Rev. Roum. Sci. Techn.– Électrotechn. et Énerg.
Vol. **71**, *2*, pp. 299–304, Bucarest, 2026

LANE BOUNDARY DETECTION IN DYNAMIC ENVIRONMENTS

YAMINA BELHADRI[1], DAHMANI MOHAMED[1], BELAL ALSHAQAQI[2], ABDULLAH SALEM BAQUHAIZEL[1]

Keywords: Collision warning systems (CWS); Advanced driver assistance system (ADAS); Hough transform; TuSimple dataset.

Lane detection is essential for collision warning systems. This technology helps vehicles maintain proper road position by identifying lane markings. This paper presents a method for lane detection within an ADAS using a monocular camera in real time. The approach uses the Hough Transform to detect lane lines in an image. The detected lines are filtered based on specific criteria, including angle, length, and symmetry relative to the image's central axis, to select the best lines representing the lane boundaries. These techniques are crucial for improving the precision of driver assistance systems by ensuring reliable lane detection, even under varying conditions. Experiments were conducted in MATLAB using the publicly available TuSimple dataset, which contains diverse road scene images. The results indicate that the proposed method achieves precise lane-line detection, which is crucial for real-time monocular vision-based driver-assistance systems, thereby significantly enhancing road safety.

1. INTRODUCTION

Lane detection and tracking are essential for collision prevention, alerting drivers to unintended lane departures and improving safety in autonomous and semi-autonomous vehicles. Classical methods rely on color segmentation, edge detection, and geometric transformations, while recent approaches use multi-stage pipelines combining filtering, adaptive thresholds, and the Hough transform to enhance robustness under challenging conditions [1,2].

However, difficult scenarios such as faded markings, curved roads, or partial occlusions require effective preprocessing and reliable feature extraction. Recent RRST studies highlight the value of multi-stage pipelines and advanced image preprocessing techniques for improving detection accuracy [3,4].

The proposed algorithm builds on these principles to detect lane markings under diverse road and marking types, while remaining optimized for real-time performance. A key challenge addressed is the reliable removal of noise while preserving relevant lane patterns, ensuring consistent and robust detection across complex driving scenarios.

2. RELATED WORKS

Various techniques have been developed for lane detection assistance, employing methods such as color-based segmentation, edge detection, and geometric transformations. [5] proposed a color-based approach using quadratic functions to locate lane boundaries in complex environments, while [6] applied multi-adaptive thresholds to sub-images to handle challenging conditions like fog and shadows. Using edge detection, [7] introduced frequency-domain features paired with deformable templates for accurate lane-marking detection. The Hough transform has been widely used, as demonstrated by [8], who achieved high accuracy in detecting lane boundaries under varying lighting conditions, and by [9], who enhanced its robustness using advanced computer vision technologies. [10] also utilized the Hough transform to extract orientation and positional parameters of lanes. Advanced techniques include [11] B-Snake-based detection and tracking algorithm, which integrates a particle filter to remove noise, and Bertozzi's GOLD system, which remaps images into a top-down view for parallel lane detection. Additionally, [12] employed a Clothoid curve and Kalman filter to ensure continuity and predict lane parameters across frames, while [13] improved lane boundary recognition by combining features such as position, orientation, and line intensity, selecting the best candidate based on proximity to the previous lane vector. These methods demonstrate significant progress in overcoming challenges like noise, environmental variability, and the need for real-time performance in lane detection systems [14–15].

3. PROPOSED METHOD

This section details the methodology used for lane detection within the framework of driver assistance systems. This approach implements the Hough Transform for line detection and symmetry for validating lane lines to provide a robust solution for lane detection, even in complex environments.

3.1 PREPROCESSING

Before line detection, several preprocessing steps are applied to the image to reduce noise and enhance the quality of lane contours:

Preprocessing is an essential and critical phase in the lane detection pipeline, as it prepares the raw image for advanced analysis.

Fig. 1 – Original image.

3.1.1. GRAYSCALE CONVERSION

The first stage of the proposed pipeline consists in converting the input RGB image (Fig. 1) into a grayscale representation. Let $I_{RGB}(x,y)$ = [R(x,y),G(x,y),B(x,y)] denote the input color image. The grayscale intensity image obtained after conversion is illustrated in Fig. 2 and is computed using a weighted linear combination of the color channels:

$$I_{gray}(x,y) = \alpha R(x,y)+\beta G(x,y)+\gamma B(x,y), \quad (1)$$

where α, β, and γ are fixed coefficients.

Grayscale representations are widely used as a

[1]Laboratory signals and images (LSI), University of Sciences and Technology of Oran -Mohamed Boudiaf-, Algeria.
[2]University of Oran 2 Mohamed Ben Ahmed, Oran, Algeria.
E-mails : amina.belhadri@univ-usto.dz;mohammed.dahmani@univ-usto.dz; abdullah.baquhaizel@univ-usto.dz, belal.alshaqaqi@univ-usto.dz

DOI : 10.59277/RRST-EE.2026.2.22

preliminary step in image processing pipelines, as they reduce data dimensionality while preserving intensity information required for feature extraction and structural analysis [1–3].

Fig. 2 – Grayscale image.

3.1.2 CONTRAST ADJUSTMENT

After grayscale conversion, a linear contrast adjustment is applied to enhance the visibility of lane markings under varying illumination conditions. Let $I_{gray}(x,y)$ denote the grayscale image. The contrast-enhanced image $I_c(x,y)$, illustrated in Fig. 3, is obtained through linear intensity normalization defined as:

$$I_c(x,y)= (I_{gray}(x,y)-I_{min}) / (I_{max}-I_{min}), \quad (2)$$

where I_{min} and I_{ma} denote the minimum and maximum intensity values in the image.

This linear contrast stretching enhances mid-range intensity variations and improves the separation between lane markings and the road surface. Such intensity transformation techniques are widely used in image preprocessing to facilitate subsequent edge detection and structural analysis stages [1–3].

Fig. 3 – Adjusted image.

3.1.3 BINARIZATION

The grayscale image is converted into a binary representation to isolate regions of interest corresponding to lane markings. Let $I_c(x,y)$ denote the contrast-enhanced grayscale image. The binarized image $I_b(x,y)$, illustrated in Fig. 4, is obtained through intensity thresholding:

$$I_b(x,y) = \begin{cases} 0, & I_c(x,y) < 0, \\ 1, & I_c(x,y) \geq T, \end{cases} \quad (3)$$

where T is the intensity threshold value, which is fixed. This technique simplifies subsequent analysis by reducing data complexity and enhancing the detectability of structural features [1,2,12,14,15].

Fig. 4 – Binary image.

3.1.4 GAUSSIAN FILTERING

Gaussian filtering is applied to the image $I_b(x,y)$ to suppress high-frequency noise while preserving the structural characteristics of lane markings. The filtered image $I_g(x,y)$, shown in Fig. 5, is computed by convolution with a Gaussian kernel G_σ:

$$I_g(x,y)=\sum_{u=-k}^{k}\sum_{v=-k}^{k} I_b(x-u,y-v)G_\sigma(u,v), \quad (4)$$

where

$$G_\sigma(u,v) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{u^2+v^2}{2\sigma^2}\right), \quad (5)$$

σ controls the smoothing strength, and k defines the kernel size.

The Gaussian filter attenuates local intensity variations caused by noise while maintaining the continuity of lane edges. The standard deviation σ is chosen to balance noise reduction and edge preservation [1,2,17,13].

The Gaussian filtering ensures that lane marking edges remain continuous and less fragmented after binarization, which is critical for the subsequent edge detection and Hough transform stages.

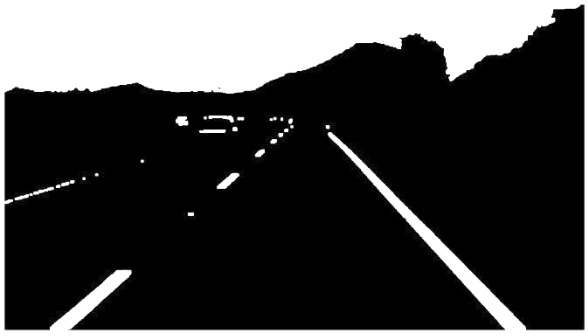

Fig. 5 –Gaussian filtering image.

3.2. EDGE DETECTION (CANNY)

The Canny edge detector is applied to extract significant edges. Let $I_g(x,y)$ be the smoothed image. The gradients in horizontal and vertical directions are computed:

$$G_x=\frac{\partial I_g}{\partial x} \ , \ G_y=\frac{\partial I_g}{\partial y} \ , \ M(x,y)=\sqrt{{G_x}^2+{G_y}^2},$$

$$\theta(x,y) = \arctan\left(\frac{G_y}{G_x}\right) \quad (6)$$

Non-maximum suppression, double thresholding (T_h, T_l), and hysteresis produce the binary edge map $I_e(x,y)$, illustrated in Fig. 6 [1,2,4,11,19].

Canny edges provide continuous and well-localized lane boundaries for subsequent processing.

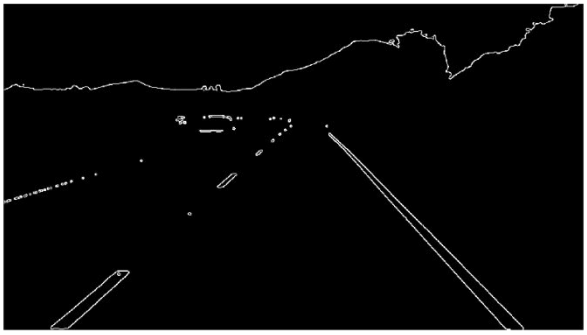

Fig. 6 – Edge image.

3.3. REDUCTION OF THE REGION OF INTEREST

To concentrate the analysis on relevant areas, a trapezoidal region of interest (ROI) is defined, corresponding to the road surface as projected in the image plane. Let $I_e(x,y)$ be the binary edge map from Canny detection. A mask $R(x,y)$ representing the trapezoidal ROI is applied:

$$I_{ROI}(x,y) = I_e(x,y)R(x,y). \quad (7)$$

Only edges within the ROI are retained, while edges outside are discarded, as illustrated in Fig. 7. The trapezoid accounts for camera perspective and road geometry, reducing false detections from off-road regions. This preprocessing ensures that subsequent operations, such as the Hough Transform, operate only on relevant lane areas, improving the robustness of lane detection [1,2,21].

Fig. 7 – Masked image.

This optimization is especially important in the context of CWS, where processing speed is crucial for issuing real-time alerts.

By combining these steps, preprocessing optimizes the input image for more efficient and robust lane detection. A rigorous, well-calibrated preprocessing phase is indispensable to ensure the system's overall performance by minimizing the impact of noise, lighting variation, and artifacts while focusing the analysis on the key area of interest.

3.4. LINES DETECTION USING THE HOUGH TRANSFORM

The Hough Transform is used to detect lines in the image. The basic parameters of the Hough transform represent the straight line and convert it to the image in the parameter space. The relationship between the image space and the Hough space is illustrated in Fig. 8.

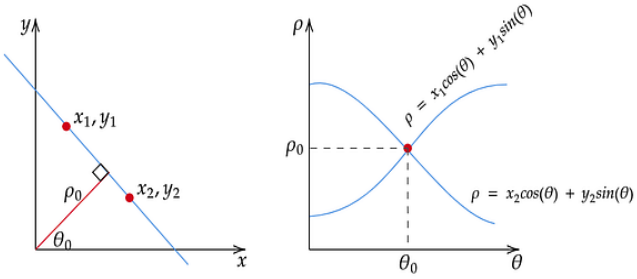

Fig. 8 – Parametric description (domain space and Hough space).

Each edge point (x_i,y_i) in the ROI maps to a sinusoid in Hough space (ρ,θ):

$$\rho = x_i\cos\theta + y_i\sin\theta. \quad (8)$$

Intersections of sinusoids indicate potential lane lines, as shown in Fig. 9. Peaks in the Hough accumulator correspond to strong line candidates, which are projected back to the image as detected lines, $I_L(x,y)$.

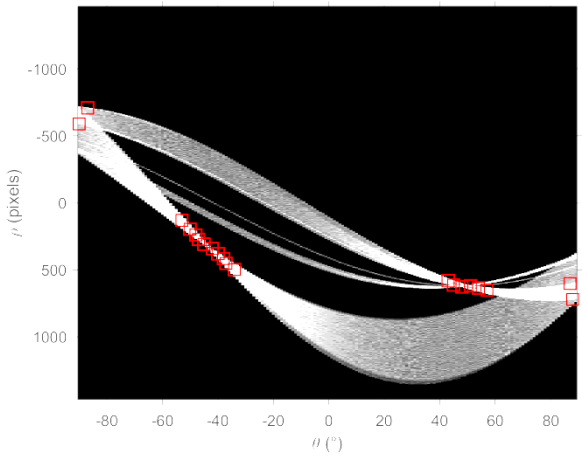

Fig. 9 – Detecting the peak pixel on the Hough transform.

By identifying peaks in the Hough accumulator, the system selects the most likely lines corresponding to lane markings, as illustrated in Fig. 10 [1,2]

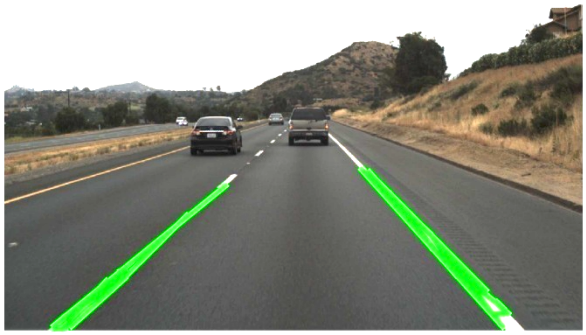

Fig. 10 – Line detection with Hough transform.

3.5. LINE FILTERING AND GROUPING

The lines detected by the Hough Transform may contain noise, duplicates, or irrelevant detections. To refine these results, a filtering and grouping procedure is applied based online parameters, primarily the slope m. Let each line be represented in slope-intercept form:

$$y = m_ix+b_i,\ i = 1,\ldots,N. \quad (9)$$

- Filtering: Lines with slopes outside a plausible range for

lane markings are discarded ($m_{min} \le m_i \le m_{max}$) to exclude overly horizontal or vertical lines.

Outlier lines that do not cluster with others in parameter space are removed.

- Grouping: Lines with similar slope and intercept values are clustered together to represent a single lane marking. Let C_k denote the k-th cluster:

$$C_k=\{(m_i,b_i) \mid |m_i - \dot{m}_k| < \varepsilon_m, \ |b_i - \dot{b}_k| < \varepsilon_b\}$$

The average line parameters of each cluster $(\dot{m}_k,\dot{b}_k)$ define the final lane representation.

This filtering and grouping step ensures a coherent and robust lane representation by removing spurious detections and consolidating multiple detections of the same marking, as illustrated in Fig. 11. [1,2,5].

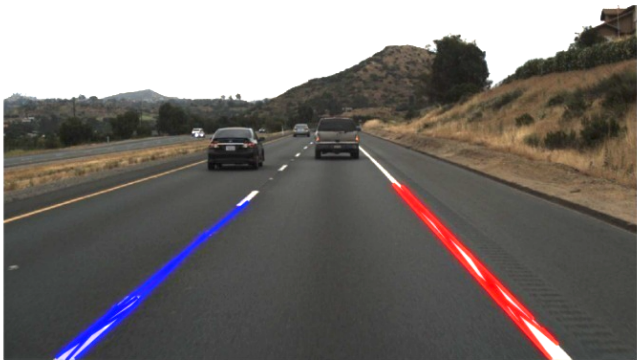

Fig. 11 – Lines filtering and grouping.

3.6. LANE LINE SELECTION: SYMMETRY CRITERIA

To select the most relevant lane boundaries, a symmetry criterion is applied based on the geometric relationship of lines relative to the image center.

Let the image width be W and let the centerline be at x_c = W/2. For each detected line Li, its average horizontal position xi is computed. The symmetry score S(Li,Lj) between two lines Li and L_j is defined as:

$$S(L_i,L_j) = 1 - \frac{|(x_i+x_j)/2 - x_c|}{x_c}. \quad (10)$$

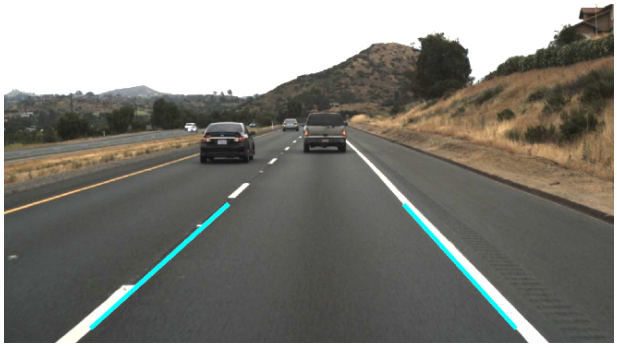

Fig. 12 – Lane boundaries selection.

- $S(L_i,L_j) \in [0,1]$, where 1 indicates perfect symmetry relative to the image center.
- Lines with higher symmetry scores are prioritized as lane boundaries.
- Selection procedure:
- Compute the symmetry score for all line pairs.
- Discard lines with extreme deviation from expected lane width or low symmetry.
- Select the two lines with the highest symmetry score as the left and right lane boundaries.
- Refine line parameters using linear regression to reduce noise.

This approach ensures robust detection even in cases of partially occluded or faded lane markings, while maintaining geometric consistency, as illustrated in Fig. 12.

4. RESULTS AND DISCUSSION

The proposed lane detection algorithm was designed and evaluated on a PC-based experimental platform featuring an Intel i7 1.7 GHz CPU, implemented in Matlab. The input images had a high resolution of 720×1280 pixels, which preserves fine details of lane markings, especially under complex road and lighting conditions.

The algorithm was applied to a set of images from the public TuSimple dataset [23]. The results are illustrated in Figs. 13-15, showing successful lane detection across various road scenarios, including curves, shadows, and partial occlusions.

Table1

The results under different road conditions.

Condition	Number of frames	False Detection Rate [%]	True Detection Rate [%]	Average execution Time [ms]
Straight road, clear markings	500	0.8	99.2	152
Slightly curved road	480	1.66	98.44	263
Faded lane markings	460	2.82	97.28	165
Highly curved road	400	1.7	98.3	290

4.1 COMPUTATION TIME AND ADAPTIVE RESOLUTION STRATEGY

The average computation time per frame measured on the CPU implementation is 150–290 ms, depending on the complexity of the scene and the number of lane markings detected, as summarized in Table 1.

Fig. 13 – Selection of the two best lines in slight curved lane.

Fig. 14 – Selection of the two best lines in a high curved lane.

Fig. 15 –Selection of the two best lines in the faded lane.

The higher computation time compared to standard benchmarks is primarily due to the large input resolution (720×1280 pixels), intentionally chosen to preserve all relevant lane features. To balance detection accuracy and real-time performance, the following adaptive resolution strategy is proposed:

- Initial processing at full resolution to accurately detect fine lane details.
- Resizing images to a lower resolution for subsequent processing stages (Canny edge detection, Hough transform, and line filtering), which significantly reduces computational load. Preliminary tests indicate that this approach can reduce the processing time to below 50 ms per frame, making the algorithm compatible with real-time ADAS requirements.

4.2 COMPARISON WITH RECENT LITERATURE

To place our results in context, Table 2 compares the computation time and platform of the proposed method with recent lane detection studies.

Table 2
Comparison of the proposed method's computation time with recent state-of-the-art lane detection methods.

Method	Platform	Image Size	Avg. Time per Frame
Proposed method	CPU i7 1.7 GHz	1280x720	150 ms
Wu et al., 2019 [6]	Intel Core i7-2th	1280x720	261.1 ms
Son et al., 2019[7]	Intel Core i7-4th	640x360	667.0 ms
Zheng et al., 2018[8]	Intel Core i7-6th	768x432	65.4 ms
Philion, 2019 [9]	NVIDIA GTX 1080, GPU	1280x720	65.3 ms
Zou et al., 2020 [10]	Intel Xeon E5-2nd GTX TITAN-X GPU	1280x720	42.0 ms

4.3 DISCUSSION

The proposed method demonstrates high accuracy across different road conditions, with true detection rates ranging from 97.28% to 99.2% and false detection rates ranging from 0.8% to 2.82%.

The initially higher computation times are justified using high-resolution images (1280×720 pixels), which preserve fine lane features for robust detection.

By employing the adaptive resolution strategy, which processes initially high-resolution images and then downsizes them for subsequent stages, execution times can be significantly reduced without sacrificing detection accuracy. Compared with published benchmarks, the proposed pipeline provides a reliable and robust baseline for lane detection and can be optimized to meet real-time ADAS standards.

4.4 CONTRIBUTION AND ORIGINALITY

This work proposes a robust lane detection pipeline for ADAS applications that combines high-resolution image processing with an adaptive optimization strategy. Unlike many existing approaches, the method initially processes high-resolution images (1280×720 pixels) to preserve fine lane details, improving detection accuracy under challenging road conditions. To address computational constraints, an adaptive-resolution strategy is introduced, enabling subsequent processing at lower resolutions without significant loss of accuracy.

The proposed multi-stage pipeline integrates preprocessing, edge detection, region-of-interest reduction, Hough-based line detection, and symmetry-guided lane selection, ensuring reliable performance across various scenarios. The method is validated on the public TuSimple dataset, achieving high true detection rates and low false detection rates. Although higher execution times are observed due to the use of high-resolution images, practical optimization strategies are identified to meet real-time ADAS requirements. Overall, the proposed approach provides a practical and validated contribution to CPU-based lane detection systems.

5. CONCLUSION AND RECOMMENDATIONS

This paper presented a structured lane boundary detection approach based on classical image processing techniques and the Hough Transform within an ADAS framework. The proposed method combines edge-based detection with geometric constraints, including symmetry and orientation, to select the most representative lane boundaries. The overall design emphasizes robustness, stability, and adaptability to real-world road conditions.

The experimental results show that the method generalizes well across different driving scenarios, including straight roads, curved lanes, and situations involving partially degraded or faded markings. The approach's effectiveness is largely influenced by the image preprocessing stage, where contrast enhancement and noise reduction significantly improve edge visibility before Hough-based line extraction, especially under challenging conditions such as shadows, blur, and uneven illumination.

The algorithm was evaluated using high-resolution images from the public TuSimple dataset. Although processing high-resolution frames increases computational load, this choice preserves fine lane details and contributes to improved detection accuracy. A favorable trade-off between accuracy and execution time is achieved, and several optimization strategies have been identified to enable real-time implementation on embedded or hardware-accelerated platforms.

Future work will focus on extending the proposed framework by incorporating curved line models to better represent complex road geometries, integrating learning-based components to enhance robustness in low-contrast or ambiguous scenarios, and expanding the system toward multi-lane tracking while accounting for dynamic elements such as vehicles and temporary occlusions.

Overall, the proposed approach provides a reliable baseline for camera-based lane detection and offers a

practical foundation for further developments in intelligent transportation systems.

ACKNOWLEDGEMENT

This work was conducted within the Signals and Images Laboratory (LSI) as part of a research project on collision detection systems. The authors would like to express their sincere gratitude to the members of the LSI laboratory for their valuable support, technical assistance, and constructive discussions throughout the development of this research. The authors also wish to thank the reviewers and the editorial board for their insightful comments and suggestions, which significantly improved the quality and clarity of this manuscript.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

YAMINA BELHADRI: writing review & editing, writing original draft, visualization, validation, software, methodology, algorithm design, and conceptualization.

MOHAMED DAHMANI: visualization, validation, supervision, project administration, methodology, and conceptualization.

BELAL ALSHAQAQI: writing original draft, visualization, validation, supervision, software, project administration, methodology, and interpretation.

ABDULLAH SALEM BAQUHAIZEL: visualization, validation, resources, project administration, methodology, and conceptualization.

Received on (month, day, year).

REFERENCES

1. R.C. Gonzalez, R.E. Woods, *Digital image processing*, 4th ed., Pearson, pp. 1–500 (2018).
2. R. Szeliski, *Computer vision: Algorithms and applications*, Springer, pp. 1–400 (2010).
3. N. Otsu, *A threshold selection method from gray-level histograms*, IEEE Trans. Syst., Man, Cybern., **9**, *1*, pp. 62–66 (1979).
4. J.F. Canny, *A computational approach to edge detection*, IEEE Trans. Pattern Anal. Mach. Intell., **8**, *6*, pp. 679–698 (1986).
5. M. Abdelaal, H.M.A. Mohamed, T. Connie, M.K.O. Goh, *Robust lane detection under varying lighting conditions using adaptive vision-based techniques*, Journal of Informatics and Web Engineering, **4**, *3*, pp. 336–355 (2025).
6. Y. Son, E. Lee, D. Kum, *Robust multi-lane detection and tracking using adaptive threshold and lane classification*, Machine Vision and Applications, **30**, pp. 111–124 (2019).
7. C.-B. Wu, L.-H. Wang, K.-C. Wang, *Ultra-low complexity block-based lane detection and departure warning system*, IEEE Trans. on Circuits and Systems for Video Technology, **29**, *2*, pp. 582–593 (2019).
8. F. Zheng, S. Luo, K. Song, C.-W. Yan, M.-C. Wang, *Improved lane line detection algorithm based on Hough transform*, Pattern Recognition and Image Analysis, **28**, *2*, pp. 254–260 (2018).
9. J. Philion, *FastDraw: Addressing the long tail of lane detection by adapting a sequential prediction network*, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, USA, pp. 11574–11583 (2019).
10. Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, Q. Wang, *Robust lane detection from continuous driving scenes using deep neural networks*, IEEE Trans. on Vehicular Technology, **69**, *1*, pp. 41–54 (2020).
11. D. Cacovean, M. Ileana, *Analyse d'imagerie neurologique avancée utilisant des techniques de tenseur de diffusion et des systèmes web distribués*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **69**, *1*, pp. 103–108 (2024).
12. M.B. Priya, C. Ramakrishnan, S. Karthik, *Classification et segmentation de l'écho 3D fœtal utilisant des caractéristiques de couleur et de texture pour la détection de TR*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **69**, *1*, pp. 109–114 (2024).
13. A. Appathurai, A.S.I. Tinu, M. Narayanaperumal, *Détection de tumeurs cérébrales basée sur des images MEG et PET à l'aide de la segmentation Otsu de Kapur et de la classification MobileNet optimisée par la Sooty*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **69**, *3*, pp. 359–364 (2024).
14. K.Y. Chiu, S.F. Lin, *Lane detection using color-based segmentation*, IEEE Proc., Intell. Veh. Symp., pp. 706–711 (2005).
15. C.Y. Chang, C.H. Lin, *An efficient method for lane-mark extraction in complex conditions*, 9th IEEE ICCP, pp. 330–336 (2012).
16. C. Kreucher, S. Lakshmanan, *Lane: A lane extraction algorithm that uses frequency domain features*, IEEE Transactions on Robotics and Automation, **15**, *2*, pp. 343–350 (1999).
17. S. Habib, M.A. Hannan, *Lane departure detection and transmission using Hough transform method*, Przeglad Elektrotechniczny, **89**, *5*, pp. 141–146 (2013).
18. V. Hardzeyeu, F. Klefenz, *On using the Hough transform for driving assistance applications*, 4th IEEE ICCP, pp. 91–98 (2008).
19. J.W. Lee, U.K. Yi, *A lane-departure identification based on LBPE, Hough transform, and linear regression*, Computer Vision and Image Understanding, **99**, *3*, pp. 359–383 (2005).
20. Y. Wang, E.K. Teoha, D. Shen, *Lane detection and tracking using B-Snake*, Image and Vision Computing, **22**, *4*, pp. 269–280 (2004).
21. K.A. Redmill, S. Upadhya, A. Krishnamurthy, Ü. Özgüner, *A lane tracking system for intelligent vehicle applications*, IEEE Intelligent Transportation Systems, Oakland, pp. 273–279 (2001).
22. Y.U. Yim, S.-Y. Oh, *Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving*, IEEE Transactions on Intelligent Transportation Systems, **4**, *4*, pp. 219–225 (2003).
23. ***TuSimple Lane Detection Challenge, pp. 1–10 (2017).