Advanced Lane Detection and Tracking Using Robust Thresholding and Frame Coherence

Saumya Srivastava and Rina Maiti

***Abstract*— Despite the development of several vision-based methods over past decades, lane detection is still a challenging issue in computer vision community. Traditional research on lane detection is mainly based on inter-frame similarity. Conversely, our methodology challenges this assumption, allowing the algorithm to adapt in diverse dynamic driving scenarios. In this paper, we proposed an advanced lane detection and tracking method using robust thresholding and frame coherence. A novel two stage feature extraction methodology for lane detection is introduced. In the initial stage, adaptive thresholding is employed based on the local statistical and geometrical constraints of lane markings at the sub-image level. Subsequently, artifacts are addressed and removed using a vertical aggregation technique combined with the Hough transform at the regional level. In this lane detection framework, tracking is integrated to establish a dynamic area of interest for lane search in the subsequent frame. For robust tracking, Kalman filter parameters are adjusted based on measurement reliability. To adapt to the non-linear variations inherent in real-world driving scenarios, outcomes of inter-frame similarity analysis are compared with periodic full-frame detection and a moving average window estimator is employed within the verification protocol. The proposed method is evaluated using Caltech dataset and our dataset, which was recorded at Indian highway and urban roads. Experiment results shows that the proposed method is robust to varying illumination conditions, dense shadows, lane change scenarios, abrupt change in lane width, and occluded lanes with a whooping average detection rate of 98.37%.**

***Index Terms*— Lane detection, Hough transform, Kalman filter, inter-frame similarity, Indian roads, advanced driver assistance system.**

I. Introduction

LANE detection plays a crucial role in many advanced driver assistance modules, such as Lane Departure Warning Systems (LDWS), Lane Keeping Assistance Systems (LKAS), and Lane Changing Assistance Systems (LCAS). Lane detection comprises of three elements in its pipeline: feature extraction, model fitting and tracking [1]. To extract lane features, many studies explored the property of white and yellow colors associated with lane marks [2], [3]. However, varying illumination and cast shadow cause hindrance in collating these color cues. To circumvent these conditions, studies have taken color-space transformations into consideration [4], [5], [6]. Another well-known method to extract lane features is based on the image gradient or edges. Some studies [7], [8] have explored gradient properties using oriented Gaussian kernels, while others have employed detectors like Canny [9], [10], Sobel [11], [12], and improved versions of Canny [13] to extract lane edges. These detectors offer low computational cost and provide valuable structural information about lane marks but tend to produce fragmented edges, especially in noisy images.

Numerous sophisticated techniques such as clustering [14], [15], scan-line test [4], [16], edge distribution function [17], line segment detector (LSD) [18], conditional random function [19] and steerable filters [20], [21] have been utilized in the literature to extract the lane features. Moreover, different local or global thresholding methods such as histogram-based segmentation [3], [14], adaptive thresholding [22], [23], have been exploited to segment lane marks. However, each of these methods possesses inherent limitations, particularly when addressing noise that exhibits similar orientation and intensity characteristics as lane marks. To improve the fitting of features with varying curvature, models such as parabolic curve, hyperbolic, and spline models [17], [24], [25] are commonly employed. However, higher-order models increase computational costs.

In order to deal with scenarios where lane markings are difficult to detect, tracking is often integrated into lane detection pipeline. It ensures the continuous monitoring of lane positions and characteristics over time, even in adverse conditions like low visibility, varying illumination, partial occlusions, etc [26], [27]. McCall and Trivedi [21] have employed the Kalman filter to track various parameters, including steering angle, vehicle velocity, lane position and its angle estimations, while others have utilized it to track parameters derived from the Hough transform, such as ρ and θ [28], [29]. However, the Kalman filter achieves optimal performance only when the noise is Gaussian-distributed. To strengthen lane detection performance in complex environments, numerous studies rely on the notion of inter-frame similarity [30], [31], [32]. Under the assumption that lane parameters remain relatively consistent between consecutive frames, researchers have developed various strategies to define a fixed region of interest (ROI) around the detected lane markings, including Λ-ROI [16], lane boundary region of interest (LBROI) [17], and adaptive ROI [33], AROI [34]. These ROIs serve as a predictive space where lane markings are expected to be located in the subsequent frame. However, real-world driving scenarios often involve non-linear events including lane change and alteration in lane attributes such as width, slope, and position of lane

Received 8 May 2024; revised 20 December 2024 and 11 February 2025; accepted 13 February 2025. Date of publication 10 March 2025; date of current version 2 June 2025. The Associate Editor for this article was L. M. Bergasa. *(Corresponding author: Saumya Srivastava.)*

The authors are with the Department of Design and Manufacturing, Indian Institute of Science, Bengaluru 560012, India (e-mail: ssaumya@iisc.ac.in; rmaiti@iisc.ac.in).

Digital Object Identifier 10.1109/TITS.2025.3545554

marks which challenge the validity of inter-frame similarity assumption and demand more robust and adaptive methods for accurate lane detection and tracking.

To overcome the challenges posed by dynamic road conditions, our methodology presents two key modules: (a) preliminary lane detection and (b) frame coherence. The lane detection module uses a two-stage feature extraction process. In the first stage, adaptive thresholds are determined at the sub-image level and validated using *contrast* and *peak offset* values. To remove artifacts caused by varying illumination conditions, such as fluctuating light levels and dense shadows, a vertical aggregation technique is combined with the Hough transform at the regional level. However, accurate lane detection remains challenging under varied road conditions, including instances of lane transitions, sudden alterations in lane width and obscured lane markings. To overcome these hurdles, the frame coherence module is introduced, incorporating Kalman tracking with parameters adjusted based on *measurement reliability*, followed by establishing a DAOI for lane search in successive frames. To handle non-linear variations, inter-frame similarity analysis is cross-checked with periodic full-frame detection using a moving average estimator. Upon the confirmation of new lane marks, the DAOI is re-established, and the Kalman filter is re-initialized. These measures significantly enhance the robustness and adaptability of the framework in diverse driving conditions.

This work presents four primary contributions: (1) Lane features are iteratively pruned at sub-image and regional levels to effectively extract details under varying lighting conditions and eliminates non-lane information from the image. (2) *Measurement reliability*, derived from comparing detection results across frames, is applied to adjust the noise covariance matrix (R) of the Kalman filter. This enables the filter to adapt its sensitivity toward measurement, thereby leading to optimal tracking result. (3) A verification protocol is implemented that enables adaptation to non-linear variations. (4) Lane changes are efficiently detected by analyzing the relative positions of lane markings to the mid-demarcation line across consecutive frames.

This paper is organized as follows: Section II briefly reviews related works on lane detection system. Section III describes the proposed lane detection framework. Section IV shows the details of the experimental results and discussion. Finally, section V concludes the paper with the inclusion of future work.

II. Related Work

Numerous studies have been presented on lane detection through the application of inverse perspective mapping (IPM) [35], [36]. This geometrical transformation eliminates the perspective effects in the road image by homogeneously remapping picture elements to a new reference frame, thereby providing a top-down view of the road scene in front of the camera. Aly [8] have applied 2D Gaussian kernels on IPM image, followed by thresholding, Hough transform and Random Sample Consensus (RANSAC) line fitting, and spline fitting for lane estimation. Similarly, Xu et al. [25] have employed a 1D filter with global thresholding over IPM, followed by connected component analysis to form ROIs and fitted a hyperbolic model. However, both of these methods [8], [25] are prone to produce high false positive in presence of lane-like noise and varying illumination conditions. Moreover, these approaches work well only under the flat road assumption and the transformation is vulnerable to vehicle vibrations and obstruction by leading vehicles, causing artifacts in the IPM image. To achieve a bird's eye view independently of camera calibration, Kang et al. [37] have estimated a dynamic homography matrix from detected lane markings. Though the method can adapt to changing pitch and yaw angles of camera, it fails in handling roads with varying curvature. On the other hand, Ying and Li [38] utilized road boundaries to generate an IPM, followed by template filtering and binarization with a fixed threshold. However, relying on a single threshold for binarization is insufficient to handle uneven lighting conditions and road texture variations. To overcome this disadvantage, Yoo et al. [39] have adaptively selected threshold for the Canny edge detector is used to address varying illumination conditions. Similarly, Son et al. [14] applied thresholds on the Y-channel and Cb channel to extract white and yellow lines, respectively. However, traditional global thresholding approaches [4], [25], whether employing single or multiple thresholds, prove inadequate for accurate lane marking identification.

To address this challenge, several authors have introduced the concept of sub-image processing to extract the feature on local level [40], [41]. Wu et al. [33] have dynamically calculated thresholds for small blocks using the Otsu thresholding [42] and remove edges with abnormal orientations through gradient constraint. While Otsu's method achieves satisfactory results with bimodal histogram, it is sensitive to noise and uneven illumination. Similarly, Yuan et al. [40] have considered statistical features of pixel values within sub-blocks to determine binarization thresholds and applied denoising operation with area and orientation constraints. However, in extracting lane edges embedded with noise, often valuable information (e.g., small lane edges of broken lane marks) is inadvertently erased, thereby altering the structural information of the lane and affecting subsequent processes. With the assumption that lane are parallel, Yoo et al. [30] have integrated an orientation constraint with the line segment detector (LSD) to extract the line segments from the image. Later, the strength of each line segment is used to calculate vanishing point by using probabilistic voting procedure. To minimize computational costs, the voting function is computed using a pre-defined lookup table. However, computed values in the lookup table are often constructed using assumptions or data acquired under specific conditions, which may not align with the real-time road variations. Further, to make system robust against outliers, they have defined score function by using geometric relationships between the line segments and the estimated vanishing point. However, methods struggles in the presence of lane-like artifacts.

To further fine-tune lane detection, inter-frame similarity techniques have been integrated with conventional methods to reduce transient artifacts from shadows, reflections, and lane-like noise by analyzing lane features over multiple frames

and averaging out noise and disturbances for more stable detection.Yoo et al. [30] have utilized inter-frame similarity to ensure location consistency of ego lanes and vanishing point. For a given frame, detection validation of the vanishing point and lane angle managed through a queue system. In [43], a spatiotemporal image is formed by integrating pixel intensities along a scanline across time. The method assumes temporal consistency in lane width, utilizing it for lane detection when one lane is present and the other is absent. However, detection becomes challenging when both the lanes are occluded. With the assumption that lane markings persist at the similar location across consecutive frames, Yuan et al. [40] have introduced AROI that dynamically adjust its size and position based on lane curvature and vehicle speed. Their method emphasize on re-identifying the starting point of the lane at regular interval to handle lane change scenarios. However, false detections after re-initialization can propagate unreliable AROIs until the next re-identification process. In [23], tracking of parameters (ρ and θ) is paused if lane markers remain undetected on the road for a predefined number of frames, until new detection occurs. However, no procedures are established to verify the detection results in subsequent frames.

Inter-frame similarity techniques often combine motion estimation algorithms, such as optical flow [44], particle filter [12] or Kalman filter [29], [33] along with established ROI to enhance lane detection performance. Wu et al. [33] have combined tracking with adaptive lane-mark ROIs and processed only sub-images with edge-links from the previous frame to reduce processing time. Their approach outperforms the traditional detection methods by ensuring consistent lane positions and adapting to changing conditions without being sensitive to short-term disturbances. To address temporary absence of lane markers, Lee and Moon [16] have employed inter-frame clustering to aggregate statistical lane parameters of candidate lane marks over a predefined number of frames. Lane positions were tracked using lane width and intersection points, with the Kalman filter predicting trajectories when markings were undetected within the Λ-ROI. The distorted trapezoidal Λ-ROI with variable boundaries ensures detection despite lane width changes or vehicle shifts beyond the typical range. However, abrupt changes in road dynamics may cause deviations from the assumed state model, leading to failures in both Kalman tracking and the established Λ-ROI.

III. Proposed Method

In the proposed work, we deal with two modules: 1) Preliminary lane detection, which involves feature extraction and curve fitting. 2) Frame coherence, where the location consistency of detected lane marks are validated. Frame coherence encompasses a tracking system, the establishment of a DAOI, and a verification protocol. It also highlight the adaptation to non-linear variation. Process starts by extracting the individual frames from the video and transforming the frames to IPM to detect lane markers. Bonnet area is removed before IPM transformation. To generate the transform, it is imperative to have the intrinsic (camera optics, skew factor) and extrinsic (position, orientation) parameters of the camera sensor. These matrices establish the relationship between the camera coordinate and the image coordinate.

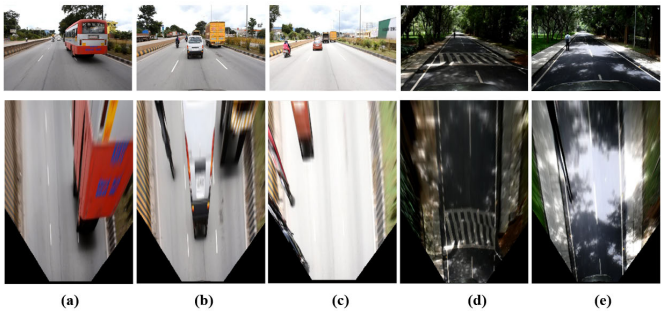

Fig. 1. Typical examples of challenges associated with lane mark feature extraction. First row displays original images for the reference and second row represent their respective IPM transformation. (a)∼(c) show the presence of leading vehicles occluding lane marks and create artifacts in IPM. Additionally, edges of various vehicle parts exhibiting similar orientation to that of lane mark (c) shows image saturation due to intense ambient light (d)∼(e) indicate severe shadow in the road scene.

Despite restoring lane marking parallelism, challenges still persist in the feature extraction process as shown in Fig. 1. These challenges include: 1) Presence of leading vehicles can occlude lane marks and create artifacts in IPM images (Fig. 1(a), (b), (c)). 2) Risk of misclassifying objects as lane mark candidates due to similarities in intensity and alignment (Fig. 1(a), (b)) 3) The varying illumination across the road scene, where for instance, intense glare induced by background light or shadow cast by trees can alter the dark-bright-dark pattern of lane marks against the road surface as shown in Fig. 1(c) and Fig. 1(d), (e) respectively.

A. Preliminary Lane Detection

To address these challenges, our methodology begins by classifying images as *(*dark) or *(*bright) based on histogram skew toward higher or lower intensity values. Images (a), (b), and (c) in Fig. 1 are classified as bright, while images (d) and (e) are classified as dark. Each IPM image is partitioned into left half section (LHS) and right half section (RHS) using the central demarcation line at x_{mid} as shown in orange line in Fig. 2(a). To extract white lane markings, RGB values are first transformed to HSV and then processed through a two-stage feature extraction. In the first stage, adaptive threshold values are applied at the sub-image level based on statistical and geometrical constraint of lane marking, while in the second stage,artifact removal operations are performed at the regional level.

1) Adaptive Thresholding: Longitudinal lane marks often present in continuous or dashed patterns, offering high visibility and contrast against the road surface. To extract the lane marks, adaptive thresholding is applied by dividing the LHS and RHS into sub-images of size $r \times c$ (60×30). A histogram is constructed for each sub-image, spanning pixel values from $[0, (L-1)]$. Assuming lane markings are brighter than the road surface, typically occupying the right extrema of the histogram, and maintaining a consistent pixel width of $w_m = 7$ pixels, the cumulative distribution function (CDF) of the sub-image is calculated such that the cumulation satisfies

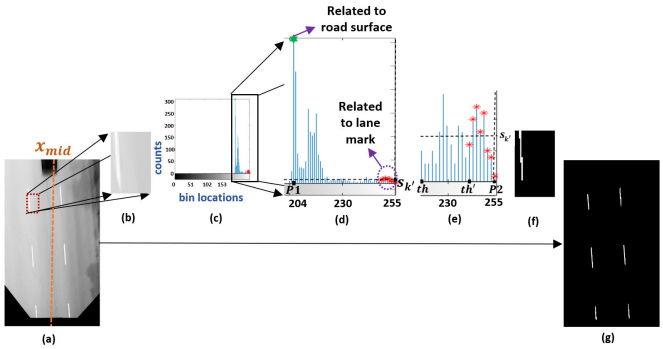

Fig. 2. Adaptive Thresholding is performed over an image (a) A sub-image is chosen from the LHS of IPM image. The central demarcation line at x_{mid} is depicted in orange (b) Zoomed-in representation of the sub-image (c) Generated histogram of the sub-image (d) Zoomed-in image of the right side of histogram (e) th' represents the pixel index at which the cumulation stops as per equation1, th is candidate threshold index and P2 denotes the highest pixel index in histogram. Peaks of accumulated pixel index from th' to P2 are highlighted in red asterisk. $s_{k'}$ represents the average cardinality for pixel intensities ranging from th' to P2. (f) Sub-image obtained after performing Adaptive Thresholding (g) Final image obtained by applying Adaptive Thresholding to the entire image.

the following constraint:

$$\text{cdf}(X) = \left\{ \sum_{k=L-1}^{0} P(X) \mid \sum_{k=L-1}^{0} P(X) \leq \frac{b \times w_m \times r}{L-1} \right\} \tag{1}$$

where P(X) is the probability density function at each pixel value defined as $P(X_k) = m_k/M$, m_k is the cardinality at gray level k, $M = r \times c$ is a total number of pixels in the sub-image, and L is the total number of gray levels in the image. The scaling factor b, set to 1/7, ensures that thresholding remains sensitive to detecting even small segments of lane markings within the sub-image. The pixel index at which the cumulation stops in equation 1, is identified (Fig. 2e) as candidate threshold, th'. Based on the th', *contrast offset* and *peak offset* are defined as

$$Contrast\ offset = \max\left(k'\right) - \arg\max_{k}\left(m_k\right) \tag{2}$$

$$Peak\ offset = \max\left(m_k\right) - s_{k'} \tag{3}$$

$$s_{k'} = \frac{1}{th' - (L-1)} \sum_{k=th'}^{L-1} m_k \tag{4}$$

where $k \in [1, L-1]$, $k' \in \left[th', L-1\right]$ and $s_{k'}$ is the average cardinality for pixel intensity ranging from $[th', L-1]$. For sample image shown in Fig. 2 (d), (e), $\arg\max_k(m_k)$ and $max(k')$ are represented by points P1 and P2 respectively. For th' to be valid both the contrast offset and peak offset exceed values of 30 and 10 respectively. Contrast offset greater than 30 signifies a high contrast between the foreground (lane marking) and the background (road surface). Similarly, peak offset threshold of 10 validates the presence of longitudinal lane marking with area $r \times w_m$ embedded within a sub-image of area $r \times c$. Once th' is validated, a margin of 20 is subtracted to establish the threshold th as $th = th' - 20$, ensuring greater adaptability to varying conditions. Pixels with intensity values above th are retained, while the sub-image is set to zero and

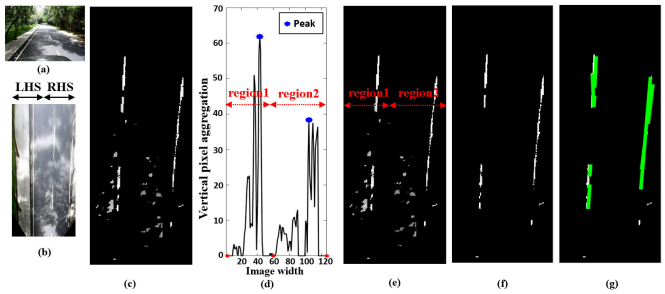

Fig. 3. Artifact removal. (a) Sample image consists of shadow on the roadway. This image is a typical case of classified *bright* image with high ambient light (b) IPM image (c) close-up of the RHS obtained after applying adaptive thresholding reveals erratic patches along with lane marks (d) Vertical pixel aggregation across the image width. Significant peaks (highlighted in blue) are identified to define regions across the image. (e) Two regions are identified, referred to as region1 and region2 (f) Image obtained after applying threshold over each region (g) Hough lines drawn on each region (highlighted in green).

classified as background if the histogram is flat or lacks a distinct peak. The outcome of local thresholding over entire image is illustrated in Fig. 2(g).

However, in complex environment with varying illumination, the intensity distributions of the road surface and lane markings can overlap and compromise the separability of lane markings from the background, as evident in Fig. 3(c), obtained after applying *Adaptive thresholding*. Therefore, results obtained from previous method are insufficient in extracting lane features and further requires an additional step of *Artifact Removal*.

2) Artifact Removal: To remove irregular intensity artifacts alongside the longitudinal white lane markings, RHS (Fig. 3(c)) is divided into potential lane mark containing regions using a histogram detailing the vertical aggregation of pixels generated across the image [25]. Under the assumption that the lane width maintains a minimum separation of at least 40 pixels between the lane marks, significant peaks are identified to define regions as shown in Fig. 3 (d). As shown in Fig. 3(e), region1 and region2 around peaks are designated and further subjected to thresholding to remove spurious noises based on following criteria. In case of *bright* images, lane markings are having higher pixel values, and the highest value is used to determine the threshold as

$$region\ threshold = \max_{i,j \in region}\left(I(i,j)\right) - c_1 \tag{5}$$

On the contrary in case of *dark* image (Fig. 1(d)), captured with reduced ambient light, like cloudy, dawn, intense shadow or night scenes, index with the maximum intensity value is chosen as threshold in dark image as

$$region\ threshold = \arg\max_{i,j \in region}\left(I(i,j)\right) - c_2 \tag{6}$$

where $I(i,j)$ represents the intensity value at pixel (i,j) in the region. Margins $c_1 = 40$ and $c_2 = 5$ are utilized to enhance the adaptability of the proposed method across diverse lighting conditions. In both cases, pixels above this threshold value are set to 1, while others are set to 0 (Fig. 3(f)), and then Hough lines are applied (Fig. 3(g)) to remove noise and choose aligned non-zero pixel. Vertical structuring element of size 10×1 is applied over object boundaries to highlight the lane

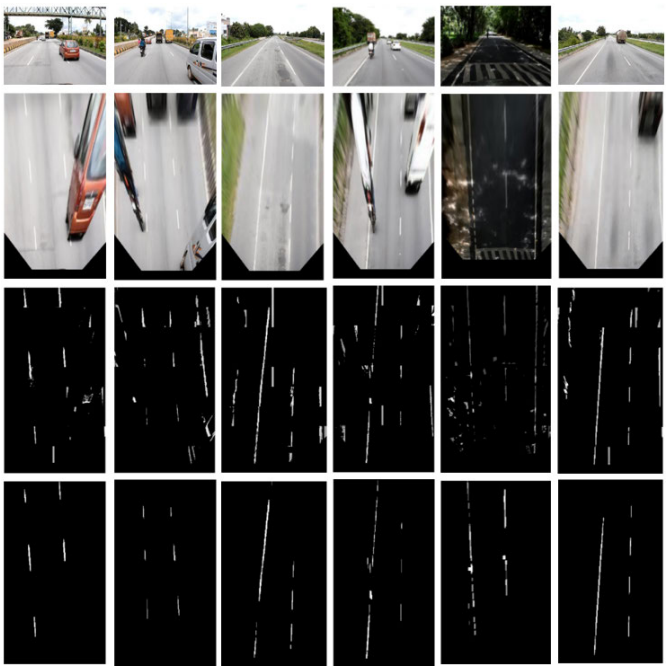

Fig. 4. The feature extraction process on our sample dataset. Row 1 depicts the original image, Row 2 demonstrates the Inverse Perspective Mapping (IPM), Row 3 showcases the result post-application of adaptive thresholding on a sub-image level, and Row 4 exhibits the final output after artifact removal.

features. A similar process is also applied on the LHS of the image. The feature extraction process on the sample dataset is illustrated in Fig. 4.

In multi-lane scenarios where both sections (LHS, RHS) of image have more than single lane marks, ego lane is defined by lane marks corresponding to farthest line and nearest line in the LHS and RHS of the image respectively. Depending on the deflection observed in the extracted features, either a straight line or a second-order polynomial function with RANSAC is employed to plot curves over each region. However, noise resembling lane markings can generate peaks, leading to the formation of pseudo lines that may be incorrectly identified as lane markings. To address this, the topmost and bottommost x-coordinates, along with the model parameters, are further validated in the *Frame Coherence* module.

B. Frame Coherence

Temporal and spatial continuity is a widely employed concept in various research endeavors [43], [45]. We make the proposed system robust by utilizing the frame to frame consistency in two aspects: 1) for tracking lane marks and establishing DAOI, and 2) for adapting to non-linear variation in road dynamics. Table I summarizes the key notations used throughout the mathematical formulation.

1) Tracking: Lane markings on paved roads are essential for traffic management and driver guidance, offering clear visual cues for drivers to maintain their position within their designated lanes. However, their visibility may be compromised in certain frames due to various external factors including glare from oncoming headlights or sunlight, cover-up by the vehicle ahead, debris or dirt accumulation on the road surface, faded lane markings etc. To make the algorithm robust in such challenging environments, two Kalman filters are used to track the detected lane, one for left lane marking and the other for right lane marking.

TABLE I

PRIMARY NOTATIONS USED IN THE MATHEMATICAL FORMULATION AND THEIR DESCRIPTION

Notation	Description
${}^{1}_{r}x_t$	Detected x-coordinates of the top-most point for right lane marking at time t
${}^{C}_{r}x_t$	Detected x-coordinates of the bottom-most point for right lane markings at t
${}^{1}_{r}\hat{x}^{+}_{t}$	Kalman estimated x-coordinates of the top-most point for right lane marking at time t
${}^{C}_{r}\hat{x}^{+}_{t}$	Kalman estimated x-coordinates of the bottom-most point for right lane marking at time t
$\hat{u}^{-}_{t}$	Priori estimate state vector at time t
$\hat{u}^{+}_{t}$	Posteriori estimate state vector at time t
P^{-}_{t}	Priori estimate error covariance at time t
P^{+}_{t}	Posteriori estimate error covariance at time t
${}^{1}_{r}\tilde{x}_t$	Average value for x-coordinate of the bottom-most for right lane marking at time $t-m$ to t
${}^{C}_{r}\tilde{x}_t$	Average value calculated for x-coordinate of the top-most for right lane marking at time $t-m$ to t

The efficacy of the Kalman filter is intricately linked to Q (noise covariance matrix) and R (process covariance matrix), which play crucial roles in the filter's estimation process. In many studies, noise in both the state and measurement is assumed to be white and Gaussian. For instance, Wu et al. [33] configured Q and R as diagonal matrices with values 10^{-5} and 0.1, respectively, whereas Liang et al. [28] set Q to 0.05 (diagonal) and R as an identity matrix. However, due to the variability in noise levels and dynamic nature of driving environments, determining fixed values for Q and R are challenging lane detection. Incorrect values of Q may lead to over-fitting or under-fitting of observed data, potentially causing divergence of the Kalman filter. Similarly, incorrect values of R can make the filter overly sensitive or insensitive to measurement errors, resulting in sub-optimal filtering, reduced accuracy in state estimation, and degraded tracking performance [46].

In the proposed work, two distinct values of R are used to update the state, based on the reliability of lane mark measurements. The *measurement reliability* is evaluated through the deviation factor (DF), which compares detection outcome obtained from *Preliminary Lane Detection* module at time t with the Kalman estimate at $t-1$

$$DF_r = \left| \left({}^{1}_{r}x_t,\ {}^{C}_{r}x_t \right) - \left({}^{1}_{r}\hat{x}^{+}_{t-1},\ {}^{C}_{r}\hat{x}^{+}_{t-1} \right) \right| \tag{7}$$

where ${}^{1}_{r}x_t$, ${}^{C}_{r}x_t$ are detected top-most and bottom-most x-coordinates of the right lane markings at time t, and ${}^{1}_{r}\hat{x}^{+}_{t-1}$ *and* ${}^{C}_{r}\hat{x}^{+}_{t-1}$ are their corresponding Kalman estimates from time $t-1$. For detected lane marks to be reliable, the DF must fall below the Tolerable Pixel Shift (TPS) of 15. TPS refers to the permissible deviation in pixel positions between consecutive frames within a video sequence. In our approach, when the DF exceeds the TPS, we employ high diagonal values of 10^2 for R which prompts the Kalman filter to rely on the predicted state in its estimation process. Conversely, if DF is within TPS then R is set to low diagonal values of

0.01 and Kalman filter incorporates measurement into its state estimation process. Throughout our study, Q remains constant, with diagonal values fixed at 0.001 to ensure consistent tracking. Kalman tracking consists of two stages, first stage predicts the present state of the target as

$$\hat{u}_t^- = A \times \hat{u}_{t-1}^+ + w \tag{8}$$

where $\hat{u}_t$ represents the state vector at time t, w is process noise, and A is a process transition matrix that establishes the relation between the previous and current state in consecutive frame. The process then updates the prior estimate error covariance P_t^-

$$P_t^- = A \times P_{t-1}^+ \times A^T + Q \tag{9}$$

Measurement vector Z at any time t for right lane marking consists of quadratic parameters

$$Z_r = [\ \alpha_0, \alpha_1, \alpha_2]_r \tag{10}$$

The second stage corrects the predicted state using measurements. The Kalman gain K is calculated to refine the estimated state and update the error covariance from P_t^- to P_t^+

$$K_t = P_t^- \times (P_t^- + R)^{-1} \tag{11}$$

$$\hat{u}_t^+ = \hat{u}_t^- + K_t(Z_r - H\hat{u}_t^-) \tag{12}$$

$$P_t^+ = (I - K_t) \times P_t^- \tag{13}$$

The right lane estimated parameters are then derived from the Kalman state-space equation as

$$\left[\widehat{\alpha_0}_t^+, \widehat{\alpha_1}_t^+, \widehat{\alpha_2}_t^+\right]_r = \hat{u}_t^+[1:3] \tag{14}$$

Using these estimated parameters, bottom-most and top-most x-coordinates of the fitted curve for the right lane marking is calculated as

$$\left({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+\right) = \left\{x \mid y = f_r\left(x, \widehat{\alpha_0}_t^+, \widehat{\alpha_1}_t^+, \widehat{\alpha_2}_t^+\right), y \in \{1, C\}\right\} \tag{15}$$

For convenience, we have provided the equations solely for tracking the right lane marking. However, the same principle applies to track the left lane marking as well. Once the Kalman filter is updated, based on state estimates at time t, a demarcated area with boundary $x \in [x_{min} : x_{max}]$ and $y \in [1 : \ length(IPM)]$ around the detected lane marking is formed as

$$x_{min} = \hat{x}_t^+ - p_w \tag{16}$$

$$x_{max} = \hat{x}_t^+ + p_w \tag{17}$$

where

$$\hat{x}_t^+ = x \mid y = f_r\left(x, \widehat{\alpha_0}_t^+, \widehat{\alpha_1}_t^+, \widehat{\alpha_2}_t^+\right), y \in [1, C]\Big\} \tag{18}$$

$p_w = 20$ defines the pixel width added on both sides of the detected lane marking to form the DAOI. Constraining the search area for lane marks in the subsequent frame not only reduces background noise but also minimizes computational overhead.

2) Non-Linear Adaptation: To adapt to non-linear variations, the proposed framework employs two detection modes: *Mode1*, which detects lane marks within the established DAOI using inter-frame similarity, and *Mode2* which performs full-frame detection. Initially, *Mode1* operates over a temporal span of $(N-1)$ consecutive frames, named as *Mode1* cycle, while at each N^{th} frame, *Mode2* scans the entire image for both LHS and RHS and the detection outcome in each mode is validated using *measurement reliability* as mentioned in equation 7. Upon moving from the $(N-1)^{th}$ to the N^{th} frame, the algorithm shift from *Mode1* to *Mode2* and then two conditions may arise:

(i) Consistent detection: where DF<TPS i.e. *Mode2* detection results at N^{th} frame align with the $(N-1)^{th}$ frame's *Mode1* detection results, then algorithm continues in Mode1 for the detection in remaining frames in *Mode1* cycle.

(ii) Inconsistent detection: where DF>TPS, can arise from two possible scenarios, either (a) transient erroneous detection or (b) introduction of new lane mark. To handle inconsistent detection, initially predicted values derived from Kalman motion model in $(N-1)^{th}$ frame is retained in place of N^{th} frame result. Simultaneously, lane model parameters from *Mode2* at N^{th} frame are stored within moving average window estimator. However, as a consequence of encountering this inconsistency, algorithm switches to *Mode2* for the subsequent $(N+1)^{th}$ frame. If the inconsistency arises from transient errors, it is anticipated that subsequent detection results will match up with the existing motion model in subsequent frames. Consequently, *Mode1* will be reinstated and persist for remaining *Mode1* cycle. On the other hand, if there is introduction of new lane marks, their location consistency over frames is evaluated based on the values stored in the moving average window estimator. A new detection is declared if it maintains parallelism in defining the ego lane and satisfy the following two conditions:

$$\left({}_r^1x_t, {}_r^Cx_t - {}_r^1\tilde{x}_t, {}_r^C\tilde{x}_t\right) < TPS \tag{19}$$

$$\left({}_r^1\tilde{x}_t, {}_r^C\tilde{x}_t\right) = \left({}_r^1\tilde{x}_{t-1}, {}_r^C\tilde{x}_{t-1}\right) \tag{20}$$

where

$$\tilde{x}_t = \frac{\sum_{t-m}^{t} x_t}{m} \tag{21}$$

here $\tilde{x}$ represents an average value calculated for x-coordinate of the bottom-most and top-most point for right lane detection while m denotes the number of observations in the moving average window. In our study, m is set to 5. This allows the system to incorporate sudden changes in lane markings and redefine the ego lane. In this case, the DAOI is re-established, and the Kalman filter is re-initialized. Subsequently, algorithm enters in *Mode1* based on the newly established DAOI for the remaining frames *Mode1* cycle. Algorithm 1 is presented for the detection of right lane marking. However, a similar procedure is independently applied for the detection of left lane marking. When a newly detected lane mark deviates from parallelism with the existing lane mark on the opposite side, it is flagged as invalid, indicating a potential detection error due to disrupted alignment between the left and right

Algorithm 1 Algorithm for lane detection for right lane marking based on *Frame Coherence*

Require: IPM, $x_{mid}, {}_r^1x_t, {}_r^Cx_t, {}_r^1\hat{x}_{t-1}^+, {}_r^C\hat{x}_{t-1}^+, Flag1_t, Flag2_t, n$

for $t = 1, 2, \ldots, T$ (T is the number of frames) **do**

$({}_r^1x_t, [\alpha_0\alpha_1\alpha_2]_r, {}_r^Cx_t)$ =PLD$(IPM,\ mode, x_m)$

$({}_r^1\widetilde{x}_t, {}_r^C\widetilde{x}_t) \leftarrow$ Update MAWE $({}_r^1x_r, {}_r^Cx_r)$

DF =calculateDifference $({}_r^1x_t, {}_r^Cx_t, {}_r^1\hat{x}_{t-1}^+, {}_r^C\hat{x}_{t-1}^+)$

▷ Mode1 applied

if $mod(t/N) \neq 0$ **or** $Flag1_t == True$ **then**

$Flag1_{t+1} = True$

if $DF < TPS$ **then**

$R = \text{diag}\left(0.1^2,\ 0.1^2, \ldots,\ 0.1^2\right)$

$({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+) \leftarrow$ Update KF $({}_r^1x_t, {}_l^cx_t)$

else

$R = \text{diag}\left(10^2,\ 10^2, \ldots,\ 10^2\right)$

$({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+) \leftarrow$ Update KF (R)

end if

end if

▷ Mode2 applied

if $mod(t/N) == 0$ **or** $Flag2_t == True$ **then**

if $DF < TPS$ **then**

$Flag1_{t+1}$=$True$

$R = \text{diag}\left(0.1^2,\ 0.1^2, \ldots,\ 0.1^2\right)$

$({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+) \leftarrow$ Update KF $({}_r^1x_t, {}_r^cx_t)$

else if $({}_r^1x_t, {}_r^Cx_t - {}_r^1\widetilde{x}_t, {}_r^C\widetilde{x}_t) < TPS$ **then**

$Flag1_{t+1} = True$

$R = \text{diag}\left(0.1^2,\ 0.1^2, \ldots,\ 0.1^2\right)$

$({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+) \leftarrow$ Re-initialize KF $({}_r^1x_t, {}_r^Cx_t)$

else

$Flag2_{t+1} = True$

$R = \text{diag}\left(10^2,\ 10^2, \ldots,\ 10^2\right)$

$({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+) \leftarrow$ Update KF (R)

end if

end if

return $({}_r^1\hat{x}_t^+, {}_r^C\hat{x}_t^+)$

end for

Variables

PLD: Preliminary Lane Detection, TPS: Total Pixel Shift, KF: Kalman Filter, MAWE: Moving Average Window Estimator, DF: Deviation Factor

lane markings. Invalid right lane marks are corrected using parameters from the left lane mark, and vice versa for invalid left lane marks.

C. Analysis for Prolonged Shadow/Occlusion

In scenarios where prolonged shadows, occlusions, or other visual obstructions complicate the detection of lane markings in the *Preliminary Lane Detection* stage, *Frame Coherence* becomes crucial. Fig. 5(a) illustrates the detected and tracked positions of the left lane marking for clip 6 (frames 150–250). In Fig. 5(b), itMode1 works for 10 frames in a row, focusing

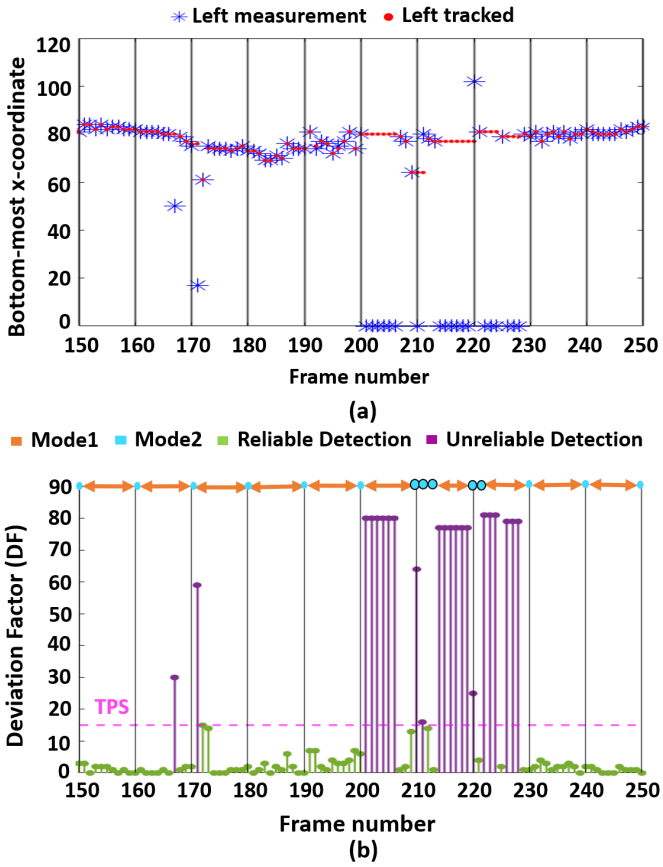

Fig. 5. (a) Detected and tracked positions of the left lane marking, represented by the bottom-most x-coordinate, for clip 6 of the dataset spanning frame numbers 150 to 250. (b) Measurement reliability for detection in each frame is assessed using the deviation factor (DF). Detections with DF>TPS are classified as unreliable and depicted in muted purple, while detections with DF<TPS are deemed reliable and shown in green.

only on the DAOI for detection while *Mode2* happens every 10th frame and checks the whole image for detection. The measurement reliability for each frame is calculated using the deviation factor (DF) relative to the threshold, TPS. Measurements with DF>TPS are categorized as unreliable, while those with DF<TPS are identified as reliable detections. As evident from Fig. 5, errors at frames 167 and 171 occur within *Mode1* and are transient in nature, effectively handled by the Kalman output estimation. Similarly, for frames 201–206, 214–219, 222–225, and 226–228 in *Mode1*, Kalman filter relies on its predictions to maintain accuracy. The system transitions from *Mode1* to *Mode2* between frames 209–210 and 219–220, where detections in frames 210 and 220 are classified as inconsistent due to DF>TPS. Although the Kalman filter prioritizes the predicted state for output estimation, this inconsistency causes the algorithm to remain in *Mode2* at frames 211 and 221. By frame 212 and 221, reliable detections align with the motion model, reinstating *Mode1*, which continues for the remainder of the cycle. It is important to note that the moving average window estimator does not play a significant role in this scenario, as no new lane marking values are consistently detected.

Noise such as shadows, construction lines, or worn-out paint can resemble lane markings if they persist for multiple frames. The moving average window estimator, designed to

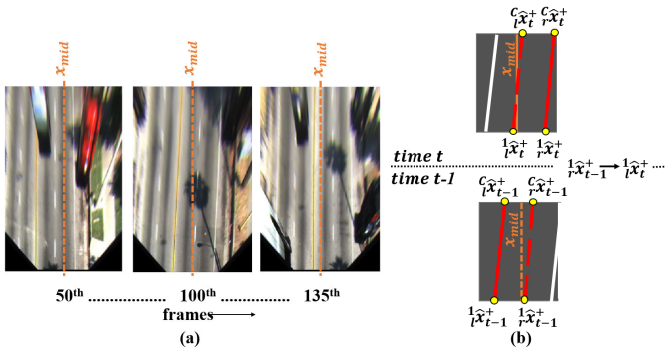

Fig. 6. (a) Lane marking appears vertical before (50^{th} frame) and after (135^{th} frame) lane transition. However it appear slanted while transitioning. (b) The orange line represents the mid-line, which divides the image into left and right sides. The green portion depicts the ego-lane for the driving car. The transition of the ego-lane from time stamp $t-1$ to t occurs as follows: before the transition, both end points of left lane marks lies left side of x_{mid} and right lane marks lies right side of x_{mid}. However, at the onset of lane transition, end points of right lane marks starts sweeping towards left side of x_{mid} and then new ego-lane is defined where the right lane marking of former ego-lane at t-1 becomes the left lane marking for new ego-lane after lane change at time t.

average recent observations, may incorporate this noise into its calculations, mistakenly identifying it as a valid lane mark. For a window size of $m = 5$, the algorithm identifies the new location only if the noise persists for more than 5 frames at the same position and the moving average computed by the window estimator at time t must remain consistent with its value at $t-1$ and align with the measurement at t (Equation 19, 20). In such cases, system continues to output false positive until *Mode2*, with correct lane detection is triggered. Once activated, *Mode2* recalibrate the system by accurately detecting the lane and updating Kalman filter and DAOI. However, such prolong noise is rare in dynamic driving situations and usually short-lived.

D. Lane Change

In real-world driving scenarios, lane change is frequent maneuver that causes shift in the camera's alignment relative to the lane markings and thereby changes its yaw angle. These changes can cause lane markings to appear slanted in IPM images for certain frames until the transition of ego vehicle from old lane to new lane is completed. As shown in Fig. 6 (a) lane marking appears vertical before (50^{th} frame) and after (135^{th} frame) lane transition. However, they appear slanted at 100^{th} frame. As shown in Fig. 6 (b) detection from LHS has top-most and bottom-most coordinates lying on the left side of the mid-demarcation x_{mid} i.e. ${}^{1}_{l}\hat{x}^{+}_{t-1} < x_{mid}$ and ${}^{C}_{l}\hat{x}^{+}_{t-1} < x_{mid}$ before transition. Similarly, detection from RHS has top-most and bottom-most coordinates lying on the right side of the mid-demarcation i.e. ${}^{1}_{r}\hat{x}^{+}_{t-1} > x_{mid}$ and ${}^{C}_{r}\hat{x}^{+}_{t-1} > x_{mid}$. However, at the onset of lane transition, end points of right lane marks starts sweeping towards left side of x_{mid} and then a new ego-lane is defined such that former right lane marking at $t-1$ becomes the left lane marking after lane change at t as

$$ {}^{1}_{l}\hat{x}^{+}_{t-1}, {}^{C}_{l}\hat{x}^{+}_{t-1} \rightarrow {}^{1}_{r}x_t, {}^{C}_{r}x_t \tag{22} $$

Fig. 7. Experimental set-up. The camera is installed above the vehicle and steadied using gimbal and rigs.

The algorithm then scan the entire RHS image to identify the new right lane marking. Similarly, when ego lane shifts to adjacent left lane at time from $t-1$ to t, the former left lane marking at $t-1$ becomes the right lane marking after lane change at t as

$$ {}^{1}_{r}\hat{x}^{+}_{t-1}, {}^{C}_{r}\hat{x}^{+}_{t-1} \rightarrow {}^{1}_{l}x_t, {}^{C}_{l}x_t \tag{23} $$

Once the lane change is confirmed, Kalman filter is reinitialized for both LHS and RHS of the image. For the next frame, new demarcation, x'_{mid} is defined to partition the image into left and right halves. If the lane changes to right side, new demarcation is defined as

$$ x'_{mid} = {}^{1}_{l}\hat{x}^{+}_{t} + w_l \tag{24} $$

where w_l is minimum distance between two lane markings, empirically estimated to be at least 40 pixels. It creates the new demarcation line to the right side of the existing mid-line, such that right lane-marking is searched in area confined with boundary $x \in [x'_{mid} : width(IPM)]$ *and* $y \in [1 : length(IPM)]$. This ensure that both the left and right lane marking detection are positioned on their respective sides relative to the new demarcation. Similarly, if the lane changes to left side, new demarcation is defined as

$$ x'_{mid} = {}^{1}_{r}\hat{x}^{+}_{t} - w_l \tag{25} $$

left lane-marking is searched in area confined with boundary $x \in [1 : x'_{mid}]$ *and* $y \in [1 : length(IPM)]$. New demarcation at x'_{mid} is utilized until both the left and right lane marking detection are positioned on their respective sides relative to the previous mid-line demarcation, x_{mid}.

IV. Results and Discussion

To validate the detection results, detected lines within every frame are compared to the hand-labeled ground truth, adhering to the evaluation methodology proposed in Aly's [8] and Yoo's [30] studies. P samples on detected line $\{l_1, l_2, l_3 \ldots l_P\}$ are selected. Then, their respective nearest P counterparts from the ground truth, represented as $g'_1, g'_2, g'_3, \ldots, g'_P$ are identified using the shortest Euclidean distance. Mean and median are derived as

$$ \bar{d}_L = \text{mean}\left(d_1{}^{L}_{G}, d_2{}^{L}_{G}, d_3{}^{L}_{G} \ldots . d_P{}^{L}_{G}\right) \tag{26} $$

$$ \tilde{d}_L = \text{median}\left(d_1{}^{L}_{G}, d_2{}^{L}_{G}, d_3{}^{L}_{G} \ldots . d_P{}^{L}_{G}\right) \tag{27} $$

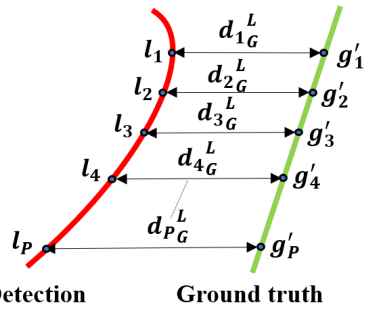

Fig. 8. Proposed evaluation criteria. P samples on the detected line (in red) are selected, and their respective nearest counterparts from the ground truth (in green) are identified using Euclidean distance.

Similarly, $\bar{d}_G$ and $\tilde{d}_G$ is derived by computing the distance between P samples on ground truth and their corresponding nearest samples on detected line. For both detection and ground truth to coincide, the following conditions have to be satisfied

$$\min\left(\bar{d}_L, \bar{d}_G\right) \leq t_1 \tag{28}$$

$$\min\left(\tilde{d}_L, \tilde{d}_G\right) \leq t_2 \tag{29}$$

where $t_1 = 10$ and $t_2 = 15$. The proposed method is evaluated on two datasets: 1) Our dataset and 2) Caltech dataset. The experiments are conducted on PC equipped with 3.00 GHz Intel Core i9 CPU and 32 GB RAM, with images scaled to 640×340.

Our Dataset: To generate the dataset, camera is mounted over the top of the vehicle and stabilized with the help of gimbal and rigs, as illustrated in Fig. 7. The recording occurs at frame rate of 30 fps on structured Indian roads, covering both highways and urban routes during daytime, spanning 200 kilometers across various road segments under diverse illumination conditions, including cloudy weather, dusk, glare, and shadows. Eight video clips were manually selected, resulting in a total of 7,113 images that capture driving environments with varying lighting conditions and road surfaces. The road types range from well-maintained sections with clear lane markings to more complex surfaces with faded or inconsistent markings, including stretches of smooth asphalt as well as areas with tar patches, repair work, rough patches, and uneven surfaces. Clip1 is recorded on cloudy day with minimal traffic on the highway. There is enough contrast of black-white-black pattern on road surface and lane markings are clearly visible. In contrast, clip2 and clip3 feature traffic scenarios with obscured lane markings due to the presence of cyclists and/or vehicles. Clip2 also exhibits frames with intense glare. Clip5 is captured under low-light conditions, while clip6 has intense shadow on the road scene. Tar patches, cracks and skid marks are evident in clip4, clip7, and clip8.

To gauge the robustness of our proposed method, we conducted a comparative analysis on our dataset against four existing methods i.e, Aly [8], Xu et al. [25], Srivastava et al. [26], and Yoo et al. [30]. Fig. 9 illustrates the comparison in detection outcomes between our method and those of Aly's, Yoo's, Xu's, and Srivastava's method on samples from our dataset. Detection rate is calculated as follows:

$$R\% = \frac{C_r + C_l}{2T} \tag{30}$$

where T is the total number of frames and C_l and C_r represent the counts of accurately detected left and right lanes, respectively. Table II shows the proposed model surpasses the performance of existing models in terms of detection accuracy, as evaluated on our dataset, which comprises Indian roads.

TABLE II
COMPARISON OF LANE DETECTION RATE ON OUR DATASET

Clips	Xu's [25]	Modified Xu's [25]	Aly's [8]	Yoo's [30]	Srivas-tava's[41]	Our Method1	Our Method2
Clip1	47.68	87.84	96.66	96.75	50.00	**97.1**	**99.67**
Clip2	54.15	75.71	71.25	89.08	43.47	93.95	**99.80**
Clip3	49.71	74.57	70.57	91.71	41.43	91.82	**99.86**
Clip4	3.61	81.12	89.74	99.01	99.46	**98.86**	**99.92**
Clip5	54.39	59.48	85.92	38.62	99.60	**98.80**	**99.10**
Clip6	55.24	59.54	95.11	35.78	97.88	87.11	**98.33**
Clip7	49.43	52.71	84.00	56.29	98.57	94.71	**100.00**
Clip8	11.29	58.71	91.28	98.00	50.00	**99.28**	**99.85**

As shown in the first column of Table II, Xu's approach [25] yields a very low detection rate. We modified their method by changing the threshold percentage at the interval of 5 and then the threshold value with maximum detection rate is considered. Different threshold values are obtained tailored to individual clips. While it has improved detection rate in clip1, clip2, clip3, clip4, and clip8, but the overall performance remains unsatisfactory with average detection rate of 68.71%, as indicated in the second column of Table II. This results emphasizes that single threshold is insufficient to handle the varying illumination levels across our different video clips. While using Xu's method, it is also observed that lane marks are easily removed by global thresholding if saturation due to bright light reduces the contrast of lane mark. Moreover, the formation of ROI is susceptible to edges originating from vehicles, stains, or any other aligned noises. These artifacts often generate significant peaks during vertical aggregation and eventually form strong connected components, thereby yielding a false positive detection results using their method.

Aly's method explore gradient properties by using oriented Gaussian kernels, which prove effective for detecting bright lane markings against a dark background, particularly on new roads, as evidenced by high detection rate of 96.66%, 95.11%, and 91.28% for clip1, clip6, and clip8 respectively. However, their methodology is prone to produce high false positives, if filters designed to detect vertical lines of a particular width erroneously identifies street writing, crosswalks, stop lines symbol, or other lane-like noise as lane marks. Moreover, in the absence of temporal integration, their methodology proves inadequate for missing or occluded lane, as evidenced by the detection results of 71.25% and 70.57% in clip2 and clip3 respectively.

Yoo's method demonstrates superior performance compared to the previous methodologies. However, detection performance suffers in scenarios characterized by intense shadows,

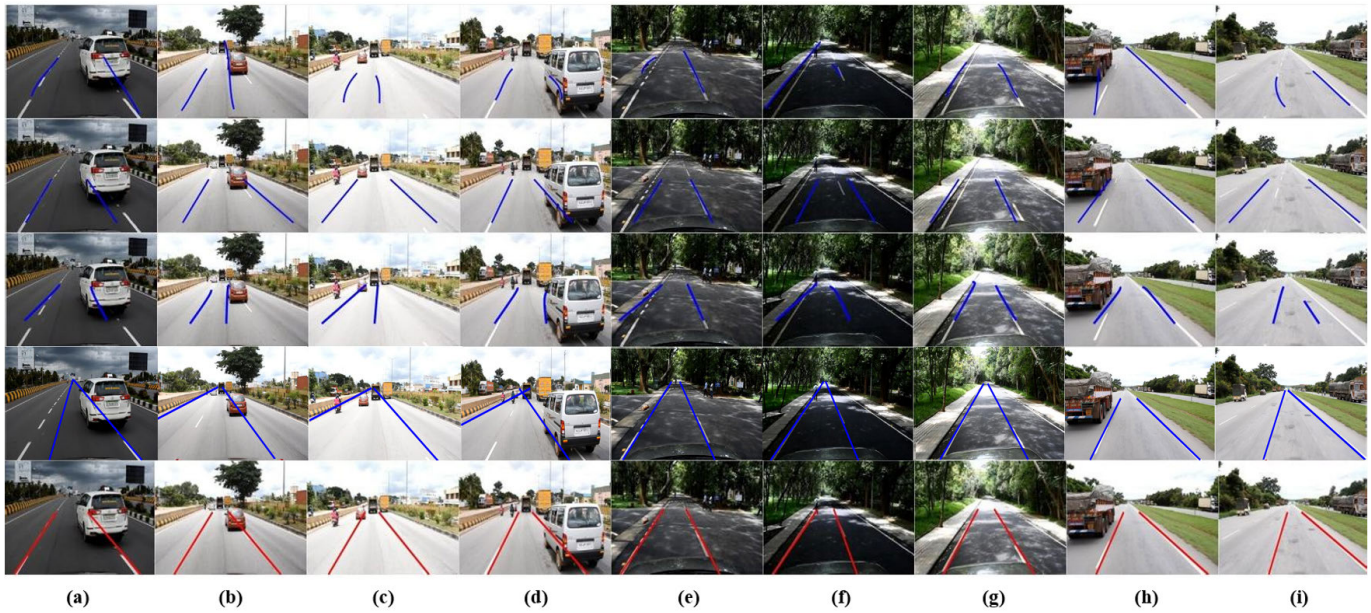

Fig. 9. Lane detection results on our dataset samples: Row 1 - Aly's method [8], Row 2 - Yoo's method [30], Row 3 - Xu's method [25], Row4- Srivastava's method [26], Row 5 - our proposed method. Detection samples show robustness of our proposed method when confronted with lane occlusion due to leading vehicle in columns (a), (b), (d), intense glare in column (c), shadows in in columns (e), (f), (g) and lane-like noises in columns (h) and (i).

as evidenced by a very low detection rate of 35% in clip6. The method inadequately addresses rapid changes in road dynamics, particularly during lane transitions, where angle constraints of ego lane are insufficient, leading to inaccurate detection. This limitation becomes apparent with a low detection rate of 35% and 56% in clip6 and clip7 respectively, both of which involve lane changes. Additionally, a lower detection rate, notably in clip2, clip3, and clip5, is due to line segments collected from vehicles with similar orientations of lane marks. This similarity disrupts the geometric relationships between line segments and the estimated vanishing point.

Srivastava's approach filters noise and irrelevant non-lane information from the given frame using sub-images. By applying an adaptive angle constraint with mark-ROI and integrating slope information, the method achieves a high average detection rate of 98.88% for clip4, clip5, clip7, and clip8, effectively detecting lane changes. However, the approach relies on the initial accumulation of lane parameters to establish a reference model and lacks a self-correction mechanism when repetitive artifacts, such as tire imprints or road repairs with slopes resembling lane markings, appear during the initial sequence. This limitation can lead to inaccurate ROI detection, as observed by low average detection rate of 46.22% for clip1, clip2, clip3, and clip8. Further, the tuning of Canny edge detection introduces trade-offs: a high threshold can omit worn-out or faint lane markings, while a low threshold increases sensitivity to artifacts like tire tracks or repairs which can mislead the algorithm in lane change scenarios.

As can be seen in Table II, our work is presented in two methods. Method1 involves solely applying the proposed method based only on *Preliminary Lane Detection*. Conversely, Method2 incorporates *Frame Coherence* along with *Preliminary Lane Detection*. The proposed method outperforms other methods with an average detection rate of 99.55% as seen in last column. In traffic scenarios where lane markings are occluded by leading vehicles, our Method1 achieves a better detection rate of 93.95% and 91.82% for clip2 and clip3, respectively, as compared with others. Relatively, a low detection rate of 87.11% is seen for clip6, that deals with intense shadow and lane change scenarios. However, in Method2 upon integrating the *Frame Coherence* module, a substantial improvement over detection rate by 5.85%, 8.04%, 11.22% is observed for clip2, clip3, and clip6, respectively.

Our method works well to maintain the definition of ego lane, even in scenarios involving lane changes, as depicted in Fig. 10(a), (b), the trajectory of the ego lane location is plotted, where the x-axis represents the frame numbers, and the y-axis denotes the bottom-most x-coordinate of the detected left and right lane marking. Fig. 10 (a) shows that ego lane undergoes a rightward shift around the 40^{th} frame, which is indicated by abrupt positional transitions of both left and right lane markings in Fig. 10 (c). At 40^{th} frame number, position of left lane mark shifts from q1 to q2, followed by the right lane mark changes from q2 to q3. Similarly, around 265^{th} frame, the ego lane undergoes a leftward shift (Fig. 10 b), with the position of right lane mark shifts from q5 to q4, followed by the left lane mark changes from q4 to point q6. Moreover, the Fig. 10 (c) provides tracking outcomes for clip7, highlighting the position with DF greater than TPS, emphasizing kalman filter towards the predicted state over the measurement during its estimation process. As a result, smoother tracking outcomes are achieved, as illustrated in Figure 10.

Caltech Dataset: The proposed methods are also applied on the Caltech dataset [8] with results shown in Fig 11. Caltech dataset consists of four clips: cordova1, cordova2, washington1 and washington2, which includes markings like deflection arrows, word messages, bifurcation arrows, rectangular blocks, and zebra crossing marks. To capture yellow lane marking in

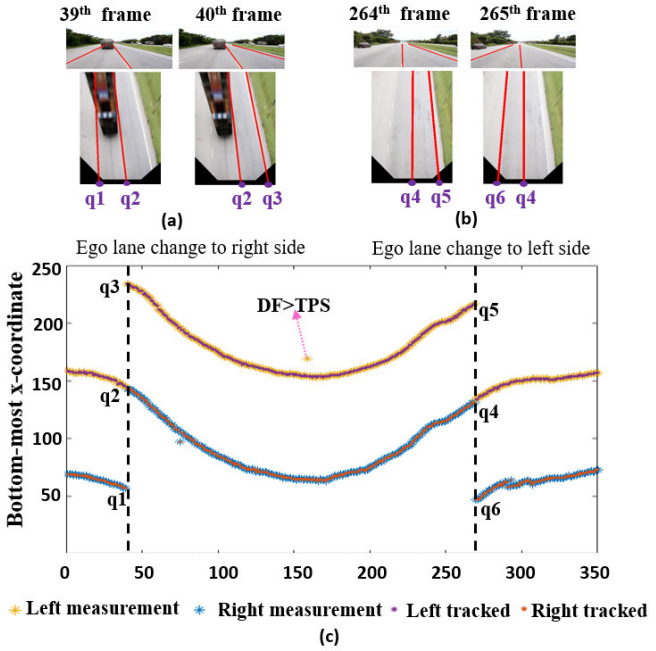

Fig. 10. (a) Ego lane change to right from frame 39^{th} to 40^{th}. (b) Ego lane change to left from frame 264^{th} to 265^{th}. (c) The detected and tracked positions of the bottom-most x-coordinate for clip7 within our dataset reveal a lane change to the right side, indicated by the abrupt transition of the left lane mark point from q1 to q2 and the right lane mark point from q2 to q3. Similarly, during a lane change to the left side, the left lane mark point transitions from points q5 to q4 and from q4 to q6. Measurements with deviation factor (DF) greater than tolerable pixel shift (TPS) are smoothed by Kalman tracking.

TABLE III
COMPARISON OF LANE DETECTION RATE ON CALTECH DATASET

Clips	Aly's [8]	Yoo's [30]	Our Method1	Our Method2
Cordova1	91.60	89.40	93.80	**98.50**
Cordova2	75.37	88.42	81.22	**91.01**
Washington1	92.43	83.97	91.39	**95.76**
Washington2	92.46	95.25	98.07	**98.83**

Caltech dataset, we empirically set the threshold band on H channel to 20° as the lower bound and 65° as the upper bound. Unlike previous approaches by Yoo et al. [30] and Aly [8], which used evaluation criteria of $t_1 = 15$ and $t_2 = 20$ for the Caltech dataset, we employed $t_1 = 10$ and $t_2 = 15$ to minimize false positive detections. Comparison of lane detection rate on Caltech dataset is shown in Table III. Our method achieves an average detection rate of 96.03% using Method2 and 91.12% using Method1, surpassing the rates achieved by Yoo's and Aly's methods, which a 89.26% and 87.96%, respectively.

Despite Yoo's method relying on inter-frame similarity for location consistency and vanishing point estimation, it exhibits a comparatively low detection rate of 88.42% and 83.97% in cordova2 and washington1, respectively. This is primarily due to limitation of line segment detector to extract accurate line segments in scenes with inconsistent illumination levels and dense shadows. Moreover, its sensitivity to noise often results in incorrect detection, particularly when aligned shadows of trees, crosswalks, or other lane-like noise coincide with lane marks, as similarly observed in case of our dataset.

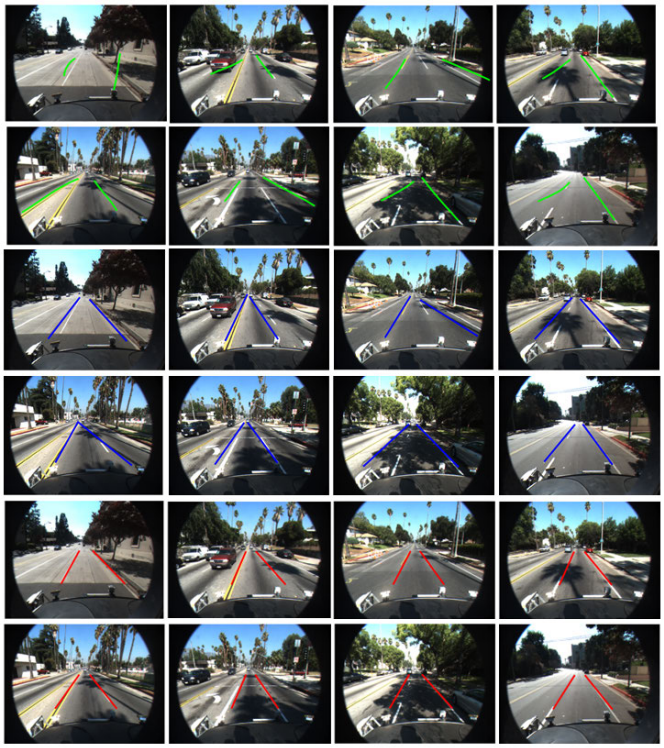

Fig. 11. Lane detection results on Caltech dataset samples: Row 1, 2 - Aly's method [8], Row 3, 4- Yoo's method [30], Row 5, 6- our proposed method.

In the cordova2 clip, the lane width undergoes abrupt changes attributed to the presence of parking lines (f00264 to f00314). Additionally, there are sudden alterations in the location of lane marks before and after the intersection. *Frame Coherence* module in the Method2 effectively accommodates such variations by periodically employing *Mode2* and verifying the current kalman state model and moving average window estimator. However, absence of right lane marking cues consistently over subsequent frame leading to incorrect detection rate in cordova2 images. Among the 407 frames analyzed, left lane marks were successfully detected in 406 frames, whereas only 334 frames showed successful detection of right lane marks. While edge detection could potentially remedy this issue, it falls beyond the scope of our current work. By effectively addressing lane changes, our Method2 achieves a high detection rate of 95.76% in washington1 images. Fig 12 shows the trajectory of ego lane over 350 frames for washington1 images. The ego lane shifts to left side around 100^{th} frame, with transition of left lane mark from q7 to q8 and right lane mark from q8 to q9. Presence of intense ambient light, combined with strong shadows results in persistent lane-like noise over frames which can inadvertently surpass through moving average window estimator (Equation 19, 20) and provide false positives measurements.

Our dataset, annotation and results are available on https://sites.google.com/view/lane-detection2024/home.
Detection on both the LHS and RHS are done independently and simultaneously. Our algorithm, with and without the *Frame Coherence* module, achieves a run-time of less than 30 ms per frame, making it well-suited for real-time applications.

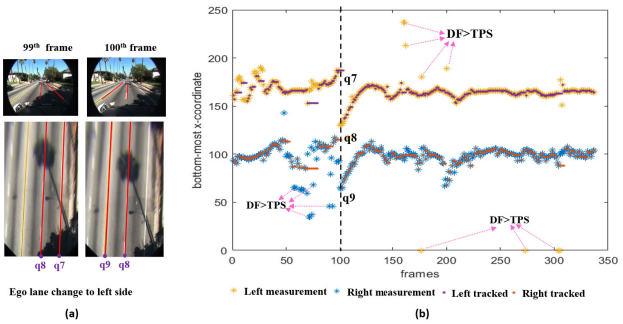

Fig. 12. The detected and tracked positions of the bottom-most x-coordinate for washington1 in Caltech dataset. Lane change to left side at 100^th frame is evident by sudden transition of right lane mark point from (a) to (b) and left lane mark point (b) to (c). Measurements with deviation factor (DF) greater than tolerable pixel shift (TPS) are smoothed by kalman tracking.

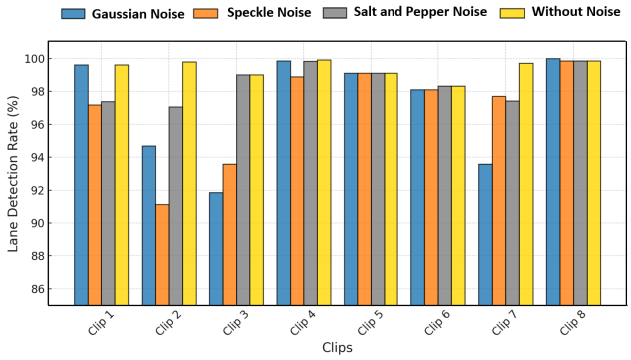

Fig. 13. Comparison of lane detection accuracy across clips degraded by Salt-and-Pepper Noise, Speckle Noise, and Gaussian Noise.

A. Analysis for Noise

Due to the dynamic nature of the environment and the motion of the vehicle, onboard cameras may experience vibrations, low-light conditions, high ISO[1] settings, and electronic disturbances. To assess the performance of the algorithm under such adverse scenarios, the dataset undergoes degradation with three types of noise: salt-and-pepper noise, speckle noise, and Gaussian noise. Salt-and-pepper noise, often associated with rain or dust particles, is introduced with a noise density of 1%. Fig. 13 shows that this noise causes a detection rate variation of less than 3%. Speckle noise, often perceived as graininess, arises from reflections from textured surfaces such as concrete or asphalt and becomes more pronounced under high ISO settings in low-light conditions. In this study, speckle noise with a variance of 0.005 is applied, leading to a reduction in detection rates by more than 8% and 6% for clip2 and clip3, respectively. The reduction is a consequence of the multiplicative nature of speckle noise, which becomes amplified in regions with intense glare, generating higher noise levels in bright areas and making their processing more challenging. Gaussian noise, which directly affects pixel intensities and is applied with a standard deviation of 0.05, leads to performance degradation of 5.12%, 8.01%, and 6.43% for clip2, clip3, and clip7, respectively. This degradation occurs because noise saturation disrupts pixel variability and corrupts statistical features which is critical for lane detection. Additional factors such as engine vibrations, uneven roads, and wind resistance can cause additional blur or distortion in the image. However, the gimbal setup, as shown in Fig. 7, stabilizes the camera by counteracting these disturbances, ensuring consistent frames in low-light conditions and avoiding unnecessary noise amplification caused by erratic movements. Overall, the results clearly demonstrate a decline in detection rates with an increase in noise variance.

V. Conclusion

The proposed method significantly enhances feature extraction performance compared to traditional techniques, thereby improving lane detection and tracking effectiveness. Our methodology diverges from the assumption of inter-frame similarity, which often overlooks variations in dynamic driving scenarios. Instead, we enable the algorithm to adapt to diverse conditions by systematically comparing the outcomes of inter-frame similarity analysis with periodic full-frame detection, utilizing a moving average window estimator within the verification protocol. The work relies on the assumption that the occlusion of lane marks and the presence of lane-like noise are transient phenomena, particularly in the context of moving vehicles. This understanding allows our algorithm to effectively differentiate between meaningful lane markings and temporary disturbances caused by factors such as passing vehicles or environmental conditions. The robustness of the proposed algorithm across diverse environmental conditions, including varying illumination, vehicle occlusion, lane-like noise, lane change and dense shadows, is demonstrated by the experimental results. However, it struggles in foggy environments due to limited visibility. Additionally, while the algorithm effectively adjusts the ego lane during abrupt variations in lane width, the moving average window estimator introduces latency in adjusting to changes in lane positions. This latency arises because the estimator calculates the average position of lane marks over a certain window of frames. As a result, it takes time for the estimator to fully incorporate new information about lane positions into its calculations. During this adjustment period, the estimated lane positions may not accurately reflect the current state of the road, leading to false positive results. While such instances are rare in complex scenarios and typically brief in duration, future improvements could involve employing sensor fusion techniques to enhance the ability of system to quickly identify and adapt to alterations in lane configurations.

[1]ISO stands for International Organization for Standardization.

References

[1] S. P. Narote, P. N. Bhujbal, A. S. Narote, and D. M. Dhane, "A review of recent advances in lane detection and departure warning system," *Pattern Recognit.*, vol. 73, pp. 216–234, Jan. 2018.

[2] K.-Y. Chiu and S.-F. Lin, "Lane detection using color-based segmentation," in *Proc. IEEE Intell. Vehicles Symp.*, Jul. 2005, pp. 706–711.

[3] J. Gonzalez and U. Ozguner, "Lane detection using histogram-based segmentation and decision trees," in *Proc. IEEE Intell. Transp. Syst.*, Oct. 2000, pp. 346–351.

[4] T.-Y. Sun, S.-J. Tsai, and V. Chan, "HSI color model based lane-marking detection," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Sep. 2006, pp. 1168–1172.

[5] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, and K.-C. Fan, "Lane detection with moving vehicles in the traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 571–582, Dec. 2006.

[6] J. M. Álvarez, A. M. López, and R. M. Baldrich, "Shadow resistant road segmentation from a mobile monocular system," in *Proc. Iberian Conf. Pattern Recognit. Image Anal.*, Jul. 2007, pp. 9–16.

[7] J. Deng and Y. Han, "A real-time system of lane detection and tracking based on optimized RANSAC B-spline fitting," in *Proc. Res. Adapt. Convergent Syst.*, Oct. 2013, pp. 157–164.

[8] M. Aly, "Real time detection of lane markers in urban streets," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2008, pp. 7–12.

[9] M. Li, Y. Li, and M. Jiang, "Lane detection based on connection of various feature extraction methods," *Adv. Multimedia*, vol. 2018, no. 1, 2018, Art. no. 8320207.

[10] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[11] D.-J. Kang and M.-H. Jung, "Road lane segmentation using dynamic programming for active safety vehicles," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 3177–3185, Dec. 2003.

[12] Y. Wang, N. Dahnoun, and A. Achim, "A novel system for robust lane detection and tracking," *Signal Process.*, vol. 92, no. 2, pp. 319–334, Feb. 2012.

[13] G. H. Chen, W. Zhou, F. J. Wang, B. J. Xiao, and S. F. Dai, "Lane detection based on improved Canny detector and least square fitting," *Adv. Mater. Res.*, vols. 765–767, pp. 2383–2387, Sep. 2013, doi: 10.4028/WWW.SCIENTIFIC.NET/AMR.765-767.2383.

[14] J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-time illumination invariant lane detection for lane departure warning system," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 1816–1824, Oct. 2014.

[15] J. Niu, J. Lu, M. Xu, P. Lv, and X. Zhao, "Robust lane detection using two-stage feature extraction with curve fitting," *Pattern Recognit.*, vol. 59, pp. 225–233, Nov. 2016.

[16] C. Lee and J.-H. Moon, "Robust lane detection and tracking for real-time applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 4043–4048, Dec. 2018.

[17] C. R. Jung and C. R. Kelber, "Lane following and lane departure using a linear-parabolic model," *Image Vis. Comput.*, vol. 23, no. 13, pp. 1192–1202, Nov. 2005.

[18] V. Nguyen, H. Kim, S. Jun, and K. Boo, "A study on real-time detection method of lane and vehicle for lane change assistant system using vision system on highway," *Eng. Sci. Technol., Int. J.*, vol. 21, no. 5, pp. 822–833, Oct. 2018.

[19] J. Hur, S.-N. Kang, and S.-W. Seo, "Multi-lane detection in urban driving environments using conditional random fields," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 1297–1302.

[20] M. Haloi and D. B. Jayagopi, "A robust lane detection and departure warning system," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2015, pp. 126–131.

[21] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.

[22] T.-T. Tran, C.-S. Bae, Y.-N. Kim, H.-M. Cho, and S.-B. Cho, "An adaptive method for lane marking detection based on HSI color model," in *Proc. Int. Conf. Intell. Comput.* Cham, Switzerland: Springer, Jan. 2010, pp. 304–311.

[23] A. Borkar, M. Hayes, and M. T. Smith, "A novel lane detection system with efficient ground truth generation," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 365–374, Mar. 2012.

[24] Y. Wang, D. Shen, and E. K. Teoh, "Lane detection using spline model," *Pattern Recognit. Lett.*, vol. 21, no. 8, pp. 677–689, Jul. 2000.

[25] S. Xu, P. Ye, S. Han, H. Sun, and Q. Jia, "Road lane modeling based on RANSAC algorithm and hyperbolic model," in *Proc. 3rd Int. Conf. Syst. Informat. (ICSAI)*, Nov. 2016, pp. 97–101.

[26] S. Srivastava and R. Maiti, "Robust lane change detection and tracking in urban environment," in *Proc. IEEE 8th Int. Conf. Signal Image Process. Appl. (ICSIPA)*, Sep. 2024, pp. 1–6.

[27] S. K. Gehrig, A. Gern, S. Heinrich, and B. Woltermann, "Lane recognition on poorly structured roads—The bots dot problem in California," in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, Jul. 2002, pp. 67–71.

[28] M. Liang, Z. Zhou, and Q. Song, "Improved lane departure response distortion warning method based on Hough transformation and Kalman filter," *Informatica*, vol. 41, no. 3, pp. 283–288, 2017.

[29] A. Borkar, M. Hayes, and M. T. Smith, "Robust lane detection and tracking with ransac and Kalman filter," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 3261–3264.

[30] J. H. Yoo, S. Lee, S. Park, and D. H. Kim, "A robust lane detection method based on vanishing point estimation using the relevance of line segments," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3254–3266, Dec. 2017.

[31] G. Liu, F. Wörgötter, and I. Markelic, "Combining statistical Hough transform and particle filter for robust lane detection and tracking," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2010, pp. 993–997.

[32] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-snake," *Image Vis. Comput.*, vol. 22, no. 4, pp. 269–280, Apr. 2004.

[33] P.-C. Wu, C.-Y. Chang, and C. H. Lin, "Lane-mark extraction for automobiles under complex conditions," *Pattern Recognit.*, vol. 47, no. 8, pp. 2756–2767, Aug. 2014.

[34] R. Yousri, M. A. Elattar, and M. S. Darweesh, "A deep learning-based benchmarking framework for lane segmentation in the complex and dynamic road scenes," *IEEE Access*, vol. 9, pp. 117565–117580, 2021.

[35] M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. Image Process.*, vol. 7, no. 1, pp. 62–81, Jan. 1998.

[36] M. Nieto, L. Salgado, F. Jaureguizar, and J. Arrospide, "Robust multiple lane road modeling based on perspective analysis," in *Proc. 15th IEEE Int. Conf. Image Process.*, 2008, pp. 2396–2399.

[37] S. N. Kang, S. Lee, J. Hur, and S. W. Seo, "Multi-lane detection based on accurate geometric lane estimation in highway scenarios," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 221–226.

[38] Z. Ying and G. Li, "Robust lane marking detection using boundary-based inverse perspective mapping," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 1921–1925.

[39] H. Yoo, U. Yang, and K. Sohn, "Gradient-enhancing conversion for illumination-robust lane detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1083–1094, Sep. 2013.

[40] C. Yuan, H. Chen, J. Liu, D. Zhu, and Y. Xu, "Robust lane detection for complicated road environment based on normal map," *IEEE Access*, vol. 6, pp. 49679–49689, 2018.

[41] S. Srivastava and R. Maiti, "Multi-lane detection robust to complex illumination variations and noise sources," in *Proc. 1st Int. Conf. Electr., Control Instrum. Eng. (ICECIE)*, Nov. 2019, pp. 1–8.

[42] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[43] S. Jung, J. Youn, and S. Sull, "Efficient lane detection based on spatiotemporal images," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 289–295, Jan. 2016.

[44] A. Gern, R. Moebus, and U. Franke, "Vision-based lane recognition under adverse weather conditions using optical flow," in *Proc. IEEE Intell. Vehicle Symp.*, vol. 2, Aug. 2002, pp. 652–657.

[45] J. Ruyi, K. Reinhard, V. Tobi, and W. Shigang, "Lane detection and tracking using a new lane model and distance transform," *Mach. Vis. Appl.*, vol. 22, no. 4, pp. 721–737, Jan. 2011.

[46] A. Almagbile, J. Wang, and W. Ding, "Evaluating the performances of adaptive Kalman filter methods in GPS/INS integration," *J. Global Positioning Syst.*, vol. 9, no. 1, pp. 33–40, Jun. 2010.

Saumya Srivastava received the B.Tech. degree in instrumentation and controls engineering from Uttar Pradesh Technical University, India, in 2014, and the M.Tech. degree from IIT-BHU in 2017. She is currently pursuing the Ph.D. degree with the Department of Design and Manufacturing, Indian Institute of Science, India. She has multidisciplinary research experience in the fields of biomedical engineering, autonomous driving, and ADAS systems. Her research interests include image processing, pattern recognition, and computer vision.

Rina Maiti received the M.Tech. and Ph.D. degrees in biomedical engineering from IIT Bombay, India, in 1995 and 2001, respectively. She has worked as a one-year Post-Doctoral Fellow with the Machine Learning Laboratory, Pohang University of Science and Technology, South Korea. She is currently an Assistant Professor with the Department of Design and Manufacturing, Indian Institute of Science, India. Her research interests include computer vision, virtual reality, biomedical signal processing, and assistive device design.