2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Bi²Lane: Bi-Directional Temporal Refinement with Bi-Level Feature Aggregation for 3D Lane Detection

Chengxin Li[1,2], Yihui Hu[1], Zewen Zheng[3,1], Xiang Gao[4,1], Yongqiang Mou[1✉], Peng Nie[1] and Jun Li[2]

***Abstract*— Monocular 3D lane detection has recently received increasing research attention in autonomous driving due to its application effectiveness and simplicity. However, depending solely on the limited semantic information from a single image makes current monocular detection methods unable to deal with complex scenarios, such as occluded, blurred, and unaligned scenes. In this study, we introduce an end-to-end framework named Bi²Lane which models temporal dependency in a continuous sequence. It recurrently utilizes detected lanes within historical frames as prior information to achieve robust lane detection. Additionally, Bi²Lane employs temporal reverse refinement together with temporal forward refinement to achieve bi-directional temporal refinement (BDTR) while maintaining a robust temporal dependency. For the refined features of different frames, we design a bi-level feature aggregation module (BLFA) to fuse them in both point-level and line-level manners, enabling a comprehensive feature representation to deal with complicated road scenes. Extensive experiments conducted on the OpenLane dataset demonstrate the superiority of Bi²Lane, achieving a notable F1 score of 63.8% using a simple ResNet50 backbone, surpassing the performance of existing state-of-the-art methods.**

I. Introduction

Accurate lane detection is of paramount importance for autonomous driving, as it directly impacts downstream driving planning and control modules. The anticipated lane detection results provide crucial information to the autonomous driving system, enabling it to securely determine the vehicle's position in complex traffic scenarios. Specifically, single-frame lane detection models are built to identify lane positions based on extracting semantic information from an individual 2D image. As depicted in Fig.1, lane detection tasks face significant challenges in complex scenarios, such as occluded, blurred, and unaligned scenes. While predicting 2D lanes on an image [1]–[7] is simple and direct, they encounter difficulties in accurately addressing these complex scenarios.

Based on 2D lane detection, 3D lane detection adds depth information estimation for lanes from the image, which can to some extent alleviate the aforementioned issues of occlusion, blurriness, and instability. As an emerging field, recent unprecedented progress has been achieved in 3D lane detection [8]–[11]. However, existing methods based on single frames still lack sufficient contextual information to handle the mentioned challenges, causing mistake predictions that heavily affect the safety of autonomous driving. Considering that the historical information (e.g., historical frames) is easily accessible in autonomous driving, it may promote lane detection performance in complex scenarios and enhance system's perception of contextual road.

Fig. 1. Motivation behind Bi²Lane: the predictions from the historical frame serve as anchors in the current frame, enhancing single-frame resilience to environmental variations. For example, (a) illustrates the supplementation of an occluded lane from the historical frame. (b) illustrates the process of patching the blurred lane from the historical frame. (c) illustrates the lane alignment achieved through acquiring historical lane information.

In light of this, we argue that temporal features within multi-frame are crucial for lane detection, and modeling temporal dependency can significantly enhance the detection model's ability to recognize complex scenarios involving occluded, blurred, and unaligned, and other disturbed scenes. The temporal dependency consists of object motions which is essential for maintaining the perception of dynamic objects and object supplements for more comprehensive features of objects. Modeling temporal dependency helps multi-frame detection models be better aware of the contextual road environment and more adaptable at predicting the positions of objects in complex scenarios. Existing approaches usually warp historical objects or features to the current frame [12]–[16] to fuse temporal features. However, these temporal approaches lack the simultaneous consideration of capturing object motions and complementing object information, failing to model temporal dependency. Moreover, existing feature fusion methods usually do not account for levels within object features, such as positional information of

✉Represent the Corresponding Author.
[1]R&D Center, Guangzhou Automobile Group Co Ltd., China, {huyihui, mouyongqiang, niepeng}@gacrnd.com.
[2]South China Academy of Advanced Optoelectronics, South China Normal University, China, lichengxin@gacrnd.com, jun.li@coer-scnu.org.
[3]School of Computer Science and Technology, Guangdong University of Technology, China, 2112105305@mail2.gdut.edu.cn.
[4]School of Information Engineering, Guangdong University of Technology, China, 2112103140@mail2.gdut.edu.cn.

point-level and structural information of line-level, which might lead to difficult learning for the network as those features have distinct distribution characteristics.

To address the above challenges, we propose a temporal 3D lane detection framework named Bi²Lane, which recurrently refines lane features guided by anchor for establishing temporal dependency in a continuous sequence. In Bi²Lane, we construct a bi-directional temporal refinement module (BDTR) to efficiently upgrade contextual information and a bi-level feature aggregation module (BLFA) to enhance lane feature representations. The BDTR consists of a forward refinement process and a reverse refinement process, better establishing temporal dependency of lane across consecutive frames. Firstly, the forward refinement encompasses the motion of lane over continuous frames, maintaining perception of lane lines. Secondly, in the reverse, by utilizing current information as a reference to mine relevant clues from historical frames, it acquires the ability to complement lane. As shown in Fig. 1, BDTR could efficiently utilize contextual information, reducing the impact of surrounding disturbances and obtaining more robust detection results than sing-frame models in complex scenarios. Furthermore, we introduce BLFA to aggregate lane features at point level and line level, respectively, enhancing feature representations for lane detection.

Our main contributions are as follows:

- We design a novel bi-directional temporal refinement module to establish temporal dependency and complement lane information in a continuous sequence, alleviating the issues in complex scenarios and improving the robustness of detection results.
- We disentangle lane features at both point-level and line-level and develop a bi-level feature aggregation module to fuse them for fine-grained representation.
- Extensive experiments conducted on the OpenLane dataset demonstrate the superiority of Bi²Lane, achieving a notable F1 score of 63.8% using a simple ResNet50 backbone, surpassing the performance of existing state-of-the-art methods.

II. Related Work

A. Monocular 2D Lane Detection

2D lane detection aims to locate lane positions from images captured by vehicle cameras. Some works [17]–[20] define 2D lane detection as a pixel-wise segmentation task, extracting semantically informative features in front view (FV) image. However, the segmentation results usually need further image post-processing to get accurate lane representation, which brings complexity and expensive computation. Therefore, flexible pre-defined anchors are employed to detect 2D lanes. LineCNN [21] defines straight rays emanating from the image boundary to fit 2D lanes. LaneATT [22] regresses lanes on the pre-defined ray-anchors. CLRNet [23] dynamically refines the start point and angle of ray-anchors through pyramidal features. The spatial-temporal layers [24] utilize recurrent neural network to extract lane features among the forward continuous frames. Despite obtaining some progress, the performance of detection model would experience a sharp decline in complex scenarios such as intersections and slopes due to the insufficient estimation of depth information.

B. Monocular 3D Lane Detection

Based on 2D lane detection, 3D lane detection focuses on reconstructing the geometric features of lane in 3D space. By estimating depth information of lanes, it attains improved performance in complex scenarios while also fostering enhanced adaptability for downstream tasks. 3DLaneNet [8] introduces an anchor-based 3D lane representation method, projecting extracted features from FV into bird's eye view (BEV). Gen-LaneNet [9] predicts lanes in BEV and maps them back to real-world coordinates. PersFormer [10] obtains BEV features with a spatial transformation module. CLGo [25] estimates camera pose to predict 3D lanes from generated top-view image. BEV-LaneDet [11] establishes a virtual camera to unify FV features. CurveFormer [26] employs a dynamic anchor point set to compute 3D lane from image features. While 3D lanes can obtain estimated depth information, these general single-frame detection models remain susceptible to occluded, blurred, and unaligned scenes in practical scenarios, would lead to inconsistency between consecutive frames in terms of missed, misjudged, or other unrobust detection results.

C. Temporal Modeling

Existing works in the field of 3D object detection have considered temporal modeling in multi-frame detection. BEVDet4D [12] utilizes ego-motion to align historical BEV features and concatenate them with current frame. DETR4D [27] fuses current objects from concatenated past objects and current objects with attention mechanisms. However, these approaches overlook the dynamic variations of objects within a continuous sequence. For 3D lane detection, StreamPETR [28] adopts a recurrent manner, using object queries to propagate features across multi-frame. Similarly, BEVFormer [16] employs a recurrent mechanism to fuse historical BEV features with the current frame. Despite incorporating dynamic object variations, it still lacks explicit temporal dependency modeling and the precise fine-grained disentanglement of lane features. In contrast, we develop a bi-directional temporal refinement module to model temporal dependency in a continuous sequence for enhancing contextual lane information and further introduce a bi-level feature aggregation module to enhance the lane feature representations at point level and line level, respectively.

III. Method

The overall architecture of our Bi²Lane is illustrated in Fig. 2. The framework solely leverages FV features in a continuous sequence for end-to-end 3D lane detection. While conventional lane detection methods regress lane coordinates in images, our framework aims to detect lane coordinates directly in the 3D space. Therefore, we define anchors in 3D

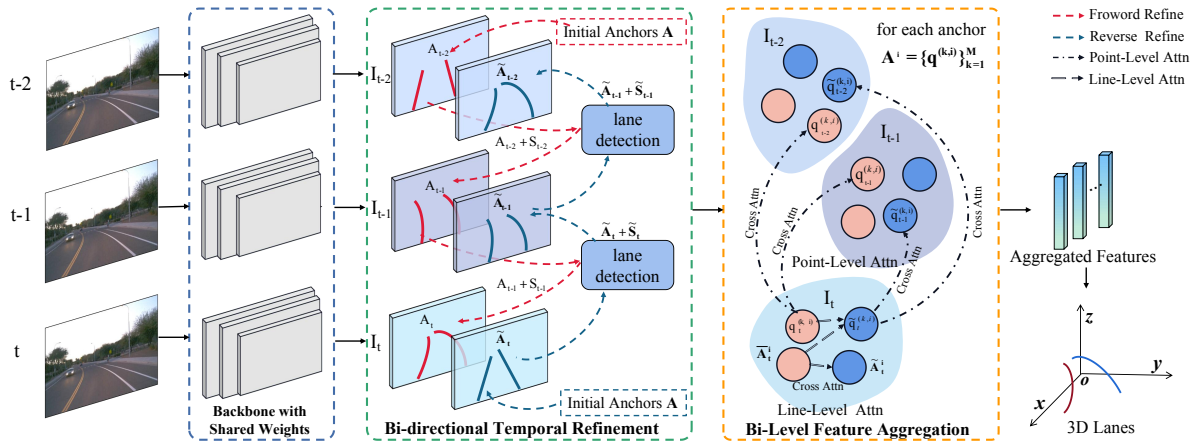

Fig. 2. The overall architecture of Bi²Lane, which mainly contains a bi-directional temporal refinement (BDTR) module to generate temporal-dependent lane features after backbone and a bi-level feature aggregation (BLFA) module to form aggregated features in consecutive frames, finally feed to head for 3D lane detection.

space and project them onto the FV feature map to sample corresponding features, which are then used to regress 3D coordinates.

Specifically, given a continuous sequence of FV images, we extract features through an image backbone. With these multi-frame features, we introduce 3D anchors to sample lane features at corresponding spatial positions across frames to generate lane feature, which is short for anchor-guided temporal lane feature. The bi-directional temporal refinement (BDTR) module establishes temporal dependency by recurrently updating the temporal feature guided by 3D anchors in forward and reverse directions in a continuous sequence. Subsequently, the bi-level feature aggregation (BLFA) module is employed to enhance lane features at fine-grained disentanglement from historical frames to current frames. Finally, the fused features are fed into the 3D lane detection head to regress lane 3D coordinates.

A. Anchor-Guided Temporal Refinement

In this subsection, we outline the process of generating and refining temporal features through anchor-guided design.

1) Generation: To avoid introducing BEV representations and generating more precise lane features, we extract the lane-corresponding temporal feature from the FV feature using anchors defined in 3D space. The extracted lane feature serves as the foundational modeling element.

Specifically, the $i-th$ anchor is designed by 3D points with M y-coordinates $\mathbf{y} = \{y^k\}_{k=1}^{M}$. Combined with $(\mathbf{x}_i^k, \mathbf{z}_i^k)$ which are the horizontal and vertical locations of $\mathbf{y}_i^k$, the 3D lane line is defined based on the 3D anchor $\mathbf{A}^i = \left\{\mathbf{q}^{(k,i)}\right\}_{k=1}^{M}$. With the time step of t to represent the current frame, a sequence of FV images is first encoded to extract multi-frame features maps $\{\mathbf{I}_{t-N}, \ldots, \mathbf{I}_t\} \in \mathbb{R}^{N \times H_I \times W_I \times C}$, where N is the number of frames and H_I, W_I, and C represent the height, width, and number of channels in each FV feature map, respectively.

At the time step of $t-1$, the lane feature $\mathbf{F}_{t-1}^i \in \mathbb{R}^{M \times C}$ is the feature sampled by anchor $\mathbf{A}_{t-1}^i$ which projected onto the FV feature map $\mathbf{I}_{t-1}$ by coordinates transformation function $\mathscr{P}_{g2I}(\cdot)$ from ground coordinates to camera coordinates. Meanwhile, the generated lane feature is pushed into a lane feature set.

$$\mathbf{F}_{t-1}^i = \mathscr{S}\left(\mathscr{P}_{g2I}(\mathbf{A}_{t-1}^i),\ \mathbf{I}_{t-1}\right) \tag{1}$$

where $\mathscr{S}(\cdot)$ stands for sampling temporal features by the anchor.

2) Refinement: Given the lane feature, we employ a regression head to output the offset of the projected anchor. With anchor alignment between two frames, it easily ensures temporal dependency consistency when taking the previous anchor and its offset that is the detected lane directly as a new anchor to extract current features for lane feature refinement.

Feeding the lane feature $\mathbf{F}_{t-1}^i$ into a regression head, the offsets $\mathbf{S}_{t-1}^i = \left\{(\Delta\mathbf{x}_{t-1}^{(k,i)}, \Delta\mathbf{z}_{t-N}^{(k,i)})\right\}_{k=1}^{M}$ from anchor to lane is predicted:

$$\mathbf{S}_{t-1}^i = \mathscr{O}(\mathbf{F}_{t-1}^i) \tag{2}$$

where $\mathscr{O}(\cdot)$ is the regression prediction head to predict anchor offset.

For anchor-guided temporal feature refinement, $\mathbf{A}_{t-1}^i + \mathbf{S}_{t-1}^i$ is aligned as anchor $\mathbf{A}_t^i$ at the time step of t and then extracts lane feature $\mathbf{F}_t^i$ on FV features $\mathbf{I}_t$:

$$\mathbf{A}_t^i = \mathbf{T}_{t-1\to t}(\mathbf{A}_{t-1}^i + \mathbf{S}_{t-1}) \tag{3}$$

$$\mathbf{F}_t^i = \mathscr{S}\left(\mathscr{P}_{g2I}(\mathbf{A}_t^i),\ \mathbf{I}_t\right) \tag{4}$$

where $\mathbf{T}_{t-1\to t} \in \mathbb{R}^{3\times 4}$ is the transformation matrix from the $t-1$-frame to the t-frame. In the end, we can obtain lane features for each frame in a continuous sequence.

B. Bi-Directional Temporal Refinement

Based on the lane feature refinement, we introduce a more comprehensive module, BDTR, as shown in Fig. 3. The BDTR consists of a forward temporal refinement process for

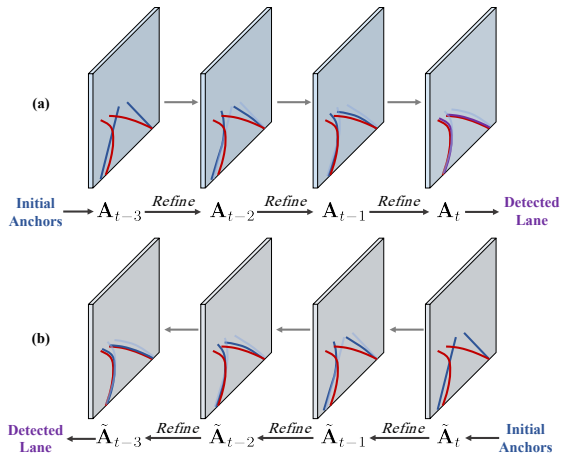

Fig. 3. Illustration of bi-directional temporal refinement process.

capturing lane motions and a reverse temporal refinement process for complementing lane spatial features. It aims to model temporal dependency to enhance contextual lane information, thus better smoothing lane detection results in complex scenarios, and providing additional constraints to enhance the performance of sparse tasks.

1) Forward Temporal Refinement: In the forward temporal refinement process, it recurrently employs the previously detected lane as anchor points to refine lane features until propagating to the current frame. Consequently, the alignment of lane features throughout a consecutive sequence closely follows the distribution of initially detected lanes, facilitating effective propagation of lane motion information and robustly capturing temporal motion patterns for lane representation.

Overall, the forward refinement is from $\mathbf{A}^i_{t-n}$ to $\mathbf{A}^i_{t-(n-1)}$, in which $n \in (N,\ N-1,\ \ldots\ ,\ 1)$:

$$\mathbf{A}^i_{t-(n-1)} = \mathscr{F}_{for}\left(\mathbf{A}^i_{t-n}, \mathbf{I}_{t-n}\right) \tag{5}$$

where $\mathscr{F}_{for}(\cdot)$ stands for the above formulas that propagate anchor in a forward manner.

The lane feature set $\left\{\mathbf{F}^i_e\right\}^t_{e=t-N} \in \mathbb{R}^{N\times M\times C}$ is established which contains refined lane features during forward refinement.

2) Reverse Temporal Refinement: In contrast to the forward temporal refinement, the reverse process aims to refine historical lane features to be progressively more consistent with the lane distribution in the current frame. The reverse refinement significantly utilizes historical lane features to complement the current lane, thus helping reduce the influence of surrounding disruptions.

In the reverse branch, the reverse temporal refinement is from anchor $\tilde{\mathbf{A}}^i_{t-\tilde{n}}$ to anchor $\tilde{\mathbf{A}}^i_{t-(\tilde{n}+1)}$, where $\tilde{n} \in (0,\ 1, \ldots\ ,\ N-1)$, and form the reverse refined lane feature set $\left\{\tilde{\mathbf{F}}^i_e\right\}^t_{e=t-N} \in \mathbb{R}^{N\times M\times C}$:

$$\tilde{\mathbf{A}}^i_{t-(n+1)} = \mathscr{F}_{rev}\left(\tilde{\mathbf{A}}^i_{t-n}, \mathbf{I}_{t-n}\right) \tag{6}$$

where $\mathscr{F}_{rev}(\cdot)$ represents propagating anchor in a reverse manner.

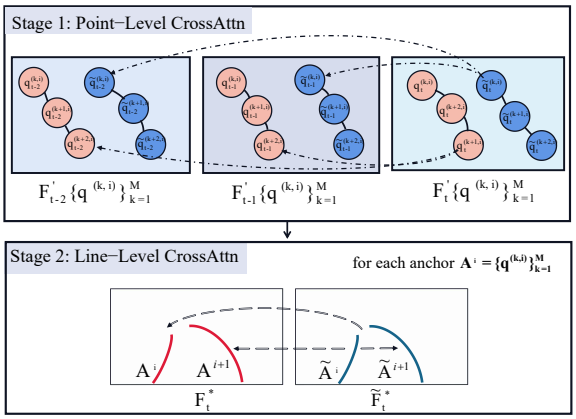

Fig. 4. Illustration of bi-level feature aggregation, where Stage 1 is the enhancement of temporal refined features at point level across frames and Stage 2 is the interaction of line-level features in the current frame.

C. Bi-Level Feature Aggregation

As the refined lane feature sets from BDTR are separated in bi-directions, it is essential to aggregate these features, enabling stronger lane feature representations. We introduce the BLFA to perform a separate fusion at point level and line level within lane features in a continuous sequence, as shown in Fig. 4.

1) Point-level Feature Aggregation: We treat k-th point $\mathbf{q}^{(k,i)}_t$ on lane feature $\mathbf{F}^i_t$ from i-th projected anchor at t-frame as the Query and the point set $\left\{\mathbf{q}^{(k,i)}_e\right\}^{t-1}_{e=t-N}$ on historical lane feature set $\left\{\mathbf{F}^i_e\right\}^{t-1}_{e=t-N}$ as both the Key and the Value for the Transformer-based [29] point-level decoder. The decoder performs cross-attention to fuse long-term dependencies and highlight relevant information at the point level, obtaining the fused feature point $\mathbf{q}^{(k,i)\ '}_t$.

$$\mathbf{q}^{(k,i)\ '}_t = \mathrm{CrossAttn}\left(\mathbf{q}^{(k,i)}_t, \left\{\mathbf{q}^{(k,i)}_e\right\}^{t-1}_{e=t-N}\right) + \mathbf{q}^{(k,i)}_t \tag{7}$$

$$\tilde{\mathbf{q}}^{(k,i)\ '}_t = \mathrm{CrossAttn}\left(\tilde{\mathbf{q}}^{(k,i)}_t, \left\{\tilde{\mathbf{q}}^{(k,i)}_e\right\}^{t-1}_{e=t-N}\right) + \tilde{\mathbf{q}}^{(k,i)}_t \tag{8}$$

2) Line-level Feature Aggregation: We concatenate features of point set $\left\{\mathbf{q}^{(k,i)\ '}_t\right\}^{M}_{k=1}$ belonging to the i-th anchor as its feature representation $\mathbf{F}^{i\ '}_t \in \mathbb{R}^{M\times C}$ at t-frame. Taking $\mathbf{F}^{i\ '}_t \in \mathbb{R}^{M\times C}$ as Query, Key and Value, an informative lane feature $\mathbf{F}^{i\ *}_t \in \mathbb{R}^{M\times C}$ is generated:

$$\mathbf{F}^{i\ *}_t = \mathrm{CrossAttn}\left(\mathbf{F}^{i\ '}_t, \mathbf{F}^{i\ '}_t\right) + \mathbf{F}^{i\ '}_t \tag{9}$$

$$\tilde{\mathbf{F}}^{i\ *}_t = \mathrm{CrossAttn}\left(\tilde{\mathbf{F}}^{i\ '}_t, \tilde{\mathbf{F}}^{i\ '}_t\right) + \tilde{\mathbf{F}}^{i\ '}_t \tag{10}$$

Then, the fused current lane features $\mathbf{F}^{i\ '}_t$ and $\tilde{\mathbf{F}}^{i\ *}_t$ is concatenated to get the final lane feature $\widehat{\mathbf{F}}^i_t$ at t-frame:

$$\widehat{\mathbf{F}}^i_t = \varepsilon\left(\mathbf{F}^{i\ *}_t, \tilde{\mathbf{F}}^{i\ *}_t\right) \tag{11}$$

where $\varepsilon(\cdot)$ refers to the operation of a concatenation.

TABLE I

COMPARISON WITH SOTA METHODS ON OPENLANE VALIDATION SET, WHERE "†" REPRESENTS MULTI-FRAME FEATURE FUSION COMBINED WITH ITERATIVE REGRESSION IN ANCHOR3DLANE, "F" DENOTES 40-100M AND "C" DENOTES 0-40M.

Method	F1(%)↑	x err/C(m)↓	x err/F(m)↓	z err/C(m)↓	z far/F(m)↓
3D-LaneNet	44.1	0.479	0.572	0.367	0.443
GenLaneNet	32.3	0.591	0.684	0.411	0.521
PersFormer	50.5	0.485	0.553	0.364	0.431
Anchor3DLane	53.1	0.300	0.311	0.103	0.139
Anchor3DLane†	54.3	0.275	0.310	0.105	0.135
BEV-LaneDet	58.4	0.309	0.659	0.244	0.631
Bi²Lane (Ours)	**63.8**(↑5.4)	**0.222**(↓0.087)	**0.241**(↓0.418)	**0.093**(↓0.151)	**0.116**(↓0.515)

TABLE II

COMPARISON IN F1 SCORES WITH SOTA METHODS ON OPENLANE VALIDATION SET OF DIFFERENT SCENARIOS, WHERE "†" REPRESENTS MULTI-FRAME FEATURE FUSION COMBINED WITH ITERATIVE REGRESSION IN ANCHOR3DLANE.

Method	Mean(%)↑	Up&Down	Intersection	Merge&Split	Extreme Weather	Night	Curve
3D-LaneNet	41.7	40.8	32.1	41.7	47.5	41.5	46.5
GenLaneNet	26.4	25.4	21.4	31.0	28.1	18.7	33.5
PersFormer	47.3	42.4	40.0	50.7	48.6	46.6	55.6
Anchor3DLane	49.3	45.5	44.2	50.5	51.9	47.2	56.2
Anchor3DLane†	50.7	47.2	45.8	51.7	52.7	48.7	58.0
BEV-LaneDet	53.8	48.7	50.3	53.7	53.4	53.4	63.1
Bi²Lane (Ours)	**60.1**(↑6.3)	**54.6**	**54.8**	**62.5**	**60.8**	**58.0**	**69.2**

D. Loss Function

We employ a classification head and a regression head based on the aggregated lane feature $\widehat{\mathbf{F}}_t^i$ from the projected i-th anchor for current lane detection to predict its lane classification probability $\mathbf{c}^i \in \mathbb{R}^B$, anchor points offsets $\Delta\mathbf{x}_t^i \in \mathbb{R}^M, \Delta\mathbf{z}_t^i \in \mathbb{R}^M$, and visibility of each point $\mathbf{vis}^i \in \mathbb{R}^M$ respectively. Combined with initial anchor $\mathbf{A}^i = \left\{\mathbf{q}^{(k,i)}\right\}_{k=1}^{M}$, the current 3D lane proposal can be generated as $\mathbf{P}_t^i = (\mathbf{c}^i, \mathbf{x}^i + \Delta\mathbf{x}_t^i, \mathbf{y}, \mathbf{z}^i + \Delta\mathbf{z}_t^i, \mathbf{vis}^i)$. With the i-th current ground-truth lane is defined as $\mathbf{G}_t^i = \left(\check{\mathbf{x}}_t^i, \check{\mathbf{z}}_t^i, \check{\mathbf{vis}}_t^i\right)$. We adopt $\{\mathbf{P}_t^i\}_{i=1}^{M_{pos}}$ for positive proposal associated with current ground-truth lanes $\left\{\mathbf{G}_t^i\right\}_{i=1}^{M_{pos}}$ to calculate the regression loss function. Particularly, we introduce a separate BEV segmentation loss $\mathscr{L}_{\text{seg}}$ to enhance the feature expression of our framework, where the BEV feature is transformed from FV feature.

$$\mathscr{L}_{\text{cls}} = -\sum_{i=1}^{M}\sum_{b=1}^{B} \alpha^b (1-c_i^b)^{\gamma} \log c_i^b, \tag{12}$$

where α^b and γ are the hyperparameters for focal loss [32].

$$\begin{aligned}\mathscr{L}_{\text{reg}} = &\sum_{i=1}^{M_{pos}}\sum_{k=1}^{M}\left\|\check{\mathbf{vis}}_t^{(k,i)} \cdot \left(\mathbf{x}^{(k,i)} + \Delta\mathbf{x}_t^{(k,i)} - \check{\mathbf{x}}_t^{(k,i)}\right)\right\|_1 \\ &+ \sum_{i=1}^{M_{pos}}\sum_{k=1}^{M}\left\|\check{\mathbf{vis}}_t^{(k,i)} \cdot \left(\mathbf{z}^{(k,i)} + \Delta\mathbf{z}_t^{(k,i)} - \check{\mathbf{z}}_t^{(k,i)}\right)\right\|_1 \\ &+ \sum_{i=1}^{M_{pos}}\sum_{k=1}^{M}\left\|\check{\mathbf{vis}}_t^{(k,i)} - \mathbf{vis}^{(k,i)}\right\|_1 .\end{aligned} \tag{13}$$

With λ_{cls}, λ_{reg}, and λ_{seg} representing the weight of the corresponding loss item, respectively, the total loss function of our Bi²Lane is the sum of above losses:

$$\mathscr{L}_{total} = \lambda_{cls}\mathscr{L}_{cls} + \lambda_{reg}\mathscr{L}_{reg} + \lambda_{seg}\mathscr{L}_{seg} \tag{14}$$

IV. EXPERIMENTS

A. Datasets and Experimental Setting

1) Datasets: OpenLane is a large-scale real-world 3D lane detection dataset constructed from the Waymo Open [30] dataset, providing annotations for 1000 road segments with lane and Closest In-Path Objects (CIPO) information. It comprises 200K frames with over 880K meticulously annotated lanes.

ONCE-3DLanes [31] is a real-world dataset designed for 3D lane detection. The camera data from ONCE-3DLanes is downsampled along with the LiDAR data recorded at a speed of 10 frames per second (FPS) and downsampled with the rate of 2 FPS. Considering the serious loss of continuity in multi-frame images due to excessive sampling frequency in ONCE-3DLanes, we conclude that it lacks temporal information. Therefore, we do not conduct experiments on it as our framework is designed to utilize temporal information for robust multi-frame lane detection.

2) Implementation Details: We conduct experiments with ResNet-50 [32] under ImageNet [33] pre-trained weights as the image backbones. Bi²Lane is trained by Adam optimizer with a batch size of 64. The learning rate is set to 2e-4. During training, we select a time window of 5 and choose the 3 nearest frames to the current frame as historical frames. During testing, we directly select the 3 nearest frames for detecting 3D lanes on the current frame.

B. Qualitative Results

We compare our Bi²Lane with other state-of-the-art 3D lane detection methods on the OpenLane validation set and different scenarios, as shown in TABLE I and TABLE II, respectively. Our Bi²Lane outperforms Anchor3DLane† and BEV-LaneDet by 9.5% and 5.4% in F1 score, respectively, indicated in TABLE I. Moreover, compared to BEV-LaneDet, our approach significantly reduces close x errors, far x errors,

TABLE III

ABLATION STUDY ON MODULES OF BI²LANE.

Model	F1(%)	x err/C(m)	x err/F(m)	z err/C(m)	z far/F(m)
Baseline	55.7	0.267	0.292	0.098	0.126
+BDTR	59.8	0.251	0.274	0.097	0.124
+BLFA (Bi²Lane)	**60.0**	**0.251**	**0.251**	**0.097**	**0.123**

TABLE IV

ABLATION STUDY ON FEATURE LEVELS OF AGGREGATION.

Feature Level	F1(%)	x err/C(m)	x err/F(m)	z err/C(m)	z far/F(m)
Point-level	59.8	0.251	0.274	0.097	0.124
Line-level	59.7	0.251	0.270	0.098	0.125
Bi-Level (Bi²Lane)	**60.0**	**0.251**	**0.251**	**0.097**	**0.123**

TABLE V

ABLATION STUDY ON DIRECTIONS OF TEMPORAL REFINEMENT.

Direction	F1(%)	x err/C(m)	x err/F(m)	z err/C(m)	z far/F(m)
Forward	58.8	**0.242**	0.284	0.098	0.128
Reverse	59.6	0.261	0.281	0.102	0.129
Bi-Directional (Bi²Lane)	**60.0**	0.251	**0.251**	**0.097**	**0.123**

close z errors, and far z errors by 28.2%, 63.4%, 61.9%, and 81.6%, respectively. In addition, our approach has achieved significant improvements in complex scenarios shown in TABLE II. Compared to Anchor3DLane† and BEV-LaneDet, our method shows an astonishing performance gain of 9.4% and 6.3%, respectively. In the Merge&Split, Extreme Weather, and Curve scenarios, temporal dependency helps the model capture lane variations involving lane merging, splitting, deformation, or curve shape changes more accurately. However, in the Up&Down, Night, and Intersection scenarios, due to road undulations, poor lighting, and complex traffic situations, the temporal dependency is limited.

C. Ablation Study and Analysis

We conduct experiments to study the effectiveness of our proposed modules based on the OpenLane validation set with Resnet-18 as the backbone. We demonstrate the significance of our different modules shown in TABLE III. It is evident that compared to the baseline, our bi-directional temporal refinement module (BDTR) results in an improved F1 score of 59.8%. Additionally, the bi-level feature aggregation module (BLFA) benefits from the separate fusion of point-level and lane-level features, improving F1 score 4.3% from baseline.

1) Bi-Level Feature Aggregation: We decompose the lane features from BDTR into point level, line level, and bi-level for lane detection, demonstrated in TABLE IV. It can be seen that Bi²Lane achieves the highest F1 score of 60.0% among different feature levels of aggregation in TABLE IV. Since the lane features contain positional information and structural information, such as shapes, lengths, and orientations, bi-level feature aggregation could enhance lane feature representations more comprehensively.

2) Bi-Directional Temporal Refinement: Compared with single-directional temporal refinements, our bi-directional temporal refinement emerged with the lowest far x errors of 0.251, shown in TABLE V. The forward temporal refinement partially disregard lane spatial feature information from subsequent frames, potentially leading to incomplete lane feature complement for current frame after a long term. Similarly, the reverse assumes that current information is the most pivotal within the sequence, potentially hindering accurately capturing lane motions when current frame faces surrounding disturbances. In contrast, our bi-directional temporal refinement propagates information forward and reversely, capturing the temporal motion and complementing the spatial features of lanes within a continuous sequence, significantly reducing detected lane position errors.

TABLE VI

ABLATION STUDY ON THE NUMBER OF HISTORICAL FRAMES.

Frame Number	F1(%)	x err/C(m)	x err/F(m)	z err/C(m)	z far/F(m)	Inference Time(ms)
1	59.1	0.256	0.278	0.098	0.126	**29**
2	59.2	**0.244**	0.275	0.098	0.126	33
3	**60.0**	0.251	**0.251**	0.097	**0.123**	37
4	59.3	0.247	0.275	**0.092**	0.127	41
5	59.3	0.256	0.278	0.097	0.126	44

TABLE VII

ABLATION STUDY ON TEMPORAL ALIGNMENT OF BI²LANE, WHERE 'BI²LANE(W/O)' MEANS WITHOUT TEMPORAL ALIGNMENT.

Model	F1(%)	x err/C(m)	x err/F(m)	z err/C(m)	z far/F(m)
Bi²Lane(w/o)	59.6	0.258	0.274	0.098	0.126
Bi²Lane	**60.0**	**0.251**	**0.251**	**0.097**	**0.123**

3) Number of Historical Frames: TABLE VI presents the evaluation results of our Bi²Lane with varying numbers of historical frames, suggesting that a short continuous sequence is not sufficient to establish temporal dependency for improving detection performance, and excessively long continuous sequences might bring disconnection between the historical and current frames. Besides, we can see that our Bi²Lane can meet the real-time requirements in the case of different frames, and the inference time would not increase significantly when increasing the number of input frames.

4) Temporal Alignment: The temporal alignment aligns the anchor points from the previous frame to the subsequent one using ego-pose data, ensuring a consistent lane positioning across frames. We contrast the performance of our Bi²Lane, with and without temporal alignment, as demonstrated in TABLE VII, and find that temporal alignment indeed helps Bi²Lane to decrease the far x errors to 0.251.

V. CONCLUSION

In this study, we propose Bi²Lane, a robust multi-frame 3D lane detection framework, migrating the disturbances in complex scenarios and improving detection robustness while not introducing post-processing computation or hard-to-obtain information. Different from the previous works, our method explores an anchor-guided temporal refinement paradigm that upgrades temporal features through recurrent anchors across frames. We develop a bi-directional temporal refinement module to establish temporal dependency in a continuous sequence, greatly complementing lane features. Furthermore, a bi-level feature aggregation module is introduced to fuse the refined lane feature sets in bi-directions at point level and line level, enhancing lane feature representations. Extensive experiments show that Bi²Lane achieves remarkable competitive performance, outperforming existing state-of-the-art lane detection approaches.

REFERENCES

[1] 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 Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 8949–8958, 2021.

[2] J. Wang, Y. Ma, S. Huang, T. Hui, F. Wang, C. Qian, and T. Zhang, "A keypoint-based global association network for lane detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1392–1401.

[3] Z. Qin, H. Wang, and X. Li, "Ultra fast structure-aware deep lane detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*. Springer, 2020, pp. 276–291.

[4] J. Han, X. Deng, X. Cai, Z. Yang, H. Xu, C. Xu, and X. Liang, "Laneformer: Object-aware row-column transformers for lane detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, 2022, pp. 799–807.

[5] L. Tabelini, R. Berriel, T. M. Paixao, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "Polylanenet: Lane estimation via deep polynomial regression," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 6150–6156.

[6] B. Wang, Z. Wang, and Y. Zhang, "Polynomial regression network for variable-number lane detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*. Springer, 2020, pp. 719–734.

[7] H. Xu, S. Wang, X. Cai, W. Zhang, X. Liang, and Z. Li, "Curvelane-nas: Unifying lane-sensitive architecture search and adaptive point blending," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*. Springer, 2020, pp. 689–704.

[8] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: end-to-end 3d multiple lane detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2921–2930.

[9] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*. Springer, 2020, pp. 666–681.

[10] L. Chen, C. Sima, Y. Li, Z. Zheng, J. Xu, X. Geng, H. Li, C. He, J. Shi, Y. Qiao *et al.*, "Persformer: 3d lane detection via perspective transformer and the openlane benchmark," in *European Conference on Computer Vision*. Springer, 2022, pp. 550–567.

[11] R. Wang, J. Qin, K. Li, and D. Cao, "Bev lane det: Fast lane detection on bev ground," *arXiv preprint arXiv:2210.06006*, 2022.

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

[13] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, X. Zhang, and J. Sun, "Petrv2: A unified framework for 3d perception from multi-camera images," *arXiv preprint arXiv:2206.01256*, 2022.

[14] Y. Li, H. Bao, Z. Ge, J. Yang, J. Sun, and Z. Li, "Bevstereo: Enhancing depth estimation in multi-view 3d object detection with dynamic temporal stereo," *arXiv preprint arXiv:2209.10248*, 2022.

[15] J. Park, C. Xu, S. Yang, K. Keutzer, K. Kitani, M. Tomizuka, and W. Zhan, "Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection," *arXiv preprint arXiv:2210.02443*, 2022.

[16] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European conference on computer vision*. Springer, 2022, pp. 1–18.

[17] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "Hdmapnet: A local semantic map learning and evaluation framework," *arXiv preprint arXiv:2107.06307*, vol. 3, no. 4, p. 7, 2021.

[18] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards end-to-end lane detection: an instance segmentation approach," in *2018 IEEE intelligent vehicles symposium (IV)*. IEEE, 2018, pp. 286–291.

[19] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[20] T. Zheng, H. Fang, Y. Zhang, W. Tang, Z. Yang, H. Liu, and D. Cai, "Resa: Recurrent feature-shift aggregator for lane detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 4, 2021, pp. 3547–3554.

[21] X. Li, J. Li, X. Hu, and J. Yang, "Line-cnn: End-to-end traffic line detection with line proposal unit," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 248–258, 2019.

[22] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 294–302.

[23] T. Zheng, Y. Huang, Y. Liu, W. Tang, Z. Yang, D. Cai, and X. He, "Clrnet: Cross layer refinement network for lane detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 898–907.

[24] Y. Dong, S. Patil, B. van Arem, and H. Farah, "A hybrid spatial–temporal deep learning architecture for lane detection," *Computer-Aided Civil and Infrastructure Engineering*, vol. 38, no. 1, pp. 67–86, 2023.

[25] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 1765–1772.

[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 *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 7062–7068.

[27] Z. Luo, C. Zhou, G. Zhang, and S. Lu, "Detr4d: Direct multi-view 3d object detection with sparse attention," *arXiv preprint arXiv:2212.07849*, 2022.

[28] S. Wang, Y. Liu, T. Wang, Y. Li, and X. Zhang, "Exploring object-centric temporal modeling for efficient multi-view 3d object detection," *arXiv preprint arXiv:2303.11926*, 2023.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

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

[31] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 143–17 152.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.