Consistent Multi-Lane Tracking with Temporally Recursive Spline Modeling

Sanghyeon Lee[1][a], Donghun Kang[2][b] and Min H. Kim[1][c]
[1]*School of Computing, KAIST, Daejeon, South Korea*
[2]*Autonomous Driving Development Team, Hyundai Motor Company, Seongnam, South Korea*
shlee@vclab.kaist.ac.kr, dhkang@hyundai.com, minhkim@vclab.kaist.ac.kr

Keywords: Lane tracking, Autonomous driving

Abstract: Lane recognition and tracking are essential for autonomous driving, providing precise positioning and navigation data for vehicles. Existing single-image lane detection methods often falter in real-world conditions like poor lighting and occlusions. Video-based approaches, while leveraging sequential frames, typically lack continuity in lane tracking, leading to fragmented lane representations. We introduce a novel approach that addresses these challenges through temporally recursive spline modeling, a robust framework designed to maintain consistent, multi-lane tracking over time. Unlike traditional methods that limit tracking to adjacent lanes, our technique models lane trajectories as temporally recursive splines mapped in world space, capturing smooth lane continuity and enhancing long-term tracking fidelity across complex driving scenes. Our framework incorporates 2D image-based lane detections into a recursive spline model, facilitating accurate, real-time lane trajectory representation across frames. To ensure reliable lane association and continuity, we integrate a Kalman filter and an adaptive Hungarian algorithm, allowing our method to enhance baseline detectors and support consistent multi-lane tracking. Experimental results demonstrate that our temporally recursive spline modeling outperforms conventional approaches in lane detection and tracking metrics, achieving superior continuous lane recognition in challenging driving environments.

1 Introduction

In real-world autonomous driving, accurate perception of the vehicle's surroundings is essential for both performance and safety, with lane recognition and tracking serving as core components. Previous research has largely focused on single-image lane detection, which is inherently limited by occlusions and challenging lighting conditions common on the road, often resulting in flickering or intermittent lane recognition when applied frame-by-frame (Figure 1-a).

In contrast, video-based methods (Zou et al., 2020; Zhang et al., 2021a; Zhang et al., 2021b; Wang et al., 2022b; Tabelini et al., 2022; Jin et al., 2023) leverage consecutive frames to improve accuracy, allowing for the detection of occluded lanes based on prior observations. However, these methods primarily enhance single-frame accuracy without achieving consistent lane identification across frames, thus failing to provide true multi-lane tracking or propagate lane continuity into future frames (Figure 1-b).

While lane tracking has seen less focus compared to detection, existing methods predominantly rely on tracking-by-detection frameworks (Borkar et al., 2009; Mammeri et al., 2014; Hou et al., 2016; Nguyen et al., 2018; Meuter et al., 2009; Zhao et al., 2012) that operate independently on each frame and use Kalman filters to estimate lane positions. These methods typically restrict tracking to immediate lanes adjacent to the ego vehicle, overlooking broader multi-lane tracking and often lacking in identity association across the scene. Further, they focus solely on per-frame detection accuracy, disregarding the importance of temporal lane consistency and identity preservation.

Traditional Kalman filter-based approaches are limited in their representation of only the lane segments visible within each frame. However, real-world lanes extend well beyond this limited view, making it essential to extrapolate lane trajectories throughout the entire driving scene. Zhao et al. and Qiao et al. introduce spline-based lane tracking, using con-

[a] https://orcid.org/0009-0007-2109-6098
[b] https://orcid.org/0009-0007-8526-6674
[c] https://orcid.org/0000-0002-5078-4005

Figure 1: (a) Single-image lane detection struggles with obstructions and lighting issues, causing flickering and missing lane markings (Liu et al., 2021). (b) Video-based methods improve accuracy using multiple frames but lack consistent lane identity tracking across frames (Zhang et al., 2021b). (c) Our method uses temporally recursive spline modeling to continuously track multiple lanes with stable identity association across frames. Distinct colors represent unique lanes, while predictive control point updates ensure precise, uninterrupted lane tracking in complex scenes.

trol points to represent lanes; however, these methods lack technical transparency or rely on hand-crafted features and relatively inaccurate 3D image-based detectors (Zhao et al., 2012; Zhijian Qiao and Shen, 2023). Moreover, reproducibility issues arise from the unavailability of datasets and code, with Qiao et al. being the only exception (Zhijian Qiao and Shen, 2023).

Our work introduces a novel approach for multi-lane detection and tracking that is both recursive and temporally robust, enabling continuous, identity-preserving lane tracking over complex scenes. Starting with initial lane measurements from 2D images, we transform these into control points within a spline function, mapping lanes in world space. As the vehicle progresses, we leverage vehicle odometry to refine lane position tracking and extend lane trajectories beyond the visible scene through spline extrapolation. Our method uniquely incorporates a lightweight neural network to predict new control points in real-time, ensuring both accuracy and temporal stability. By coupling this with the Kalman filter and Hungarian algorithm, we achieve consistent lane association across frames.

Additionally, we introduce multi-object tracking metrics to rigorously evaluate lane tracking based on identity association, generating unique lane ID labels for the OpenLane-V dataset (Jin et al., 2023) to support this evaluation.

In summary, our contributions are:

- We enhance single-image-based lane detectors by introducing a robust, multi-lane tracking framework that provides consistent lane identification across frames.
- Our method employs neural networks to enable intuitive yet precise spline extrapolation for continuous, temporally stable lane tracking.
- We adopt multi-object tracking metrics to assess identity-preserving lane tracking performance and expand the OpenLane-V dataset with lane ID annotations to support reproducible benchmarking.
- Our comprehensive experiments demonstrate the effectiveness of our *temporally recursive spline modeling* in achieving continuous, identity-preserving multi-lane tracking, even in complex driving environments.

To promote transparency and foster future advancements, we will release our code and enriched dataset, complete with multi-lane ID annotations.

2 Related Work

2.1 Video-based lane detection

Video-based lane detection methods have emerged to address the inherent limitations of single-image lane detection, primarily by incorporating temporal information from sequential frames. Recent approaches utilize recurrent neural networks, such as LSTM (Zou et al., 2020; Jin and Kim, 2024) and GRU (Zhang et al., 2021a), to enhance detection accuracy by leveraging past frames. Jin et al. introduced an occlusion-aware feature refinement model that integrates obstacle mask estimation to improve robustness under occlusions (Jin and Kim, 2024). Other methods include Zhang et al.'s attention-based network (Zhang et al., 2021b), Wang et al.'s use of temporal and geometrical consistency constraints (Wang et al., 2022b), Tabelini et al.'s anchor-based feature pooling over multiple frames (Tabelini et al., 2022), and Jin et al.'s recursive approach, which refines features based on the motion field between consecutive frames (Jin et al., 2023).

Despite these advancements, existing methods primarily focus on enhancing detection accuracy within the current frame by drawing on previous frames, often neglecting the continuity of lane identities across the sequence. Prior works like RVLD (Jin et al., 2023) and OMR (Jin and Kim, 2024) propagate information through time but do not establish or maintain consistent lane identification across frames.

Our approach introduces a significant departure from prior works by not only enhancing detection accuracy but also ensuring persistent, multi-lane track-

ing with consistent lane identities throughout the entire driving sequence. By employing a novel temporally recursive spline model, our method associates and tracks each lane identity across frames, overcoming the limitations of frame-dependent refinement methods. This approach provides a robust framework for continuous lane recognition that adapts to occlusions and scene changes, making it highly effective for real-world driving conditions.

2.2 Lane tracking

Lane tracking methods based on tracking-by-detection have traditionally relied on gradient-based features followed by Hough transformations to identify lanes in images (Borkar et al., 2009; Meuter et al., 2009; Zhao et al., 2012; Mammeri et al., 2014; Hou et al., 2016). More recent work by Nguyen et al. introduced a method that refines segmentation maps generated by FCNs with gradient-based edge maps and Hough transformation, offering improved accuracy in lane detection (Nguyen et al., 2018).

Many studies have also integrated the Kalman filter (Kalman, 1960) to track lanes over time, typically assuming constant lane velocity (Borkar et al., 2009; Mammeri et al., 2014; Hou et al., 2016; Nguyen et al., 2018; Meuter et al., 2009; Zhao et al., 2012). This approach incorporates a first derivative term to model lane motion, but it uses a static prediction matrix in the Kalman filter, limiting adaptability to the real motion dynamics of the ego vehicle (Borkar et al., 2009; Mammeri et al., 2014; Hou et al., 2016; Nguyen et al., 2018). Alternative methods address this by incorporating the vehicle's yaw rate and velocity directly into lane prediction, although the predictive capability remains limited by the lane model used (Meuter et al., 2009; Zhao et al., 2012).

Various representations for the Kalman filter's lane state have been explored: Borkar et al. use a straight line parameterized by ρ and θ (Borkar et al., 2009), Mammeri et al. define by endpoints (Mammeri et al., 2014), Hou et al. use vanishing points and angles (Hou et al., 2016), while Nguyen et al. combine multiple lines (Nguyen et al., 2018). However, lane shapes in the real world are often complex, demanding more adaptable models. To address this, Meuter et al. use second-order polynomials (Meuter et al., 2009), while Zhao et al. introduced a Catmull-Rom Spline (CRS) (Catmull and Rom, 1974), representing lanes using control points and lane width for smoother lane tracking (Zhao et al., 2012). Historically, lane tracking has been performed in the image plane, focusing on lane shape estimation within each frame without continuity across the driving scene. Zhao et al. extended lane tracking to world space and extrapolated splines using control points, though their extrapolation strategy remains undefined (Zhao et al., 2012).

Our approach, in contrast, provides a well-defined and robust updating strategy for spline control points, using a neural network for adaptive control point prediction and lifecycle management of lanes. Qiao et al. propose a SLAM-based method that integrates 3D lane detection (Chen et al., 2022) and tracks lanes by associating control points as landmarks within a point cloud (Zhijian Qiao and Shen, 2023). However, 3D lane detection relying solely on RGB images suffers from depth ambiguity, leading to reduced accuracy when projected onto the 2D image plane. To address this, our method employs a 2D lane detector, providing accurate initial lane measurements on the image plane as input to our algorithm.

Most existing tracking methods are limited to two lanes—the ones immediately adjacent to the ego vehicle. Although Zhao et al. and Qiao et al. extend their tracking to multiple lanes (Zhao et al., 2012; Zhijian Qiao and Shen, 2023), Zhao et al.'s evaluation is restricted to the two nearest lanes. Moreover, previous methods have concentrated on per-frame detection accuracy, largely overlooking the association of consistent lane IDs across frames. To better assess tracking performance with ID continuity, we adopt multi-object tracking metrics, assigning unique IDs to lanes for robust multi-lane tracking evaluation.

Despite progress in the field, reproducibility remains challenging, as publicly available code and datasets are limited, with Qiao et al. being one of the few exceptions (Zhijian Qiao and Shen, 2023). Our work advances this field by providing a method that reliably tracks multiple lanes with consistent identity association across the driving scene, leveraging a novel temporally recursive spline model and providing our code and dataset annotations for future research.

3 Temporally Recursive Spline Modeling

We introduce a temporally recursive spline model (TRSM), a novel framework for detecting and tracking multiple lanes continuously over time. This model, illustrated in Figure 2, enables robust lane tracking by recursively updating lane trajectories as the vehicle progresses through a sequence. By leveraging vehicle odometry, lane measurements, and color images as inputs for each frame, our method dynamically extrapolates lane positions, ensuring continuous tracking across the scene. As the vehicle

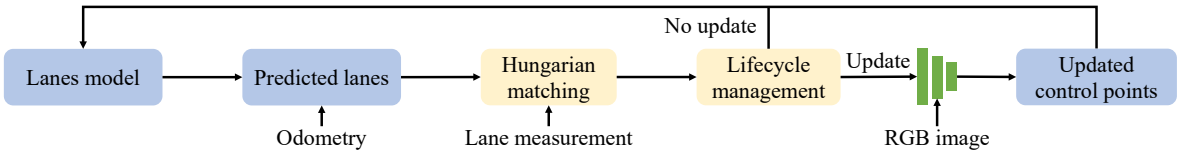

Figure 2: Overview. Our lane detection system utilizes a novel temporally recursive spline model to continuously represent lane trajectories over time. By leveraging vehicle odometry and the Hungarian algorithm, the model accurately associates new lane measurements with previously tracked lanes, preserving identity across frames. Intelligent spline extrapolation predicts new control points recursively, drawing on current and past frame data to ensure stable, identity-preserving multi-lane tracking, even in challenging conditions.

moves forward, our method extrapolates and refines these initial estimates, seamlessly integrating spatial and temporal information for reliable multi-lane tracking in real-world driving environments.

3.1 Lane Representation

To represent lane detection in our work, we use the classical Catmull-Rom Spline (Catmull and Rom, 1974). This is an interpolating spline that is defined by a sequence of k control points $\{C_i\}_{i=0}^{k-1}$. To describe a curve between C_i and C_{i+1}, we use four consecutive control points, which are denoted as $[C_{i-1}, C_i, C_{i+1}, C_{i+2}]$.

$$P(t) = \begin{bmatrix} 1 & t & t^2 & t^3 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1/2 & 0 & 1/2 & 0 \\ 1 & -5/2 & 2 & -1/2 \\ -1/2 & 3/2 & -3/2 & 1/2 \end{bmatrix} \begin{bmatrix} C_{i-1} \\ C_i \\ C_{i+1} \\ C_{i+2} \end{bmatrix}, \tag{1}$$

where t is bounded to $[0,1]$.

In our method, lane labels are initially annotated on the image plane, serving as input for both dataset preparation and initial lane measurement. However, for reliable lane tracking and continuous control point updates, it is essential to ensure that these control points are equidistant in world space. To achieve this, we project the lane points from the image plane onto a consistent bird's-eye view (BEV) plane, defined as $z = -(\text{camera's height})$, which we derive using the camera's extrinsic parameters.

The transformation uses the intrinsic matrix $\mathbf{K}$ of the camera to map each 2D lane measurement point into 3D world coordinates on the BEV plane:

$$p^{3D} = h\mathbf{K}^{-1}p^{\text{img}}, \tag{2}$$

where h is a scaling factor between the camera frame and the BEV plane. This results in a set of 3D lane measurement points, $\{p^{3D}\}_{j=0}^{n-1}$, which provide an accurate spatial representation of the lane.

Our method's innovation lies in fitting these 3D points to spline control points, $\{C_i\}_{i=0}^{k-1}$, by optimizing the spline curve equation defined in Equation 1. To ensure smooth and proportionate lane representation, we sample the parameter set $\{t_j\}_{j=0}^{n-1}$ across the range $[0,1]$ for each curve segment, with sampling intervals proportional to the distances between the projected lane points.

A distinctive feature of our approach is its capacity for curve extrapolation, enabling us to extend lane tracking by appending control points in real-time. This capability allows our model to dynamically predict and update new control points continuously within the driving scenario, ensuring persistent and adaptive lane tracking, even in extended or complex driving sequences.

3.2 Updating Spline Control Points

Our approach introduces a robust strategy for updating spline control points, enabling accurate and dynamic lane tracking in world space. As illustrated in Figure 3, we use vehicle odometry to predict the spline control points from the previous frame into the current frame within the world coordinate system. This innovation allows us to maintain stationary control points in world space, which then appear to shift backward in the vehicle frame as the vehicle moves forward, providing a stable reference for lane tracking.

When the first spline segment, defined by control points $[C_0, C_1, C_2, C_3]$, moves out of the image plane, we extend the lane by extrapolating the spline. We release the initial control point C_0—removing it from immediate influence on the lane shape in the image plane, yet maintaining it as part of the lane's continuous structure. Additionally, we discard the last control point C_{k-1} to avoid inaccuracies in lane representation, a common pitfall in prior methods that retain extraneous endpoints. After releasing and discarding these points, we reindex the remaining $k-2$ spline control points, $\{C_i\}_{i=0}^{k-3}$, ensuring smooth lane continuity.

Our method's novelty is further highlighted in the prediction of the last two spline control points, $\hat{C}_{k-2}$

and $\hat{C}_{k-1}$. Using the remaining control points from the previous frame, we generate Gaussian heat maps to represent these points in alignment with the current image, inspired by Carreira et al. (Carreira et al., 2016). This mapping encodes spatial positions with high fidelity, supporting precise and temporally consistent extrapolation.

In training our model, we employ two specialized loss functions: shape loss and order loss. The shape loss $\mathcal{L}_{\text{shape}}$ minimizes deviations from the ground-truth curve, focusing on the last two segments, which are directly affected by the predicted control points, following Feng et al. (Feng et al., 2022):

$$\mathcal{L}_{\text{shape}} = \frac{1}{N}\sum_{l}^{N} \left\| P(t_l) - \hat{P}(t_l) \right\|_2^2, \tag{3}$$

where t_l is sampled uniformly over $[0,1]$.

The order loss $\mathcal{L}_{\text{ord}}$ enforces correct sequential ordering of control points in the v-coordinate on the uv-image plane, penalizing any reversals that compromise lane continuity:

$$\begin{aligned}\mathcal{L}_{\text{ord}} = \max\left(0, \left(\hat{C}^v_{k-1} - \hat{C}^v_{k-2}\right)\right) \\ + \max\left(0, \left(\hat{C}^v_{k-2} - C^v_{k-3}\right)\right).\end{aligned} \tag{4}$$

Our overall loss function is defined as a weighted sum of these components:

$$\mathcal{L} = w_{\text{shape}}\mathcal{L}_{\text{shape}} + w_{\text{ord}}\mathcal{L}_{\text{ord}}, \tag{5}$$

where $w_{\text{shape}} = 1.0$ and $w_{\text{ord}} = 1.0$ balance shape accuracy and order fidelity.

Through this updating mechanism and loss framework, our model achieves consistent, accurate lane tracking that extends naturally with the driving scene. By combining stationary world-space control points, dynamic extrapolation, and a structured loss approach, our method enables continuous multi-lane tracking with high spatial fidelity and temporal stability—an advancement over traditional lane-tracking methods.

3.3 Lane Lifecycle Management

Our method introduces a new lane lifecycle management strategy to maintain accurate lane tracking and correct potential mapping errors over time. We begin by obtaining an initial lane measurement from a color frame using a 2D image-based lane detector, providing input as pixel coordinates.

To consistently match multiple tracked lanes with their respective measurements, we employ the Hungarian algorithm with Intersection over Union (IoU) as the distance metric (Bewley et al., 2016). In this process, each lane is represented as a 1.2-meter wide

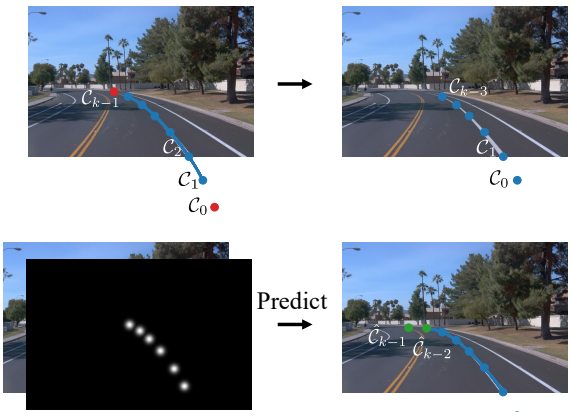

Figure 3: In our method, as the first spline segment exits the image plane, the first and last control points are removed to maintain lane continuity and accuracy. The remaining control points are encoded into a Gaussian heat map, aligned with the image plane, and input into the prediction network for extrapolation. Although the control points appear evenly spaced on the image plane, they are precisely spaced in 3D world space, ensuring consistent representation and stability in lane tracking.

curve on the BEV plane for IoU calculation, following the evaluation approach by Pan et al. (Pan et al., 2018). When a match is established, our lane model is dynamically updated using the Kalman filter, detailed in Section 3.4, enhancing temporal accuracy and maintaining spatial alignment.

For unmatched predictions and measurements, we implement an innovative lifecycle policy inspired by Wojke et al. (Wojke et al., 2017) but tailored to handle realistic lane dynamics. Instead of immediate deletion following a mismatch, as in prior multi-object tracking methods that assume constant object velocity, our approach reflects the natural variability in lane trajectories. A lane is only initialized after consistent measurement across a specified number of consecutive frames, n_{init} (Section 3.1). Conversely, tracking only terminates after n_{del} consecutive frames without a match, allowing our system to adapt to complex real-world lane behavior without premature deletions.

This lifecycle management framework introduces robust handling of lane persistence, ensuring stability and reliability in multi-lane tracking across diverse driving scenarios, a marked improvement over conventional methods with simplified assumptions on lane dynamics.

3.4 Kalman Filter

To achieve accurate lane tracking, our method integrates lane predictions with corresponding measurements through an advanced Kalman filter-based framework. The filter's state, defined as $(\mathbf{x}, \mathbf{\Sigma})$, con-

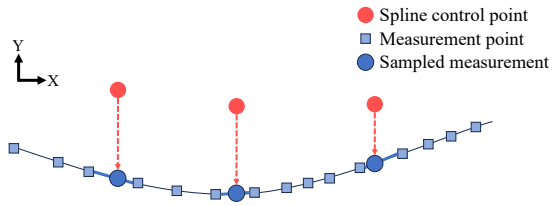

Figure 4: Sampling measurement points. When given as a set of points, the measurement state of the Kalman filter is sampled by linearly interpolating the two nearest points from the predicted spline control point.

sists of the mean $\mathbf{x}$ and covariance $\mathbf{\Sigma}$. Here, $\mathbf{x}$ represents a concatenation of spline control points in world coordinates:

$$\mathcal{C}_i = [\mathcal{C}_i^x, \mathcal{C}_i^y, \mathcal{C}_i^z]^T, \tag{6}$$

$$\mathbf{x} = [\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_{k-1}]^T, \mathbf{x} \in \mathbb{R}^{k\times 3}. \tag{7}$$

Our novel approach builds a state-space model that governs the evolution of lane state and measurement vectors over time:

$$\mathbf{x}_t = \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t), \tag{8}$$

$$\mathbf{z}_t = \mathbf{C}_t \mathbf{x}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t), \tag{9}$$

where $\mathbf{x}_t$ represents the lane state vector at time t, and $\mathbf{z}_t$ denotes the measured lane vector at the same time. The state transition matrix $\mathbf{A}_t$ and the measurement matrix $\mathbf{C}_t$ are computed at each time step, with $\mathbf{w}_t$ and $\mathbf{v}_t$ representing the Gaussian noise in prediction and measurement, respectively. We derive $\mathbf{A}_t$ using the rigid body transformation (RBT) matrix from vehicle odometry, allowing our model to adapt dynamically to the vehicle's motion.

A key innovation of our approach lies in the construction of $\mathbf{z}_t$. We employ a sampling strategy on a sequence of measurement points, interpolating between the two nearest points along the X-axis for each spline control point (Figure 4). This method enables accurate correspondence between predicted and measured control points on the BEV plane, further reinforced by setting $\mathbf{C}_t$ as an identity matrix $\mathbf{I}$ to maintain consistency between state and measurement spaces.

Using the state-space model, we recursively estimate the lane state $(\mathbf{x}_t, \mathbf{\Sigma}_t)$, incorporating the following Kalman filter equations:

$$\bar{\mathbf{x}}_t = \mathbf{A}_t \mathbf{x}_{t-1}, \tag{10}$$

$$\bar{\mathbf{\Sigma}}_t = \mathbf{A}_t \mathbf{\Sigma}_{t-1} \mathbf{A}_t^T + \mathbf{Q}_t, \tag{11}$$

$$\mathbf{K}_t = \bar{\mathbf{\Sigma}}_t \mathbf{C}_t^T \left(\mathbf{C}_t \bar{\mathbf{\Sigma}}_t \mathbf{C}_t^T + \mathbf{R}_t\right)^{-1}, \tag{12}$$

$$\mathbf{x}_t = \bar{\mathbf{x}}_t + \mathbf{K}_t \left(\mathbf{z}_t - \mathbf{C}_t \bar{\mathbf{x}}_t\right), \tag{13}$$

$$\mathbf{\Sigma}_t = \left(\mathbf{I} - \mathbf{K}_t \mathbf{C}_t\right) \bar{\mathbf{\Sigma}}_t. \tag{14}$$

Our approach specifically prioritizes the Y-axis accuracy of control points on the BEV plane, as it is more critical to tracking performance than the X-axis. Importantly, we exclude newly predicted control points outside the measurement range from Kalman filtering, maintaining model integrity. To ensure a control point contributes to valid lane detection, it must be processed by the Kalman filter.

4 Results

4.1 Dataset Annotation

We use the Waymo Open Dataset (Sun et al., 2020), which is a real-world dataset for autonomous driving. It includes color camera and LiDAR sensor data for each frame, along with object labels and vehicle odometry. Chen et al. (Chen et al., 2022) introduced OpenLane, a dataset that enriches the Waymo Open Dataset with lane annotations. OpenLane provides lane markings in both 2D image and 3D world coordinates, along with lane IDs. However, it only annotates the visible area of the lane, and the same line is sometimes annotated separately (Jin et al., 2023). This constraint restricts its applicability in video-based lane detection and tracking.

To tackle these challenges, Jin et al. introduce OpenLane-V, which enhances the 2D image plane lane annotations using a matrix completion method to ensure a more comprehensive and continuous lane representation (Jin et al., 2023). However, it is worth noting that OpenLane-V does not provide a ground truth label for lane IDs. To address this, we manually annotated a unique ID for each lane throughout the entire scene. We also filtered out some mislabeled lanes during the process.

In summary, our experiments and evaluations are conducted on a modified OpenLane-V dataset that includes vehicle odometry data from the Waymo Open Dataset. On average, each validation scene contains 149 frames, maintaining a frame rate of 10 FPS.

Meanwhile, MonoLaneMapping utilize image-based 3D lane detection as their input, training their detector, PersFormer (Chen et al., 2022), on the OpenLane (Zhijian Qiao and Shen, 2023). Since OpenLane-V doesn't support 3D lane annotations and our research is conducted on the 2D image plane, we projected the results from MonoLaneMapping onto the image plane for comparative analysis (Zhijian Qiao and Shen, 2023). The evaluation is done with the 2D lane annotation of OpenLane-V, which is a feasible approach due to the shared scenes between OpenLane and OpenLane-V from the Waymo Open Dataset.

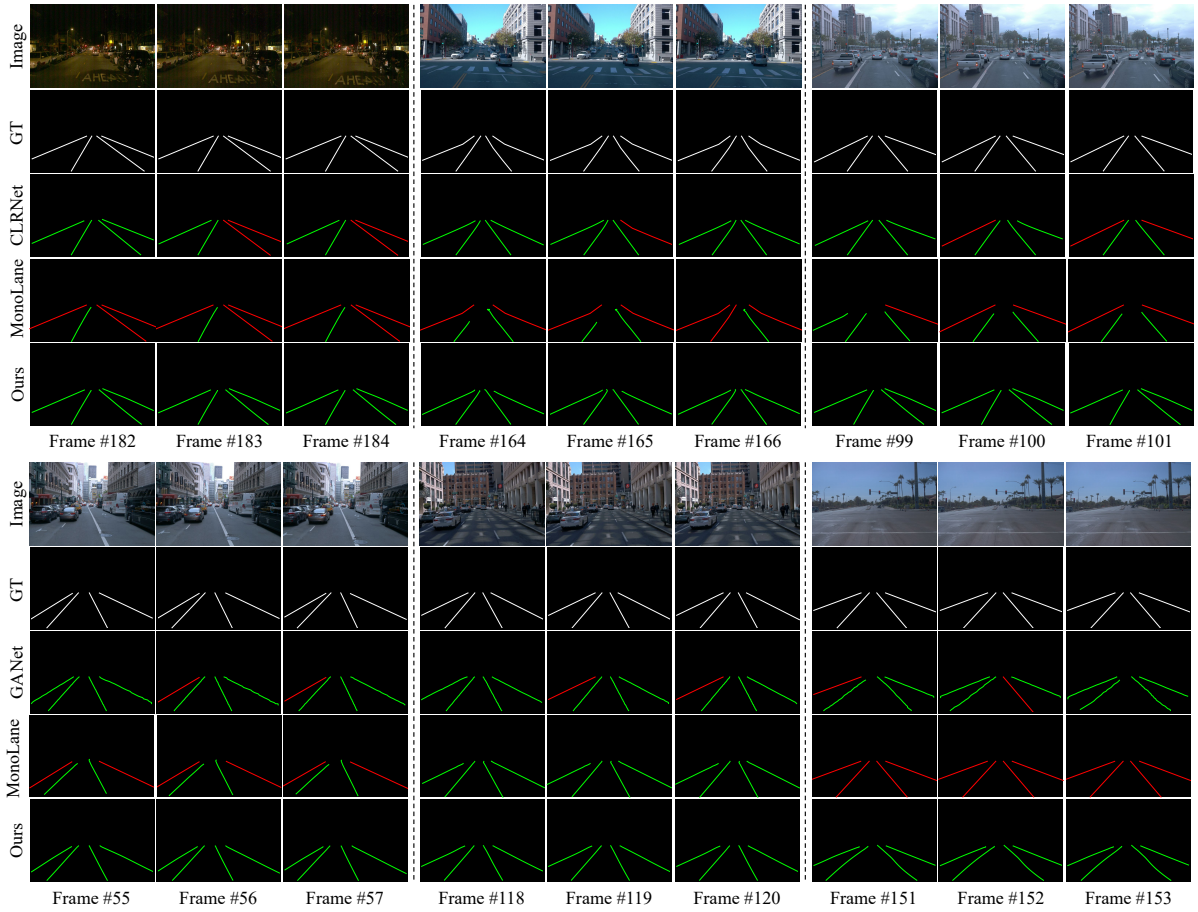

Figure 5: Qualitative comparison of our method with CLRNet (Zheng et al., 2022), GANet (Wang et al., 2022a), and MonoLaneMapping (Zhijian Qiao and Shen, 2023) on the OpenLane-V dataset across three consecutive frames. The top row shows input images, the second row shows ground truth lane markings, the third row shows baseline results, and the last row shows our method's results, built on the baseline. Green lines denote true positives, and red lines indicate false negatives. Quantitative results are in Table 1; refer to the supplemental video for temporal consistency.

4.2 Implementation Details

In the ablation study, we compare three state-of-the-art lane detectors, CLRNet (Zheng et al., 2022), GANet (Wang et al., 2022a), and CondLaneNet (Liu et al., 2021), to generate input lane measurements as initialization for our method. They are trained on the OpenLane-V dataset using the official code released by the authors. To train the network for updating the spline control points, we use DLA-34 (Yu et al., 2018) as the backbone network. The network takes both RGB images and grayscale heat maps as input. We introduce padding to accommodate spline control points that may fall outside the image plane. Specifically, we apply a padding of 480 pixels at the bottom and 960 pixels on both the left and right sides of the image. When a control point extends beyond this padded region, we reposition it to the border, aligning it towards the subsequent control point. It is important to note that this adjustment only applies to the first control point, $\mathcal{C}_0$, and has a negligible impact on the lane shape. In the lane tracking algorithm, we initialize the lane with $k = 8$ control points. We conduct lifecycle management of the lane using parameters set at $n_{\text{init}} = 2$ for initiation and $n_{\text{del}} = 3$ for deletion.

4.3 Evaluation Metrics

4.3.1 Lane detection metrics

To evaluate our results, we use the official evaluation metric of SCNN (Pan et al., 2018). A lane is considered as a 30-pixel width line on the image. If the IoU of the predicted lane with the ground truth is larger than a threshold, it is considered as a true positive (TP). The evaluation metric we use is the F1 measure, which is defined as $\text{F1} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$, where $\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}$ and $\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}$. We

Table 1: Comparison of lane detection on OpenLane-V, categorized by image-based lane detectors, video-based lane detectors, and lane tracking methods. The best results are emphasized in boldface.

Method		F1	Precision	Recall
Image-based	BezierLaneNet	76.37	76.86	75.88
	GANet	78.22	79.41	77.07
	CondLaneNet	78.32	83.01	74.12
	CLRNet	80.20	<u>83.15</u>	77.45
	ADNet	75.96	76.84	75.10
Video -based	MMA-Net	55.39	67.59	46.93
	RVLD	79.97	80.09	79.84
	OMR	<u>81.21</u>	81.90	**80.53**
Lane tracking	MonoLaneMapping	59.59	62.41	57.01
	Ours	**81.59**	**83.35**	<u>79.89</u>

Table 2: Comparison of lane tracking on OpenLane-V. The best results are emphasized in boldface.

Method	HOTA	MOTA	MOTP	IDF1
MonoLaneMapping	34.15	21.54	61.56	51.48
Ours	**52.81**	**63.33**	**70.24**	**73.10**

have set the threshold for IoU calculation to be 0.3.

4.3.2 Multi-lane tracking metrics

Currently, there is no standard metric for evaluating multi-lane tracking. Therefore, we utilize multiple object tracking (MOT) metrics, including HOTA (Luiten et al., 2021), MOTA, MOTP (Bernardin and Stiefelhagen, 2008), and IDF1 (Ristani et al., 2016) to evaluate lane tracking while considering the ID association. MOTA penalizes false positives, false negatives, and ID switches. This metric is highly dependent on the accuracy of detection since the number of ID switches is usually much lower than false positives and false negatives. MOTP measures the localization accuracy by averaging IoU across all correct matches. In contrast, IDF1 emphasizes the accuracy of ID association. Lastly, HOTA provides a balanced assessment across detection, ID association, and localization. For computational details of these metrics, please refer to the supplemental document. Following the standards of lane detection metrics, we consider matches with an IoU greater than 0.3 as true positives for MOTA, MOTP, and IDF1.

Table 3: Ablation study on lane detection performance with respect to the extrapolation method and the Kalman filter.

Extrapolation	Kalman filter	F1	Precision	Recall
Baseline		48.39	48.60	48.17
Ours		53.06	53.01	53.11
Baseline	✓	77.80	79.17	76.48
Ours	✓	81.59	83.35	79.89

Table 4: Ablation study of lane detection performance using different initialization. Numbers in parentheses show improvement over initial measurements on an independent frame.

Baseline	F1	Precision	Recall
GANet	79.60 (+1.38)	79.66 (+0.25)	79.54 (+2.47)
CondLaneNet	79.81 (+1.49)	83.12 (+0.11)	76.76 (+2.64)
CLRNet	81.59 (+1.39)	83.35 (+0.20)	79.89 (+2.44)

4.4 Quantitative Results

4.4.1 Lane Detection Results

Table 1 presents a comparative analysis of our lane detection accuracy against leading 2D image-based lane detectors such as BezierLaneNet (Feng et al., 2022), CLRNet (Zheng et al., 2022), GANet (Wang et al., 2022a), CondLaneNet (Liu et al., 2021), and ADNet (Xiao et al., 2023), video-based detectors such as MMA-Net (Zhang et al., 2021b), RVLD (Jin et al., 2023), and OMR (Jin and Kim, 2024), as well as the lane tracking method MonoLaneMapping (Zhijian Qiao and Shen, 2023) on the OpenLane-V dataset. Our approach consistently surpasses these methods across most metrics, underscoring its superior capacity for reliable and precise lane detection in complex driving scenarios.

4.4.2 Lane Tracking Results

In lane tracking metrics, shown in Table 2, our method significantly outperforms MonoLaneMapping (Zhijian Qiao and Shen, 2023) across all baselines, underscoring the robustness and effectiveness of our tracking approach. Our model excels in maintaining lane ID associations across frames, ensuring enhanced temporal consistency and reliability in multi-lane tracking. This capability is particularly valuable in real-world driving, where stable, identity-preserving lane tracking is essential for safety and navigation.

4.5 Qualitative Results

Figure 5 presents a visual comparison between the baseline 2D detectors (Zheng et al., 2022; Wang et al.,

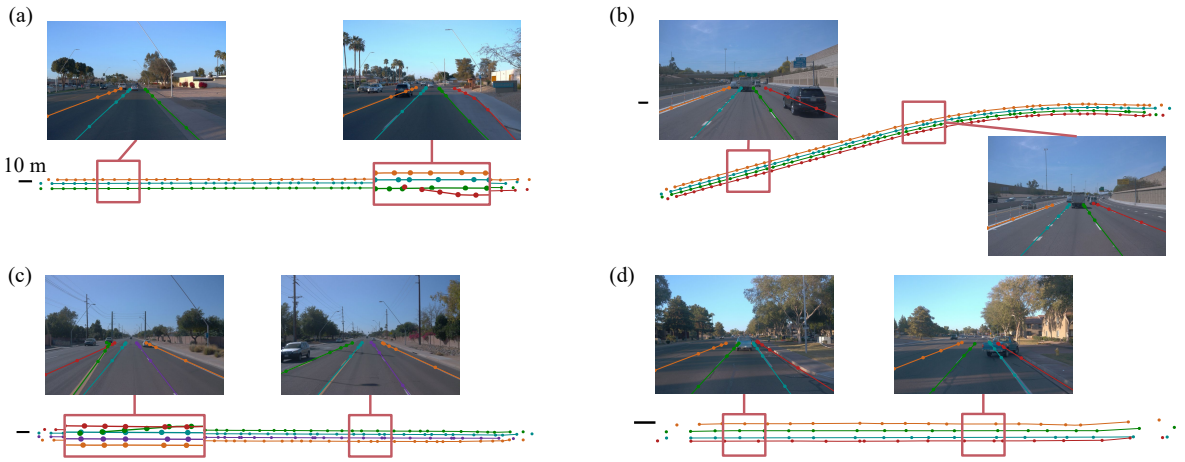

Figure 6: Results of our proposed method for lane mapping across different driving scenarios. The lane maps are shown in BEV space, with spline control points represented by points on the lines. We also provide corresponding driver perspective scenes for each scenario. To aid visualization, a scale is included on the left side of each subfigure, and more complex maps have been magnified. For more results, refer to the supplemental video.

2022a), MonoLaneMapping (Zhijian Qiao and Shen, 2023) and our enhanced method, which leverages the baseline's initial lane measurements. To demonstrate the effectiveness of our approach, we illustrate results across three consecutive frames for each scene. While the baseline results reveal issues such as flickering and lane omissions, especially in challenging scenarios with occlusions and varying lighting, our method presents substantial improvements. Our approach effectively mitigates these inconsistencies, providing smoother and more reliable lane continuity across frames.

In Figure 6, our approach demonstrates clear lane mapping using individual splines that capture each lane's trajectory along the entire driving path. This representation underscores our model's capability to accurately maintain the continuous shape and identity of lanes, even in complex situations involving lane splitting, merging, curves, and occlusions. Additionally, our robust extrapolation scheme ensures dependable lane tracking while preserving lane IDs, enabling high-fidelity lane association over time. These visual improvements validate our method's robustness in real-world conditions, where consistent and continuous lane tracking is critical.

4.6 Ablation studies

We conducted ablation studies to evaluate the contributions of our extrapolation method and the Kalman filter. For the baseline, we implemented a linear extrapolation method based on the last two control points. Specifically, the new control point was positioned such that the vector between the last two control points matched the vector between the new and the last control point. In addition, we experimented without the Kalman filter. In this setup, lane measurements from each frame were used solely for lifecycle management (initialization and termination) by the Hungarian matcher without updating the tracked lanes.

The quantitative results are presented in Table 3, showing lane detection metrics with OpenLane-V. When using the baseline extrapolation method, the F1 score decreased by 3.79 and 4.67 compared to our method, with and without the Kalman filter, respectively. These results confirm the effectiveness of our extrapolation method in accurately estimating the next control point. Moreover, the experiments with the Kalman filter demonstrated improved performance, underscoring the importance and effectiveness of the refinement process applied to the lane model by the Kalman filter.

Table 4 provides an ablation study that evaluates the influence of different initialization methods. We tested initializations with GANet (Wang et al., 2022a), CondLaneNet (Liu et al., 2021), and CLRNet (Zheng et al., 2022). Our model consistently improves upon each baseline across all metrics, with the highest performance observed when initialized with CLRNet, which we adopt as the preferred initialization. Notably, our method achieves substantial gains in recall, demonstrating its robustness in addressing inconsistencies like flickering caused by occlusions and single-frame limitations.

5 Conclusions

We have introduced a novel approach to lane tracking that employs temporally recursive spline model-

ing to achieve consistent, multi-lane tracking across time. Our method effectively addresses the limitations of single-image lane detectors by ensuring continuous, identity-preserving lane associations over extended sequences.

5.1 Limitation

While our method demonstrates substantial improvements, there are areas for future enhancement. A primary limitation is the absence of a graph-based global optimization, which may result in minor drift errors over time. To mitigate this, we incorporate a Kalman filter to enhance tracking stability, yet further research could strengthen the robustness of our approach through more sophisticated optimization techniques.

Additionally, our current method assumes a BEV plane for lane projections, which may not fully accommodate scenarios with sloping or uneven road surfaces. Developing adaptive models capable of handling diverse road geometries and conditions to improve tracking accuracy across varying terrains would be interesting future work.

Acknowledgements

Min H. Kim acknowledges the main support of the Hyundai Motor Company for this research project, in addition to the partial support by the Korea NRF grant (RS-2024-00357548) and the MSIT/IITP of Korea (RS-2022-00155620, RS-2024-00398830, 2022-0-00058, and 2017-0-00072).

REFERENCES

Bernardin, K. and Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10.

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B. (2016). Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE.

Borkar, A., Hayes, M., and Smith, M. T. (2009). Robust lane detection and tracking with ransac and kalman filter. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 3261–3264. IEEE.

Carreira, J., Agrawal, P., Fragkiadaki, K., and Malik, J. (2016). Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742.

Catmull, E. and Rom, R. (1974). A class of local interpolating splines. In *Computer aided geometric design*, pages 317–326. Elsevier.

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

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*, pages 17062–17070.

Hou, C., Hou, J., and Yu, C. (2016). An efficient lane markings detection and tracking method based on vanishing point constraints. In *2016 35th Chinese Control Conference (CCC)*, pages 6999–7004. IEEE.

Jin, D. and Kim, C.-S. (2024). Omr: Occlusion-aware memory-based refinement for video lane detection. In *ECCV*.

Jin, D., Kim, D., and Kim, C.-S. (2023). Recursive video lane detection. In *ICCV*.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.

Liu, L., Chen, X., Zhu, S., and Tan, P. (2021). Condlanenet: a top-to-down lane detection framework based on conditional convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3773–3782.

Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., and Leibe, B. (2021). Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129:548–578.

Mammeri, A., Boukerche, A., and Lu, G. (2014). Lane detection and tracking system based on the mser algorithm, hough transform and kalman filter. In *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pages 259–266.

Meuter, M., Muller-Schneiders, S., Mika, A., Hold, S., Nunn, C., and Kummert, A. (2009). A novel approach to lane detection and tracking. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–6. IEEE.

Nguyen, T.-P., Tran, V.-H., and Huang, C.-C. (2018). Lane detection and tracking based on fully convolutional networks and probabilistic graphical models. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1282–1287. IEEE.

Pan, X., Shi, J., Luo, P., Wang, X., and Tang, X. (2018). Spatial as deep: Spatial cnn for traffic scene understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ristani, E., Solera, F., Zou, R., Cucchiara, R., and Tomasi, C. (2016). Performance measures and a data set for multi-target, multi-camera tracking. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II*, pages 17–35. Springer.

Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., and Anguelov, D. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tabelini, L., Berriel, R., De Souza, A. F., Badue, C., and Oliveira-Santos, T. (2022). Lane marking detection and classification using spatial-temporal feature pooling. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Wang, J., Ma, Y., Huang, S., Hui, T., Wang, F., Qian, C., and Zhang, T. (2022a). A keypoint-based global association network for lane detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1392–1401.

Wang, M., Zhang, Y., Feng, W., Zhu, L., and Wang, S. (2022b). Video instance lane detection via deep temporal and geometry consistency constraints. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2324–2332.

Wojke, N., Bewley, A., and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE.

Xiao, L., Li, X., Yang, S., and Yang, W. (2023). Adnet: Lane shape prediction via anchor decomposition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6404–6413.

Yu, F., Wang, D., Shelhamer, E., and Darrell, T. (2018). Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412.

Zhang, J., Deng, T., Yan, F., and Liu, W. (2021a). Lane detection model based on spatio-temporal network with double convolutional gated recurrent units. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6666–6678.

Zhang, Y., Zhu, L., Feng, W., Fu, H., Wang, M., Li, Q., Li, C., and Wang, S. (2021b). Vil-100: A new dataset and a baseline model for video instance lane detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15681–15690.

Zhao, K., Meuter, M., Nunn, C., Müller, D., Müller-Schneiders, S., and Pauli, J. (2012). A novel multi-lane detection and tracking system. In *2012 IEEE Intelligent Vehicles Symposium*, pages 1084–1089. IEEE.

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

Zhijian Qiao, Zehuan Yu, H. Y. and Shen, S. (2023). Online monocular lane mapping using catmull-rom spline. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Zou, Q., Jiang, H., Dai, Q., Yue, Y., Chen, L., and Wang, Q. (2020). Robust lane detection from continuous driving scenes using deep neural networks. *IEEE Transactions on Vehicular Technology*, 69(1):41–54.

APPENDIX

A Lane Tracking Metrics

To evaluate the performance of lane-tracking systems, we utilize several metrics for multiple object tracking (MOT). These metrics include HOTA (Luiten et al., 2021), MOTA, MOTP (Bernardin and Stiefelhagen, 2008), and IDF1 (Ristani et al., 2016).

MOTA is a metric that measures the performance of a tracking system by penalizing three types of errors: false positives (FP), false negatives (FN), and identity switches (IDSW). This is calculated using the following formula:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FP}_t + \text{FN}_t + \text{IDSW}_t)}{\sum_t \text{GT}_t}, \qquad (15)$$

where GT refers to the total number of ground truth objects present in the sequence, while the sums run over all time frames in the sequence, MOTA is heavily influenced by detection accuracy since the number of identity switches is typically much less than the number of false positives and false negatives.

MOTP measures the accuracy of object localization by averaging intersection over union (IoU) across all correct matches. It is calculated using the following formula:

$$\text{MOTP} = \frac{1}{|\text{TP}|} \sum_{\text{TP}} \text{IoU}, \qquad (16)$$

where TP is the total number of true positives, and the sum runs over all true positives.

IDF1 is a metric that measures the accuracy of identity (ID) association throughout an entire sequence based on a trajectory. In simple terms, it calculates bipartite matching between the set of ground truth and predicted trajectories. IDF1 determines the number of detections that belong to matched trajectories, as well as the number of false negatives and false positives from unmatched ground truth and predicted trajectories, respectively. The calculation of IDF1 is based on these values as follows:

$$\text{IDF1} = \frac{\text{IDTP}}{\text{IDTP} + 0.5 \cdot \text{IDFN} + 0.5 \cdot \text{IDFP}}. \qquad (17)$$

HOTA is a modern metric that provides a well-rounded evaluation of detection, ID association, and localization. To calculate the detection accuracy

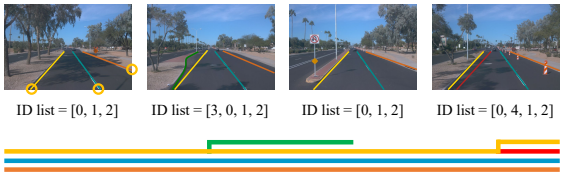

Figure 7: Lane ID Labeling. The figure shows positional relationships between lanes, not their actual shapes. Images are included for frames where lane IDs change.

(DetA), HOTA uses the IoU calculation to find the ratio of true positives (TPs) to the sum of TPs, false negatives (FNs), and false positives (FPs). Upon the matched detections, HOTA assesses the consistency between the ground truth and predicted tracks across the entire sequence based on their IDs. True positive associations (TPAs) are the correctly matched detections between two tracks. False negative associations (FNAs) and false positive associations (FPAs) count the unmatched detections in the ground-truth track and predicted track, respectively. The association accuracy (AssA) is the ratio of TPAs to the sum of TPAs, FNAs, and FPAs. Finally, HOTA calculates the geometric mean of two accuracies as follows:

$$\text{DetA} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|}, \tag{18}$$

$$\text{AssA} = \frac{1}{|\text{TP}|} \sum_{d \in \text{TP}} \frac{|\text{TPA}(d)|}{|\text{TPA}(d)| + |\text{FNA}(d)| + |\text{FPA}(d)|}, \tag{19}$$

$$\text{HOTA}_\tau = \sqrt{\text{DetA}_\tau \cdot \text{AssA}_\tau}, \tag{20}$$

$$\text{HOTA} = \frac{1}{19} \sum_\tau \text{HOTA}_\tau, \tag{21}$$

where τ is a threshold of true positive in IoU calculation ranging from 0.05 to 0.95 at 0.05 intervals.

B Labeling Lane IDs on OpenLane-V Dataset

Lane ID annotations are not available in the OpenLane-V dataset (Jin et al., 2023). Therefore, we take the initiative to manually label the lane IDs for lane tracking evaluation mentioned in Section A. To accomplish this task, we sort the lanes based on the u-coordinate of the bottom point of each lane within the uv image coordinate system. This methodology proves to be advantageous as it ensures that the order of these u-coordinates remains largely consistent across consecutive frames. The lanes are assigned IDs in ascending order of u-coordinates, resulting in a compiled list of IDs.

Critical moments occur when lanes either emerge or conclude on the road. These moments are typically identified by shifts in the total number of lanes. We adjust the ID list to reflect these changes while maintaining the order based on the u value in such frames. Refer to Figure 7 for a visual illustration of this process and the resulting lane ID annotations.