JCRES
Journal of Current Research in Engineering and Science
Bi-Annual Online Journal ( ISSN: 2581 - 611X )

VOLUME:8, ISSUE:1, FEBRUARY 2025, PAPER NUMBER:76

# Investigation on Advanced Techniques in Lane Line Detection Using Machine Learning

**KAVINAYA E [1], JANANI R [2], KARTHIKA S M.E[3],**

*[1,2] Students, and [3] Faculty*
*Dept. of Artificial Intelligence and Machine*
*Learning, Bannari Amman Institute of*
*Technology, Erode, India.*
*kavinaya.al21@bitsathy.ac.in*
*Janani.al21@bitsathy.ac.in*
*karthikas@bitsathy.ac.in*

Abstract: The rapid development of self-driving technology and Advanced Driver Assistance Systems (ADAS) has highlighted the importance of enhancing lane detection systems for vehicle safety purposes. This study delves into the exploration of advanced methodologies for lane line detection through the utilization of Convolutional Neural Networks (CNN) and machine learning techniques such as Edge Detection and Hough Transform. Lane line detection bears significant importance in the realm of autonomous vehicles and driver assistance systems. The research emphasizes the application of CNN to accurately and efficiently detect lane lines across diverse road and weather conditions. The study encompasses comprehensive experimentation and evaluation to underscore the efficacy of the proposed approach in real-world scenarios. By harnessing the innate capability of CNN to autonomously learn features from raw pixel data, the aim is to make the system more efficient and able to detect in all conditions. Furthermore, the research probes into the deployment of machine learning algorithms to augment the generalization and adaptability of lane line detection models amidst varying environmental conditions. The outcomes of this study hold promise in significantly advancing the domain of autonomous driving systems by addressing the intricacies associated with accurate and real-time lane line detection. The proposed methodologies seek to elevate the safety and dependability of autonomous vehicles, thereby contributing to the widespread integration of self-driving technologies. In essence, this study offers invaluable insights into leveraging CNN and machine learning for sophisticated lane line detection, thus paving the way for the evolution of more autonomous driving systems.

Keywords— Self-driving technology, Advanced Driver Assistance Systems (ADAS), Edge Detection, Hough Transform, Lane Line Detection, Feature Extraction, Deep Learning, Convolutional Neural Network (CNN), Preprocessing Techniques, Model Generalization, Autonomous Driving Systems.

## 1. INTRODUCTION

The increasing urban traffic has amplified the significance of traffic safety. Lane departure contributes to approximately 30% of all highway accidents, with most of these incidents attributed to driver distraction and fatigue. Consequently, the development of a system capable of alerting drivers to potential dangers holds immense potential for saving numerous lives [1]. Lane detection stands out as a focus area as it plays a crucial role, in ensuring safe navigation and reducing the risks of lane departure incidents. In dealing with challenging driving scenarios, like shadows and road deterioration or when vehicles obstruct the view lane detection systems that depend on geometric models or supervised learning techniques encounter difficulties.

Monocular cameras are commonly used for understanding road environments. Face challenges with lighting changes and weather conditions that can make detecting lanes tricky in specific situations [4]. Although laser radar has its benefits it is expensive and sensitive to influences making it less common than cameras for lane detection technology. As this technology progresses from supporting drivers in ADAS to being vital for vehicles (AVs), ensuring precise and dependable performance, in various settings becomes crucial. In this study, we delve into the advancements in detecting lane lines with a focus on addressing obstacles and enhancing the reliability of systems used in real-world scenarios, for self-driving vehicles [7-8].

Lane detection is a crucial project for independent motors and Advanced Driver Assistance Systems (ADAS), as it provides essential data regarding lane obstacles to ensure secure navigation. Road lanes commonly appear as non-stop line systems, both strong or dashed, and are surprisingly correlated among neighboring frames in using scenes [27]. This correlation allows for advanced lane detection by using more than one frame, even in cases where lanes are occluded, degraded, or obscured by shadows. Recently, deep gaining knowledge of methods, in particular Convolutional Neural Networks (CNN) have demonstrated modern overall performance in numerous laptop imaginative and prescient duties, including lane detection [35]. CNNs excel at feature extraction from photographs and motion pictures are powerful for time-series prediction. CNN is used for extracting the features of the input data by simplifying and analyzing the features over the frames of the input data to improve and increase efficiency and accuracy. This technique enhances the reliability of identifying the lanes in various challenging conditions which include shadows, distractions, and deteriorated road markings [39]. Moreover, the effectiveness and real-time performance of this method render it a viable option for integration, into world autonomous driving situations. Two fresh sets of data have been collected for assessment to showcase how well this hybrid approach works in practice. This study delves into methods for lane detection using CNN structures to strike a balance, between precise detection rates and minimal computational load.

To better solve lane detection, a new method is presented that utilizes deep CNN to capture global features in different scales through spatial convolution and crop grid region classification. The final CNN architecture is a fully convolutional network (FCN) type that includes both encoder and decoder networks, which will capture the complex shapes in the scenario while retaining high-resolution lane markings to detect them [15]. The encoder network receives inputs and extracts hierarchical semantic features, in response to the decoder, these are reconstructed spatial feature maps TO accurately find out lane lines.

This architecture is specially made for tough road conditions so that much more accurate lane detection is possible than conventional methods [17]. The CNN tackles lane detection in challenging driving scenarios by learning frame-by-frame, making a substantial difference to typical single-frame used with conventional know-how. These advanced techniques offer a robust solution for accurate lane line detection.

## 2.LITERATURE SURVEY

In the 2012 look, F. Mariut et al. delivered an algorithm designed to robotically hit upon lane markings in digital pictures of the usage of the Hough remodel [1]. This method specializes in emphasizing lane marks and figuring out their traits, including the ability to determine the direction of travel. The fundamental principle of the Hough Remodel algorithm is to detect and extract the inner margins of the lane line whereas it improves the accuracy of the lane detection. The algorithm is very effective on straight roads because the straight lane lines are simpler to detect whereas on the other hand, the algorithm struggles with curved roads. The system performs very well on straight lanes whereas it struggles on curved lanes. These challenges highlight the need to improve the model to deal with such situations more robustly.

In 2014, Jiun Kim and Minho Lee proposed a detector that employs CNNs for final layer feature extraction followed by further clustering using RANSAC [3]. Architecture of CNN (8 layers/3 convolutional & 2 subsampling/3 fully connected), The training dataset is images that have been selected with ROI and mapped to edges. The last fully connected layer outputs the predicted image (10,015) where white pixels are lanes.

A method presented by Yao et al. (2022) to enhance lane detection, especially in low-lighting environments where shadows and road color changes are observed [4]. The objective of their approach is to combine these properties in a single network, which encompasses both the learnable abstract feature representations and detailed spatial information by incorporating a two-branch network architecture. A Spatial Attention Module (SAM) is provided to refine the spatial relationships of features which improves lane boundary detection in further and occluded areas. To detect weak lane lines, we utilize explicit boundary regression (EBR), which helps utilize Sobel edge detection and focal loss for end-to-end segmentation.

Liu et al. employed mobile edge computing to help the learning process, by reducing latency [6]. John et al. applied a CNN to the task of lane line detection in images and showed CNNs can learn powerful features even with occlusion or absence of detail from road images [8]. They also trained a tree-based regression model on the inferred features by CNN for an accurate lane line location.

Kim et al. To achieve this, serial end-to-end transfer learning for directly predicting the left and right lanes in different road conditions to construct a dataset that can be effectively utilized by the CNN model [9]. Pan et al. The spatial CNN (SCNN)was also proposed with convolution extended across the feature map, so that information is transferred between rows and columns of pixels.

N. Phaneendra et al. (2013) presented a vision-based lane departure warning system. The primary objective of this model is to develop an algorithm that can process the image and detect lanes on the road it also gives a textual warning at lane departure [2]. Lane Departure decision-making has a simple methodology that relies on the distance between lanes and the center of the bottom in the captured image coordinate, which needs fewer parameters. Kalman filter (as opposed to our usual Hough transform) has enhanced the lane detection performance. This new model was both effective and practical compared to other systems and when the road conditions would become more complicated, that system still could not detect markers of lanes correctly.

Pan et al. The spatial CNN (SCNN) proposed with the convolutional expansion on the feature map, enables information exchange between rows and columns of pixels as well [10]. Zhang et al. Li et al. proposed to embed a Convolutional Neural Network (CNN)-based deep learning architecture for monocular vision instance segmentation adopting orderless scene-level object representation which provides marginal gains in performance compared to real-world scenarios and restricted robust constraints [12].

Shriyash Chougule et al. pointed out that current semantic segmentation-based approaches for lane detection put too much emphasis on pixel-level classification and overlook the specification of shape [5]. However, segmentations-based approaches obtained remarkable results despite these significant challenges. The work of Chiu Kuo-Yu in 2005, considered lane detection as an image segmentation task but was still using pre-deep learning algorithms. Older research used GCN for lane surrounding recognition (Wenhui Zhang and Tejas Mahale) as well as Mask-RCNN with Kalman filtering-based tracking to detect the departure of lanes in the case of Riera Luis et. On real-time applications, Shriyash Chougule et al. [5] designed a ten-layer CNN, and Pizzati Fabio et al. modified ERFNet to detect drivable areas and road classes by employing DBSCAN to group pixel-wise free space points into polygons.

## 3.MATERIALS AND METHODS

### A. Data Collection

The methodology for lane line detection is developed by using various datasets like TuSimple, KNU database, and Caltech database, serving to train the detection system. Training images were meticulously labeled according to defined standards and aimed to ensure representative and heterogeneous local driving environments, to be then used for serving the real-time group speed estimation system on lane lines. Initially, they gathered just 2000 images and extended the dataset using image mirroring to reduce overfitting and improve training results. This ensured the diverse data exposure.

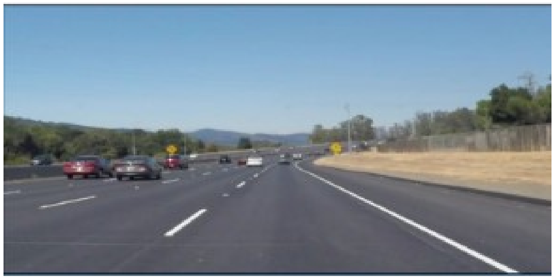


Fig 1.1 Original Images

### B. Image Preprocessing

This method is applied to the images to enhance the contrast of the images, it is useful when lane markings are light or when illumination conditions are poor. Histogram equalization makes lane markings clearer by increasing their contrast with the surrounding road surface through pixel intensity distribution adjustment [28]. To keep things consistent and aid in processing images are resized to 256×256 pixels. This step ensures consistency and makes the image suitable for computational efficiency by standardizing the input dimensions. The chosen resolution strikes a balance by keeping lane line details while also reducing the computing resources needed for analysis efficiently and quickly without compromising on the accuracy of the lane line specifics Use a brighter threshold for images in low-light or foggy conditions that would make lane lines hard to see.

### C. Color Space Conversion

The color space used to convert all the images is RGB; this will allow our algorithm to become more robust against lighting changes. The V channel (for Value) is the least variant, therefore allowing easier detection of lane markings in a scene-invariant way. Except in the case of changing brightness (shadows, tunnels, sunshine, etc.) where then they are detected by concentrating on a single channel (V) since lane lines will continue to be detectable from the overall contrast [18]. To enhance the CNN's generalization ability, the input data undergoes data augmentation before being fed into the network. Techniques like random rotations, flipping, and contrast adjustments are applied to artificially expand the dataset. This helps the network become robust to various road conditions, including different lighting, weather, and road quality.

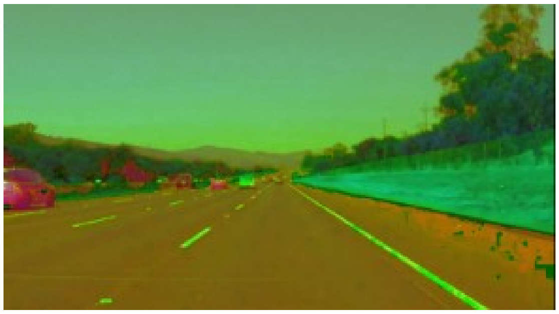


Fig 1.2 Color Space Conversion image

### D. Noise Reduction and Gray Scaling

After increasing the contrast using histogram equalization, noise is reduced by applying a Gaussian filter. Because we don't want random pixel intensity fluctuation in the image to influence lane detection, we apply this filter The Gaussian filter has smoothed the image, but still kept the important edges of lane lines.

Grayscale conversion simplifies the image by reducing the three-channel RGB data to a single-channel image, where each pixel represents a shade of gray [20]. This step significantly reduces the computational complexity by decreasing the amount of data that needs to be processed, allowing the lane detection algorithm to run faster and more efficiently.

### E. Feature Extraction

Feature extraction in detecting lane lines is primarily concerned with pinpointing attributes in input images that differentiate lane lines, from the road surface. The process starts by identifying edges through techniques, like the Canny Edge Detector or Sobel Filter that emphasize the shifts in intensity that mark lane boundaries [17]. As the CNN goes deeper, the filters can recognize extremely complex and abstract features useful in lane detection. This involves the separation of dashed and solid lane lines owing to the recurrent patterns within the feature maps, edge detection to determine lane boundaries that mark lane segregation as well and finding curvatures in all four lines which are essential when driving through winding roads [14]. The CNN is trained on extracted road images annotated with lane line markings in a supervised manner, and while being trained, the weights and biases of the CNN update through backpropagation to minimize a loss function (e.g., MSE for lane position regression or cross-entropy for lane type classification). The network is repeatedly shown the small images and told whether or not they contain lane lines, which lets the network learn good filters for detecting lane lines even in the presence of varying conditions including different light intensities/conditions, weather (clear vs foggy), road texture, etc.

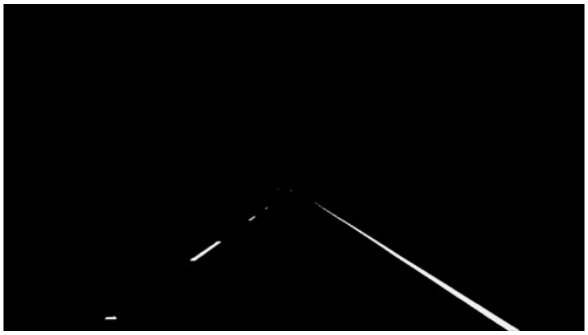


Fig 1.3 Feature Extraction Image

### F. Region of Interest (ROI) Selection:

Selecting a Region of Interest (ROI) is a critical aspect of lane line detection as it enables the restriction of the analysis to the image area where lane lines are most likely to be present, typically the lower half or a trapezoidal section corresponding to the road [21]. This method helps exclude irrelevant image components such as the sky, nearby vehicles, or roadside objects, thereby reducing noise and enhancing model accuracy. By focusing processing efforts on a smaller, pertinent section, ROI selection not only reduces computational overhead but also allows the model to concentrate on lane lines, thereby improving detection precision and efficiency, particularly in challenging environments with distractions such as shadows or road debris.

### G. Hough Transform

The Hough Transform method is commonly utilized to extract features, for image analysis and digital image processing purposes. The conventional Hough Transform primarily focuses on detecting lines within images. In automated digital image analysis tasks in the past, it was challenging to identify lines or circles. To address this issue in image analysis processing stages often involved using edge detection techniques to locate points on an image that align with the desired curve [17], however, sometimes certain pixels along the desired curve were missing due to imperfections, in the image. So, for solving this problem Hough Transform is used.

To detect lines in an image, the first thing we do is convert it to a binary image with some threshold. After that, we add the necessary samples to the dataset. The main part of Hough Transform is the Hough space. Each point (d, T) from the Hough Space is paired with a line at the angle of T and distance d from the origin. A point on a line corresponds to a value of the (Hough space) function. For each point, you want to consider every line that goes through that point at discrete set angles priority in order. The accumulator is an array for spotting the lines from the Hough transform Accumulator should have the same dimension as several unknown Hough transform parameters [40]. At first, those lines are generated which can move through every point. When a line intersects with the lines from other points, it votes for those (d, T) parameters. In the end, the (d, T) pair that gets most of the votes is all selected as major lines present on the image plane based on points lies along this line.

### H. Convolutional Neural Networks Architecture

Convolutional Neural Networks (CNN) used for Image Classification and Recognition. They work pretty well in tasks such as Image recognition, classification, and object detection. CNNs can learn to extract features from images for us, and thus Convolutional Neural Network (CNN) has become the most commonly used technique in studying lane line detection. CNNs to recognize highway lines/ related road (e.g. photos-->image), on-road scene: evidence of angle from road (e.g. motorway outlines/patterns…).

The first layer in CNN of this network is convolutional; it runs filters (kernels) that recognize low-level features such as edges and types of field marks. In Convolutional Layers, filters slide over the input image performing convolution and calculating a weighted sum for every region.

$Zij=(X*W)ij+b$ (1)

where X is the input image, W is the filter (kernel), b is the bias term, and Zijis the output feature map. This leads, to feature maps highlighting lane boundaries, road surfaces, colors, etc. against a background of particular lighting conditions.

Post the convolution operation in the architecture of the model, a non-linear activation function like ReLU (Rectified Linear Unit) is performed. ReLU a will instantly give it non-linearity for its transformation to the output of every layer, as well as ReLU.

$f(x)=\max(0,x)$ (2)

ReLU uses these negative values, zeroing them out to reveal just positive ones, enabling the model to learn more sophisticated patterns - like particularly curved lane lines.

Pooling layers (generally Max Pooling) are incorporated to decrease the spatial dimensions of the feature maps, to make the model computationally efficient. Pooling: Pooling keeps only the most important features (ex-lane line edges) and reduces noise. Max pooling is defined as:

$Zij=\max(\{Xi+m,j+n:0\leq m,n<k\})$ (3)

here, k is the size of the pooling window. This eventually reduces parameters and keeps key points responsible for lane detection.

The CNN can also use multiple convolutional and pooling layers after one another to detect more advanced features. The filters become more complex as the network goes deeper, looking for dashed and solid lines, curves, or lane boundaries.

The 2D feature maps are passed through multiple Convolution and Pooling layers, and then they are reshaped to that at the end of them a Flattening layer converts the 2D feature maps to a single 1-D vector where it can be passed through fully connected or dense layers. The final, several fully connected layers to classify or regress the output of the extracted features (e.g., Dashed vs solid lane or Predicted lane position).

Fully connected layer which works as

$y=W\cdot x+by$ (4)

where x is the input feature vector, W is the weight matrix, b is the bias, and y is the output prediction.

For lane classification (e.g., dashed vs. solid lane), a softmax function is used to predict the class probabilities:

Lane position regression (predictions of the actual lane line positions in an image) could spew out a bunch of coordinates that define the lane lines. A CNN contains a big dataset of driving images which is labeled by humans. Backed by backpropagation and optimization techniques like Stochastic Gradient Descent (SGD) or Adam, it minimizes a loss function such as cross-entropy for classification or mean squared error (MSE) for regression.

### I. Image Classification and Regression

After CNN goes through the image and extracts appropriate features, it tries to do lane detection either by classification or regression. The CNN outputs a probability distribution over classes--dashed or solid, curved or straight for every lane segment in classification tasks using a SoftMax classifier in the last layer. This allows the system to label and separate each lane line detected, which is important for Advanced Driver Assistance Systems (ADAS), so they know what type of lane marking they are reading. On the other hand, when a CNN-only architecture detects lane lines in an image, it predicts pixel coordinates where a pair of lane lines exist from input images generating continuous values representing where they are located for regression tasks. The output could be a set of (x, y) coordinates that can be superimposed on the original image for visualization. To smooth these lane line coordinates, a polynomial curve fit can be used in post-processing to create a continuous and visually consistent way of visualizing the specific locations of lane lines of the detected lanes.

### J. Lane Visualization

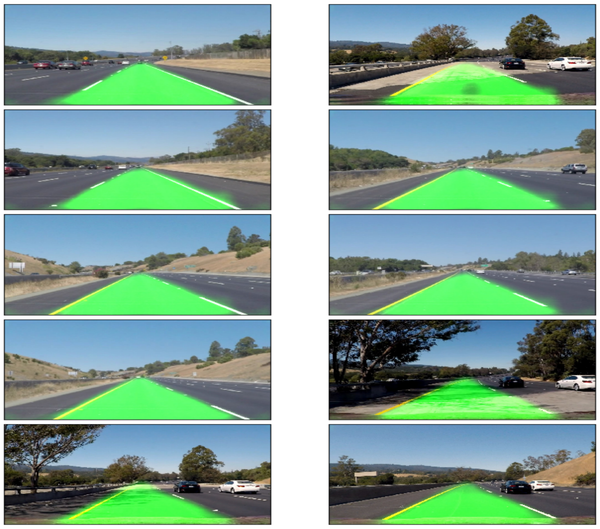
After the CNN outputs the lane line predictions, post-processing techniques are employed to enhance the accuracy and visualization of the detected lanes. One key step is smoothing the lane prediction algorithms like moving average filters to mitigate slight jitters in the predicted lane line positions, ensuring that the detected lanes appear more stable and natural in the driving scene, the coordinates of the detected lane lines are projected back onto the original image, allowing the system to superimpose the predicted lanes over the road, providing a clear visual representation crucial for autonomous driving systems. In this visualization, the lane color is highlighted in green, making it easily identifiable and enhancing the overall clarity of the detected lanes in the driving environment.

## 4. RESULTS AND DISCUSSION

The lane line detection was a carried out using a CNN model that performed the lane detection task, trained on a mix of public datasets TuSimple, CULane and KITTI. These datasets cover a wide variety of driving scenes such as highways, city streets, and between trees, random lighting cum weather conditions, different lane level visibility. To do this, the CNN model processes each frame individually to extract lane features and therefore can successfully recognize different lane markings across a range of road types and conditions.

The training data came from the TuSimple, CULane (133K images) and KITTI (7481 images with the ground truth). The images were rescaled to 1280x720 pixel resolution for quicker processing with minimum resampling-induced data loss. The datasets were from different types of lanes (dashed or solid) and under increasing difficulties such as rainy, dry, night-time snow. We split the data using 80% for training and 20 % for testing. In this respect, data augmentation is used to reduce the effect of dataset bias and more generally learn representations that can be expected extrapolated to real-world conditions in this project random rotations, flipping, and brightness adjustment have been applied. It allowed the model to be consistently reliable in different environment perturbations such as shadow, reflection, and faded lanes.

We use an Adam Optimizer with an initial learning rate of 0.001 and a batch size of 32, and the model was trained for 200 epochs. Mean squared error (MSE) was used as the loss function for regression-based lane fitting tasks, while Binary Cross Entropy loss was used for pixel-wise classification of lanes region. If the validation loss does not improve then it reduced learning rate after 50 epochs of training. It was a schedule related to the learning rate and made the model converge well.



The CNN architecture with residual layers and dilated convolutions enables us to capture both the fine-grained features and lane structures at various scales. Our CNN model is evaluated on the tested U-Net, LaneNet and machine learning models. Table I shows results

Table I. Comparison of Accuracy, Precision, Recall, F1 Score, and Model Loss for Traditional Methods vs. CNN Model.

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** | **Model Loss** |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| Hough Transform | 85.12 | 80.34 | 79.45 | 79.89 | 0.0257 |
| Canny Edge Detection | 76.15 | 72.30 | 74.12 | 73.20 | 0.0312 |
| Proposed CNN model | 92.45 | 91.23 | 91.67 | 91.45 | 0.0136 |
| Vanilla CNN | 88.34 | 85.12 | 86.45 | 85.78 | 0.0203 |
| U – Net | 90.56 | 88.67 | 89.12 | 88.89 | 0.0118 |
| LaneNet | 90.37 | 88.23 | 89.17 | 89.00 | 0.0156 |

In the next steps, a few tests were conducted to ensure that the CNN model properly recognized lane lines under different scenarios and its robustness against new environments, considering shadows, light changes and lane disappearing. The lane structure and curvature were perfectly captured by convolutional layers, resulting in stable lane segmentation across a variety of road types.

Curved lanes were segmented at a precision of 89.12% by CNN model, and the straight lane segmentation precision obtained as 93.45%. Although faded lane lines at night was still reflected as a crash, the model had an important boost in performance overall (as can be seen in Table II).

Table II. Lane detection results for different lane types.

| Lane Type | Number of Images | Accuracy |
|---|---|---|
| Straight | 2,100 | 93.45 |
| Curved | 1,800 | 89.12 |
| Night – time | 1,000 | 85.78 |
| Wet roads | 700 | 82.45 |

Moreover, the CNN model was able to identify lane lines in different real-life settings especially various factors as shadows, lighting variation and disappearance of lanes. As can be seen in the figure above, the CNN captured both lane structure and curvature and are capable of providing consistent lane segmentation across all types of roads.

CNN model has shown good quality of results by detecting the lane lines that performed well and achieved an overall score accuracy of 91.45%. In general (the new lane split testing) we noticed that the model had a decent ease in generalizing between straight lanes (93.45% accuracy) to corners of roads (89.12% accuracy) in different light settings compared to previous edge detection methods.

But its results were a bit worse than some in the most challenging of tests, night-time driving with occluded or poorly visible lane lines – that is when they could measure how well systems detect obstacles not in their lane, such as children running into the street. Possible future refinements will be data augmentation or adding more sensor inputs such as LIDAR, RADAR to enhance robustness on real world.

These results show CNNs (Convolutional Neural Networks) to be a strong candidate when it comes to lane detection in a host of applications ranging widely from autonomous driving and ADAS, with the potential for real-time deployment across various road conditions.

## 5. CONCLUSION

CNN (Convolutional Neural Network) algorithms have also shown advantage in the field of object detection and recognition, thus can help in Lane line detection with high accuracy and robustness. Additional methods like greater sandwich architectures, transfer learning and data augmentation can provide CNN models superior efficiency and flexibility in challenging physical conditions such as night-time or adverse weather. CNNs have very high scalability and real-time processing capabilities which can be easily integrated into autonomous vehicles and advanced driver-assistance systems (ADAS) that are imperative for preserving road safety, reliability, and efficiency.

CNN-based lane detection holds high promise in not only augmenting driving accuracy but also significantly decrementing human errors that can guarantee safer and effective navigation while being driven. As this technology continues to mature, there are ethical implications for data privacy and bias in training datasets that need to be considered so it can be used responsibly. Lane detection has great commercial value which includes important aspects of autonomous driving, ADAS and traffic control as well as making it imperative for CNN-based models to play a crucial role in shaping the future of automotive innovation.

In this paper, a thorough analysis of CNN-based lane detection methods is provided and demonstrates their significant advances over traditional edge detection techniques. The proposed CNN model achieved an accuracy of 91.45% when lane lines had to be detected under different lighting and road conditions. While the performance dropped in some challenging situations, the good robustness of our model promises that it can be applied to real-time lane detection tasks. Future investigations could improve these models with longer architecture, data augmentation, and multi-sensor fusions which may consolidate CNNs as the key building blocks in safe and intelligent driving.

## 6. REFERENCES

[1] F. Mariut, C. Fosalau, and D. Petrisor, “Lane Mark Detection Using Hough Transform", In IEEE International Conference and Exposition on Electrical and Power Engineering, pp. 871 - 875, 2012.

[2] Nalla, Phaneendra, GCL AbhiramanGoud, and V. Padmaja. "ACCIDENT AVOIDING SYSTEM USING LANE DETECTION." IJRECE 1, no. 1 (2013): 01-04.

[3] J. Kim and M. Lee, “Robust lane detection based on con-convolutional neural network and random sample consensus.” in proceedings of the International Conference on Neural Information processing, pp.454-461, Neural Information processing. Kuching, Malaysia, November 2014.

[4] Zhiting Yao, Xiyuan Chen, “Efficient Lane Detection Technique Based on Lightweight Attention Deep Neural Network”, Journal of Advanced TransportationVolume 2022, Issue 1, 09 March 2022

[5] Shriyash Chougule, Nora Koznek, Asad Ismail, Ganesh Adam, Vikram Narayan, and Matthias Schulze, “Reliable multilane detection and classification by utilizing CNN as a regression network”, Munich. In ECCV Workshop, 2018.

[6] Sheng Liu, Weiwei Chen, Weixing Wang, Kevin Wang, Zhaoying Li, Huan Li, “Lane departure warning systems and lane line detection methods based on image processing and semantic segmentation”, Journal of Traffic and Transportation Engineering, December 2020.

[7] Cho, Jae-Hyun, Young-Min Jang, and Sang-Bock Cho. "Lane recognition algorithm using the Hough transform with applied accumulator cells in multichannel ROI." In Consumer Electronics (ISCE 2014), The 18th IEEE International Symposium on, pp. 1- 3. IEEE, 2014..

[8] John V, Liu Z, Guo C, Mita S, Kidono K (2015) Real-time lane estimation using deep features and extra trees regression In Image Video Technol, 721–733. Springer.

[9] Kim J., Lee M. “ Robust lane detection based on convolutional neural network and random sample consensus”. Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8834 (2014), pp. 454-461.

[10] X. Pan, J. Shi, P. Luo, et al. “Spatial as deep: spatial CNN for traffic scene understanding”. arXiv, 1712.06080, 2018

[11] T. Zhang, “Research on Complex Background Lane Line Detection Based on Fractional Theory and Least Square Method”, Chang’an University, Xi’an (2017)

[12] Y Zhang, Z Lu, D Ma, et al. Ripple-GAN: Lane line detection with ripple lane line detection network and Wasserstein GAN. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[13]Sun-Woo Baek,Myeong-Jun Kim, Upendra Suddamalla, Anthony Wong, Bang-Hyon Lee, Jung-Ha Kim, “Real-Time Lane Detection Based on Deep Learning”, 9 September 2021

[14] Erkan Oğuz, Ayhan Kucukmanisa, Ramazan Duvar, Oguzhan Urhan, “A deep learning based fast lane detection approach”, February 2022, DOI: 10.1016/j.chaos.2021.111722

[15] S. Srivastava, R. Singal, and M. Lumb, “ Efficient Lane Detection Algorithm using Different Filtering Techniques”, International Journal of Computer Applications, vol. 88, no.3, pp. 975-8887, 2014.

[16] A. Borkar, M. Hayes, M.T. Smith, and S. Pankanti, “A Layered Approach To Robust Lane Detection At Night”, In IEEE International Conference and Exposition on Electrical and Power Engineering, Iasi, Romania, pp. 735 - 739, 2011.

[17] K. Ghazali, R. Xiao, and J. Ma, “Road Lane Detection Using H-Maxima and Improved Hough Transform”, Fourth International Conference on Computational Intelligence, Modelling, and Simulation, pp: 2166-8531, 2011.

[18] Z. Kim, “Robust Lane Detection and Tracking in Challenging Scenarios”, In IEEE Transactions on Intelligent Transportation Systems, vol. 9, no. 1, pp. 16 - 26, 2008.

[19] M. Aly, “Real-time Detection of Lane Markers in Urban Streets”, In IEEE Intelligent Vehicles Symposium, pp. 7 - 12, 2008.

[20] J.C. McCall and M.M. Trivedi, “Video-based Lane Estimation and Tracking for Driver Assistance: Survey, System, and Evaluation”, IEEE Transactions on Intelligent Transportation Systems, vol.7, pp.20-37, 2006.

[21] Y.Wang, E. K. Teoh and D. Shen, “Lane Detection and Tracking Using B-snake,” Image and Vision Computing, vol. 22, pp. 269-280, 2004.

[22] A. Broggi and S. Berte, “Vision-based Road Detection in Automotive Systems: a Real-time Expectation-driven Approach”, Journal of Artificial Intelligence Research, vol.3, pp. 325-348, 1995.

[23] M. Bertozzi and A. Broggi, “GOLD: A Parallel Real-time Stereo Vision System for Generic Obstacle and Lane Detection”, IEEE Transactions of Image Processing, pp. 62-81, 1998.

[24] S.G. Jeong, C.S. Kim, K.S. Yoon, J.N. Lee, J.I. Bae, and M.H. Lee, “Real-time Lane Detection for Autonomous Navigation”, IEEE Proc. Intelligent Transportation Systems, pp. 508–513, 2001.

[25] Y.Wang, E.K. Teoh and D. Shen, “Improved Lane Detection and Tracking Using B-snake,” Image and Vision Computing, vol. 20, pp. 259-272, 2005.

[26] C. R. Jung and C. R. Kelber, “A Lane Departure Warning System Using Lateral Offset with Uncalibrated Camera,” Proc. IEEE Conf. on Intelligent Transportation Systems, pp.102-107, 2005.

[27] D.J. Kang, J. W. Choi, and I.S. Kweon, “Finding and Tracking Road Lanes Using Line-snakes”, Proceedings of Conference on Intelligent Vehicle, pp. 189-194, 1996.

[28] Z. Kim, “Real-time lane tracking of the curved local road”, Proc. IEEE Conf. on Intelligent Transportation System, pp. 1149-1155, 2006.

[29] Y. Wang, D. Shen, and E.K. Teoh, “Lane Detection Using Spline Model”, Pattern Recognition Letters vol.21, pp. 677-689, 2000.

[30] Q. Lin, Y. Han, and H. Hahn, “Real-time lane departure detection based on extended edge-linking algorithm”, In IEEE 2nd International Conference on Computer Research and Development, pp. 725-730, 2010.

[31] O.O. Khalifa and A.H.A Hashim, “Vision-Based Lane Detection for Autonomous Artificial Intelligent Vehicles”, In IEEE International Conference on Semantic Computing, pp. 636 - 641, 2009.

[32] K. Ghazali, R. Xiao, and J. Ma, “Road Lane Detection Using H-Maxima and Improved Hough Transform”, Fourth International Conference on Computational Intelligence, Modelling, and Simulation, pp: 2166-8531, 2012.

[33] T. T Tran, C. S Bae, Y. N. Kim, H.M. Cho, and S.B. Cho, “An Adaptive Method for Lane Marking Detection Based on HSI Color Model”, ICIC, CCIS 93, pp. 304– 311, 2010.

[34] D. Pomerleau and Jochem, “Rapidly Adapting Machine Vision for Automated Vehicle Steering, IEEE, 1996.

[35] B. M, Broggi, “GOLD: A parallel real-time stereo Vision system for generic obstacle and lane detection”, IEEE Transactions on Image Processing, pp. 4-6, 1998.

[36] C. Kreucher and S. K. Lakshmanan, A Driver warning System based on the LOIS Lane detection Algorithm, Proceeding of IEEE International Conference On Intelligent Vehicles. pp. 17 -22, 1998.

[37] Zielinski A., Cupek R., Erdogan H., and Waechter S., A survey of ADAS technologies for the future perspective of sensor fusion, Computational Collective Intelligence, *International Conference on Computational Collective Intelligence*, 2016.

[38] Ni J., Han J., and Dong F., Multivehicle cooperative lane change control strategy for intelligent connected vehicle, *Journal of Advanced Transportation*. (2020).

[39] Chen W, Wang W, Wang K, et al., Lane departure warning systems and lane line detection methods based on image processing and semantic segmentation–a review, Journal of Traffic and Transportation Engineering. (2020).

[40] GurjyotKaur and Gagandeeop Singh, “A review of lane detection techniques”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 02 Issue: 03 | June-2015.