Ensemble Based Approach for Road Lane Detection

A.DURGA - 221FA04051
Department of Computer Science And Engineering
Vignan's Foundation for Science, Technology and Research
Guntur, India
durgaasodi14@gmail.com

M.SHANMUKH - 221FA04062
Department of Computer Science And Engineering
Vignan's Foundation for Science, Technology and Research
Guntur, India
shanmukhmunaga@gmail.com

K.NIKHIL - 221FA04108
Department of Computer Science And Engineering
Vignan's Foundation for Science, Technology and Research
Guntur, India
nikhilkamepally108@gmail.com

G.VENKATA KRISHNA - 221FA04696
Department of Computer Science And Engineering
Vignan's Foundation for Science, Technology and Research
Guntur, India
venkatakrishnaguntanala03@gmail.com

I. Abstract

This work offers an end-to-end framework for detecting road lanes based on image classification and clustering that combines deep feature extraction using convolutional neural networks with state-of-the-art machine learning classifiers. Pretrained deep convolutional neural networks like ResNet50, Xception, and MobileNetV2 are used to derive high-level semantic features from images of roads. These deep features are strong representations for input of further unsupervised and supervised learning tasks. K-Means clustering is utilized to identify intrinsic patterns within the feature space to help infer lane structures and variations between images. Gradient boosting models such as XGBoost, LightGBM, and AdaBoost are utilized for effective classification of lane types. In order to treat the class imbalance problem in the dataset, during training, Synthetic Minority Over-sampling Technique (SMOTE) is applied. Ensemble methods Voting and Stacking improve the prediction capability of the system. Models are compared with accuracy, precision, recall, and F1-score for rigorous performance evaluation. The experiments verify the utility of integrating deep feature extraction with ensemble learning methodologies to achieve efficient road lane detection that is both robust and accurate.

II. Introduction

Lane detection on roads is a core element of autonomous driving and Advanced Driver-Assistance Systems (ADAS), and it is a crucial factor in ensuring vehicle stability, safety, and efficient driving. It is a prerequisite for many functionalities like lane-keeping, lane departure warning, and adaptive cruise control, all of which help in curbing road accidents and improving overall driving experience. With the growing popularity of autonomous and semi-autonomous vehicles, robust and real-time lane detection is one of the central research topics in the domain of intelligent transportation systems (ITS).

Conventional computer vision-based techniques, such as edge detection, Hough transform, and morphological processing, have been popularly applied to lane detection. These are often based on hand-crafted visual features, e.g., lane markings and road edges, for extracting informative features from images. Yet, they tend to perform poorly in difficult driving situations, including changing road textures, occlusions due to vehicles or pedestrians, low lighting (e.g., night driving), inclement weather (rain, fog, snow), and worn-out or absent lane markings. These limitations call for the creation of more robust and adaptive lane detection methods.

The arrival of machine learning and deep learning has revolutionized lane detection to a large extent by allowing models to learn sophisticated patterns straight from raw input data. Convolutional Neural Networks (CNNs) have proved to be especially effective in learning spatial hierarchies and lane feature detection in a more generalized way. Recent developments in deep learning, including Transformer-based architecture and attention mechanisms, further strengthen lane detection by capturing contextual knowledge and long-range dependency features more efficiently. This has allowed lane detection models to generalize well to a wide range of road conditions, vastly reducing the necessity for manual feature engineering.

In addition, sensor fusion methods have enhanced lane detection reliability even further. The use of data from multiple sources—e.g., cameras, LiDAR, radar, and GPS—allows for the recognition of lane boundaries with greater accuracy in challenging urban conditions involving occlusion, shadows, or dynamically changing road geometries. Sensor fusion allows the system to better distinguish between lane markings, road curbs, and obstacles, thus providing a safer and more attentive driving experience.

Since real-time performance is essential for autonomous driving use cases, making deep learning models more efficient is an area of significant research. Methods involving model quantization, pruning, and hardware-based acceleration (involving GPUs and TPUs) are in the works so that lane detection models can process at high frames per second while maintaining low latency. Additionally, datasets like TuSimple, CULane, and LLAMAS have furnished large-scale labeled lane images to speed up building more robust and generalized models.

In this project, we investigate the developments in machine learning-based lane detection with emphasis on deep learning architectures, real-time processing optimizations, and sensor fusion methodologies. Our objective is to create a highly accurate and efficient lane detection system that can accommodate varied environmental conditions, ultimately contributing to the development of autonomous driving technologies and intelligent transportation systems.

III. Literature Survey

The field of road lane detection has seen significant advancements with the adoption of machine learning and deep learning techniques. Various studies have explored different approaches to enhance lane detection accuracy, robustness, and real-time performance.A real-time lane detection system utilizing YOLOv8 has been developed and implemented through a Streamlit interface, demonstrating its effectiveness in dynamic driving environments. Further research extended this study by incorporating additional datasets such as RAVDESS and TESS, using Support Vector Machines (SVM), Multi-Layer Perceptrons (MLP), and feature extraction techniques including Mel Frequency Cepstral Coefficients (MFCC), Mel Spectrogram, Chroma, and Tonnetz features, achieving an accuracy of 86.5 percentage despite computational complexity challenges [1][2]. A CNN-based method for lane detection on complex roads has been introduced to improve accuracy in challenging scenarios such as curves, broken lanes, and missing lane markings [3]. Another study proposed a deep learning framework for optimizing lane detection and steering in self-driving cars, utilizing a virtual sandbox environment in Unity3D to simulate various road scenarios [4].An integrated road lane and vehicle detection system has also been developed, employing YOLO and SSD for vehicle detection and Canny Edge Detection with Hough Line Transform for lane identification [5]. Further research has explored deep neural network designs for traffic control applications, analyzing models such as ResNet-50, Xception, and MobileNet-V2 on the KITTI road dataset to enhance road surface interpretation [6].A comprehensive review of deep learning-based lane detection techniques has highlighted advancements in lane marking detection, lane boundary detection, and lane type classification, along with ongoing challenges such as real-time processing and generalization across different environments [7].In addition, LVLane, an end-to-end lane detection and classification system, has been developed using a custom dataset covering challenging scenarios. A CNN-based classification branch integrated into the detector facilitates lane type identification, supporting ADAS functionalities [8].To improve lane tracking, RONELDv2, a lightweight and faster lane detection method, has been introduced, enhancing detection accuracy while reducing computational complexity, making it suitable for real-time applications in autonomous vehicles and lane departure warning systems [9].A multi-lane detection approach based on Affinity Fields has also been explored to detect various lanes without assuming a fixed number, leveraging novel decoding techniques to improve detection efficiency in complex driving environments [10].These advancements underscore the growing role of AI in autonomous driving and ADAS, offering improved lane detection capabilities for enhanced road safety and vehicle autonomy.

IV. Proposed Methodology

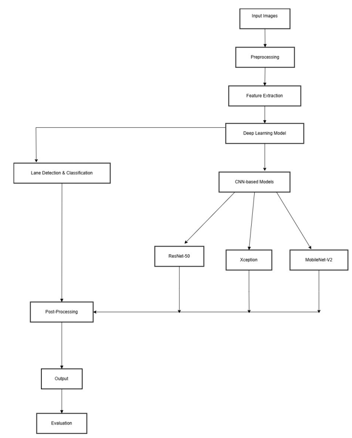

Fig. 1. flow chart of model

This study presents a deep learning-based approach for real-time road lane detection. The proposed system utilizes convolutional neural networks (CNNs) to accurately identify and classify lane markings, enabling improved autonomous driving and driver-assistance capabilities. The methodology follows a structured pipeline, which includes data preprocessing, feature extraction, model training, and evaluation.

a) Data acquisition: The system takes input images captured from road scenes, which may include multiple lane types, varying weather conditions, and different road structures. The dataset is curated from publicly available road datasets and custom image collections to ensure diverse lane scenarios.

b) Preprocessing: To enhance feature extraction and improve model accuracy, the input images undergo the following preprocessing steps:

Resizing: Standardizing image resolution to ensure compatibility with deep learning models. Normalization: Scaling pixel values for stable training and faster convergence. Contrast Enhancement: Using techniques such as histogram equalization or CLAHE to improve visibility under poor lighting conditions. Noise Reduction: Applying Gaussian or median filtering to remove unwanted noise and enhance lane visibility.

c) Feature Extraction: Feature extraction is performed using a combination of deep learning-based methods:

Deep Feature Extraction:

Pre-trained CNN models such as ResNet-50, Xception, and MobileNet-V2 are employed to extract hierarchical feature representations from input images. The models are initialized with ImageNet weights, and the extracted feature vectors are utilized for lane classification. Lane-Specific Feature Extraction:

Traditional edge detection methods (such as Canny Edge Detection) and Hough Line Transform are used to identify lane structures in challenging conditions.

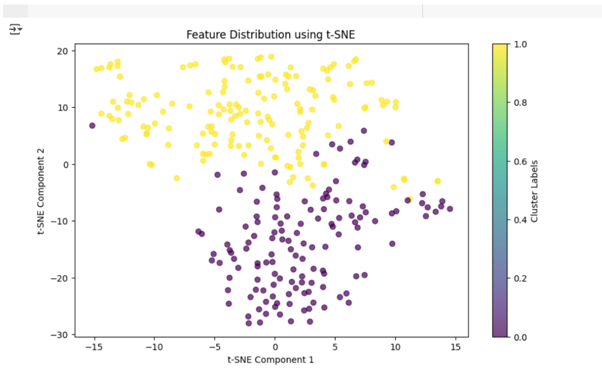

Fig. 2. Feature Distribution

d) K-means Clustering: K-Means is an unsupervised clustering algorithm that groups similar data points based on their feature similarity using Euclidean distance. In this case, deep features extracted from CNNs (ResNet50, Xception, and MobileNetV2) are clustered to identify patterns in the images. The results are visualized using t-SNE to show how well K-Means separates the images into distinct groups.

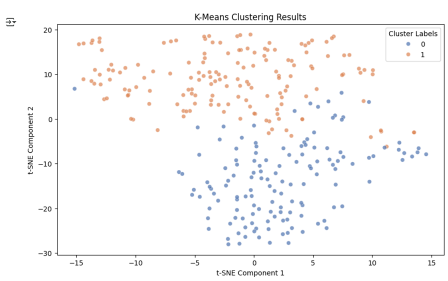

Fig. 3. Clustering Results

This model successfully categorizes road images into lane-marked and non-lane-marked clusters using clustering techniques. Future work may involve deep learning for enhanced accuracy.

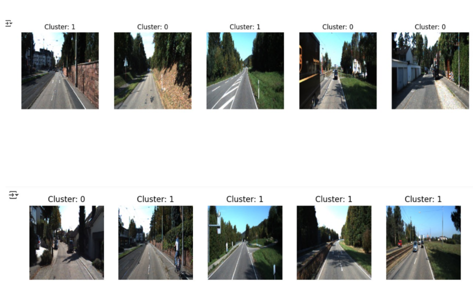

Fig. 4. Clustering Results

e) Lane Detection Classification: The processed features are passed to a deep learning model for lane detection and classification. CNN-based models analyze lane structures and categorize them based on visibility, lane types, and road curvature.

f) Output Evaluation: The final output consists of:

Detected Lane Boundaries: Visualized on input images for real-time applications. Evaluation Metrics: The system is evaluated using classification metrics, including: Accuracy: Measures the overall performance of lane detection. Precision Recall: Determines how well the system identifies lane markings. F1-Score: Ensures a balance between precision and recall. G. Visualization Implementation Feature Maps: Visual representations of extracted lane features from CNNs. Performance Graphs: Comparisons of different CNN architectures in terms of lane detection accuracy and speed. Implementation Details: The system is implemented using Python, TensorFlow, and OpenCV, and it is deployed with Streamlit for real-time interaction. H. Expected Outcome The proposed approach aims to:

Achieve high accuracy in lane detection across varying road conditions. Reduce false lane detections using a combination of deep learning and traditional feature extraction. Improve real-time performance to support autonomous driving and Advanced Driver-Assistance Systems (ADAS).

V. Implementation

A.Machine Learning Models Used:

a) (Extreme Gradient Boosting):: A decision-tree-based ensemble learning method.

Handles missing values well and provides high accuracy.

b) LightGBM (Light Gradient Boosting Machine):: Faster training time due to leaf-wise growth.

Suitable for large datasets.

c) AdaBoost (Adaptive Boosting):: Focuses on misclassified samples in iterative training.

Helps improve weak classifiers.

B.The models are evaluated using standard classification measures such as:

- **Accuracy:** Accuracy calculates the overall accuracy of the model by determining the proportion of correctly predicted observations (true positives and true negatives) to the total number of observations. It provides a general sense of how frequently the classifier is correct.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- **Precision and Recall:** Precision indicates how many of the predicted positive cases were actually correct. It is a measure of the model's ability to avoid false positives and Recall measures how well the model identifies actual positive cases. It reflects the model's ability to detect all relevant instances in the dataset, minimizing false negatives.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

- **F1-Score:** The F1-score is the harmonic mean of precision and recall. It provides a balanced evaluation when both false positives and false negatives are crucial, especially in imbalanced datasets.

$$F1{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

```
XGBoost Accuracy: 0.9167, Precision: 0.9286, Recall: 0.8966, F1-score: 0.9123
LightGBM Accuracy: 0.9333, Precision: 0.9630, Recall: 0.8966, F1-score: 0.9286
AdaBoost Accuracy: 0.9500, Precision: 1.0000, Recall: 0.8966, F1-score: 0.9455
```

Fig. 5. Performace metrics results

VI. Results

A. Performance Comparison of Model Ensembles

```
[LightGBM] [Warning] No further splits with positive gai
Voting Ensemble Accuracy: 0.9333
/usr/local/lib/python3.11/dist-packages/sklearn/utils/de
  warnings.warn(
```

Fig. 6. Accuracy using voting classifier

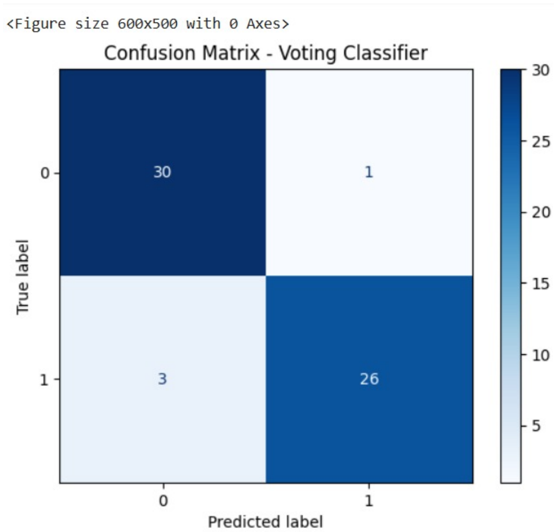

Fig. 7. voting classifier

```
  warnings.warn(
Stacked Model Accuracy: 0.9333
/usr/local/lib/python3.11/dist-package
  warnings.warn(
```

Fig. 8. Accuracy using stack model

References

[1] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–77

[2] Kim, J., Canny, J. (2017). Interpretable Learning for Self-Driving Cars by Visualizing Causal Attention. In Proceedings of the IEEE International Conference on Computer Vision (ICCV).

[3] Huval, B., Wang, T., Tandon, S., et al. (2015). An Empirical Evaluation of Deep Learning on Highway Driving. arXiv preprint arXiv

[4] Pan, X., Shi, J., Luo, P., Wang, X., Tang, X. (2018). Spatial As Deep

[5] Cordts, M., Omran, M., Ramos, S., et al. (2016).

[6] Agarwal, S., Haloi, M., Sharma, D. (2020).

[7] TensorFlow. (n.d.). An end-to-end open source platform for machine learning.

[8] PyTorch. (n.d.). An open source machine learning framework

[9] KITTI Vision Benchmark Suite. (n.d.). Autonomous driving dataset for computer vision research.

[10] Chakravarthi, M., Yadav, V. (2021).

[11] GitHub Repository - Ultra Fast Lane Detection. (2020).

[12] Google Colab. (n.d.). Free GPU-powered Jupyter Notebooks.

[13] OpenCV. (n.d.). Open Source Computer Vision Library