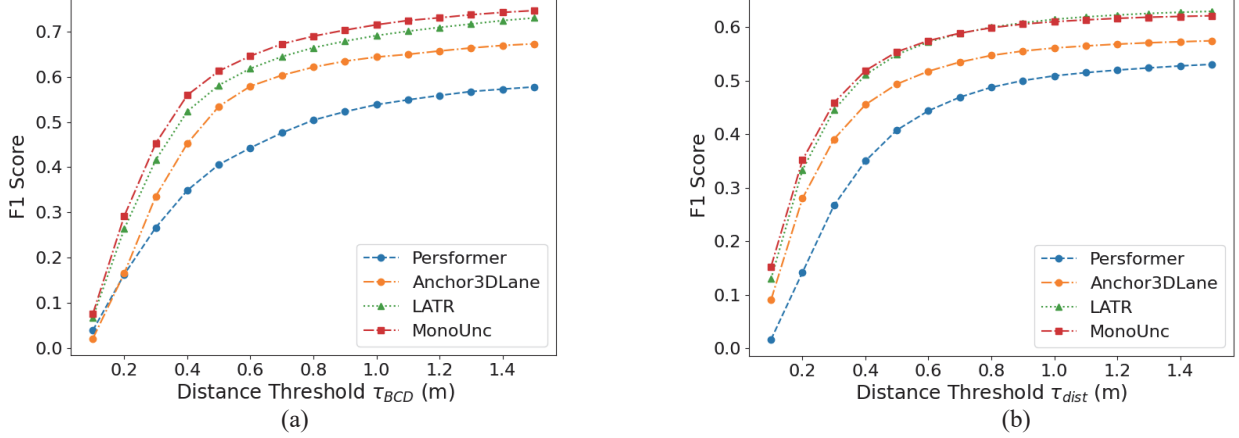


Fig. 6. F1 score vs. distance threshold. (a) ONCE-3DLanes. (b) OpenLane.

FP counts are determined as described in Algorithm 1. The number of false negatives (*FN*) is obtained by subtracting the number of *TP* from the total number of ground-truth lanes. Both ground-truth and predicted lanes are densely interpolated to N points, with N set to 100 in our experiments. The bidirectional Chamfer distance quantifies the similarity between the two unordered point sets. A predicted lane is considered a *TP* if its average bidirectional Chamfer distance to its best-matched ground-truth lane is below the predefined threshold τ_{BCD} .

The maximum bidirectional Chamfer distance (MBD) focuses on local geometric errors. Specifically, the matching degree between ground-truth lanes $\mathcal{G} = \{G_1, \dots, G_{N_g}\}$ and predicted lanes $\mathcal{P} = \{P_1, \dots, P_{N_p}\}$ is determined by computing the IoU, following the definition in the ONCE-3DLanes [8] dataset. Based on the matching correspondences ϵ , the maximum bidirectional Chamfer distance of the k -th ground truth G_k and its matched prediction $P_{\epsilon(k)}$ is computed to evaluate worst-case geometric errors. Notably, MBD does not exclude overly large errors, making it a stricter and more discriminative evaluation criterion.

OpenLane evaluation metrics. OpenLane formulates the evaluation as a bipartite matching problem, which is solved using the minimum-cost flow, following the protocol of GenLaneNet [11]. A lane prediction is considered a true positive if more than 75% of its points lie within the specified distance threshold τ_{dist} .

B. Implementation Details

All input images are first resized to a spatial resolution of $(H, W) = (720, 960)$. The resized images are then nor-

malized, and fed into MonoUnc. ResNet-50 [44] serves as the backbone to extract 3-scale features with spatial reduction ratios of $[\frac{1}{8}, \frac{1}{16}, \frac{1}{32}]$ relative to the input resolution. On top of the multi-scale feature representations, a 4-layer FPN is constructed to enhance the semantic information of features across scales, which are aggregated into a feature map F of size $(H', W') = (90, 120)$ and passed through the point decoder with $L = 6$ layers. The number of curve queries and point queries is set to $K = 12$ and $J = 20$ for the ONCE-3DLanes dataset, and $K = 40$ and $J = 20$ for the OpenLane dataset to accommodate the scene and annotation complexity. MonoUnc is trained on 4 Nvidia RTX 3090 GPUs with a total batch size of 32, using the AdamW optimizer [45] with an initial learning rate of 2×10^{-4} , weight decay of 0.01, and a cosine annealing schedule [46]. Loss coefficients γ_1 to γ_6 are set to 0.5, 2, 10, 3, 5, and 2, respectively.

C. Comparisons with State-of-the-Art Methods

Quantitative comparison on ONCE-3DLanes. The quantitative results on the ONCE-3DLanes dataset are reported in Table I and visualized in Fig. 6 (a). Since the camera extrinsic parameters are not provided, we adopt the same camera configuration as used in Persformer [9] to ensure fair comparison and reproducibility. Under the official evaluation metrics provided by ONCE-3DLanes, our proposed MonoUnc demonstrates clear superiority over existing approaches. Specifically, MonoUnc outperforms LATR with a 3.70% improvement in F1 score and a reduction of 0.003 m in CDE, indicating more accurate geometric localization. Furthermore, MonoUnc achieves a higher F1_B score of 3.67% than LATR under the

TABLE II

QUANTITATIVE COMPARISON ON OPENLANE. E_x AND E_z DENOTE ERRORS ALONG x AND z AXES, MEASURED IN THE NEAR RANGE ($[0, 40]$ METERS) AND FAR RANGE ($[40, 100]$ METERS). RESULTS WITHOUT * ARE REPORTED IN THE ORIGINAL PAPERS; * INDICATES THE BEST-PERFORMING MODELS PROVIDED IN THEIR OFFICIAL CODEBASE.

Method	F1 (%) $\uparrow$	E_x (near/far) $\downarrow$	E_z (near/far) $\downarrow$
$\tau_{dist} = 0.1$ m			
Persformer* [9]	1.7	0.190 / 0.120	0.103 / 0.076
Anchor3DLane* [15]	9.0	0.109 / 0.104	0.061 / 0.070
LATR* [14]	13.0	0.101 / 0.099	0.055 / 0.071
MonoUnc (Ours)	15.2	0.100 / 0.095	0.055 / 0.066
$\tau_{dist} = 0.5$ m			
Persformer* [9]	40.8	0.284 / 0.263	0.117 / 0.114
Anchor3DLane* [15]	49.4	0.183 / 0.203	0.077 / 0.102
LATR* [14]	54.9	0.169 / 0.202	0.070 / 0.100
MonoUnc (Ours)	55.4	0.162 / 0.187	0.069 / 0.092
$\tau_{dist} = 1.5$ m			
Persformer [9]	50.5	0.485 / 0.553	0.364 / 0.431
Persformer* [9]	53.1	0.361 / 0.328	0.124 / 0.129
Anchor3DLane [15]	53.7	0.276 / 0.311	0.107 / 0.138
Anchor3DLane* [15]	57.5	0.230 / 0.244	0.080 / 0.107
LATR [14]	61.9	0.219 / 0.259	0.075 / 0.104
LATR* [14]	63.0	0.209 / 0.250	0.073 / 0.105
MonoUnc (Ours)	62.1	0.204 / 0.227	0.071 / 0.096

TABLE III

QUANTITATIVE COMPARISON ON OPENLANE USING THE PROPOSED BIDIRECTIONAL CHAMFER DISTANCE METRICS. “ P_B ”, “ R_B ”, “ $F1_B$ ”, AND “MBD” ARE COMPUTED USING THE BIDIRECTIONAL CD UNDER DIFFERENT DISTANCE THRESHOLDS $\tau_{BCD} \in \{0.1, 0.3\}$ METERS, WHERE “MBD” INDICATES THE MAXIMUM BIDIRECTIONAL CHAMFER DISTANCE. * INDICATES THE BEST-PERFORMING MODELS IN THEIR OFFICIAL CODEBASE.

Method	Backbone	$F1_B$ (%) $\uparrow$	P_B (%) $\uparrow$	R_B (%) $\uparrow$	MBD (m) $\downarrow$
$\tau_{BCD} = 0.1$ m					
Persformer* [9]	EfficientNet-B7 [47]	7.98	8.85	7.26	1.043
Anchor3DLane* [15]	ResNet-50 [44]	24.94	27.74	22.66	0.952
LATR* [14]	ResNet-50 [44]	32.57	34.82	30.59	0.845
MonoUnc (Ours)	ResNet-50 [44]	34.13	37.53	31.30	0.754
$\tau_{BCD} = 0.3$ m					
Persformer* [9]	EfficientNet-B7 [47]	48.10	53.36	43.79	1.043
Anchor3DLane* [15]	ResNet-50 [44]	61.19	68.04	55.59	0.952
LATR* [14]	ResNet-50 [44]	68.08	72.77	63.95	0.845
MonoUnc (Ours)	ResNet-50 [44]	68.07	74.84	62.42	0.754

distance threshold of $\tau_{BCD} = 0.3$ m and a reduction of 0.126 m in MBD compared with Anchor3DLane. As illustrated in Fig. 6 (a), MonoUnc consistently achieves the highest F1 scores across all evaluated distance thresholds on the ONCE-3DLanes dataset.

Quantitative comparison on OpenLane. The standard evaluation protocol of the OpenLane dataset adopts a distance threshold of $\tau_{dist} = 1.5$ m to determine correct lane predictions, which may be insufficiently strict for safety-critical autonomous driving scenarios, as it still tolerates notable discrepancies in both shape and localization between predictions and ground truths. Therefore, we provide a more comprehensive evaluation under a series of progressively tighter distance thresholds, as presented in Fig. 6 (b) and Table II. Across all thresholds, the proposed MonoUnc consistently outperforms other methods in both lateral x and vertical z error metrics, demonstrating more precise 3D geometry estimation. Under stricter thresholds of $\tau_{dist} = 0.1$ m and $\tau_{dist} = 0.5$ m, MonoUnc improves F1 scores over LATR by 2.2% and 0.5%, respectively. Fig. 6 (b) further illustrates that as the distance threshold decreases, the performance gap between MonoUnc and other methods becomes more pronounced, highlighting its superior capability in modeling uncertainty within local lane structures.

TABLE IV

COMPARISONS OF MODEL COMPLEXITY AND INFERENCE EFFICIENCY. ALL MODELS ARE TESTED ON A SINGLE NVIDIA RTX 3090 GPU WITH A BATCH SIZE OF 1, AND PARAMETER COUNTS ARE REPORTED IN MILLIONS (M).

Method	Backbone	Resolution	Params (M)	FPS
LATR* [14]	ResNet-50	720 $\times$ 960	46.76	12.6
MonoUnc (Ours)	ResNet-50	720 $\times$ 960	43.29	13.9

To further validate the effectiveness and reliability of the proposed bidirectional Chamfer distance metrics, additional quantitative experiments on the OpenLane [9] dataset are conducted. Table III summarizes the results under two different distance thresholds $\tau_{BCD} \in \{0.1, 0.3\}$ m. As the distance threshold τ_{BCD} becomes stricter, the superiority of our MonoUnc on OpenLane is further amplified, demonstrating consistency with the trends observed in the official OpenLane evaluation.

Model complexity and inference efficiency. Comparisons of model parameters and inference speed with the state-of-the-art method LATR [14] are presented in Table IV. Experimental results demonstrate that our proposed MonoUnc achieves an inference speed of 13.9 Frames Per Second (FPS), outperforming LATR (12.6 FPS) while using fewer parameters (43.29 M).

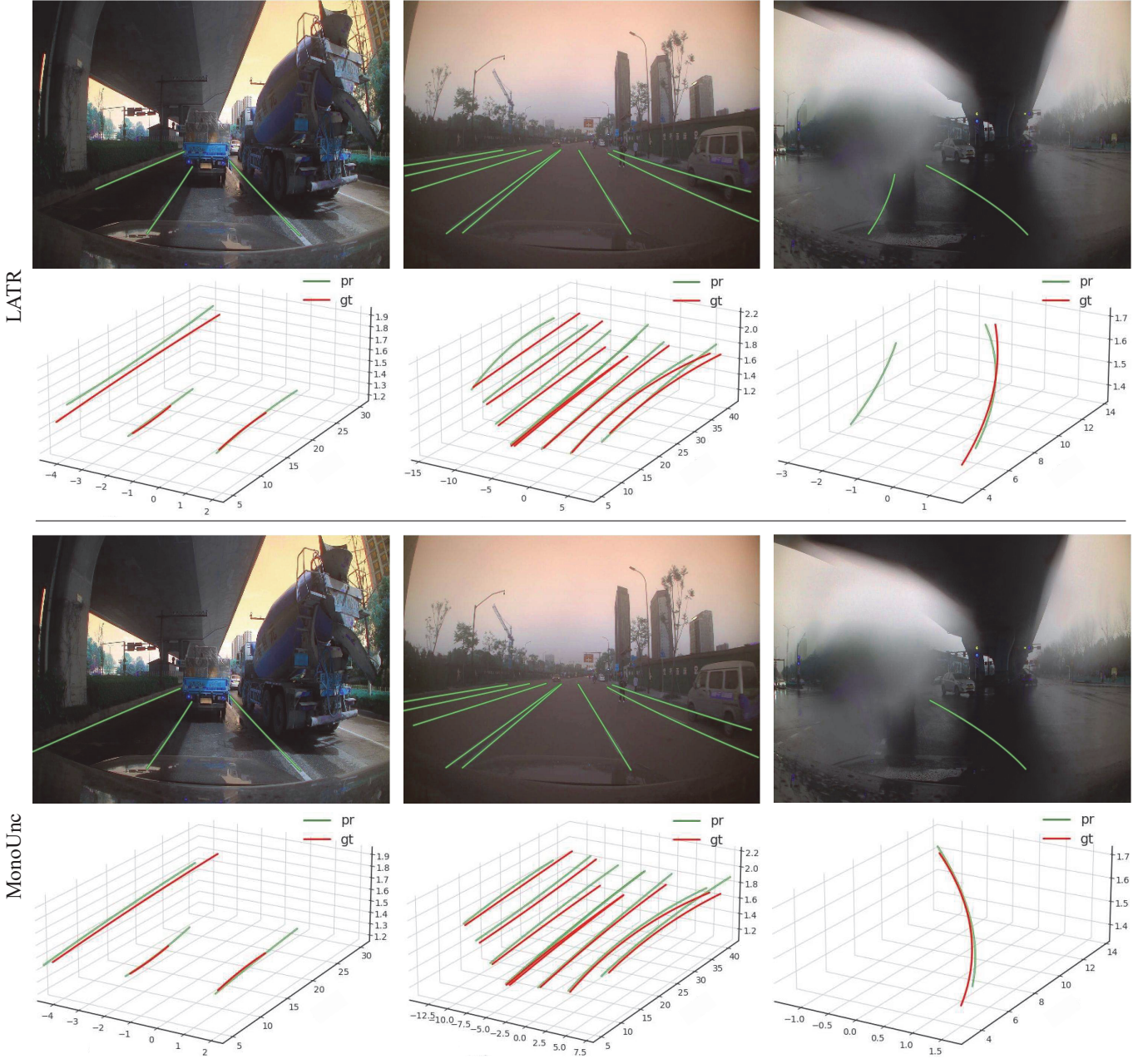


Fig. 7. Qualitative comparison on ONCE-3DLanes. Ground truths and predictions are colored red and green, respectively. Best viewed in color and zoomed in for details.

This indicates that MonoUnc attains a more favorable trade-off between computational efficiency and model compactness, enabling real-time inference without compromising accuracy.

Qualitative comparison on ONCE-3DLanes. Fig. 7 illustrates a qualitative comparison between LATR (upper row) and MonoUnc (lower row) on the ONCE-3DLanes dataset. Although both methods produce visually comparable detection results in the front-view images, MonoUnc generates more accurate and geometrically consistent 3D predictions across various scenarios, including curves, occlusions, and distant lanes. Additional qualitative results, covering challenging cases such as rainy conditions, heavy occlusions, uphill/downhill roads, fork/merge scenes, and curved lanes, are provided in the

supplementary materials.

D. Ablation Studies

We conduct ablation studies on the ONCE-3DLanes dataset to systematically evaluate the contributions of different components in our methods, including the impact of different query embeddings, the effectiveness of the proposed 3D Gaussian matching loss, and the design choices related to uncertainty modeling.

Different query embeddings. Table V presents a comparison of different query embeddings. The combination of “ $Q^c + Q^p$ ” demonstrates superior performance over “IAM + Q^p ”, achieving an improvement of 2.28% in $F1_B$ and a reduction

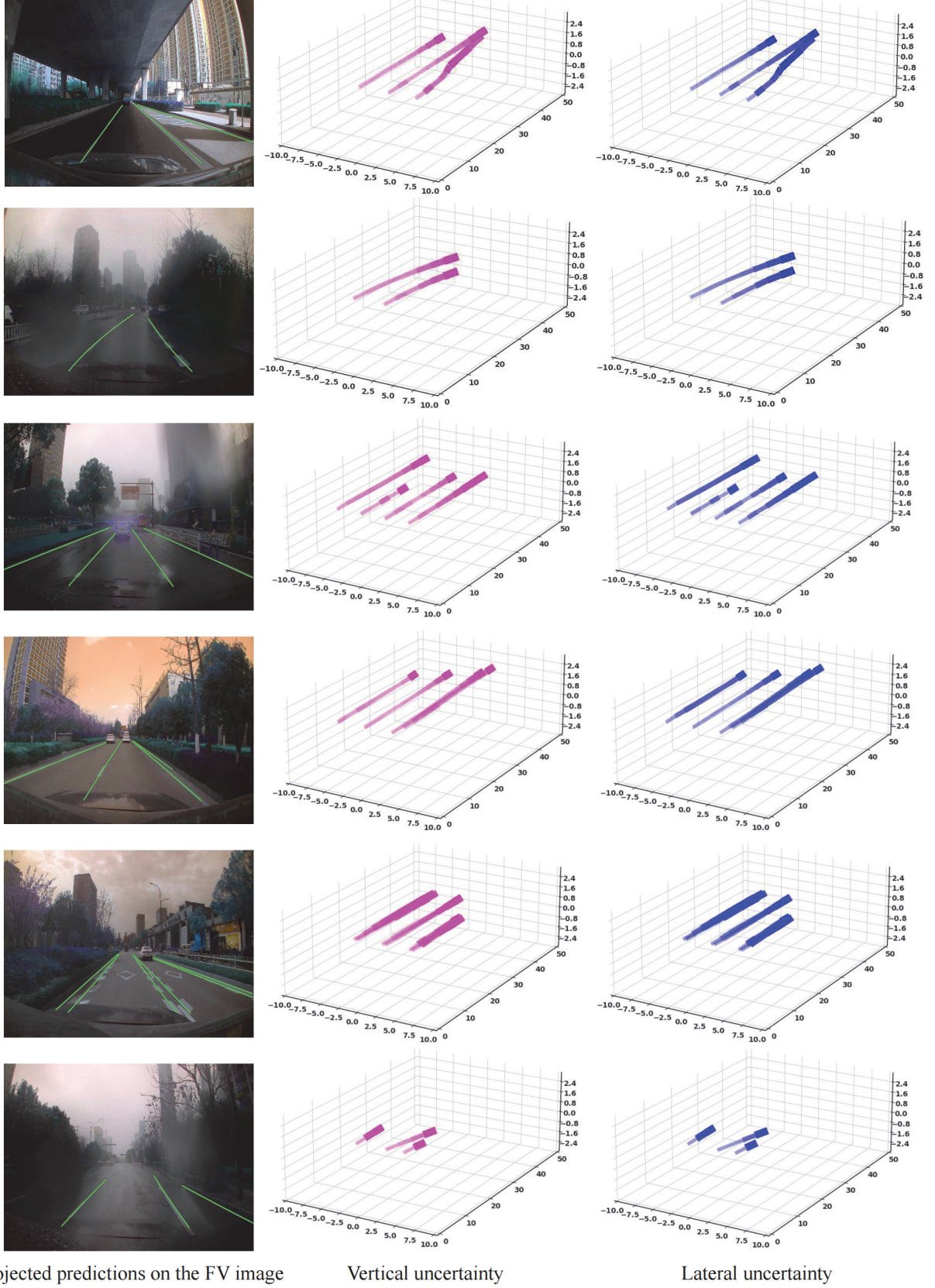


Fig. 8. Visualization of intermediate results. Vertical and lateral uncertainties are shown in pink and blue, respectively. Thicker lane segments with higher opacity indicate greater uncertainty.

TABLE V
ABLATION STUDIES ON QUERY EMBEDDINGS AND UNCERTAINTY MODELING.

Q	L_{unc}	$F1_B(\%)$	$P_B(\%)$	$R_B(\%)$	MBD (m)
IAM+ Q^P	-	41.62	44.58	39.03	2.432
Q^c+Q^P	-	43.90	49.37	39.52	2.211
Q^c+Q^P	✓	45.29	51.10	40.66	2.179

TABLE VI
ABLATION STUDIES ON THE 3D GAUSSIAN MATCHING LOSS.

γ_1	$F1_B(\%)$	$P_B(\%)$	$R_B(\%)$	MBD (m)
0	43.90	49.37	39.52	2.211
0.25	44.45	51.65	39.01	2.051
0.5	45.29	51.10	40.66	2.179
0.75	45.19	49.87	41.32	2.150
1	41.59	47.62	36.91	2.381

of 0.221 m in MBD. Here, “IAM” refers to the lane-level queries generated by the instance activation map [48]. The reason is that Q^c is capable of providing a parametric shape prior for lanes, which helps the model better capture the overall geometric structure and curvature of 3D lanes.

3D Gaussian matching loss. We further analyze the effect of the 3D Gaussian matching loss L_{unc} by varying its weight γ_1 , as reported in Tables V and VI. Compared to setting γ_1 to 0, assigning an appropriate weight of $\gamma_1 = 0.5$ improves $F1_B$ by 1.39%. However, increasing γ_1 to 1 overemphasizes uncertainty, leading to a decline in 3D point localization accuracy. These observations highlight the importance of balancing the contribution of the 3D Gaussian matching loss to achieve optimal performance in 3D lane detection.

Uncertainty modeling choices. To clarify our choice of modeling aleatoric uncertainty in local lane structures, which includes lateral λ^w and vertical λ^h uncertainties, we compare shared uncertainty $\lambda^w = \lambda^h$ and separate uncertainty $\lambda^w \neq \lambda^h$. As shown in Fig. 9, using shared uncertainty results in a noticeable performance drop of 4.78% in $F1_B$ compared to separate uncertainty. This approach even performs worse than the baseline without uncertainty modeling (40.51% vs. 43.90% in $F1_B$ and 2.482 m vs. 2.211 m in MBD). The reason is that lateral and vertical directions exhibit different observation noise characteristics, requiring separate modeling. Fig. 8 visualizes intermediate results, depicting lanes as a series of segments, where thicker segments with higher opacity indicate greater uncertainty. Both lateral and vertical uncertainties increase with depth.

V. CONCLUSION

In this work, we propose MonoUnc, a Transformer-based framework for monocular 3D lane detection. By leveraging curve-point queries, MonoUnc effectively captures the multi-scale geometric structures of lanes, enabling accurate 3D localization. Furthermore, local-structure-aware aleatoric uncertainty is modeled as a 3D Gaussian and optimized via a 3D Gaussian matching loss to learn spatially varying noise. Beyond lane detection, our method shows promise for applications such as camera-based 3D object detection and 3D semantic scene completion. In future work, multi-modal

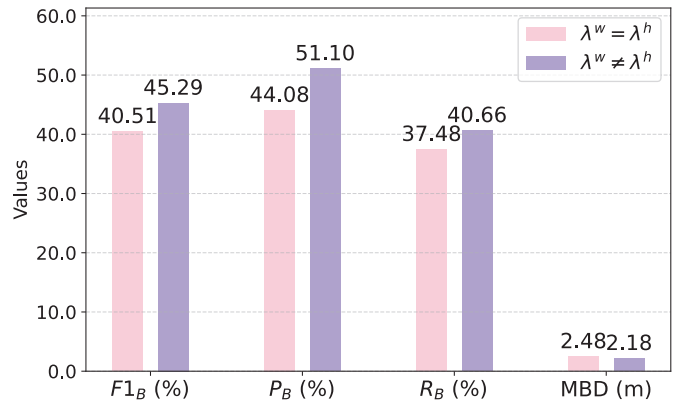


Fig. 9. Ablation studies on uncertainty modeling.

and multi-frame data will be incorporated to enhance the robustness and scalability.

REFERENCES

- [1] Q. Li, Y. Wang, Y. Wang, and H. Zhao, “Hdmapnet: An online hd map construction and evaluation framework,” in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2022, pp. 4628–4634.
- [2] B. Liao, S. Chen, X. Wang, T. Cheng, Q. Zhang, W. Liu, and C. Huang, “MapTR: Structured modeling and learning for online vectorized HD map construction,” in *Int. Conf. Learn. Represent. (ICLR)*, 2023.
- [3] R. Liu and Z. Yuan, “Compact hd map construction via douglas-peucker point transformer,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 4, 2024, pp. 3702–3710.
- [4] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, “Trafficpredict: Trajectory prediction for heterogeneous traffic-agents,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 01, 2019, pp. 6120–6127.
- [5] H. Chen, J. Wang, K. Shao, F. Liu, J. Hao, C. Guan, G. Chen, and P.-A. Heng, “Traj-mae: Masked autoencoders for trajectory prediction,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, October 2023, pp. 8351–8362.
- [6] C. Cao, X. Chen, J. Wang, Q. Song, R. Tan, and Y.-H. Li, “Cctr: Calibrating trajectory prediction for uncertainty-aware motion planning in autonomous driving,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 19, 2024, pp. 20949–20957.
- [7] A. Kendall and Y. Gal, “What uncertainties do we need in bayesian deep learning for computer vision?” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017.
- [8] F. Yan, M. Nie, X. Cai, J. Han, H. Xu, Z. Yang, C. Ye, Y. Fu, M. B. Mi, and L. Zhang, “Once-3dlanes: Building monocular 3d lane detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 17 143–17 152.
- [9] L. Chen, C. Sima, Y. Li, Z. Zheng, J. Xu, X. Geng, H. Li, C. He, J. Shi, Y. Qiao, and J. Yan, “Persformer: 3d lane detection via perspective transformer and the openlane benchmark,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 550–567.
- [10] N. Garnett, R. Cohen, T. Pe’er, R. Lahav, and D. Levi, “3d-lanenet: end-to-end 3d multiple lane detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 2921–2930.
- [11] Y. Guo, G. Chen, P. Zhao, W. Zhang, J. Miao, J. Wang, and T. E. Choe, “Gen-lanenet: A generalized and scalable approach for 3d lane detection,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 666–681.
- [12] R. Liu, D. Chen, T. Liu, Z. Xiong, and Z. Yuan, “Learning to predict 3d lane shape and camera pose from a single image via geometry constraints,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 2, 2022, pp. 1765–1772.
- [13] R. Wang, J. Qin, K. Li, Y. Li, D. Cao, and J. Xu, “Bev-lanedet: An efficient 3d lane detection based on virtual camera via key-points,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 1002–1011.
- [14] Y. Luo, C. Zheng, X. Yan, T. Kun, C. Zheng, S. Cui, and Z. Li, “Latr: 3d lane detection from monocular images with transformer,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, October 2023, pp. 7941–7952.

- [15] S. Huang, Z. Shen, Z. Huang, Z.-h. Ding, J. Dai, J. Han, N. Wang, and S. Liu, "Anchor3dlane: Learning to regress 3d anchors for monocular 3d lane detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2023, pp. 17 451–17 460.
- [16] S. Huang, Z. Shen, Z. Huang, Y. Liao, J. Han, N. Wang, and S. Liu, "Anchor3dlane++: 3d lane detection via sample-adaptive sparse 3d anchor regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–14, 2024.
- [17] Z. Zheng, X. Zhang, Y. Mou, X. Gao, C. Li, G. Huang, C.-M. Pun, and X. Yuan, "Pvalane: Prior-guided 3d lane detection with view-agnostic feature alignment," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 7, 2024, pp. 7597–7604.
- [18] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018.
- [19] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards end-to-end lane detection: an instance segmentation approach," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, 2018, pp. 286–291.
- [20] N. Homayounfar, W.-C. Ma, S. K. Lakshmikanth, and R. Urtasun, "Hierarchical recurrent attention networks for structured online maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2018.
- [21] Y. Ko, Y. Lee, S. Azam, F. Munir, M. Jeon, and W. Pedrycz, "Key points estimation and point instance segmentation approach for lane detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8949–8958, 2022.
- [22] R. Liu, Z. Guan, Z. Yuan, A. Liu, T. Zhou, T. Kun, E. Li, C. Zheng, and S. Mei, "Learning to detect 3d lanes by shape matching and embedding," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, January 2023, pp. 4291–4299.
- [23] L. Tabelini, R. Berriel, T. M. Paixao, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "Keep your eyes on the lane: Real-time attention-guided lane detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2021, pp. 294–302.
- [24] T. Zheng, Y. Huang, Y. Liu, W. Tang, Z. Yang, D. Cai, and X. He, "Clrnet: Cross layer refinement network for lane detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2022, pp. 898–907.
- [25] R. Liu, Z. Yuan, T. Liu, and Z. Xiong, "End-to-end lane shape prediction with transformers," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, January 2021, pp. 3694–3702.
- [26] Y. Bai, Z. Chen, Z. Fu, L. Peng, P. Liang, and E. Cheng, "Curveformer: 3d lane detection by curve propagation with curve queries and attention," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2023, pp. 7062–7068.
- [27] Y. Bai, Z. Chen, P. Liang, and E. Cheng, "Curveformer++: 3d lane detection by curve propagation with temporal curve queries and attention," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 6, pp. 7909–7920, 2025.
- [28] Z. Feng, S. Guo, X. Tan, K. Xu, M. Wang, and L. Ma, "Rethinking efficient lane detection via curve modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2022, pp. 17 062–17 070.
- [29] H. Chen, M. Wang, and Y. Liu, "Bsnet: Lane detection via draw b-spline curves nearby," *arXiv preprint arXiv:2301.06910*, 2023.
- [30] M. Pittner, A. Condurache, and J. Janai, "3d-splinenet: 3d traffic line detection using parametric spline representations," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, January 2023, pp. 602–611.
- [31] M. Pittner, J. Janai, and A. P. Condurache, "Lanecpp: Continuous 3d lane detection using physical priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2024, pp. 10 639–10 648.
- [32] Y. Chen, L. Tai, K. Sun, and M. Li, "Monopair: Monocular 3d object detection using pairwise spatial relationships," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 12 093–12 102.
- [33] H. Chen, Y. Huang, W. Tian, Z. Gao, and L. Xiong, "Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 10 379–10 388.
- [34] Y. Lu, X. Ma, L. Yang, T. Zhang, Y. Liu, Q. Chu, J. Yan, and W. Ouyang, "Geometry uncertainty projection network for monocular 3d object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 3111–3121.
- [35] L. Yan, P. Yan, S. Xiong, X. Xiang, and Y. Tan, "Monocd: Monocular 3d object detection with complementary depths," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 10 248–10 257.
- [36] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, July 2017.
- [37] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2961–2969.
- [38] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," in *Int. Conf. Learn. Represent. (ICLR)*, 2021.
- [39] H. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logist. Quart.*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [40] Z. Guan, R. Liu, Z. Yuan, A. Liu, K. Tang, T. Zhou, E. Li, C. Zheng, and S. Mei, "Flexible 3d lane detection by hierarchical shape matching," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 1, 2023, pp. 694–701.
- [41] J. Mao, M. Niu, C. Jiang, H. Liang, J. Chen, X. Liang, Y. Li, C. Ye, W. Zhang, Z. Li *et al.*, "One million scenes for autonomous driving: Once dataset," *arXiv preprint arXiv:2106.11037*, 2021.
- [42] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 2446–2454.
- [43] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
- [44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2016.
- [45] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Int. Conf. Learn. Represent. (ICLR)*, 2019.
- [46] —, "SGDR: Stochastic gradient descent with warm restarts," in *Int. Conf. Learn. Represent. (ICLR)*, 2017.
- [47] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Int. Conf. Mach. Learn. (ICML)*. PMLR, 2019, pp. 6105–6114.
- [48] T. Cheng, X. Wang, S. Chen, W. Zhang, Q. Zhang, C. Huang, Z. Zhang, and W. Liu, "Sparse instance activation for real-time instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 4433–4442.



Ruixin Liu received the B.S. degree in computer science and technology from Tianjin University, Tianjin, China, in 2019. She is currently pursuing the Ph.D. degree in control science and engineering under the supervision of Dr. Zejian Yuan in Xi'an Jiaotong University, Xi'an, China. Her research interests include computer vision and deep learning.



Zejian Yuan (Member, IEEE) received the M.S. degree in electronic engineering from the Xi'an University of Technology, Xi'an, China, in 1999, and the Ph.D. degree in pattern recognition and intelligent systems from Xi'an Jiaotong University, China, in 2003. He is currently a Professor with the College of Artificial Intelligence, Xi'an Jiaotong University, and a member of the Chinese Association of Robotics. His research interests include image processing, pattern recognition, and machine learning in computer vision.