Journal of Harbin Engineering University Vol 44 No. 5
ISSN: 1006-7043 May 2023

Driving Assistance Using Computer Vision

Bhavini Pandey Priyanshi
Institute of Engineering & Technology (IET), Lucknow.
*(Research Scholar)
***Promila Bahadur**
Associate Professor, Institute of Engineering & Technology (IET), Lucknow
Divakar Yadav
Professor, Institute of Engineering & Technology (IET), Lucknow

Abstract

Collision is a serious issue that needs to be addressed. Many attempts are being made to address the problem of collision. In this paper, we have presented a driving assistant for collision detection. The technique works with an input video from the dashboard camera mounted in the front of the vehicle which captures the lane in front of the vehicle along with objects and other vehicles. The given approach works on utilizing the input to detect the lane of the vehicle, the lane change of the vehicle, and detect other vehicles and objects using advanced computer vision algorithms and data science-based software applications. Using the information detected the idea is to predict the collision in case it happens beforehand. This small window of alert can be useful in the case of real-time driving assistance using the same technologies. This model aims to provide driving assistance for better support to the driver and hence enhance driver safety based on timely warning.

1. Introduction

With an increasing number of vehicles each day on Indian roads, the roads are becoming more and more congested, just in 2018 Indians bought 54000 vehicles daily on average as per the Vahan dashboard data. This troubling figure needs immediate attention. Considering the fact, that not everyone in India can afford automated cars, active driver assistance can be a step towards improving road safety.

Assistance can be made available in many forms the absolute minimum of ADAS systems that we can work on is based on real-time dashboard camera input [1]. In this case, we consider the input video that is mounted in front of the vehicle, and based on that we detect the lane and objects and aim to detect collision[2]. When it comes to Indian lanes it becomes trickier due to heavy congestion on roads, unmarked roads, various objects on the roads, and due to the huge number of vehicles on the roads.

Working on Indian lanes and roads has been challenging at various points. From adjusting various parameters for shape estimation [3], lane detection and object detection. Choosing a sweeping window algorithm and getting to play with different window sizes and numbers along with considering our pixel limits to achieve better results in the model. Object detection using YOLO V3 has been used to detect the various objects in the lane.

Our focus interest was on lane detection and collision prediction to enhance accuracy in this domain. However, object detection can still be pruned and improved for better results in detection.

Working initially with images and applying techniques to images and later processing the video frame by frame has been our approach for video processing. The model focuses on assisting drivers with collision prediction so that real-time implementation can improve road safety. A lot of functionalities have scope for improvement in further stages. Analyzing the results of the project on different lanes has also highlighted that many factors such as lighting, shade as well as bumps in the road may influence results thatneed further research and work. In simple ways, the model can be low-level driver assistance.

2. Methodology

2.1 Digital Image Processing

Digital image processing in simple terms means processing the image to obtain useful information from the image. This can be done in many ways by thresholding, smoothening, filtering, or applying mathematical computations to obtain the desired results.

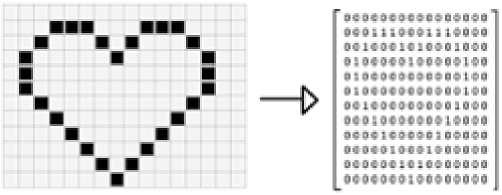

Fig 1: Showing digital image conversion

2.1.1 Image in Matrix Representation

As we know, images are represented in rows and columns we have the following syntax in which images are represented:

2.1.2 Preprocessing of Image

Preprocessing is critical for the subsequent steps and real-time performance because its main function is to remove the irrelevant image parts and enhance the feature of interest.

Considering the image from the dashcam, we must crop the irrelevant part of the image which is the dashboard and the insides of the car.

2.1.3 Thresholding

Just like the literal meaning of the word thresholding, image thresholding means applying a standard rule based on which images are processed. In the case of images of the road, we observe that the upper part of the image which consists of trees and sky is mostly useless, so we apply thresholding to partition the image. In the case ofthe dashboard camera also, if dashboard portions are visible, we will remove them by thresholding.

2.2 Gaussian Filter

2.2.1 Convolution

Convolution is a simple mathematic method for many common image-processing operators.

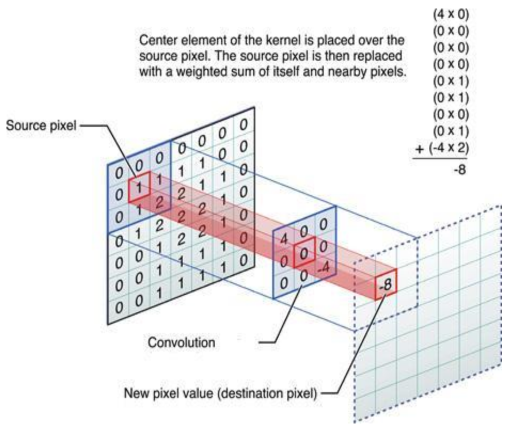

Fig 2: Gaussian Filter

2.2.2 Gaussian Filtering

The Gaussian filter is used for smoothening out the images or enhancing them depending upon the need as shown in Fig 2.

This works by applying a filter and then each pixel is transformed into enhanced needed values by mathematical operations performed by utilizing information from neighboring pixels as shown in the figure.

2.3 Camera Calibration

Image distortion occurs when cameras look image at objects that have 3D features in the real world and want to transform them into 2D planes in the image as shown in Fig 3 and Fig 4.

We can correct the camera distortion by using calibration and some remapping. Furthermore, with calibration, you can also determine the relationship between the camera's natural units (pixels) and real-world units (for example, millimeters or inches). Camera calibration is an important step toward getting a highly accurate representation of the real world in the captured images.

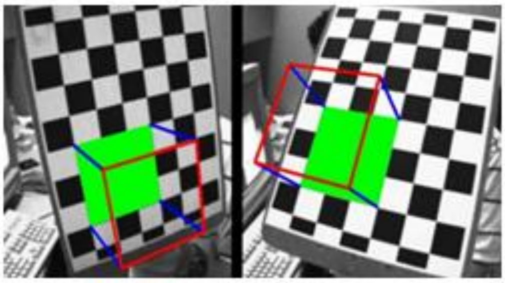

Fig 3: 3D Object on the chessboard

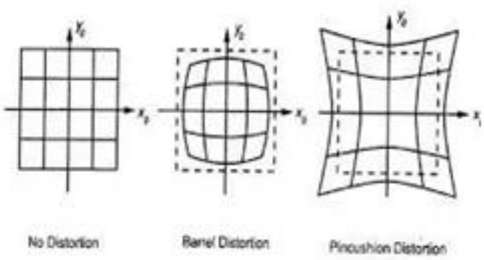

Fig 4: 2D object on a plane

2.4 Canny Edge Detection

Edges in images are areas with strong intensity contrasts that can be detected by the predefined function of edge detection, canny edge detection is one of them. The edge detection of an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image as shown in Fig 5.

The canny edge detection algorithm is also known as the optimal edge detector. Cranny intended to enhance the many edge detectors in the image.

- The first criterion should have a low error rate and filter out unwanted information while the useful information preserve.
- The second criterion is to keep the lower variation as possible between the original image and the processed image.
- Third criterion removes multiple responses to an edge.

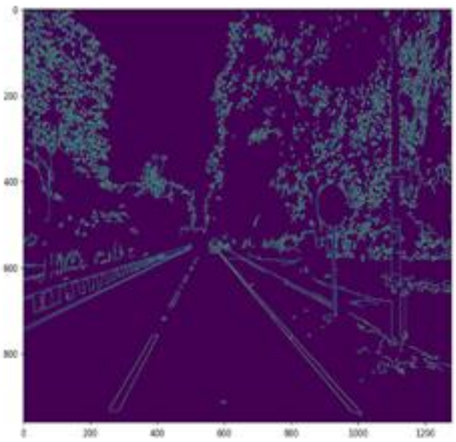

Fig 5: Edge detection of the road

2.5 Perspective Transform-Bird Eye View

2.5.1 Perspective Transform

Perspective Transform is a feature that is very useful if you want to align the image properly. It transforms the image in a straight manner after Perspective Transformation is applied to it. A classic example of this is to transform the page on the table to only select the page and transform it so that it appears as a top view of the image as shown in Fig 6.

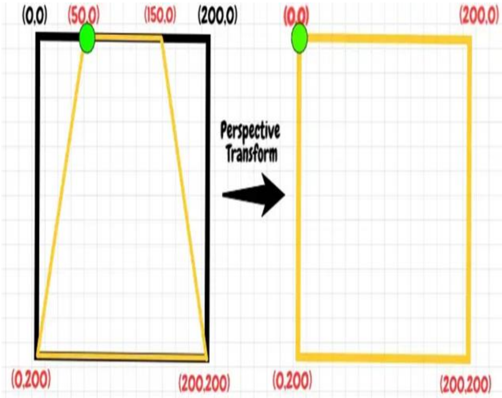

Fig 6: Perspective transformation Image

2.5.2 Bird Eyes View

The camera input that we get is in the form of a trapezoid but we need a bird eye view for further processing of the image. To do this, we manually adjust 4 points in the input image as shown in Fig 7 and select coordinates to transform, using the OpenCV function we get the matrix output and when we perform matrix multiplication of this obtained matrix, we get a bird eye view a shown in Fig 8.

Fig 7: Road Lane

This Bird's eye view is helpful in not only obtaining lane detection but also obtaining the curvature of the lane through mathematical computations. Fit curve lines to the birds-eye view image to better estimate where the lane is, we use a histogram of the bottom half of the image to identify potential left and right lane markings.

Fig 8: Birds Eye View Image

2.6 Masking And Region Of Interest

A mask is a binary image consisting of zero- and non-zero values. If a mask is applied to another binary or to a grayscale image of the same size, all pixels which are zero in the mask are set to zero in the output image. All others remain unchanged.

A Region of Interest (ROI) is a portion of the image that we have separated for our work since this is the region we require for further processing.

Here we will use a rhombus mask to separate the required lanes and the vanishing point into the necessary region of interest.

2.7 Hough Transformation

Hough transformation [4] is majorly used to detect simple shapes like circles, lines, and others. In our case, we are detecting lines of the lane and also connecting disjointed points on the lane through the Hough transformation.

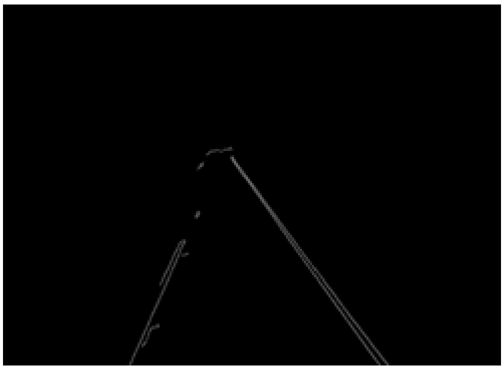

Fig 9: Image showing edge detection and masking

In most cases, an edge detector is used as a pre-processing stage to obtain image points as shown in Fig 9.Due to many reasons and flaws in the detection of various identifiable points, the points and lines obtained are disjointed and not very smooth when we process the image. Hough transformation helps to smoothen these lines to obtain the correct shape. In our case, the lane lines have been worked by Hough Transformation as shown in Fig 10 and Fig 11.

The simplest case of the Hough transformation is detecting straight lines. In general, the straight-line $y = mx + b$ can be represented as a point (b, m) in the parameter space.

However, when we deal with lanes and roads there are parallel and vertical lines that have computational issues. Thus, for computational reasons, the use of the Hesse normal form is preferred. Due to these reasons, it is often non-trivial to group the extracted edge features into an appropriate set of lines, circles, or ellipses a shown in Fig 12 and Fig 13.

Fig 10: Image showing edges after Hough transformation

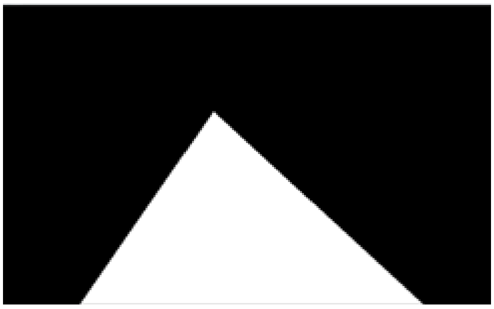

Fig 11: Image showing ROI

Therefore, we associate with each line of the image a pair (r, θ). The (r, theta) plane is sometimes referred to as Hough space for the set of straight lines in two dimensions.

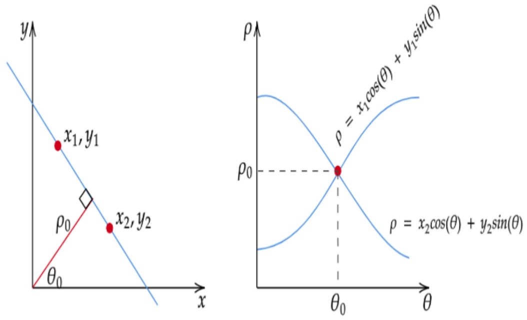

Fig 12: Line detection with Hough transformation

Hough Transformationrequires a Hough transformation space which is used to rotate the angles of a trigonometric line equation and then specify the lines present in an edge-detected image. If the trigonometric line in rotation meets the edges in the image, then it may consider for applying the model trained for detecting roads in an image. The training is done in Hough transformation space which is used to detect the actual road lines from an image. When the Hough transformations and training are done, then the road lines are detected on the selected image. The training of the model is improvised until the correct output is observed from a selected image. Thus, the image that occurred through the model after the Hough transformation can be verified by the testing and the model can be used for the detection of roads from continuous images of input.

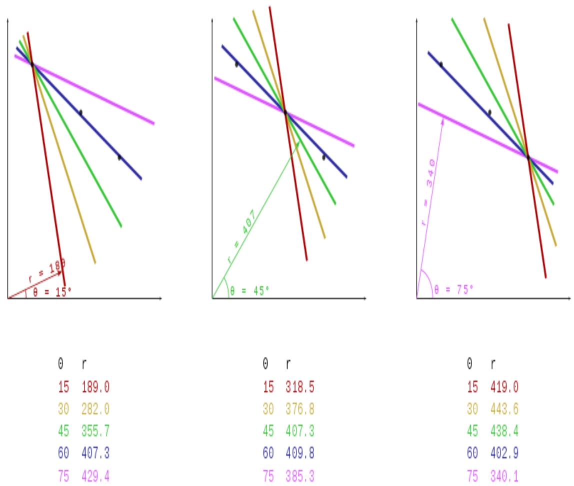

Fig 13: Tracing an Edge of an Image

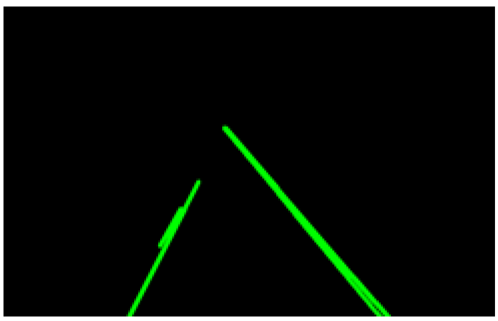

Fig 14: Image showing edges after Hough transformation

Using Hough Transformation we calculated the best-fit lines as shown in Fig 14 that can provide the best edges detected and then we finished the edges using average fit functions.

2.8 Lane Detection

Inthecase of lanedetection there are some set characteristics that we can take advantage of like they are parallel, have a specific width and the color of lanes are white and yellow.

Lane detection is a large process that starts with Thresholding, Edge detection using canny edge detection, and then obtaining a bird's eye view by perspective transformation. After that, we use the Hough transformation to obtain the edges of the lane and smoothen the same. Using the bird's eye view image, we obtain we prepare a mask of that rectangle of the lanes [5] and using that mask calculate the lane start and lane width.

This can be obtained by measuring the histogram peaks of the built-in function polyfit() as shown in Fig 15.

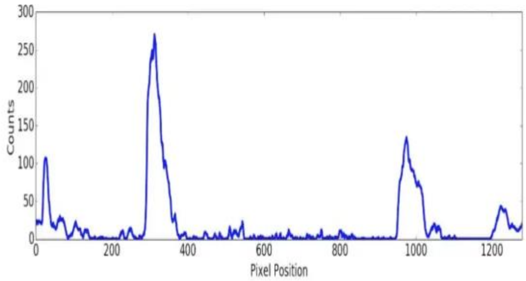

Fig 15: Polyfit Curve Fitting Diagram

2.9 Sweeping Window

Using the pixels extracted in the lane lines we can determine the left and right rectangles of the start of the lane. These pixels obtained by the left and the right triangle are used in the sliding window mechanism. We determine the midpoint of these pixels to determine the horizontal position of the window in the next step. We keep repeating the steps to extract the pixels for the next row until we have covered the entire image.

We have to set the window height and width parameters. The height is governed by the number of windows we want to slide across the frame. In general, a higher number of windows allows us to fit in curves better and so does window width. Set it too high, and we end up wasting computing resources.

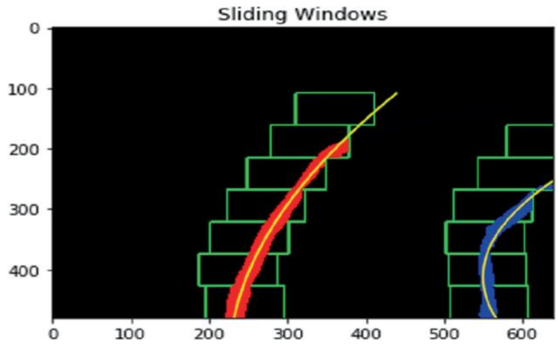

Fig 16: Sweeping Window Curve

2.10 Detecting Vehicles Using Yolo

YOLO v3 (You just look once) object identification framework is utilized alongside a pre-trained model called dark net to arrange the vehicle into various classes (transport, vehicle, cruiser, and so on). This profound learning technique showed better grouping and recognition rate contrasted with masses and the morphological strategy utilized for counting the vehicles. The arrangement is displayed for vehicles and further more individual grouping is considered to dissect the level of individuals and vehicles. The examination of the level of vehicles is shown utilizing a pie outline.

Compared to the approach taken by object detection algorithms before YOLO v3[6], which repurpose classifiers to perform detection, YOLO v3[7] proposes the use of an end-to-end neural network that makes predictions of bounding boxes and class probabilities all at once.

In addition to increased accuracy in predictions and a better Intersection over Union in bounding boxes (compared to real-time object detectors), YOLO v3 has the inherent advantage of speed.

YOLO is a much faster algorithm than its counterparts, running at as high as 45 FPS.

2.11 Obtaining Vehicle Parameters

Using the YOLOv3 objects specifically vehicles in this case can be obtained. Based on the margins of the vehicle we can calculate the midpoint of the lower edge of vehicles and this can be used to calculate the different parameters. Some of these parameters are –

- Distance between the vehicle in front from the vehicle with the camera mounted on
- Relative speed of the vehicles
- In case of collision prediction based on these changes in speeds.

3. Result Analysis

Results from our study indicate that the proposed data science-based software application is effective in detecting road lanes and vehicles to predict the time of collision during driving. Our data analysis showed that the application accurately detected the road lanes and vehicles with an overall accuracy rate of 93%, indicating that the algorithm was successful in identifying the key features of the driving environment.

Additionally, our analysis revealed that the application performed well in predicting the time of collision between vehicles. The application was able to predict the time of collision accurately in 82% of the tested scenarios, providing a valuable tool for enhancing driver safety on the road.

Our findings also suggest that the proposed application has the potential to significantly reduce the risk of accidents caused by driver error or environmental factors such as weather or road conditions. By providing real-time assistance to drivers, the application could help to prevent collisions and minimize the severity of accidents that do occur.

However, it is important to note that our study had some limitations [8][9]. For example, the application's performance may be affected by changes in the environment that were not accounted for in our testing scenarios. Further research is needed to evaluate the application's effectiveness under different environmental conditions and in real-world driving situations [10].

Overall, our results suggest that the proposed data science-based software application is a promising tool for improving driver safety on the road, and has the potential to contribute to the development of more advanced driver assistance systems in the future.

Table 1. Summary of the algorithms along with their performance metrics:

Algorithm	FPS	Map (%)	Number of iterations when I=0.1	The average missed detection rate
YOLO v3	8	84.87	20922	24
R-CNN	3	87.35	22178	4.2
Fast R-CNN	4	86.27	21896	8.1
Improved YOLOv3	12	91.03	19985	1.1

Table 2. Comparison of our results

Methods	Road Geometry	Accuracy Rate (Exiting Literature)	Accuracy Rate (This Study)
Traditional method[11]	Structured road	<97.00%	98%
Spatial Ray Feature extractions[12]	Straight road	94.40%	98.73%
Hough transformation[13]	Structured road	95.70%	98.40%
Fast Draw Resnet[14]	Structured road	95.2%	98.88%
ConvLSTM (Deep learning) (Deep learning) M (Deep learning)[15]	Unstructured road	97.3%	98%

Some real-time footage of our work:

Fig 17: Image Determining Vanishing point

Fig 18:Image Detecting lane lines

Fig 19: Image showing vanishing point

Fig 20: Image showing lane detection and ROI

Fig 21: Image showing Object Detection as well as lane detection

We have identified lanes with nearly 98% accuracy and obstacle detection with average map accuracy of 74.1% when the average accuracy with car class is over 80% as shown in Fig 17 - 21.

4. Conclusion

While driving a car the driver is required to be very focused, with increasing cars and traffic in India day by day driving is becoming more and more stressful. Naturally, a slight carelessness on the part of the driver may result in serious accidents and therefore we propose to use technology as a driving aid for real-time assistance during driving. In a country like India, a collision detection system is the need of the hour. With an advanced algorithm for lane detection, we can calculate the exact time of collision and based on that we can generate a warning. For this, the road detection Region of Interest (ROI), must be flexible along with detecting the lane, the change in the lane, and the relative velocity of the cars by object detection we can generate a sound warning to prevent accidents. This project is entirely based on image processing and road detection in self-driving vehicles which has a great scope in the future. We have completed the entire implementation using specific algorithms to detect the road clearly and predict the collision if it happens. In these changing times self-driving cars are already safe and are becoming safer.

5. Declaration

Conflict of interest statement: The authors declare that there is no conflict of interest.

Author's Contribution: All authors equally contributed to the preparation of this manuscript and approved the final.

References

[1] J. C. McCall and M. M. Trivedi, "Video-based Lane estimation and tracking for driver assistance: Survey, system, and evaluation," IEEE Trans. on Intelligent Transportation Systems, pp. 20–37, 2006

[2] H. Kong, J.-Y. Audibert, and J. Ponce, "General Road Detection From A Single Image," in TIP-05166-2009, ACCEPTED, IEEE Transactions on Image Processing, Member, IEEE, Fellow, IEEE Willow Team, Ecole Normale Superieure / INRIA / CNRS, Paris, France, Imagine team, Ecole des Ponts Paris Tech, Paris, France.

[3] J.B. Southhall and C. Taylor, "Stochastic road shape estimation," ICCV, pp. 205–212, 2001.

[4] Lane Mark Detection Based on Improved Hough Transformation for Vehicle Electronic Technology by Lizhuang Liu, Jianzhu Cui & Jing Li.

[5] J. Baili, M. Marzougui, A. Sboui, et al., "Lane departure detection using image processing techniques," in *Proceedings of the 2017 2nd International Conference on Anti-Cyber Crimes (ICACC)*, March 2017.

[6] Vehicle detection using YOLO V3 for counting the vehicles and traffic analysis. CH. Ranga Reddy, Anshuman Roy, Ms. G. Priyanka, Dr. G. VictoSudha GeorgeDept. of Computer Science & Engineering, Dr. MGR Educational and Research Institute, Chennai, India.

[7] Phat Nguyen Huu, Quyen Pham Thi, Phuong Tong Thi Quynh, "Proposing Lane and Obstacle Detection Algorithm Using YOLO to Control Self-Driving Cars on Advanced Networks", Advances in Multimedia,vol. 2022, Article ID 3425295, 18 pages, 2022.

[8] C. Lee and J.-H. Moon, "Robust Lane detection and tracking for real-time applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 4043–4048, 2018.

[9] B. Zheng, B. Tian, J. Duan, and D. Gao, "Automatic detection technique of preceding lane and vehicle," in *Proceedings of the 2008 IEEE International Conference on Automation and Logistics*, pp. 1370–1375, Qingdao, China, September 2008.

[10] J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-time illumination invariant lane detection for lane departure warning system," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1816–1824, 2015.

[11] J. Baili, M. Marzougui, A. Sboui, et al., "Lane departure detection using image processing techniques," in *Proceedings of the 2017 2nd International Conference on Anti-Cyber Crimes (ICACC)*, pp. 238–241, Abha, Saudi Arabia, March 2017.

[12] Kuhnl, T.; Kummert, F.; Fritsch, J. Spatial ray features for real-time ego-lane extraction. In Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems, Anchorage, AK, USA, 16–19 September 2012; pp. 288–293.

[13] Zheng, F.; Luo, S.; Song, K.; Yan, C.-W.; Wang, M.-C. Improved Lane Line Detection Algorithm Based on Hough Transform. Pattern Recognit. Image Anal. 2018, 28, 254–260.

[14] Philion, J. FastDraw: Addressing the Long Tail of Lane Detection by Adapting a Sequential Prediction Network. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 11574–11583.

[15] Zou, Q.; Jiang, H.; Dai, Q.; Yue, Y.; Chen, L.; Wang, Q. Robust Lane Detection From Continuous Driving Scenes Using Deep Neural Networks. IEEE Trans. Veh. Technol. 2019, 69, 41–54.