Engineering and Technology Journal e-ISSN: 2456-3358

Volume 11 Issue 03 March-2026, Page No.- 9247-9253

DOI: 10.47191/etj/v11i03.18 I.F. – 8.595

A Real-Time, Training-Free Lane Detection Approach Robust to Day–Night Illumination Changes On CPU-Only Platforms

Marwa K. Hasan[1*], Lana O. Alhamawndi[2], Alaan Ghazi[3], Lamees S. Ahmed[4], Ali J. Mohammed[5]

[1,2,4]Technical Engineering College - Kirkuk, Northern Technical University, Kirkuk, 36001, Iraq

[3]Department of Cybersecurity Techniques Engineering, College of Computer Engineering Technology and Artificial Intelligence - Kirkuk, Northern Technical University, Mosul, 41001, Iraq

[5]College of Oil and Gas Technology Engineering / Kirkuk, Northern Technical University, Kirkuk, Iraq

ABSTRACT: This paper presents a lightweight, training-free lane-line detection pipeline for daytime and night-time driving videos implemented in Python using OpenCV. The method performs illumination-aware pre-processing using CLAHE, and applies mild gamma correction in LAB space for low-light scenes. A trapezoidal road region-of-interest is then masked to suppress background clutter, followed by adaptive edge extraction using Canny with data-driven thresholds. Candidate lane segments are obtained via the probabilistic Hough transform and separated into left/right groups using slope-based filtering. The final lane boundaries are estimated by length-weighted averaging of line parameters and rendered as an overlay to visualize the drivable lane region and heading direction. Experiments on day and night recordings captured on the same road segments demonstrate stable lane detection under shadows, headlight bloom, and worn markings. The proposed approach is suitable for CPU-only embedded/mobile deployment and can support lane-departure warning and lane-keeping assistance in ADAS and autonomous driving applications.

KEYWORDS: Lane-Line Detection; Night-Time Driving; Advanced Driver-Assistance Systems; Region-Of-Interest Masking; Autonomous Driving.

I. INTRODUCTION

Lane-line detection is a fundamental perception task for Advanced Driver-Assistance Systems (ADAS) and autonomous driving, as it supports lane-departure warning and lane-keeping assistance by estimating road boundaries and vehicle lateral position. Regulatory and consumer-information programs increasingly emphasize these functions; for example, NHTSA's New Car Assessment Program (NCAP) update adds lane keeping assist (LKA) and enhances the performance evaluation of crash-avoidance technologies through objective criteria and a phased roadmap [1], [2].

Despite sustained progress, reliable lane detection remains difficult in real-world operation because the visual appearance of lane markings varies significantly with illumination and road-surface conditions. Daytime scenes may include strong shadows and glare, whereas nighttime scenes commonly suffer from low contrast, sensor noise, and headlight bloom that disrupt edge continuity. Worn paint and heterogeneous pavement textures further reduce the saliency of lane boundaries. These challenges motivate practical approaches that can adapt to illumination changes while remaining computationally efficient for embedded or mobile platforms.

Recent literature broadly follows two directions. Deep learning methods can achieve high accuracy by learning robust representations and global context, but typically require labeled data, careful domain adaptation, and compute resources that may not be available for CPU-only deployment [3], [4]. In contrast, classical computer-vision pipelines remain valuable for training-free, transparent, and low-latency operation, especially when they combine illumination normalization, region-of-interest (ROI) masking, adaptive edge extraction, and geometry-based line estimation.

In this work, we present a lightweight, training-free lane-line detection pipeline implemented in Python using OpenCV. The method applies illumination-aware preprocessing (CLAHE for contrast normalization, with mild gamma correction in LAB space for low-light scenes), restricts processing to a trapezoidal road ROI, and extracts lane candidates using adaptive Canny edge detection followed by the probabilistic Hough transform. Detected line segments are separated into left/right groups via slope-based filtering, and the final lane boundaries are estimated using length-weighted averaging of line parameters to stabilize the overlay across frames. The resulting lane visualization (lane area and heading indication) is designed for real-time CPU execution and can support practical LDW/LKA-style functionality in ADAS and autonomous driving applications [5], [6].

II. LITERATURE REVIEW

Recent lane-detection research spans benchmarks and datasets, multi-task perception, improved 2D lane representations, temporal stabilization for video, and

3D/topology-aware mapping. On the benchmark side, OpenLane-V2 extends lane perception beyond frame-wise 2D markings by introducing 3D lane detection together with topology reasoning between lanes and traffic elements, enabling more structured scene understanding for planning and HD-map consistency [7]. Complementarily, a recent comprehensive review surveys over 30 public lane-detection datasets and analyzes their annotation schemes and environmental coverage (lighting, weather, sensors), which directly influence the robustness and generalization of proposed methods [8].

A second theme is unified perception for intelligent driving. YOLOPv3 demonstrates a one-stage multi-task network that jointly performs traffic object detection, drivable-area segmentation, and lane detection, aiming to reduce overall computation by sharing features while maintaining real-time inference in practical settings [9]. Within dedicated lane-detection architectures, several works focus on improving representation and reliability under weak visual cues. CLRerNet highlights the importance of calibrated confidence by using a LaneIoU-based assignment/loss to better align confidence scores with geometric lane overlap [10]. Lane2Seq transforms the problem of lane detection into a sequence generation problem by utilizing a transformer encoder-decoder in order to integrate all forms of lane representation into a single output interface [11]. LDTR uses an anchor-chain representation of a lane in a transformer setting in order to simplify the complexity of post-processing while simultaneously improving robustness in cases of low visibility or occlusion [12]. In the case of curve-based representation, BSNet re-examines the problem of lane detection through B-spline representation and a curve-distance loss function while simultaneously exhibiting high runtime efficiency [13].

Temporal consistency in video data is also important, especially in the presence of motion blur, occlusions, and noise in the night scene. LaneTCA is designed to incorporate long-range and short-range temporal context through attention mechanisms to provide stable lane prediction. In parallel to this development, topology- and geometry-conscious designs for understanding 3D lanes keep improving. Topo2D promotes improved 3D lane topological understanding through the use of reliable 2D lane priors to initialize the 3D queries and the addition of 2D features for relationship classification. The LATR system detects 3D lanes from monocular images through transformer-based lane-aware queries with dynamic 3D ground positional embeddings to attempt to eliminate the need for BEV feature warping.

The problem of robustness to low-light conditions is still an issue of practical relevance. Zhang et al. propose a dedicated dataset for low-light lanes (NightLane) and an efficient real-time system incorporating low-light enhancement and attention-driven fusion to address issues of discontinuity and false positives due to headlight bloom and contrast reduction. Info-FPNet, on the other hand, addresses multi-scale robustness by using an informative feature pyramid and cross-layer refinement to improve detection performance even for real-world appearances. In general, although learning-based approaches are accurate, their training and computation complexity drive the need for lightweight approaches as additional tools, especially in resource-scarce environments where CPU-only execution is also an option [14].

The trade-offs involved in the design of the platforms demand a delicate balance between the quality of preprocessing, representation, and computational complexity. The effectiveness of denoising algorithms has been observed to improve the structural integrity and edge preservation capabilities of grayscale images under the influence of structured noises, specifically beneficial for low-light and nighttime driving applications [15]. Although deep learning-based pattern recognition algorithms have been observed to perform remarkably well for a range of vision applications, the need for extensive training data sets and computational complexity makes them less adaptable for CPU-only real-time systems [16]. Further, the importance of robustness, stability, and sensitivity control, often stressed within the context of deterministic algorithmic paradigms, underlines the need for safe operating characteristics under different operating conditions [17].

III. ARCHITECTURE

A forward-facing RGB video stream is processed on CPU in real time using a training-free OpenCV pipeline. Each frame is restricted to a trapezoidal road ROI, then (nighttime) enhanced using CLAHE on LAB-L with optional mild gamma brightening. The ROI is converted to grayscale, denoised with Gaussian smoothing, and passed through Canny edge detection. Candidate lane primitives are extracted using the probabilistic Hough transform (HoughLinesP), filtered by slope, and separated into left/right sets. A representative boundary is estimated for each side by aggregating the retained segments. The lane centerline and lateral offset are computed at a reference row, and the detected lanes/corridor are rendered as an overlay for visualization and reporting.

IV. METHODOLOGY

A. Data Acquisition and Study Design

This study implements a lightweight, training-free lane-line detection pipeline in Python/OpenCV for daytime and nighttime driving videos captured by a forward-facing monocular camera mounted near the vehicle windshield. To enable a controlled comparison across illumination regimes, the same roadway segments were recorded during day and night to preserve similar viewpoints and lane-marking layout. All processing is executed frame-by-frame on a standard CPU, consistent with real-time constraints of embedded and mobile platforms.

B. Region of Interest and Illumination Handling

To reduce computation and suppress false detections from non-road regions, processing is restricted to a fixed trapezoidal region of interest (ROI) that covers the drivable roadway while excluding sky, vehicle hood, and roadside clutter. Because lane appearance varies with illumination and surface reflectance, preprocessing is mode-dependent. Daytime frames are processed with standard intensity handling (optionally with mild contrast normalization when required). Nighttime frames apply contrast enhancement using CLAHE on the luminance channel (LAB-L), optionally followed by gentle gamma brightening to improve visibility of under-exposed lane markings and stabilize subsequent edge extraction under low contrast and headlight-related artifacts.

C. Edge Extraction and Candidate Segment Detection

Within the ROI, each frame is converted to grayscale and smoothed using a Gaussian filter to reduce sensor noise and fine pavement texture. Lane-candidate edges are extracted using Canny edge detection. Candidate lane segments are then detected using the probabilistic Hough transform (HoughLinesP), which provides a compact set of line primitives suitable for real-time processing and remains effective when markings are partially fragmented.

D. Geometric Filtering and Left/Right Lane Estimation

Detected Hough segments are filtered using slope-magnitude criteria to remove near-horizontal clutter and non-lane structures. Remaining segments are separated into left and right lane sets based on the sign of the slope (negative for left, positive for right), reflecting the expected image-plane geometry for forward-facing road scenes. For each side, a single representative lane boundary is estimated by aggregating the candidate segments (e.g., averaging of line parameters, optionally weighted by segment length). This aggregation reduces sensitivity to short noisy detections and produces a stable boundary estimate for visualization and offset measurement.

E. Lane-Center Offset and Output Rendering

At frames where both lane boundaries are detected, the lane centerline $x_c(y)$ is obtained from the left and right lane estimates. To quantify lateral deviation in pixel space, we compute the lane-center offset at a reference row $= yy_0$ near the bottom of the ROI

$$\frac{w}{2} = x_{\mathrm{veh}}, \quad \Delta x = x_{\mathrm{c}(y_0)} - x_{\mathrm{veh}} \qquad (1)$$

where x_{veh} denotes the vehicle/camera forward axis (image center) and W is the frame width. Fig. 1 illustrates the detected left/right boundaries and the corresponding estimated centerline used for computing Δx.

Detected left and right lane boundaries are used to estimate the lane centerline $x_c(y)$. The lane-center offset Δx is computed at a reference row $y = y_0$ relative to the image/vehicle center $x_{\mathrm{veh}} = W/2$.

To provide a visual example of the final system output, representative daytime and low-light/night results are presented in Fig. 2, showing the detected lane boundaries and the estimated drivable corridor under different illumination conditions.

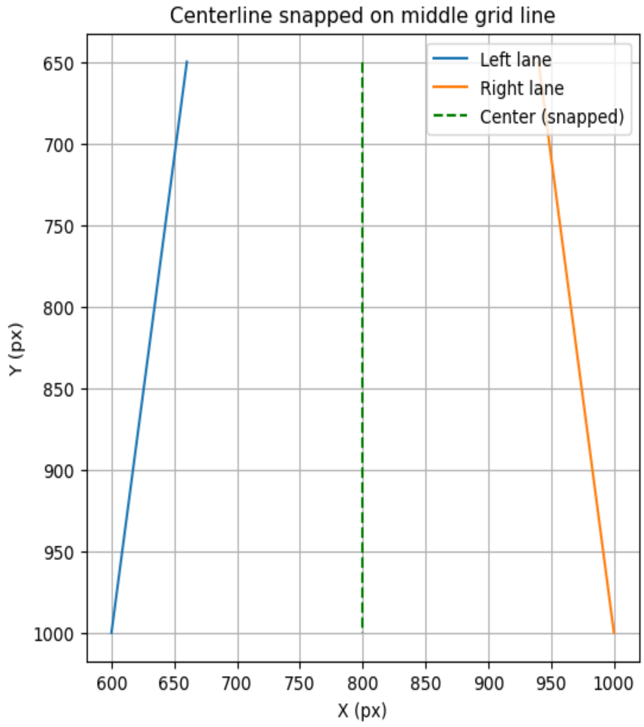

Fig.1. Centerline estimation and lane-center offset computation.

F. Parameter Setting, Reproducibility, and Real-Time Considerations

The pipeline is implemented in Python using OpenCV and NumPy and operates deterministically given fixed parameters. For a given camera setup, the same processing stages and parameter set are retained across day and night sequences, with illumination handling enabled for nighttime clips. Real-time performance is achieved by restricting expensive operations to the ROI and relying on efficient classical primitives (Gaussian smoothing, Canny, and HoughLinesP), making the approach suitable for CPU-only embedded or mobile deployment.

V. RESULTS AND DISCUSSION

To evaluate the proposed training-free lane-detection pipeline, we report qualitative outputs and quantitative stability indicators from paired daytime and nighttime (low-light) driving sequences captured on the same road segments. Qualitative examples emphasize the visual coherence of the detected left/right boundaries and the continuity of the drivable corridor overlay under shadows, mild curvature, and surrounding traffic. Quantitative analyses focus on lateral stability using the lane-center offset Δx, including distributional comparisons and categorical summaries (Left/Center/Right) derived from a small deadband around zero. Together, these results assess robustness across illumination regimes while maintaining CPU-friendly real-time operation.

Day vs. low-light qualitative comparison. Fig. 3 highlights that daytime benefits from higher lane-marking contrast, whereas the illumination-aware night branch preserves coherent boundaries and a stable corridor overlay despite reduced contrast and headlight-related artifacts.

Lane-position category counts (nighttime). To provide a concise discrete assessment of lateral behavior under low illumination, each frame in the nighttime sequence was assigned to one of three categories (Left, Center, Right) based on the lane-center offset Δx using a small deadband around zero. As shown in Fig. 4, the distribution is dominated by Left and Right assignments, whereas fewer frames fall within the Center band. This pattern is consistent with the broader spread observed in the nighttime offset distribution and indicates increased variability under low signal-to-noise conditions and headlight-related artifacts. Importantly, despite this variability, the detected lane boundaries remain continuous and geometrically plausible, supporting stable corridor visualization and downstream ADAS-style interpretation.

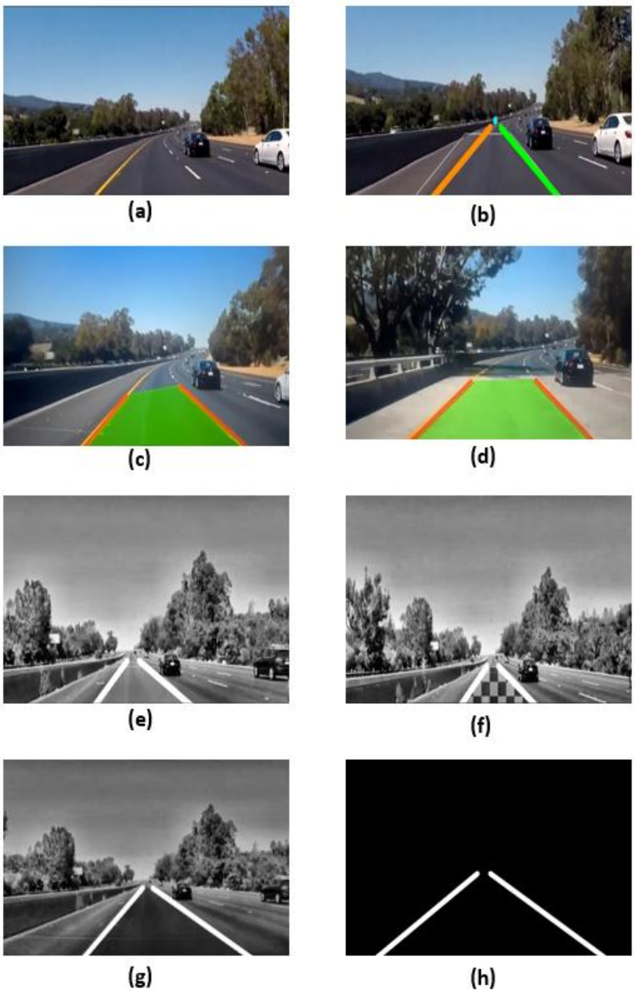

Fig.2. Qualitative results of the proposed lane-detection pipeline in daytime and nighttime conditions (a–h). Daytime examples (a–d) show the raw input, extracted boundaries, and corridor overlay under typical daylight variations, while nighttime examples (e–h) demonstrate the illumination-aware branch (CLAHE, optionally mild gamma) using print-safe visualization and the corresponding ROI-restricted lane-candidate map used for HoughLinesP seeding.

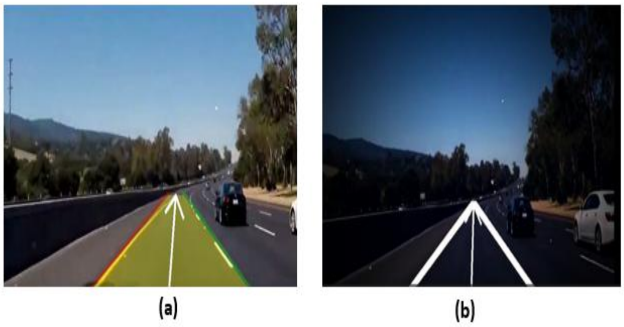

Fig.3. Qualitative comparison between daytime and low-light operation (a–b). (a) Daytime lane boundaries and drivable corridor overlay with a stable center guidance cue. (b) Low-light/night output showing coherent lane boundaries and stable guidance after illumination-aware preprocessing.

Lane center offset and apparent lane width. To verify the robustness of our lane-detection pipeline in lighting conditions, we have investigated frame-by-frame lane center offset and apparent lane width obtained from the detected left and right lanes. The results of lane center offset for the daytime and night-time datasets are shown in Fig. 5 and 6, respectively. Fig. 7 shows the apparent lane width at image-row levels.

Lane center offset distribution (day versus night). In the daytime video sequence (Fig. 5), the distribution of the lane center offset has a narrow peak around zero. This means that the estimated lane centerline is close to the vehicle’s front axis in most images. The small spread and the lighter tails of this distribution indicate a stable retention of the vehicle within the lane during daytime conditions. In the night video sequence (Fig. 6), although the distribution is centered around zero, it has a larger spread compared to daytime. This is expected due to the lower intensity of the image and the effects of vehicle headlights. However, most of the distribution is within a small spread around zero.

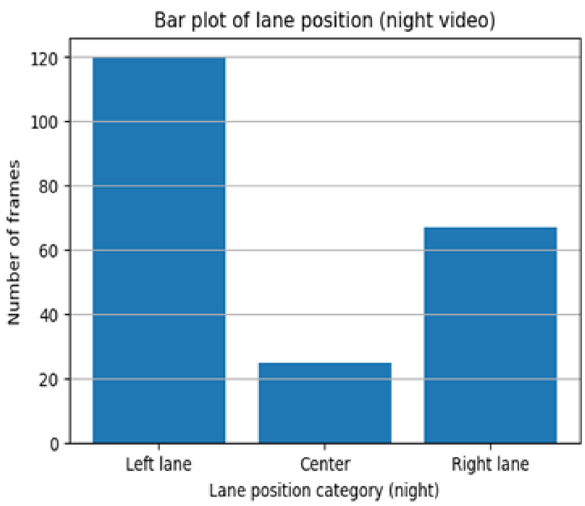

Fig.4. Lane-position category counts for the nighttime sequence.

Frame-wise counts of Left, Center, and Right categories computed from the lane-center offset $\boldsymbol{\Delta x}$ using a deadband threshold around zero.

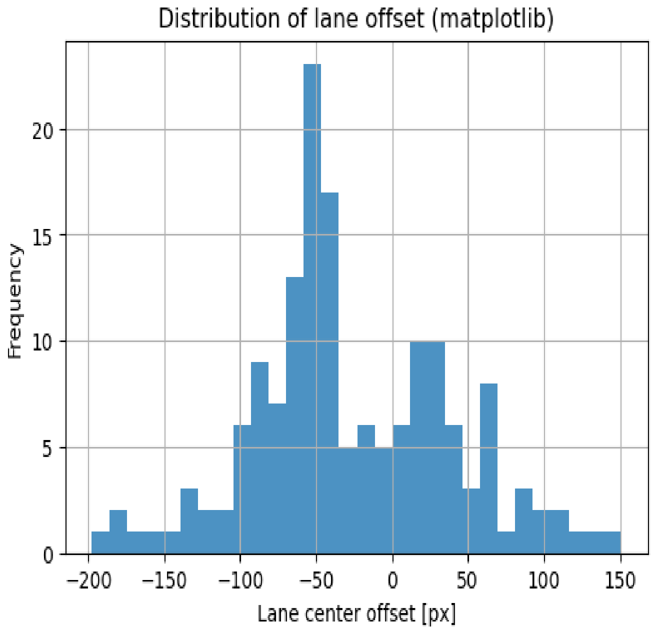

Fig.5. Lane-center offset distribution (day). Histogram of frame-wise Δx for the daytime sequence.

Apparent lane width as a function of image position. Fig. 7 shows the apparent lane width at various image row positions, Y. Note that the width of the lane increases along with the image row position, Y, as expected by perspective projection in the camera view, meaning that the lane width appears wider at the bottom of the image (closer to the camera) and narrower towards the top of the image (farther away). This is an important observation, as it verifies the geometric validity of the detected left and right lane boundaries and, more importantly, verifies that the processing pipeline retains realistic inter-lane separation in the region of interest, which is important for accurate estimation of lane center and offset positions. In general, these results show that the proposed training-free processing pipeline is able to provide robust estimates of the lane center offset positions, even during nighttime scenes, although there is an increase in the variance of the estimates.

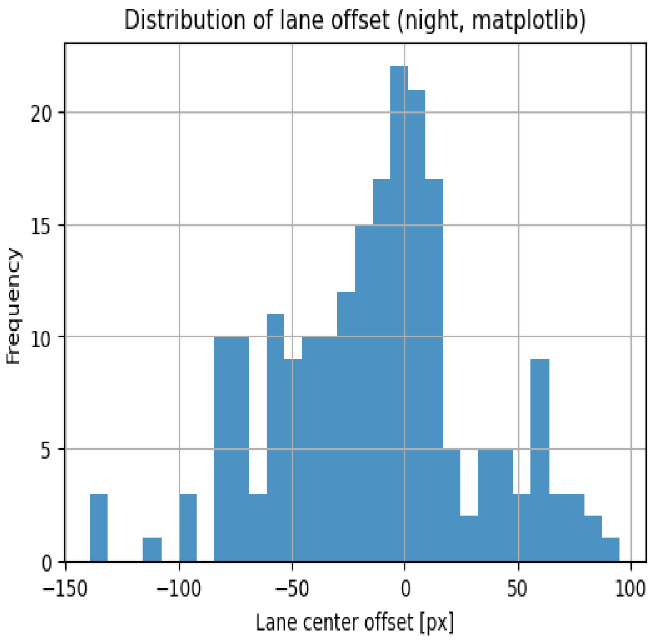

Fig.6. Lane-center offset distribution (night). Histogram of frame-wise Δx for the nighttime sequence.

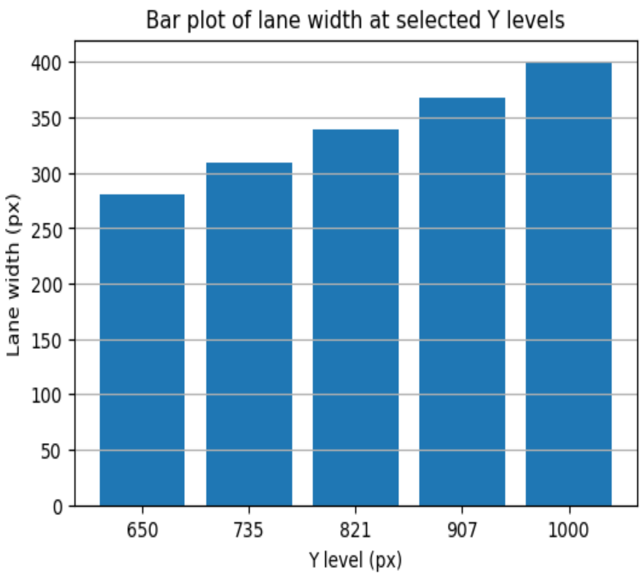

Fig.7. Apparent lane width at selected image-row positions (Y). Bar plot of the estimated inter-lane separation measured at selected image-row positions; the near-field widening is consistent with perspective projection.

Joint distribution of ROI brightness and lane center offset (night). To investigate whether there is systematic bias due to residual changes in illumination, we plot the joint distribution of mean ROI brightness and lane center offset Δx. Fig. 8 shows that, according to the density contours, there is no systematic shift of Δx along the brightness values, which indicates that the preprocessing is robust to global brightness changes, and the remaining errors are more likely to be due to local effects such as headlight bloom, specular reflections, and faded lane markings.

Lateral Offset Stability over illumination conditions. In Fig. 9(a), it is clear that Δx, representing lane center offsets for corresponding day and night sequences, is centered around zero with respect to a normalized frame number. This implies that, for most frames, the predicted lane center line is close to the vehicle's central axis. There seem to be larger oscillations around the center line during night illumination.

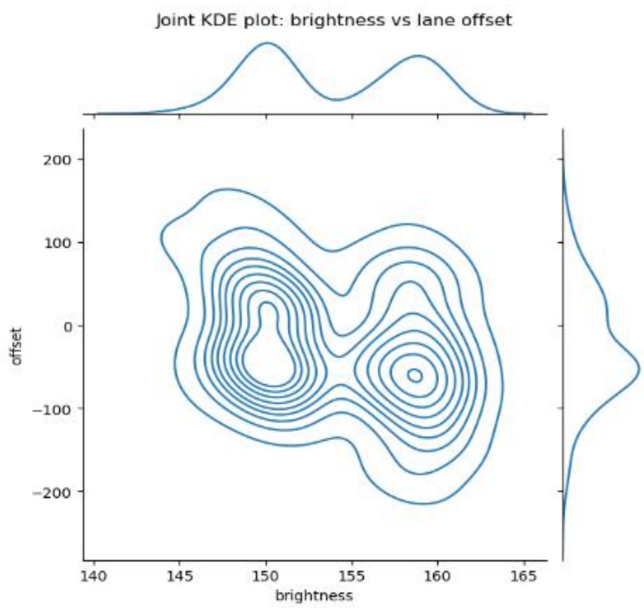

Fig.8. Joint distribution of ROI brightness and lane-center offset (night).

Joint KDE contours between mean ROI brightness and lane-center offset Δx for the nighttime sequence, with marginal density plots.

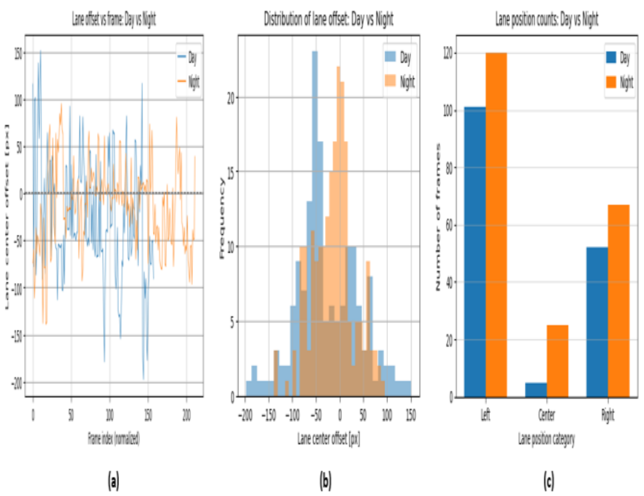

Fig.9. Day–night comparison of lane-center offset and lane-position categories: (a) frame-wise lane-center offset Δx for daytime and nighttime sequences (normalized frame index); (b) overlaid histograms of Δx for day vs. night; and (c) frame counts by lane-position category (Left/Center/Right) computed from Δx using a zero-centered deadband threshold.

Distributional Comparison. The histogram overlay on top of the bar chart in Fig. 9(b) confirms this observation: there is a more compact distribution for daytime values, with a slightly more spread-out distribution for night values. More significantly, both distributions are centered around zero, with no lateral bias introduced by the change in illumination conditions.

Discrete Categorical Summary. In a concise summary of the findings, Fig. 9(c) provides the counts of the lane position categories (Left/Center/Right), obtained from Δx with a deadband around zero. The comparison between the daytime and nighttime data shows that the categorical pattern is similar but with the nighttime variation of the estimated positions reflected in the increased counts for positions off the center lane. Overall, from Fig. 9(a)-9(c), the findings suggest that the training-free system provided by this paper does not deviate much from the estimated positions of the lane markers during the daytime and nighttime.

VI. CONCLUSIONS

This study presented a lightweight, training-free lane-line detection pipeline implemented in Python/OpenCV that operates in real time on a CPU-only platform and remains effective across both daytime and low-light/night driving. The approach confines processing to a trapezoidal road region of interest (ROI) and employs illumination-aware enhancement for nighttime operation (CLAHE with optional mild gamma brightening) to improve the visibility of under-exposed lane markings.

The candidates for the lane are then extracted by converting the image to grayscale, Gaussian smoothing, and adaptive edge detection by the Canny edge detection algorithm. The HoughLinesP function is used for the probabilistic line segment detection. Geometric filtering, involving slope-magnitude and left/right separation based on the sign of the slope, is used to obtain a reliable lane boundary.

Experiments on matched day/night sequences shot along the same track showed consistent estimates of lane boundaries and visible drivable regions for both day and night conditions. Daytime estimates produced clean overlays despite shadows and clutter, and night estimates were robust to preprocessing despite loss of contrast and headlight artifacts. In summary, this system offers a clear, deployable classical vision baseline that does not require any training data and features low latency by restricting computations to the region of interest.

The key limitations of our system include dependence on visible lane markings, vulnerability to heavy, rain, fog, and glare, and performance under heavily worn and hidden lane markings. Future research will include advanced stabilization techniques, model extension to curve-based representations to handle roads with higher curvatures, and expanded evaluations on public benchmarks and embedded targets.

ACKNOWLEDGMENT

The authors would like to thank the Northern Technical University, Kirkuk, Iraq, for providing the academic environment and facilities that supported this research. No external sponsorship was received for this work.

REFERENCES

1. D. Kim, S. Kim, Y. Kim, and K. Lee, “Review and Analysis of Driver Control Assistance Systems (DCAS) Regulation,” Journal of Auto-vehicle Safety Association 2025 17:3, vol. 17, no. 3, pp. 88–95, Sep. 2025, doi: 10.22680/kasa2025.17.3.088.
2. J. Wishart, “Safety Impacts of Active Safety and Driving Automation Features,” 2025, Accessed: Mar. 11, 2026. [Online]. Available: https://www.sae.org/gsdownload/?prodCd=EPR2025013
3. X. He et al., “Monocular Lane Detection Based on Deep Learning: A Survey,” Dec. 2024, [Online]. Available: http://arxiv.org/abs/2411.16316
4. J. Zhao, Y. Wu, R. Deng, S. Xu, J. Gao, and A. Burke, “A Survey of Autonomous Driving from a Deep Learning Perspective,” ACM Comput. Surv., vol. 57, no. 10, May 2025, doi: 10.1145/3729420.
5. Odeh and N. Odeh, “OpenCV and its Applications in Artificial Intelligent Systems,” 2024 International Conference on Intelligent Computing, Communication, Networking and Services, ICCNS 2024, pp. 242–249, 2024, doi: 10.1109/ICCNS62192.2024.10776047.
6. G. Bradski, “The OpenCV Library,” Dr. Dobb’s Journal: Software Tools for the Professional Programmer, vol. 25, no. 11, pp. 120–125, Nov. 2000, [Online]. Available: https://opencv.org/links/
7. H. Wang et al., “OpenLane-V2: A Topology Reasoning Benchmark for Unified 3D HD Mapping,” Adv. Neural Inf. Process. Syst., vol. 36, Oct. 2023, Accessed: Mar. 11, 2026. [Online]. Available: http://arxiv.org/abs/2304.10440
8. J. Gamerdinger, S. Teufel, and O. Bringmann, “Datasets for Lane Detection in Autonomous Driving: A Comprehensive Review,” Apr. 2025, Accessed: Mar. 11, 2026. [Online]. Available: https://arxiv.org/pdf/2504.08540v1
9. J. Zhan, J. Liu, Y. Wu, and C. Guo, “Multi-Task Visual Perception for Object Detection and Semantic Segmentation in Intelligent Driving,” Remote Sensing 2024, Vol. 16, Page 1774, vol. 16, no. 10, p. 1774, May 2024, doi: 10.3390/rs16101774.
10. H. Honda and Y. Uchida, “CLRerNet: Improving Confidence of Lane Detection with LaneIoU,” Proceedings - 2024 IEEE Winter Conference on Applications of Computer Vision, WACV 2024, pp. 1165–1174, May 2023, Accessed: Mar. 22, 2026. [Online]. Available: http://arxiv.org/abs/2305.08366
11. K. Zhou, “Lane2Seq: Towards Unified Lane Detection via Sequence Generation,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 16944–16953, Feb. 2024, Accessed: Mar. 22, 2026. [Online]. Available: http://arxiv.org/abs/2402.17172
12. Z. Yang et al., “LDTR: Transformer-based Lane Detection with Anchor-chain Representation,” Comput. Vis. Media (Beijing)., vol. 10, no. 4, pp. 753–769, Mar. 2024, Accessed: Mar. 22, 2026. [Online]. Available: http://arxiv.org/abs/2403.14354
13. H. Chen, M. Wang, and Y. Liu, “BSNet: Lane Detection via Draw B-spline Curves Nearby,” Jan. 2023, Accessed: Mar. 11, 2026. [Online]. Available: https://arxiv.org/pdf/2301.06910
14. G. Lian, “Robust Lane Detection Based on Informative Feature Pyramid Network in Complex Scenarios,” Electronics 2025, Vol. 14, Page 3179, vol. 14, no. 16, p. 3179, Aug. 2025, doi: 10.3390/electronics14163179.
15. D. M. Witefee and A. Al-Kharaz, “A dual-phase hybrid framework for real-time grayscale image denoising in structured noise,” International Journal of Advances in Intelligent Informatics, vol. 11, no. 4, pp. 636–652, Nov. 2025, doi: 10.26555/ijain.v11i4.2158.
16. M. Y. Pusadan, J. R. Sasuwuk, S. A. Pratama, and R. Laila, “Pattern recognition for facial expression detection using convolution neural networks,” International Journal of Advances in Intelligent Informatics, vol. 11, no. 4, pp. 622–635, 2025, doi: 10.26555/ijain.v11i1.1602.
17. Makmun, M. T. Suryadi, and S. Madenda, “LC Map: A robust chaotic function for enhancing cryptographic security through key sensitivity and randomness analysis.,” International Journal of Advances in Intelligent Informatics, vol. 11, no. 4, p. 667, Nov. 2025, doi: 10.26555/ijain.v11i4.1854.