ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 11 Issue 3 March 2024

Lane and Curve Detection Using Deep Learning

[1] Arindol Das, [2] Soumili Dutta, [3] Prodipta Bhowmik, [4] Aparajita Bandyopadhyay, [5] Yash Kumar

[1] [2] [3] [4] [5] Department of IT, Techno Main Salt Lake, Kolkata West Bengal

Corresponding Author Email: [1] arindoldas@gmail.com, [2] duttasamiran518@gmail.com, [3] prodipta.infotech2021@gmail.com, [4] aparajitabandyopadhyay5@gmail.com, [5] kryash90@gmail.com

Abstract— ***We aim to design an accurate and robust lane detection system, especially curved lane detection, to increase safety and reduce accidents on structured roads. Intelligent cars may dramatically lower the number of traffic collisions and our focus has been to design an algorithm as accurate as possible to make vehicles intelligent.***

In this project, an improved lane detection system using Convolutional Neural Networks (CNN) is proposed. As part of the pre-processing, different filters are applied on the image dataset like Gaussian Blur to remove noise, Sobel and Laplacian operator to calculate the gradient followed by Canny Edge detection. The aim remains to detect the lane pixels, determine the curvature of the lane (curves) and vehicle position with respect to the center. The detected lane boundaries are then wrapped back onto the original images of roads to highlight the lanes and curves accurately. A numerical estimation of the lane curvature is also displayed.

Index Terms— ***Lane detection, curved lane detection, Normalization of pixel, Gaussian filter, Canny Edge detection, Convolutional Neural Networks (CNN).***

I. INTRODUCTION

Lane and curve detection are crucial components in various advanced driver assistance systems (ADAS) and autonomous vehicles. They serve several important purposes:

Safety: Lane detection helps vehicles stay within their designated lanes, reducing the risk of collisions due to unintentional drifting or veering off the road. It provides warnings to the driver or autonomously assists in keeping the vehicle centered within the lanes.

Navigation: Knowing the lanes and curves in the road helps in accurate navigation. It allows the vehicle to anticipate upcoming turns, curves, and intersections, enabling better route planning and decision-making.

Autonomous Driving: For self-driving vehicles, lane and curve detection is essential. These technologies enable autonomous cars to understand the road layout, make decisions about lane changes, turning, merging, and overall maneuvering while ensuring safety and compliance with traffic rules.

Accident Prevention: By detecting lanes and curves accurately, vehicles can prevent accidents caused by human error, fatigue, distractions, or poor visibility. Systems can alert drivers accordingly.

In our project, we have used lane detection techniques like canny edge detection followed by the Convolutional Neural Network algorithm. The advantages of employing deep learning for lane and curve detection lie in its ability to handle complex visual data, adapt to diverse road conditions, and improve accuracy over traditional rule-based methods. Moreover, these algorithms can continuously learn and improve with more exposure to varied driving environments, contributing to the ongoing evolution of safer and more efficient transportation systems.

II. RELATED WORK

Z. Wang, Y. Fan and H. Zhang [2], proposed that to detect lanes, we need to detect the edges from the image. Sobel Method and Canny Method are the most extensively used methods for edge detection. The Sobel operator scans an image, looking for areas where the brightness changes rapidly in different directions. These rapid changes usually happen at the edges of objects, so the Sobel operator highlights those edges, making them easier to see. Disadvantage of Sobel operator could be sensitivity to the noise. The magnitude of the edges would degrade as the level of noise present in image increases. As a result, Sobel operator accuracy suffer as the magnitude of the edges decreases.

Maya, P., and C. Tharini [3], used a partial Hough parameter space for detecting lanes and the approach was verified with different image sets. This lane detection method starts by focusing on a specific area of the image (region of interest), then converts it to grayscale to remove color information. Next, it identifies edges within the grayscale image. Finally, it uses a special technique called "proposed partial Hough transform" to pinpoint the actual lane lines among the detected edges. Disadvantages could be: Hough transform can be computationally expensive, making it less suitable for real-time applications. The Hough transform could be sensitive to noise in the image, leading to potential false detections.

The sliding window algorithm was proposed by Keerti Chand Bhupathi and Hasan Ferdowsi [4]. The algorithm used to perform localized analysis on an image or a signal. It involved systematically moving a fixed-size window across the image, processing the contents of the window at each position. It has some few limitations. They were computationally expensive to process large images with small window sizes, and they may miss small or partially occluded objects.

A robust lane detection from continuous driving scenes using deep neural networks was proposed by J. Wu, H. Cui and N. Dahnoun [5]. The authors proposed a hybrid deep learning architecture that combines the Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN). The CNN block was used to extract information from each frame, while the RNN block was used to learn features from the CNN features of multiple continuous frames. Disadvantage could be: training CNNs for lane and curve detection required a large dataset with accurately labeled lane and curve markings. Generating such datasets can be challenging.

The usage of the computer vision concepts using the OpenCV library was proposed by S. Singh, S. Malik and R. K. Nath [10]. The advantages were: open-source, a cross-platform library that could be used in real-time applications. It was also useful in object detection and image processing and supported deep learning frameworks. Disadvantage could be: Many algorithms in OpenCV require manual parameter tuning]. OpenCV's methods might not adapt well to different road types, lane markings, or traffic conditions.

S. Pokale and D. V. Niture [12], proposed the idea of creating an algorithm that could draw the edges of any object in the image, using the canny edge detection algorithm. It was an algorithm with multiple stages like noise reduction, finding intensity gradient of the image, non-maximum suppression and hysteresis thresholding. Disadvantage could be: Canny edge detection could be sensitive to noise in the image, which could result in false edges being detected, impacting the accuracy. In complex road scenes, Canny edge detection might detect multiple edges for a single lane or curve.

A novel lane detection method using deep learning and SSIM method was proposed C. Ren, X. Huang and H. Ogai [13]. This method can find lanes in two ways at once, each using deep learning. It then compares the results with real lane markings captured on camera (ground truth) using a special image similarity measure called SSIM. The most accurate match wins and becomes the final output. Tests show that this method works well in different challenging situations, both finding lanes accurately and doing so quickly. There are some limitations of SSIM when used with medical images uniform pooling, distortion underestimation near hard edges, instabilities in regions of low variance and insensitivity in regions high intensities.

N. Sukumar and P. Sumathi [14], proposed an algorithm Random Sample Consensus (RANSAC) to develop a lane detection algorithm. It includes steps like selecting Fewer fitting points randomly from the whole point set. Then the model was made fit based on the selected points by the least-square method. First step is repeated until the maximum iterative number is reached. This method improved fitting efficiency. Disadvantage could be: randomly selecting fitting points from whole feature points is a bad strategy. RANSAC required the setting of certain parameters, such as the maximum distance threshold and the number of iterations. Finding the optimal parameter values can be challenging and may require experimentation.

A convolutional neural network-based method for recognizing lanes in driving video was proposed by N. J. Zakaria, M. I. Shapiai, R. A. Ghani, M. N. M. Yassin, M. Z. Ibrahim and N. Wahid [15]. The expectation line represented an autonomous vehicle's driving behavior in greater detail. Using the long short-term memory-based approach, the predicted line was then used to estimate the vehicle's future trajectory. Due to prior information, autonomous cars may drive smoothly by combining a convolutional neural network with long short-term memory-based techniques (LSTM).

The deep segmentation methods represent lane lines as segmented binary features, which greatly limit the speed increase. To solve this problem, scholars from Zhejiang University proposed a new detection method that explicitly used the prior information of lane lines. Their method is proposed to select locations of lanes at predefined rows of the image using global features instead of filtering each pixel feature, which significantly improves detection speed [7]. However, this detection method cannot detect curved lane lines.

Jinsheng Xiao, Wenxin Xiong, Yuan Yao, Liang Li, Reinhard Klette[8] in their paper, implemented a lane detection technique using the structure of the road and an extended Kalman filter. In this method, the lane parameters such as the lane markings are detected using the coordinates of the lane boundary points.

Mohamed Fakhfakh, Lotfi Chaari, and Nizar Fakhfakh [9] In their paper, they proposed a vision-based lane detection method by using the Bayesian framework for the estimation of multi hyperbola parameters.

III. METHOD

Lane and curve detection can be achieved in mainly two broad ways : using OpenCV libraries for edge detection or by using deep learning methods like CNN. While deep learning algorithms yield more accurate results, they need humongous datasets to be trained. On the other hand, using OpenCV libraries makes the process of object identification faster and less dependent on datasets, but accuracy and a low loss function cannot be assured.

Thus, in this project, we propose to use the OpenCV libraries in pre-processing steps to simplify our input data followed by feeding the simplified, labelled input dataset into deep learning algorithms like CNN.

A. Step 1: Data Collection

To gather a diverse dataset of annotated images or video

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 11 Issue 3 March 2024

sequences with curve lane labels [11].

B. Step 2: Data Preprocessing

To prepare the data to be used for training and evaluation: RGB images are converted into grayscale images

Gradient Calculation using Sobel's Operator: The gradient magnitude and direction are computed at each pixel to identify areas of rapid intensity changes. Sobel's operator uses first order derivative to achieve this.

Gradient Calculation using Laplacian Operator: The Laplacian operator uses second order derivative to identify areas of rapid intensity changes.

Gaussian Smoothing: The image is first convolved with a Gaussian filter to reduce noise and eliminate small variations in pixel intensity.

The pre-processed image is passed through the Canny edge detection algorithm. This involves the following steps:

- Non-Maximum Suppression: Only local maxima in the gradient direction are kept, while non-maximum values are suppressed.
- Double Thresholding: The gradient magnitude image is thresholded to classify pixels as strong edges, weak edges, or non-edges.
- Edge Tracking by Hysteresis: Weak edges that are connected to strong edges are retained as part of the edge.

C. Step 4: Labelled dataset generation

Region Of Interest masking is achieved using bitwise AND operator to focus only on the required portion of captured frame

Hough transform is applied over the images to detect any shape, if that shape can be represented in mathematical form

Detected lines are appended to obtain lane boundaries which are in turn mapped back on the images

D. Step 5: Model Selection

The model would be trained using the CNN architecture. We are going to a FCN (Fully Convolutional Network) model for creating a binary segmentation map for detecting each lane pixels.

E. Step 6: Model Training

Once we obtain the labelled dataset, we can train the CNN model. Experimenting with different hyperparameters (learning rate, batch size, etc.) and employing techniques like early stopping and learning rate scheduling can prevent overfitting and improve convergence.

F. Step 7: Model Evaluation

To evaluate the trained model on the validation set to monitor its performance.

G. Step 8: Testing and Validation

To evaluate the final model on the test set and real-world data.

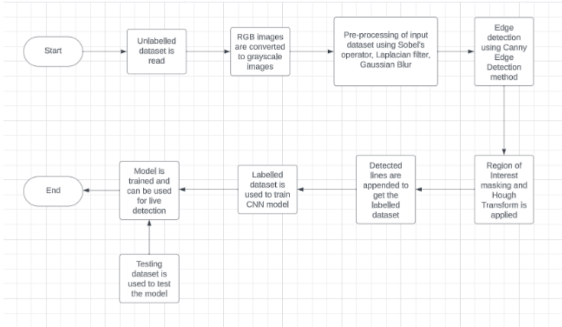

Fig. 1. Workflow of proposed procedure

IV. CONCLUSION

The combination of traditional computer vision techniques and deep learning proved to be a robust approach for curve and lane detection, leveraging the strengths of both methodologies. In the field of autonomous vehicles and self-driving cars, detection of lane and detection of curve on the lane plays an important role. A self-driven car needs to understand its surroundings so that it can navigate its way on roads with minimal human assistance. Also increase in number of vehicles in recent years, led to increase in traffic collisions and accidents. The achieved accuracies indicate the potential of a developed lane and curve system for real-world applications, can assist vehicles to detect lanes and curves accurately which can help in significant reduction of traffic complications and accidents. Also, further optimizations and fine-tuning could potentially enhance the model's performance even more, making it a valuable tool for tasks related to road scene understanding and navigation.

REFERENCES

[1] J. Ciberlin, R. Grbic, N. Teslić and M. Pilipović, "Object detection and object tracking in front of the vehicle using front view camera," 2019 Zooming Innovation in Consumer Technologies Conference (ZINC), Novi Sad, Serbia, 2019, pp. 27-32, doi: 10.1109/ZINC.2019.8769367

[2] Z. Wang, Y. Fan and H. Zhang, "Lane-line Detection Algorithm for Complex Road Based on OpenCV," 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2019, pp.1404-1407, doi:10.1109/IMCEC46724.2019.8983919.

[3] Maya, P., and C. Tharini. "Performance analysis of lane detection algorithm using partial Hough transform." 2020 21st International Arab Conference on Information Technology (ACIT). IEEE, 2020.

[4] Keerti Chand Bhupathi, Hasan Ferdowsi "An Augmented Sliding Window Technique to Improve Detection of Curved Lanes in Autonomous Vehicles" 2020 IEEE International Conference on Electro Information Technology (EIT) ISSN: 2154-0373

[5] J. Wu, H. Cui and N. Dahnoun, "Gradient Map Based Lane Detection Using CNN and RNN," 2020 22th International Conference on Digital Signal Processing and its Applications (DSPA), Moscow, Russia, 2020, pp. 1-5, doi: 10.1109/ DSPA48919.2020.9213294.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 11 Issue 3 March 2024

[6] Yin, R., Yu, B., Wu, H., Song, Y., & Niu, R. (2020). FusionLane: Multi-Sensor Fusion for Lane Marking Semantic Segmentation Using Deep Neural Networks. ArXiv, abs/2003.04404.

[7] Qin, Zequn, H. Wang and X. Li. “Ultra Fast Structure-aware Deep Lane Detection.” ArXiv abs/2004.11757 (2020): n. pag.

[8] Lane Detection Algorithm Based on Road Structure and Extended Kalman Filter Jinsheng Xiao, Wenxin Xiong, Yuan Yao, Liang Li, Reinhard Klette June 2020.

[9] Bayesian curved lane estimation for autonomous driving, Mohamed fakhfakh, Lotfi chaari, Nizar Fakhfakh, January 2020

[10] S. Singh, S. Malik and R. K. Nath, "Lane and Curve Detection using Deep Learning," 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2021, pp. 331-333, doi: 10.1109/ICAC3N53548.2021.9725528.

[11] TuSimple dataset, 2021. Available: https://www.kaggle.com/ datasets/manideep1108/tusimple

[12] S. Pokale and D. V. Niture, "On-board Camera-based Lane and Traffic Sign Detection for two Wheeler HeadLamp," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-6, doi: 10.1109/I2CT54291.2022.9824519.

[13] C. Ren, X. Huang and H. Ogai, "Lane Detection Based on Deep Learning and SSIM Method," 2022 14th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, China, 2022, pp. 48-51, doi: 10.1109/IHMSC55436.2022.00020.

[14] N. Sukumar and P. Sumathi, "A Robust Vision-based Lane Detection using RANSAC Algorithm," 2022 IEEE Global Conference on Computing, Power and Communication Technologies (GlobConPT), New Delhi, India, 2022, pp. 1-5, doi: 10.1109/GlobConPT57482.2022.9938320.

[15] N. J. Zakaria, M. I. Shapiai, R. A. Ghani, M. N. M. Yassin, M. Z. Ibrahim and N. Wahid, "Lane Detection in Autonomous Vehicles: A Systematic Review," in IEEE Access, vol. 11, pp. 3729-3765, 2023, doi: 10.1109/ACCESS.2023.3234442.