# LANE DETECTION USING CNN

AMAN KHANNA
*Computer Science and Engineering*
*Manipal Institute of Technology*
Manipal, Karnataka, India
aman.khanna@learner.manipal.edu

SWATHI MOHAN
*Computer Science and Engineering*
*Manipal Institute of Technology*
Manipal, Karnataka, India
swathi.m11@learner.manipal.edu

CALVIN JOHN MACHADO
*Computer Science and Engineering*
*Manipal Institute of Technology*
Manipal, Karnataka, India
calvin.machado@learner.manipal.edu

CHITRA A C
*Computer Science and Engineering*
*Manipal Institute of Technology*
Manipal, Karnataka, India
chitra.c@learner.manipal.edu

*Abstract*— **Lane keeping is a fundamental task in autonomous driving, yet it poses a significant challenge for computers. Unlike humans, computers lack an inherent understanding of road markings and pixel shifts in video feeds. This paper explores the application of deep learning, specifically Convolutional Neural Networks (CNNs), to enhance lane detection. By leveraging the capabilities of CNNs and training on driving video frames, we aim to develop a robust and efficient model for accurate lane detection.**

## I. INTRODUCTION

The ability to keep a vehicle in its lane is second nature to human drivers, but a complex task for computers. They lack the intuitive understanding of road markings and pixel nuances in video feeds. Traditional computer vision techniques demand significant manual input.

This project builds on this foundation by exploring deep learning, particularly Convolutional Neural Networks (CNNs), for lane detection. CNNs excel in image tasks, making them ideal for training on driving video frames. By leveraging their mathematical properties, we aim to outperform traditional computer vision methods.

This transition not only represents a technical leap but also addresses consumer acceptance. Demonstrating a computer's consistent lane sensing, possibly through in-vehicle displays, is pivotal in building trust in automated driving systems. In summary, this project introduces a shift from manual-intensive techniques to a deep learning-based approach for lane detection, with the goal of surpassing human proficiency and advancing autonomous driving capabilities.

## II. METHODOLOGY

### A. Data Collection and Preprocessing

The data collection process involved capturing video frames, including challenging scenarios like night and rainy conditions. A significant portion of the initial image data was discarded due to poor quality.

To address potential overfitting, only one out of every ten images was used for training. Labels were generated by outlining lanes in red. An issue arose with the initial code for sliding windows, particularly on curves, which was rectified to improve accuracy.

Label distribution was uneven, especially for curved lines. To address this, the process was re-run on a subset of images, focusing on videos with predominantly curved roads. Additional data from a separate video source was introduced to train the model for different distortions, necessitating a distinct perspective transformation.

Histogram analysis identified images on the periphery of the data distribution. These were used to generate "new" data through rotations, aimed at reducing the risk of overfitting to straight lines.

The first approach involved using images that had undergone several transformations. This included perspective transformation, which adjusted the view to look as if it was from above, resizing to make them smaller, converting to grayscale which removes color information, and normalizing the images, adjusting their values to a standard range.

In contrast, the fully convolutional model took a simpler route. It used road images that were only resized to be smaller. Batch Normalization was applied, which helps stabilize and speed up the training process. Additionally, this approach generated lane drawings directly as labels for the model.

Essentially, the two approaches differ in the level of preprocessing applied to the images before feeding them into the model. The first approach involved more complex transformations, while the second approach kept the images closer to their original form with just resizing and some normalization.

Data augmentation techniques, including rotations, horizontal flips, and merging lane image labels, were employed to expand the training dataset.

Total number of images - 12764

### B. Convolution Neural Network Architecture

A Convolutional Neural Network (CNN) is a type of deep learning model specifically designed for processing grid-like data, such as images or videos. CNNs are highly effective in tasks like image recognition, object detection, and more due to their ability to automatically learn hierarchical features from the data.

- Input: This is where the image data is fed into the network. The number of channels (e.g., Red, Green, and Blue for RGB images) depends on the image type.
- Conv: Short for Convolutional, this layer applies filters to the input. These filters learn to recognize different features in the image.
- ReLU: Rectified Linear Unit is an activation function that introduces non-linearity into the network. It helps the model learn more complex relationships.

- Pool: Pooling layers down-sample the spatial dimensions of the data. This reduces computational complexity and helps in focusing on the most important features.
- Flatten: This layer converts the multi-dimensional data from the previous layers into a one-dimensional vector. This is necessary before passing it to fully connected layers.
- Fully Connected: Also known as dense layers, these layers connect every neuron from the previous layer to every neuron in the current layer. They are used for making predictions based on the learned features.
- Output: The final layer provides the network's output. For example, in a classification task, this layer may output class probabilities.

The specific architecture and the number of layers can vary greatly depending on the complexity of the task and the available data. More complex models may have additional layers, while simpler tasks might require fewer layers.

### C. *Training and Evaluation*

The training and evaluation process for the lane detection CNN involved several stages and refinements:

1. Initial CNN Model with Perspective Transformed Images:
   - The first model used perspective-transformed images as input.
   - It consisted of four convolutional layers with decreasing filter sizes, followed by pooling, flattening, and four fully-connected layers, ending in six outputs for the lane coefficients.
   - ReLU activation function was used throughout the layers.
   - Dropout and image augmentation techniques were applied to prevent overfitting.
   - Mean-absolute error was found to be a more effective loss function than mean-squared error.
   - Data was shuffled for better representation of different videos.
   - The dataset was split into training and validation sets for performance assessment.
2. Transition to Road Images:
   - The second model used regular road images without perspective transformation.
   - A Crop layer was added to remove the top third of the image.
   - The model demonstrated effectiveness even without perspective transformation.
3. Challenges with Lane Drawing Transformation:
   - Despite successful coefficient prediction, drawing the lanes in a perspective-transformed space for final output caused issues for generalization.
   - Requiring a specific transformation matrix for new data posed a significant challenge.
4. Activation Map Exploration:

An activation map, is a visual representation of the areas in an input image that triggered the activation of specific neurons in a particular layer of the network. When an image is fed into a neural network, each layer contains neurons that respond to specific features or patterns. The activation map shows which regions of the input image were most influential in causing a particular neuron to activate. This helps us understand what features the network is focusing on during the learning process.

   - The possibility of directly observing activation maps to understand the CNN's focus was explored.
   - However, it was observed that the model activated differently for curves and straight lines, making direct use of activation maps challenging.

5. Transfer Learning for Activation Maps:
   - Transfer learning attempt: An effort was made to utilize a pre-trained model, which had learned to recognize complex features in data from a previous task. This model was considered potentially more robust in identifying patterns compared to training a new model from scratch.
   - Replacement of final fully-connected layer: To adapt the pre-trained model for a new task, the final fully-connected layer, which is responsible for making predictions, was replaced. This allows the model to be fine-tuned for the specific new task.
   - Elusive consistent activation patterns: Despite these efforts, achieving consistent activation patterns, which indicate which areas of the input data were most influential for specific neurons, proved to be challenging. This suggests that the adaptation of the pre-trained model for the new task may not have been entirely successful, or that the new task may have unique complexities that were not adequately addressed by the transfer learning approach.

6. Exploration of Image Segmentation with SegNet:
   - SegNet, a fully convolutional neural network specifically designed for segmenting different components of a road in an output image, was considered.
   - It was recognized that training the model to directly predict lane drawings as output could eliminate the necessity for perspective transformation.

Training the CNN to predict lane drawings directly, without being affected by factors like camera mounting or lane spacing variations, increased the model's robustness and adaptability to different scenarios. This approach ensured accurate lane predictions without the need for specific transformation matrices, simplifying the model's implementation and making it more versatile for real-world applications.

## III. RESULTS

Fully Convolutional Neural Network (CNN) implemented for lane detection, overcoming challenges related to layer operations and size adjustments. The model was designed to output lane drawings in the green channel, which were later merged with the original road image.

The chosen input size of 80x160x3 proved optimal for accurate lane predictions, with gray scaling omitted to retain visibility of yellow lines on light pavement. Labels were normalized to a range of 0 to 1 for 'G' pixel values.

### A. Final model

- RELU Activation: This activation function was used to introduce non-linearity into the model, enabling it to learn complex patterns.
- Specific Convolution Parameters: Parameters like strides and padding were carefully chosen for the convolutional layers, contributing to the model's performance.
- Extensive Dropout: Dropout, a regularization technique, was applied to prevent overfitting. It helps by randomly "dropping out" a certain percentage of neurons during training, reducing reliance on specific neurons.
- Loss Function: Mean Squared Error (MSE) was chosen as the loss function. It measures the average of the squared differences between predicted and actual values. Surprisingly, it performed better than other options in this context.
- Image Augmentation Findings: Contrary to expectation, applying image augmentation techniques did not lead to enhanced model robustness. In fact, omitting augmentation resulted in better performance.

This combination of activation function, convolution parameters, dropout, loss function, and augmentation strategy contributed to the effectiveness of the final model.

### B. Evaluation and Validation

After 10 epochs, the model achieved impressive Mean Squared Error values of 0.0064 for training and 0.0054 for validation, demonstrating significant improvement over previous models.

The model was tested on various videos, excelling even on challenging segments with obstructed lanes. It performed exceptionally well on areas that previous models struggled with.

The model successfully performed on the test video, even though it had never been trained on a single frame of it. It surpassed the first benchmark by outperforming the Computer Vision-based model, which had previously failed on this video. The deep learning model's performance, both in robustness and speed, clearly marked an improvement over traditional CV-based technique.

Overall, the Fully Convolutional Neural Network demonstrated superior performance and efficiency in lane detection, showcasing its potential as an advancement in this domain.

## IV. CONCLUSION

This project encompassed a comprehensive journey from video data collection to training a neural network for lane detection. Key steps included data curation, image processing, labeling, and refining the lane-drawing process.

Additional efforts were made to balance label distributions and augment data for better model performance. The transition to a fully convolutional neural network marked a significant breakthrough, greatly enhancing speed and accuracy.

Two major challenges included dataset curation, a time-consuming task with difficulty in assessing sufficiency, and selecting the optimal model architecture. Ultimately, the adoption of a fully convolutional approach proved pivotal.

This project exemplifies the power of deep learning in lane detection, emphasizing the critical roles of dataset curation and model choice. The implementation of a fully convolutional approach addressed critical issues, leading to a robust and efficient lane detection system.

## V. IMPROVEMENT

One prospective enhancement for the model could involve the implementation of a Recurrent Neural Network (RNN). In the current model version, a smoothing technique averaging across five frames is applied to mitigate issues in single-frame detection. However, an RNN has the capability to directly consider previous frames, allowing it to understand the relevance of prior detections to the current frame. This could lead to a reduction in erratic predictions.


## ACKNOWLEDGMENT

We would like to express our deepest gratitude to God for providing us with the strength, wisdom, and perseverance throughout this project.

We are also immensely thankful to our Head of Department in Computer Science and Engineering, Mr. Krishna Moorthy, for his constant support and guidance.

A special thanks to our Computer Vision teacher, Ms. Smitha Pai, for her invaluable knowledge and mentorship that greatly contributed to the success of this project.


## REFERENCES


[1] M. Bojarski et al., "End-to-End Learning of Lane Following for Autonomous Vehicles," arXiv, 2016. [Online]. Available: https://arxiv.org/abs/1604.07316

[2] A. Eldesokey, et al., "Robust Lane Detection and Tracking in Challenging Scenarios," in IEEE Intelligent Vehicles Symposium (IV), 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7995733

[3] G. Oliveira and W. Burgard, "Towards End-to-End Lane Detection: an Instance Segmentation Approach," arXiv, 2020. [Online]. Available: https://arxiv.org/abs/2008.06944