VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI-590014

Project Phase II Report
on

"REAL TIME ROAD LANE DETECTION SYSTEM USING COMPUTER VISION"

Submitted in the partial fulfillment of the requirement for the award of

Bachelor of Engineering

in

InformationScience and Engineering

Submitted By

R AJAY	**(1DT17IS063)**
R SHREYAS	**(1DT17IS064)**
SAI KARTHIK P K	**(1DT17IS072)**

Under the guidance of

Dr. KARTHIK S A
Assistant Professor
Dept. of Information Science& Engineering,
DSATM, Bangalore.

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
All B.E branches Accredited 3 years by NBA, New Delhi (Validity : 26-07-2018 to 30-06-2021)
DAYANANDA SAGAR ACADEMANY OF TECHNOLOGY& MANAGEMENT
Udayapura, Kanakapura Main Road, Opp. Art of Living, Bangalore-82
2020 -2021

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

(Affiliated to Visvesvaraya Technological University,Belagavi & Approved by AICTE,New Delhi)
All B.E branches Accredited 3 years by NBA, New Delhi (Validity : 26-07-2018 to 30-06-2021)
Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore- 560082

Department of Information Science and Engineering

CERTIFICATE

This is to certify that the Project Phase II and Seminar (17ISP85) report entitled **"REAL TIME ROAD LANE DETECTION SYSTEM USING COMPUTER VISION"** is a bonafide work carried out by **R AJAY(1DT17IS063), R SHREYAS(1DT17IS064), SAI KARTHIK P K (1DT17IS072)** in the partial fulfillment of the requirement for the award of degree in **Bachelor of Engineering in Information Science and Engineering** for Visvesvaraya Technological University, Belagavi, for the year **2020-2021**. The report has been approved as it satisfies the academic requirements with respect of the Project phase I and Seminar prescribed for Bachelor of Engineering Degree.

Dr. Karthik S A	Dr. Sumithra Devi K. A	Dr. B.R. Lakshmikantha
Assistant Professor Project Guide Dept of ISE DSATM	Dean Academics, Professor & Head of Dept. of ISE DSATM	Principal DSATM

ABSTRACT

Detection of lanes is an essential module for autonomous vehicles and advanced driver assistance systems (ADAS). Many state of the art methods for lane detection have been suggested in recent years. Although, these techniques focus on identifying the lane from a single frame, and they usually provide arguably dissatisfying performance in dealing with certain extreme situations such as degradation of the lane line, large shadows, significant occlusion of vehicles, noisy inputs of images, etc. Practically, lanes are supposed to be on-road continuous line structures. Hence, a lane that cannot be precisely detected in the live frame can be extrapolated from the information of previous frames. Therefore, we expect to use multiple frames from a continuous driving scenario to approach lane detection.

For this reason, a hybrid architecture- combination of a convolution neural networks (CNN) and a recurrent neural networks (RNN). Keeping in mind the general poor conditions of roads and lanes across the Indian subcontinent, in an attempt to train the model for optimum robustness, we intend to perform comprehensive experiments on two massive datasets and then our own dataset containing driving scenes across multiple Indian roads.

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of the project phase I and seminar would be incomplete without the mention of those people, who made it possible through their guidance, support and encouragement. We are thankful to all the people who have directly and indirectly involved in this project work.

We would like to thank the **VTU, Belagavi,** for having this project work as part of its curriculum, which gave us a wonderful opportunity to work on our research and presentation abilities.

We extend our deep sense of sincere gratitude to **Dr. B R Lakshmikantha, Principal,** DSATM, Bengaluru, for providing us an opportunity to finish the necessary work.

We express our heartfelt sincere gratitude to **Dr. Sumithra Devi K.A. Dean Academics, Prof. & Head, Department of Information Science and Engineering**, DSATM, Bengaluru, for her valuable suggestions and support.

We thank our project coordinators **Dr. M Kusuma, Professor, Dept of ISE and Dr. G Manjula, Associate professor, Dept of ISE** for all the support extended to us.

We wish to express our sincere thanks to our internal guide **Dr. Karthik S A, Assistant Professor, Department of Information Science and Engineering**, DSATM for the expert, sincere and valuable guidance and encouragement for the successful completion of the project phase-1 & seminar.

Finally, we would like to thank all the faculty members of Department of Information Science and Engineering, DSATM, Bengaluru, for their support.

R AJAY (1DT17IS063)
R SHREYAS (1DT17IS064)
SAI KARTHIK P K (1DT17IS072)

TABLE OF CONTENTS

ABSTRACT		**i**
ACKNOWLEDGEMENT		**ii**
CONTENTS		**iii**
CHAPTER No.	**CHAPTER NAME**	**PAGE No.**
Chapter 1	**INTRODUCTION**	**1**
	1.1 OVERVIEW	1
	1.2 MACHINE LEARNING	2
	1.3 DEEP LEARNING	5
	1.4 CONVOLUTIONAL NEURAL NETWORKS	5
	1.5 PROBLEM STATEMENT	6
Chapter 2	**LITERATURE SURVEY**	**7**
	2.1 Lane Detection for Prototype Autonomous Vehicle.	7
	2.2 Video Based Lane Departure Warning System using Hough Transform.	7
	2.3 CNN Based Lane Detection with Instance Segmentation	8
	2.4 Accurate and Robust Lane Detection based on Dual-View Convolutional Neutral Network.	8
	2.5 Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scenes.	9
	2.6 Road Lane Detection Using H-Maxima And Improved Hough Transform.	9
	2.7 Lane Detection and Classification for Forward Collision Warning System Based on Stereo Vision	10
	2.8 Towards Self-driving Car Using Convolutional Neural Network and Road Lane Detector	10
	2.09 VPGNet: Vanishing Point Guided Network for Lane and Road Marking Detection and Recognition	10
	2.10 Robust Lane Detection from Continuous Driving Scenes Using Deep Neural Networks	11

Chapter 3	**SYSTEM REQUIRMENT**	**12**
	3.1 SOFTWARE REQUIRMENTS	12
	3.2 HARDWARE REQUIRMENTS	12
	3.3 SOFTWARE DESCRIPTION	12
Chapter 4	**SYSTEM DESIGN**	**13**
	4.1 EXISTING METHODOLOGIES	13
	4.2 PROPOSED METHODOLOGY	16
	4.3 SYSTEM ARCHITECTURE	18
Chapter 5	**RESULTS**	**22**
	5.1 RESULT COMPARISON	22
Chapter 6	**CONCLUSION**	**26**
	6.1 CONCLUSION	26
Chapter 7	**PUBLICATIONS AND CERTIFICATES**	**27**
	7.1 SYMPOSIUM CERTIFICATES	27
	7.2 NCCSTM CERTIFICATES	28
	7.3 SYMPOSIUM PAPER (LINO Journal)	30
	7.4 FINAL PAPER (IJARIIE)	38
REFERENCES		**45**

LIST OF FIGURES

CHAPTER No.	FIGURE No.	FIGURE NAME	PAGE No.
4	4.1	Hough Line Transformation Architecture [11]	13
4	4.2	CNN Architecture [12]	14
4	4.3	Dual-View Convolutional Neural Network Architecture [4]	15
4	4.4	Flow Diagram for the proposed DCNN model	17
4	4.5	Deep-Convolutional Neural Network Architecture [10]	18
5	5.1	Output visual for a test video showing all the steps in lane detection procedure.	24
5	5.2	The Track-bars interface	24
5	5.3	Printing output values on terminal	25

LIST OF TABLES

CHAPTER No.	TABLE No.	TABLE NAME	PAGE No.
5	1	Comparison of the results of the papers	22
5	2	Performance analysis of DCNN model for TuSimple dataset.	23
5	3	Performance analysis of DCNN model for CULane dataset.	23

CHAPTER 1

INTRODUCTION

Our understanding of real-time driving scenarios has become increasingly realistic, given the substantial developments in high-precision optical sensors and electronic sensors, high-precision computer vision and effective machine learning algorithms. Amidst various other features of autonomous driving vehicles, road lane detection is the principal and most significant one. The vehicle will realize where to move when the position of the lanes are obtained, thereby avoiding the risks of overstepping into other lanes. As reported in the relevant works, there are a number of modern approaches presented with smooth and sophisticated performance. They include lane detection with geometric models some of which include such techniques focused on deep machine learning. Some also map out issues related to energy minimization and some use certain supervised learning strategies and so on to segment the road lane. Most of the above mentioned methodologies restrict their results by detecting road lanes from a single, current driving scenario frame and result in poor performance while handling extreme driving scenes such as large shadows, substantial road lane line degradation, and significant vehicle occlusion. Considering these cases, the lane could possibly be predicted with inaccuracy or projected in the wrong direction, it can be partly detected, or it cannot even be detected. The important reason for this being, the knowledge presented by the current image frame is not nearly sufficient.

1.1 Overview

The proposed method aims in using a sequence of continuous driving scenario images, combining Deep Convolution Neural Networks (DCNN) and Recurrent Neural Network (RNN), we present a hybrid deep neural network for road lane detection. The proposed model comprises of a DCNN from a wider perspective that incorporates different sequential images as a feedback and predicts the lane path in a svm classification manner in the current frame. In order to attain this segmentation objective, a fully deep convolution (DCNN) approach is used. It consists of a network of encoders and also a network of decoders, assuring that the final feature map is exactly the same size as the source images. From a local point of view, the encoder network’s summarized features of a Deep CNN are further explicated by a RNN. Following this, to handle the time series of encoded features, a long short-term memory (LSTM) network is used. The DRNN output should fuse the continuous input frame information and then be loaded into the network of DCNN decoders to assist in forecasting road lane routes.

We plan to build the network in the form of an encoder network -decoder network model in an effort to integrate CNN and RNN as a complete, end-to-end training network. Both the encoder and decoder CNN are fully convolution networks. The encoder-CNN processes each of the frames with a sequence of continuous frames as input, and derives a time series of feature vectors. The feature vectors are then given to the LSTM network as sources for the lane-information prediction. In order to generate a probability distribution for the lane prediction, the LSTM output is then loaded into the decoder-CNN. The probability vector for the lane is the same scale as the source images.

1.2 Machine Learning

Machine learning is a branch of artificial intelligence (AI) focused on building applications that learn from data and improve their accuracy over time without being programmed to do so. In data science, an algorithm is a sequence of statistical processing steps. In machine learning, algorithms are 'trained' to find patterns and features in massive amounts of data in order to make decisions and predictions based on new data.

The better the algorithm, the more accurate the decisions and predictions will become as it processes more data. Today, examples of machine learning are all around us. Digital assistants search the web and play music in response to our voice commands. Websites recommend products and movies and songs based on what we bought, watched, or listened to before. Robots vacuum our floors while we do something better with our time. Spam detectors stop unwanted emails from reaching our inboxes. Medical image analysis systems help doctors spot tumors they might have missed and the first self-driving cars are hitting the road. We can expect more. As big data keeps getting bigger, as computing becomes more powerful and affordable, and as data scientists keep developing more capable algorithms, machine learning will drive greater and greater efficiency in our personal and work lives.

Working of Machine Learning:

There are four basic steps for building a machine learning application (or model). These are typically performed by data scientists working closely with the business professionals for whom the model is being developed.

Step 1: Select and prepare a training data set

Training data is a data set representative of the data the machine learning model will ingest to solve the problem it's designed to solve. In some cases, the training data is labeled data 'tagged' to call out features and classifications the model will need to identify. Other data is unlabeled, and the model will need to extract those features and assign classifications on its own.

Step 2: Choose an algorithm to run on the training data set

Again, an algorithm is a set of statistical processing steps. The type of algorithm depends on the type (labeled or unlabeled) and amount of data in the training data set and on the type of problem to be solved.

Common types of machine learning algorithms for use with labeled data include the following:

Regression algorithms: Linear and logistic regression are examples of regression algorithms used to understand relationships in data. Linear regression is used to predict the value of a dependent variable based on the value of an independent variable. Logistic regression can be used when the dependent variable is binary in nature: A or B. For example, a linear regression algorithm could be trained to predict a salesperson's annual sales (the dependent variable) based on its relationship to the salesperson's education or years of experience (the independent variables.) Another type of regression algorithm called a support vector machine is useful when dependent variables are more difficult to classify.

Decision trees: Decision trees use classified data to make recommendations based on a set of decision rules. For example, a decision tree that recommends betting on a particular horse to win, place, or show could use data about the horse (e.g., age, winning percentage, pedigree) and apply rules to those factors to recommend an action or decision.

Instance-based algorithms: A good example of an instance-based algorithm is K-Nearest Neighbor or k-nn. It uses classification to estimate how likely a data point is to be a member of one group or another based on its proximity to other data points.

Step 3: Training the algorithm to create the model

Training the algorithm is an iterative process–it involves running variables through the algorithm, comparing the output with the results it should have produced, adjusting weights and biases within the algorithm that might yield a more accurate result, and running the variables again until the algorithm returns the correct result most of the time.

The resulting trained, accurate algorithm is the machine learning model—an important distinction to note, because 'algorithm' and 'model' are incorrectly used interchangeably, even by machine learning mavens.

Step 4: Using and improving the model

The final step is to use the model with new data and, in the best case, for it to improve in accuracy and effectiveness over time. Where the new data comes from will depend on the problem being solved. For example, a machine learning model designed to identify spam will ingest email messages, whereas a machine learning model that drives a robot vacuum cleaner will ingest data resulting from real-world interaction with moved furniture or new objects in the room.

Machine learning methods fall into three primary categories:

Supervised machine learning:

Supervised machine learning trains itself on a labeled data set. That is, the data is labeled with information that the machine learning model is being built to determine and that may even be classified in ways the model is supposed to classify data. For example, a computer vision model designed to identify purebred German Shepherd dogs might be trained on a data set of various labeled dog images.

Unsupervised machine learning:

Unsupervised machine learning ingests unlabeled data—lots and lots of it—and uses algorithms to extract meaningful features needed to label, sort, and classify the data in real-time, without human intervention. Unsupervised learning is less about automating decisions and predictions, and more about identifying patterns and relationships in data that humans would miss. Take spam detection, for example—people generate more email than a team of data scientists could ever hope to label or classify in their lifetimes. An unsupervised learning algorithm can analyze huge volumes of emails and uncover the features and patterns that indicate spam.

Semi-supervised learning:

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data to train a supervised learning algorithm.

1.3 Deep Learning

Deep learning is a subset of machine learning (all deep learning is machine learning, but not all machine learning is deep learning). Deep learning algorithms define an artificial neural network that is designed to learn the way the human brain learns. Deep learning models require large amounts of data that pass through multiple layers of calculations, applying weights and biases in each successive layer to continually adjust and improve the outcomes. Deep learning models are typically unsupervised or semi-supervised. Reinforcement learning models can also be deep learning models. Certain types of deep learning models—including convolutional neural networks (CNNs) and recurrent neural networks (RNNs)—are driving progress in areas such as computer vision, natural language processing (including speech recognition), and self-driving cars.

1.4 Convolutional Neural Networks

A CNN is type of a DNN consists of multiple hidden layers such as convolutional layer, RELU layer, pooling layer and fully connected a normalized layer. CNN shares weights in the convolutional layer reducing the memory footprint and increases the performance of the network. The important features of CNN lie with the 3D volumes of neurons, local connectivity and shared weights. A feature map is produced by convolution layer through convolution of different sub regions of the input image with a learned kernel. Then, anon-linear activation function is applied through ReLu layer to improve the convergence properties when the error is low. In pooling layer, a region of the image/feature map is chosen and the pixel with maximum value among them or average values is chosen as the representative pixel so that a 2x2 or 3x3 grid will be reduced to a single scalar value.

This results a large reduction in the sample size. Sometimes, traditional Fully-Connected (FC) layer will be used in conjunction with the convolutional layers towards the output stage. In CNN architecture, usually convolution layer and pool layer are used in some combination. The pooling layer usually carries out two types of operations viz. max pooling and means pooling. In mean pooling, the average neighborhood is calculated within the feature points and in max pooling it is calculated within a maximum of feature points. Mean pooling reduces the error caused by the neighborhood size limitation and retains background information. Max pooling reduces the convolution layer parameter estimated error caused by the mean deviation and hence retains more texture information.

1.5 Problem Statement

Today there is a lot of research on ADAS where everything from “Lane Departure Warning (LDW)” to “Full autonomous driving” is investigated. However, there is a need for research about the integration of safety critical applications and non-safety critical applications on a mixed criticality platform where the two applications are isolated from each other using virtualization. For an example Autosar, which is a partnership for development of software founded by major players in the automotive industry does address mixed criticality systems in the sense that they recognize that the standards must be supported on their platforms

The Deep Convolutional Neural Networks, with superior object detection performance in natural images, is considered as the best methodology in medical imaging and object detection applications as well. Neural systems have been utilized widely in the detection of road lanes. Be that as it may, for the specific instance of lane detection, CNNs are progressively fitting: since disease is basically recognized by the calcification designs, nearby associated examples are increasingly significant for the order procedure. Be that as it may, on the off chance that we utilize a completely associated neural system, where all regions of the picture influence one another, over the top superfluous loads are figured and off base functionalities start to bring into the detection methodology. Deep-learning has become a dominant method in a variety of complex tasks such as image classification and object detection. The proposed Deep CNN model provides better accuracy and achieves better performance.

Lane detection is a challenging problem. It has attracted the attention of the computer vision community for several decades. Essentially, lane detection is a multifeature detection problem that has become a real challenge for computer vision and machine learning techniques. Although many machine learning methods are used for lane detection, they are mainly used for classification rather than feature design. But modern machine learning methods can be used to identify the features that are rich in recognition and have achieved success in feature detection tests. However, these methods have not been fully implemented in the efficiency and accuracy of lane detection.

CHAPTER 2

LITERATURE SURVEY

Number of researches have been made in the field of lane detection and prediction. We have referenced the following works to understand the methodologies that they have implemented, how they are advantageous in solving the problem statement and to probe their disadvantages and limitations in an effort to overcome them in our implementation.

2.1 Lane Detection for Prototype Autonomous Vehicle [1].

In the design of the algorithm, firstly the mutation from RGB color space to HSV color space is made. The cost of detecting ROI in the image was reduced by the Grayscale image conversation method and the process was accelerated. Canny edge detection, Hough transformation and Sobel filter methods are the methods used to find the lane lines. Canny edge detection is a method of edge detection. The lanes are lines and have edges. Hough transformation method is a method that finds and shows shapes. Since the lane lines have a shape, lane lines can be found by the Hough transform method. Sobel filter method is a separate method used to find the edge. It is seen in all the studies obtained the use of the edge detection algorithm in images with a grayscale color space is closer to the edge information in the actual image.

In this paper, an autonomous vehicle prototype model that detect lanes via image processing techniques, which are a major part of autonomous vehicle technology is presented. Some image processing algorithms such as canny edge detection, Sobel filter, etc. are used to provide autonomous movement capability. They were implemented and tested successfully on the prototype vehicle.

2.2 Video Based Lane Departure Warning System using Hough Transform [2].

This paper proposed the following methods to warm drivers when they are going out of their lane. The methods used are: Smoothing: It is a process where the noise is eliminated using filters like 2d FRI filter to obtain a binary edge map and finally hough transform is used to detect edges and boundaries. Detecting line: we use hough transform for detecting lane lines which are subject to scenarios like short brakes in the lane which is caused due to noise and various other parameters. Tracking lane: we calculate distance between the lanes to find out the road width that a vehicle follows through its journey.

The proposed method achieved an accuracy of 96.5% under cloudy weather conditions and 98.7% under sunny weather conditions for detecting a straight lane. The same reduced to 90.4% and 94.3% while detecting a curved road lane.

2.3 CNN Based Lane Detection with Instance Segmentation [3].

In lane detection, in order to save computing resources, they binarize the image. The segmented branch of the two-branch network outputs a binary segmentation map, which divides the lane line and non-lane line parts. bounded inverse class weighting is used for lane pixel segmentation. Then, clustering is completed by iteration and combining the clustering algorithm with the loss function. Output of the two-branch network is a set of pixels for each lane line. It is not ideal to fit a polynomial with these pixels in the input image space, because a higher-order polynomial is needed to process the curve lane. In curve fitting the image is first transformed by inverse perspective and projected into the “bird’s eye view", where the lanes are parallel to each other, so the curve lane can be fitted by a second to third order polynomial. This model obtained an AUC score of 0.9757 on normal roads and AUC of 0.9410 for curved roads.

2.4 Accurate and Robust Lane Detection based on Dual-View Convolutional Neutral Network [4].

In this paper, they propose a Dual-View Convolutional Neutral Network (DVCNN) framework for lane detection. First, to improve the low precision ratios of literature works, a novel DVCNN strategy is designed where the front-view image and the top-view one are optimized simultaneously. In the front-view image, they exclude false detections including moving vehicles, barriers and curbs, while in the top-view image non-club-shaped structures are removed such as ground arrows and words. Second, they present a weighted hat-like filter which not only recalls potential lane line candidates, but also alleviates the disturbance of the gradual textures and reduces most false detections. Third, different from other methods, a global optimization function is designed where the lane line probabilities, lengths, widths, orientations and the amount are all taken into account. After the optimization, the optimal combination composed of true lane lines can be explored.

This method achieves an average recall value of 92.80% and a precision of 95.48%. The DVCNN detection module occupies 90% computation time, which can be optimized by several techniques, such as multi-threading, GPU speedup and so on.

2.5 Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene [5].

First, they develop a multitask deep convolutional network, which simultaneously detects the presence of the target and the geometric attributes (location and orientation) of the target with respect to the region of interest. Second, a recurrent neuron layer is adopted for structured visual detection. The recurrent neurons can deal with the spatial distribution of visible cues belonging to an object whose shape or structure is difficult to explicitly define. Both the networks are demonstrated by the practical task of detecting lane boundaries in traffic scenes. The multitask convolutional neural network provides auxiliary geometric information to help the subsequent modeling of the given lane structures. The recurrent neural network automatically detects lane boundaries, including those areas containing no marks, without any explicit prior knowledge or secondary modeling. Both the CNN and RNN detectors have been shown to be effective in detecting lanes in practical traffic scenes outperforming conventional detectors, with an average AUC score of 0.99 and 0.94 for the RNN and CNN units respectively.

2.6 Road Lane Detection Using H-Maxima And Improved Hough Transform [6].

A fast and improved algorithm with the ability to detect unexpected lane changes is aimed in this paper. A short segment of a long curve has relative low curvature which is approximated as a straight line. Based on the characteristics of physical road lane, this paper presents a lane detection technique based on H-MAXIMA transformation and improved Hough Transform algorithm which first defines the region of interest from input image for reducing searching space; divided the image into near field of view and far field of view. In near field of view, Hough transform has been applied to detect lane markers after image noise filtering. The proposed method has been developed using image processing programming language platform and was tested on collected video data. Promising result was obtained with high efficiency of detection.

The whole processing time of the proposed system is about 0.0337 seconds on average and the error rate is 4.77%. The simulation has been processed on a DELL Inspiron laptop with CPU N450@1.66GHz. The result shows that on average the efficiency has been raised significantly.

2.7 Lane Detection and Classification for Forward Collision Warning System Based on Stereo Vision [7].

This paper presents a lightweight stereo vision based driving lane detection and classification system to achieve the ego-car's lateral positioning and forward collision warning to aid advanced driver assistance systems (ADAS). For lane detection, they design a self-adaptive traffic lanes model in Hough Space with a maximum likelihood angle and dynamic pole detection region of interests (ROIs), which is robust to road bumpiness, lane structure changing while the ego-car's driving and interferential markings on the ground. Besides, the 3-D information acquired by stereo matching is used to generate an obstacle mask to reduce irrelevant objects' interfere and detect forward collision distance. For lane classification, a convolutional neural network is trained by using manually labeled ROI from KITTI data set to classify the left/right-side line of host lane so that it can provide significant information for lane changing strategy making in ADAS.

2.8 Towards Self-driving Car Using Convolutional Neural Network and Road Lane Detector [8].

YOLO (You Only Look Once) is one of the real-time CNN methods that aims to detect objects from images. On the other hand, Road Lane Detector is used to detect road track from video's frames and to provide additional information that can be helpful for the decision-making process of the self-driving car. In this paper, they use YOLO as the object detector and polynomial regression as the road guidance in the real-world driving video simulations. They use NVIDIA GTX 1070 with 8 GB of RAM for the computations. The result shows a matching pair between those two methods for self-driving car environment and road lane guidance. The proposed method's implementations and observations are computed using Python programming language and CUDA. The model has produced accuracy of 0.87. Recall rate of the model is 0.98 and Precision of the model is 0.81.

2.9 VPGNet: Vanishing Point Guided Network for Lane and Road Marking Detection and Recognition [9].

In this paper, they propose a unified end-to-end trainable multi-task network that jointly handles lane and road marking detection and recognition that is guided by a vanishing

point under adverse weather conditions. They tackle rainy and low illumination conditions, which have not been extensively studied until now due to clear challenges. For example, images taken under rainy days are subject to low illumination, while wet roads cause light reflection and distort the appearance of lane and road markings. At night, color distortion occurs under limited illumination. As a result, no benchmark dataset exists and only a few developed algorithms work under poor weather conditions. To address this shortcoming, they build up a lane and road marking benchmark which consists of about 20,000 images with 17 lane and road marking classes under four different scenarios: no rain, rain, heavy rain, and night. They train and evaluate several versions of the proposed multi-task network and validate the importance of each task.
The resulting approach, VPGNet, can detect and classify lanes and road markings, and predict a vanishing point with a single forward pass.

2.10 Robust Lane Detection from Continuous Driving Scenes Using Deep Neural Networks [10].

The lanes that cannot be predicted accurately in one current frame may potentially be inferred out by gathering information from previous frames. Hence, lane detection by using multiple frames of a continuous driving scene is proposed by using a hybrid deep architecture of combination of CNN and Recurrent Neural Network (RNN). The idea is to extract features of continuous images using CNNs and these features of multiple frames, holding the properties of time series, are then fed into RNN block for feature learning and lane prediction. To increase the accuracy of the obtained model, smoothing techniques are implemented. The simplest measurement criterion is accuracy, which tests the overall performance of the classification based on correctly categorized pixels. The model has achieved 93.48% of accuracy with a recall value of 0.8636

CHAPTER 3

SYSTEM REQUIRMENTS

3.1 Software Requirements

1. Operating System: Windows 8 or above

2. Programming Language: Python 3 or above

3. GUI Development: Anaconda Navigator

3.2 Hardware Requirements

1. Processor: Intel i5 or above

2. Ram: 8GB or above

3. Hard Disk: 4GB or above

4. GPU: GTX1050 or above

3.3 Software Description

Step 1: Environment Setup

• Anaconda Navigator/PyCharm (version 2021.1)

• Python (Preferred version is 3.8.3)

Step 2: Dependencies required

• In Anaconda Navigator create a new environment called "rldcnn".

• In that Environment install the Dependencies using conda install or pip install.

CHAPTER 4

SYSTEM DESIGN

There is a vast assortment current techniques used for Lane detection, it is observed that even though there are several methods that achieved a noticeable improvement in the result and detection of the lane lines, there are several challenges that are not addressed by many of these methodologies. Such as: poor road conditions and lane line degradation, low illumination during night time, challenging weather conditions, etc. Here are some of the existing methods that are noteworthy.

4.1 EXISTING METHODOLOGIES

Frame Masking and Randomized Hough Line Transformation:

To detect white markings in the lane, first, we need to mask the rest part of the frame. We do this using frame masking. The frame is nothing but a NumPy array of image pixel values. To mask the unnecessary pixel of the frame, we simply update those pixel values to 0 in the NumPy array. After making we need to detect lane lines. The technique used to detect mathematical shapes like this is called Hough Transform. Hough transformation can detect shapes like rectangles, circles, triangles, and lines.

The next step is to find a collection of straight lines in the binary image using the Randomized Hough Transform (RHT). The Randomized Hough Transform operates iteratively by randomly sampling a set of points to compute a single location in the Hough space that is incremented. The architecture of Hough Line Transform is shown in Fig 4.1. To elaborate, in each iteration of the RHT, two non-zero pixels at (x1, y1) and (x2, y2) are randomly selected without replacement from the binary image.

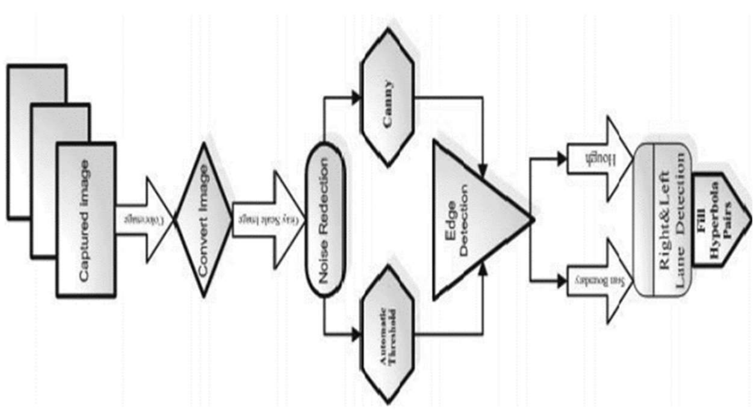


Figure 4.1: Hough Line Transformation Architecture [11]

Disadvantages:

- It still takes a huge amount of time to train the network as you would have to classify 2000 region proposals per image.
- It cannot be implemented real time as it takes around 47 seconds for each test image.
- The selective search algorithm is a fixed algorithm. Therefore, no learning is happening at that stage. This could lead to the generation of bad candidate region proposals.

CNN (Convolutional Neural Networks):

CNN includes various stack layers of convolution which learns automatically considering useful information from the input data without concentrating on pre-processing or any feature engineering-procedures. The main components of Convolution Neural Network include Layer of convolution, Layer of pooling and fully-connected layer. The general CNN architecture is shown below in Fig 4.2.

The feature-map is created in the convolution layer by applying dot product numerical operation of grids of weights all through whole substance of each example of input information. The next operation is pooling, which usually applied-after-the convolution-process. After the application of convolution process the dimensionality of the result increases hence to reduce it pooling is carried out.

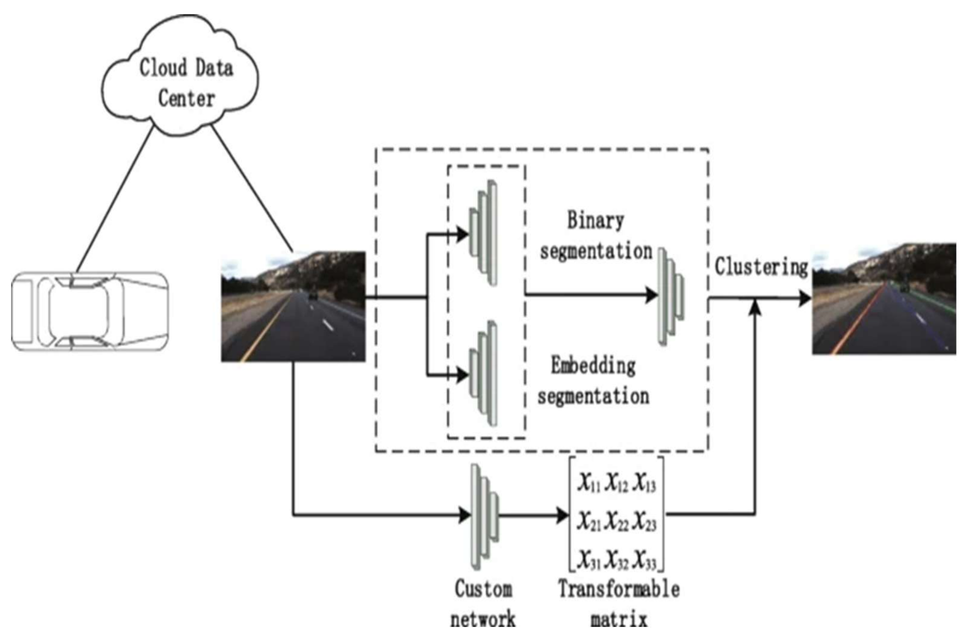


Figure 4.2: CNN Architecture [12]

Disadvantages:

- Less accuracy due to its more general nature.
- No proper region isolation thus may cause false negatives.
- Lower efficiency and higher error rate.

DVCNN (Dual View Convolutional Neutral Network):

The top-view image is inverse-perspective-mapped from the front-view one, where lane line candidates are extracted with a weighted hat-like filter. The front-view and top-view patches are utilized as input to the DVCNN framework simultaneously. The outputs of the framework correspond to the probabilities of the true lane lines. All achieved results, including lane line probabilities, lengths, widths, orientations and the amount are combined for the global optimization, where the final outputs are refined. In Fig. lane lines marked by the green color (green means the high probability) are detected accurately and robustly. The Siamese network joins two sub-network at the head, to estimate the probability of the true lane line. The data layer accepts the front-view and top-view patches as inputs, where sizes are normalized into 128 × 128 and 64 × 64 pixels respectively. The 2 sub-networks are similar, both composed of several layers of convolution, ReLU, pooling and ending with a fully-connected one. The DVCNN framework is depicted in the below Fig 4.3.

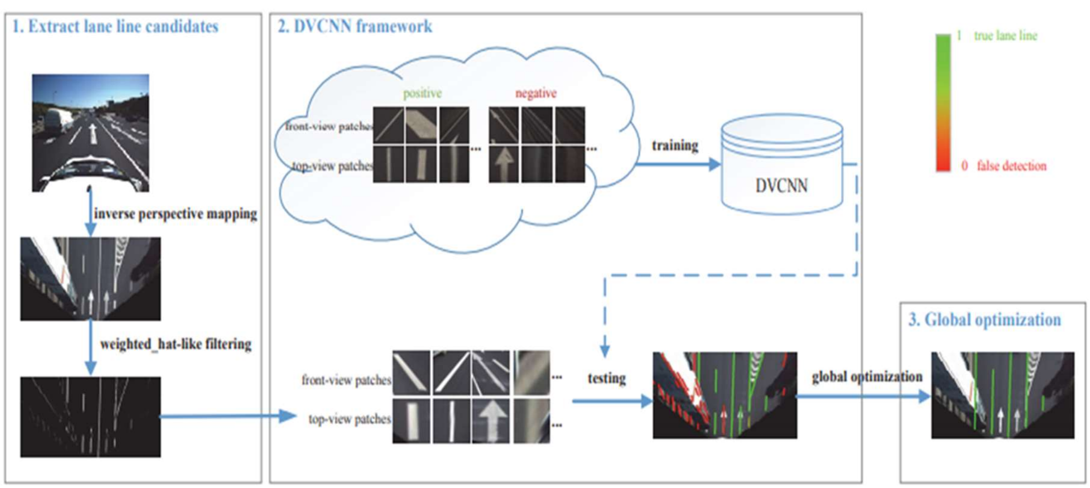


Figure 4.3: Dual-View Convolutional Neural Network Architecture [4]

Disadvantages:

- Longer processing times needed as large volume of data must be processed.
- Complex structure of algorithm makes it difficult to program.

4.2 PROPOSED METHODOLOGY

Deep-Convolutional Neural Networks (CNN):

The proposed method uses Deep Convolutional Neural Networks to detect road lanes based on a continuous input of frames.

The proposed model involves the following stages:

Pre-processing & Segmentation: In the first stage, lane regions are extracted from current frame and in that region each slice is segmented to get the region of interest (ROI).

Convolutional Neural Networks (CNN):

A CNN is composed of several kinds of layers:

Convolutional layer: creates a feature map to predict the class probabilities for each feature by applying a filter that scans the whole image, few pixels at a time.

Pooling layer (down-sampling): scales down the amount of information the convolutional layer generated for each feature and maintains the most essential information (the process of the convolutional and pooling layers usually repeats several times).

Fully connected input layer: flattens the outputs generated by previous layers to turn them into a single vector that can be used as an input for the next layer.

Fully connected layer: Applies weights over the input generated by the feature analysis to predict an accurate label.

Fully connected output layer: Generates the final probabilities to determine a class for the image.

In this method, using a sequence of continuous driving scenario images, combining Deep Convolution Neural Networks (DCNN) and Recurrent Neural Network (RNN), we present a hybrid deep neural network for road lane detection (RNN). The proposed model comprises of a DCNN from a wider perspective that incorporates different sequential images as a feedback and predicts the lane path in a svm classification manner in the current frame. In order to attain this segmentation objective, a fully deep convolution (DCNN) approach is used. It consists of a network of encoders and also a network of decoders, assuring that the final feature map is exactly the same size as the source images. From a local point of view, the encoder network's summarised features of a Deep CNN are further explicated by a RNN. Following this, to handle the time series of encoded features, a long short-term memory (LSTM) network is used. The DRNN output should fuse the continuous input frame information and then be loaded into the network of DCNN decoders to assist in forecasting road lane routes. The Flow Diagram for the proposed DCNN model is shown in Figure 4.4 below:

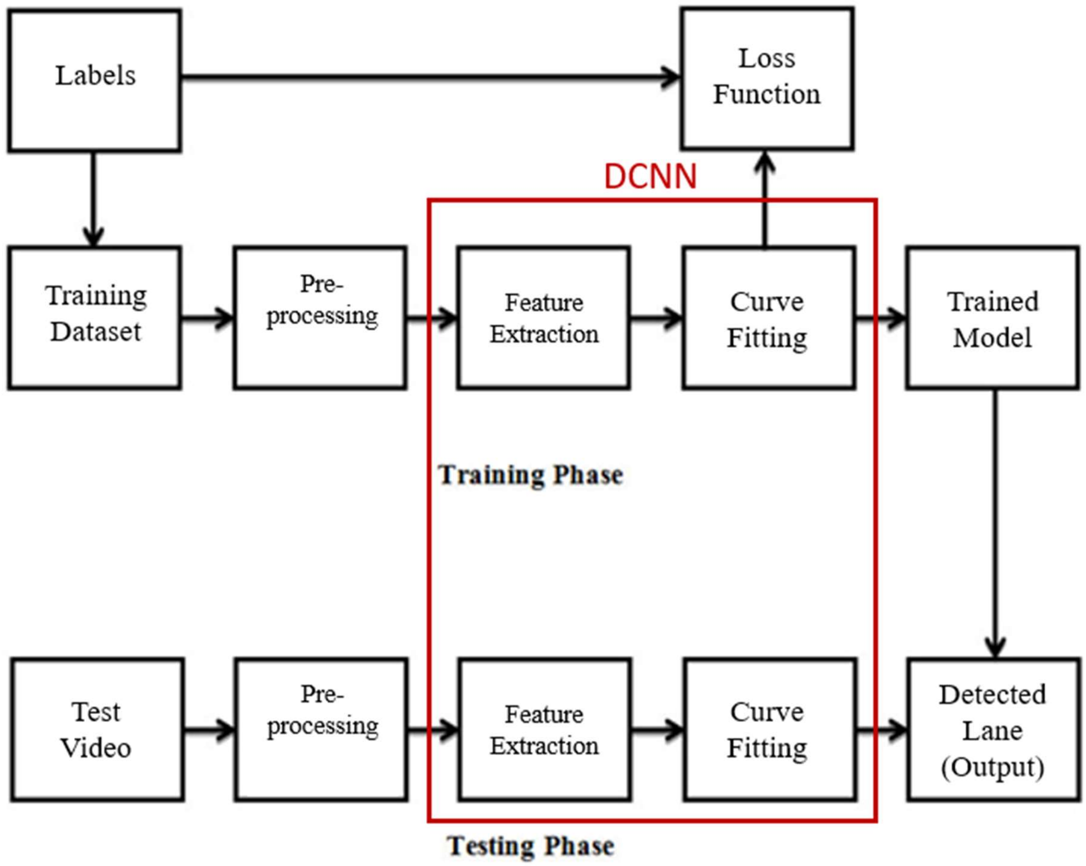


Figure 4.4: Flow Diagram for the proposed DCNN model

LSTM Network: The RNN unit in the presented network architecture considers feature vectors produced by the encoder-CNN over each image as the feedback for modelling the sequence of continuous images of driving scenarios as a time series. Various types of RNN models, such as LSTM and GRU, have been suggested to tackle the variable time-series results. An LSTM network is used in this model, which typically outclasses the conventional RNN architecture with its ability to forget insignificant details and recall only the critical characteristics by using network cells to determine whether or not a segment of information is essential. With the first unit for simultaneous feature extraction and the second for assimilation, a dual-layer LSTM is implemented.

The activations of a general ConvLSTM cell at time t can be formulated as:

$Ct = ftoCt - 1 + ito \tanh(Wxc * Xt + Whc * Ht - 1 + bc)$

$ft = \sigma(Wxf * Xt + Whf * Wt - 1 + WcfoCt - 1 + bf)$

$ot = \sigma(Wxo * Xt + Who * Wt - 1 + WcooCt - 1 + bo)$

$it = \sigma(Wxi * Xt + Whi * Wt - 1 + WcioCt - 1 + bi)$

$Ht = oto \tanh(Ct)$

where Xt denotes the input feature maps extracted by the encoder CNN at time t. Ct, Ht and Ct1 , Ht1 denote the memory and output activations at time t and t − 1, respetively. Ct, it, ft and ot denote the cell, input, forget and output gates, respectively. Wxi is the weight matrix of the input Xt to the input gate, bi is the bias of the input gate. The meaning of other W and b can be inferred from the above rule. σ(·) represents the sigmoid operation and tanh(·) represents the hyperbolic tangent non-linearities. '∗' and 'o' denote the convolutio operation and the Hadamard product, respectively.

Advantages:

- The large database helps to easily detect road lanes.
- The rate of error and efficiency is much higher than that most other methods.
- The performance is much better compared to existing methods.
- The inaccuracies caused by weathering effects are greatly reduced.

4.3 SYSTEM ARCHITECTURE

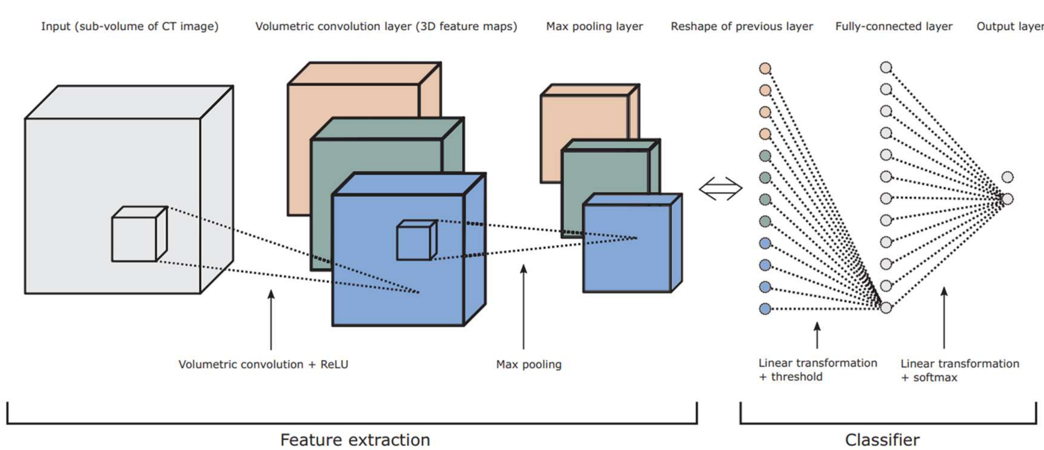


Figure 4.5: Deep-Convolutional Neural Network Architecture [10]

A Deep-CNN is type of a DNN consists of multiple hidden layers such as convolutional layer, RELU layer, Pooling layer and fully connected a normalized layer as depicted in the above architecture Fig 4.4. CNN shares weights in the convolutional layer reducing the memory footprint and increases the performance of the network. The important features of CNN lie with the 3D volumes of neurons, local connectivity and shared weights. A feature map is produced by convolution layer through convolution of different sub regions of the input image with a learned kernel. Then, anon-linear activation function is applied through ReLu layer to improve the convergence properties when the error is low. In pooling layer, a region of the image/feature map is chosen and the pixel with maximum value among them or average values is chosen as the representative pixel so that a 2x2 or 3x3 grid will be reduced to a single scalar value.

This results a large reduction in the sample size. Sometimes, traditional Fully-Connected (FC) layer will be used in conjunction with the convolutional layers towards the output stage. In CNN architecture, usually convolution layer and pool layer are used in some combination. The pooling layer usually carries out two types of operations viz. max pooling and means pooling. In mean pooling, the average neighborhood is calculated within the feature points and in max pooling it is calculated within a maximum of feature points. Mean pooling reduces the error caused by the neighborhood size limitation and retains background information. Max pooling reduces the convolution layer parameter estimated error caused by the mean deviation and hence retains more texture information.

In order to generate a probability distribution for the lane prediction, the LSTM output is then loaded into the decoder-CNN. The probability vector for the lane is the same scale as the source images.

The proposed method follows these stages:

Data Set:

The dataset we plan to use the lane scenes provided by TuSimple and CULane Datasets. The TuSimple lane data set comprises of 6,408 image sequences. These images are the front of the expressways in the United States of America. There are 20 continuous image sequences captured in one second in each series. The final image, that is, the 20th picture, is labelled with lane ground truth for each sequence. Additionally, keeping in mind the previously discussed challenges posed by Indian road lanes, we plan to create our own data set consisting of at least 20-30 image sequences, captured from a mounted camera on a car that is driven along various Indian road scenes covering a wide range of road conditions.

Image Segmentation:

The segmentation of photographs is the phase where the visual image is partitioned into several parts. This normally helps to identify artifacts and boundaries. The aim of segmentation is to simplify the transition in interpretation of a picture into concrete picture that can be clearly interpreted and quickly analyzed.

Pre-Processing:

In preprocessing stage, the median filter is used to restore the image under test by minimizing the effects of the degradations during acquisition. Various preprocessing and segmentation techniques of road lanes are discussed in. The median filter simply replaces each pixel value with the median value of its neighbors including itself. Hence, the pixel values which are very different from their neighbors will be eliminated.

Convolutional Neural Networks:

A CNN is type of a DNN consists of multiple hidden layers such as convolutional layer, RELU layer, Pooling layer and fully connected a normalized layer. CNN shares weights in the convolutional layer reducing the memory footprint and increases the performance of the network. The important features of CNN lie with the 3D volumes of neurons, local connectivity and shared weights. A feature map is produced by convolution layer through convolution of different sub regions of the input image with a learned kernel. Then, anon-linear activation function is applied through ReLu layer to improve the convergence properties when the error is low. In pooling layer, a region of the image/feature map is chosen and the pixel with maximum value among them or average values is chosen as the representative pixel so that a 2x2 or 3x3 grid will be reduced to a single scalar value. This results a large reduction in the sample size.

Deep Learning:

Deep learning composed of several layers of nonlinear nodes, combine input data with a set of weights so that assigning significance to inputs for the corresponding task the algorithm is attempting to learn in supervised and/or unsupervised behavior.

The sum of product of these input and weights is passed through activation function of nodes. The output of each layer’s is fed simultaneously as input to the subsequent layer starting from input layer. Learning can be performed in multiple levels of representations correspond to various levels of abstraction.

LSTM network:

The RNN block in the proposed network accepts feature map extracted on each frame by the encoder CNN as input. Specifically, LSTM network is employed, which generally outperforms the traditional RNN model with its ability in forgetting unimportant information and remembering the essential features. Once the end-to-end trainable neural network is designed, a back-propagation method may train the network to predict ground truth by updating the weight parameters of the convolutionary kernels and the ConvLSTM matrix.

Training:

Back-propagation algorithm is used to train the Deep CNN to detect road lanes in frame of size 5×20×20. It consists of two phases. In the first phase, a CNN consists of multiple volumetric convolution, rectified linear units (ReLU) and max pooling layers is used to extract valuable volumetric features from input data. The second phase is the classifier. It has multipleFC and threshold layers, followed by a LSTM network to perform the high-level reasoning of the neural network. During training, the random sub-volumes extracted from the frames of the training set and are normalized according to an estimate of the normal distribution of the voxel values in the dataset.

CHAPTER 5

RESULTS

The CNN model is trained with both assorted driving scenes from TuSimple dataset and from Tvt dataset. The CNN model is trained with 3000 various driving scene images at 640x480 Resolution and around 133,000 classified driving scenes under shadow and occlusion conditions from CULane Dataset.

5.1 RESULT COMPARISON

Recent Studies on Road Lane Detection using CNN.

Authors	Model	Result
Shivakumar et al	D-CNN	Accuracy = 91.84 precision = 0.4262 recall = 0.8085
Jiyong et al	SCNN and ConvGRUs	Accuracy = 97.98 precision = 0.8674 recall = 0.9562
Xianpeng et al	Dual View –CNN (DVCNN)	Precision = 95.49 Recall = 92.80
Rama Sai et al	CNN with Google Street View	Accuracy = 0.9610 precision = 0.9889 recall = 0.8732
Ze Wang et al	CNN with LSTM	TPR = 87.3% FPR = 7.7%
Wei Wang et al	CNN with instance segmentation	ACU = 0.9785
Seokju Lee et al	VPGNet	Detection Score = 0.77

Table 1. Comparison of the results of the papers.

The above table provides a comparison between the models proposed by various authors. It describes the accuracy, precision and recall values achieved by each of these proposed models. Thus, giving us a better overview of the results obtained from the mentioned methodologies.

Basic measures derived from the confusion matrix:

Error rate (ERR) is calculated as the number of all incorrect predictions divided by the total number of the dataset. The best error rate is 0.0, whereas the worst is 1.0.

Error Rate=(FP+FN)/(P+N)

Accuracy (ACC) is calculated as the number of all correct predictions divided by the total number of the dataset. The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – ERR.

Accuracy = (TP+TN)/(P+N)

Precision (PREC) is calculated as the number of correct positive predictions divided by the total number of positive predictions. It is also called positive predictive value (PPV). The best precision is 1.0, whereas the worst is 0.0.

Precision = TP/TP+FP

False positive rate (FPR) is calculated as the number of incorrect positive predictions divided by the total number of negatives. The best false positive rate is 0.0 whereas the worst is 1.0. It can also be calculated as 1 – specificity.

FP= FP/TN+FP

The performance analysis results for our model:

Sl. No	1
Dataset	Driving Scenes from TuSimple Dataset
Data Set Split	Total=6,408 Training=3,626 Testing=2,782
Error Rate (ERR)= FP+FN/P+N	4%
Precision =TP/FP+FN	0.964
Accuracy =(1-ERR)	±96%

Table 2. Performance analysis of DCNN model for TuSimple dataset.

Sl. No	2
Dataset	Driving Scenes from CULane Dataset
Data Set Split	Total= 133,235 Training= 88,880 Testing= 34,680
Error Rate (ERR)= FP+FN/P+N	18%
Precision =TP/FP+FN	0.827
Accuracy =(1-ERR)	±82%

Table 3: Performance analysis of DCNN model for CULane dataset.

The output window consists of four different windows as shown in figure 5.1. The first window depicts the final output video of the detected lane as a live video pipeline, another window shows the corresponding live video pipeline for all the processes that the test video is subjected to. This is done to increase transparency on the process of lane detection. The third window as depicted in figure 5.2 shows the track bar parameters that allow us to adjust the dimensions of the area of transformation in real time so that we can try to increase the accuracy during difficult lane conditions in real time. The fourth window as depicted in figure 5.3 shows the live detected lane curvature parameters. Also, the predicted position of the vehicle and the supposed directed of steering according to the detected lane is printed live every 0.5 seconds on the terminal. This can be used as an input for an external automated vehicle steering module. The model is also tested with multiple test videos with varying degrees of complexity and is observed to be robust in most cases.

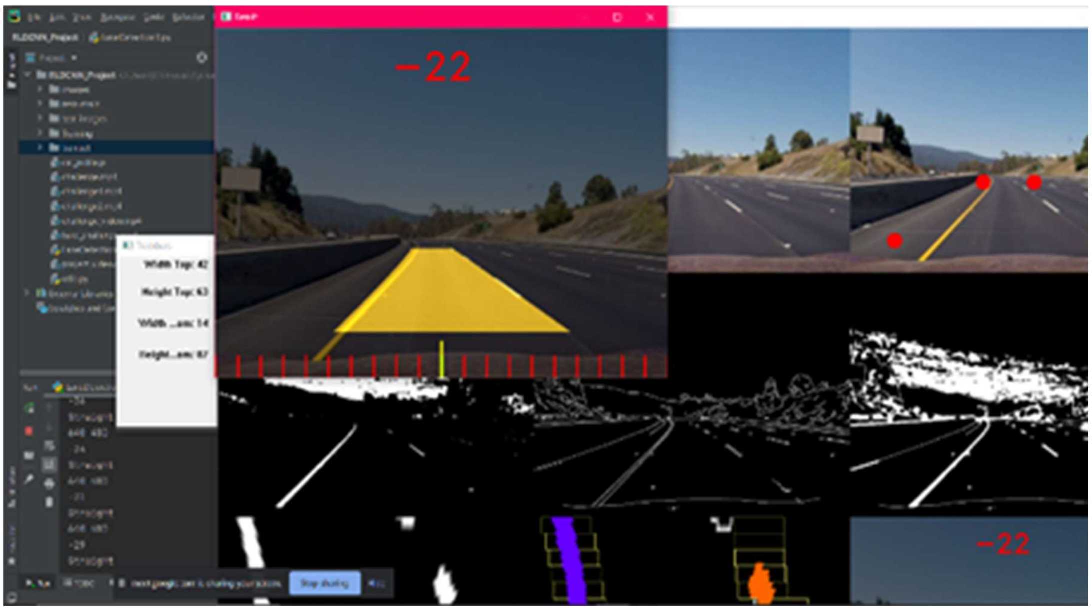


Figure 5.1: Output visual for a test video showing all the steps in lane detection procedure.

Figure 5.2: The Track-bars interface- they allow us to define the dimensions of area of transformation in real time, this providing higher accuracy during excessive road curvatures.

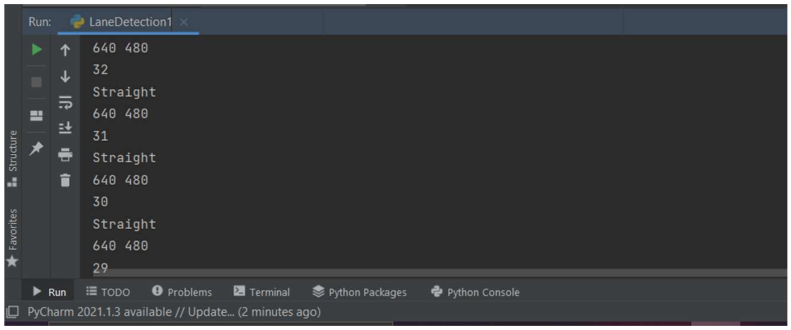


Figure 5.3: Printing output values on terminal- The output window size, the lane curvature value and the expected driving value to maintain centring is printed on the terminal every 0.5 seconds.

CHAPTER 6

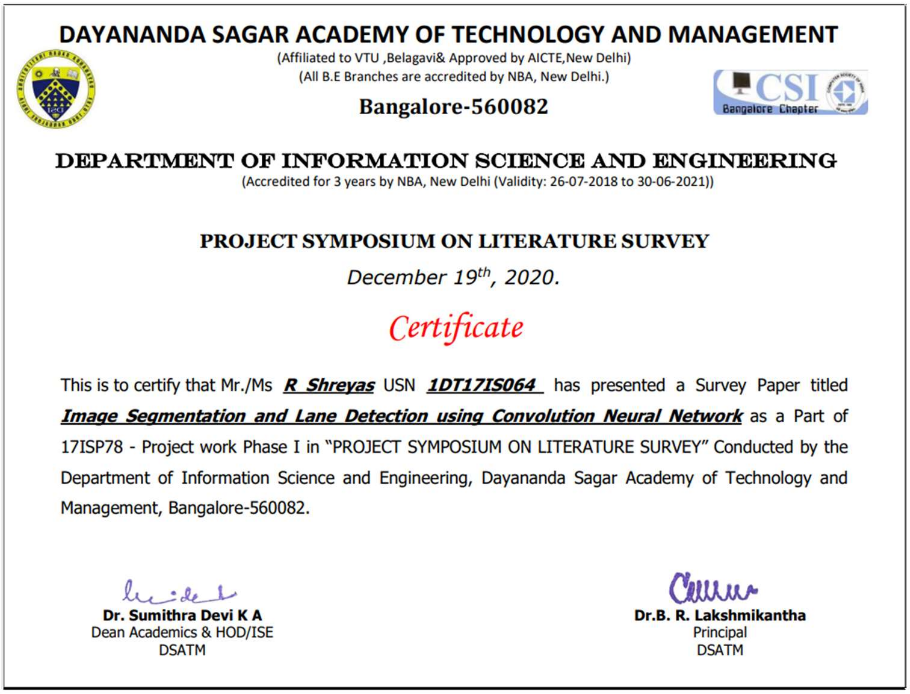
CONCLUSION

6.1 Conclusion

It is evident that recent advances have been made in the detection of road lanes on driving scenes. Even though there are several methods that achieved significant advancements using high precision and efficient methodologies, there are still many challenges pertaining to extreme driving conditions that have to be addressed. To overcome all these challenges we are proposed an optimized and hybrid combination of Deep CNN and RNN. The reason we choose CNN is that it can extract the spatial from the data using kernels, which other networks are not capable of. The proposed method uses a combination of DCNN and RNN to predict road lanes using a continuous sequence of frames as an input. Through vigorous experimentation and testing, our model has achieved a significant accuracy of 96% and a precision of 0.964. We have also demonstrated the step by step inner workings of our model in our output, for a better visual understanding of the lane detection process by our model. Also we have implemented a novel feature of real time adjustment of ‘area of transformation’ by providing a track bar interface to vary the dimensions of the area in order to achieve high accuracy when the lane conditions are worsened.

CHAPTER 7
PUBLICATIONS AND CERTIFICATES

7.1 Symposium Certificates



DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
(Affiliated to VTU ,Belagavi& Approved by AICTE,New Delhi)
(All B.E Branches are accredited by NBA, New Delhi.)
Bangalore-560082

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
(Accredited for 3 years by NBA, New Delhi (Validity: 26-07-2018 to 30-06-2021))

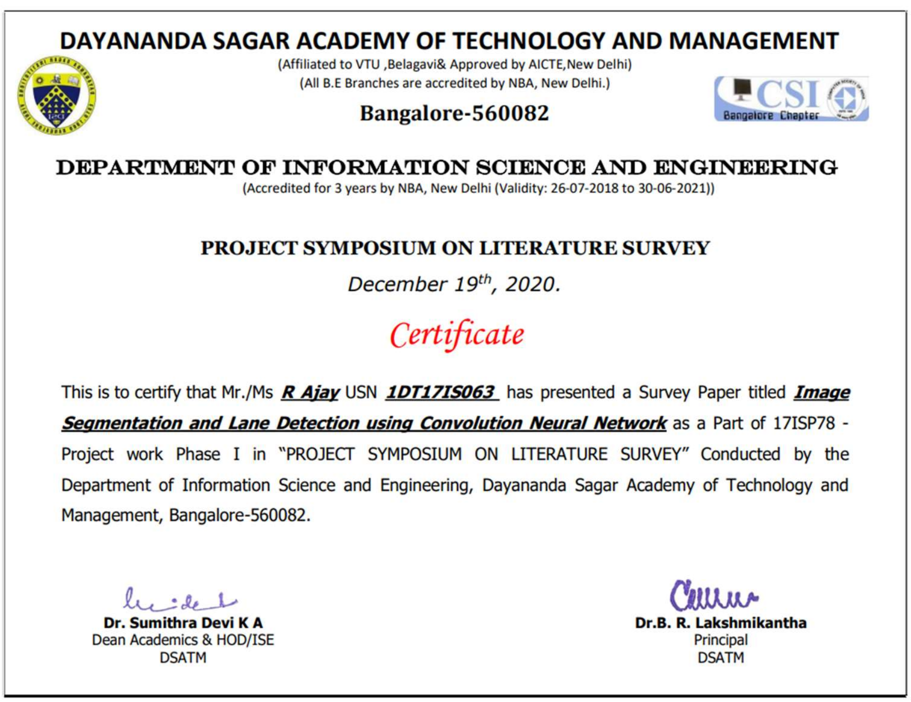
PROJECT SYMPOSIUM ON LITERATURE SURVEY
December 19th, 2020.

Certificate

This is to certify that Mr./Ms ***R Shreyas*** USN ***1DT17IS064*** has presented a Survey Paper titled ***Image Segmentation and Lane Detection using Convolution Neural Network*** as a Part of 17ISP78 - Project work Phase I in "PROJECT SYMPOSIUM ON LITERATURE SURVEY" Conducted by the Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD/ISE
DSATM

Dr.B. R. Lakshmikantha
Principal
DSATM



DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
(Affiliated to VTU ,Belagavi& Approved by AICTE,New Delhi)
(All B.E Branches are accredited by NBA, New Delhi.)
Bangalore-560082

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
(Accredited for 3 years by NBA, New Delhi (Validity: 26-07-2018 to 30-06-2021))

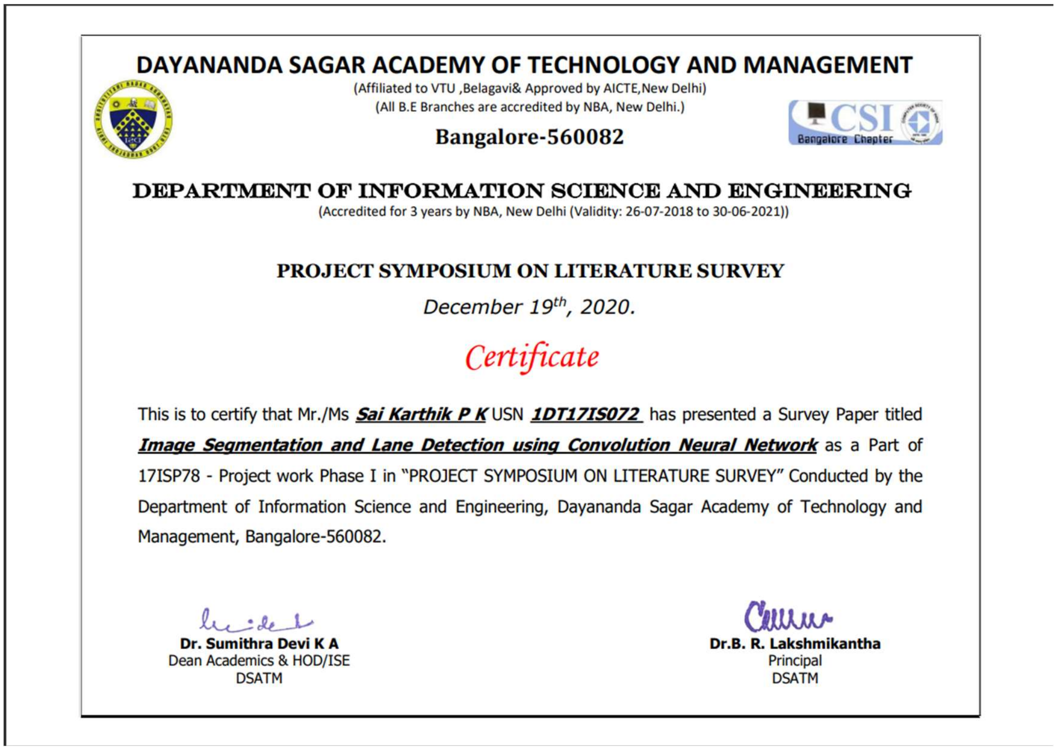
PROJECT SYMPOSIUM ON LITERATURE SURVEY
December 19th, 2020.

Certificate

This is to certify that Mr./Ms ***R Ajay*** USN ***1DT17IS063*** has presented a Survey Paper titled ***Image Segmentation and Lane Detection using Convolution Neural Network*** as a Part of 17ISP78 - Project work Phase I in "PROJECT SYMPOSIUM ON LITERATURE SURVEY" Conducted by the Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD/ISE
DSATM

Dr.B. R. Lakshmikantha
Principal
DSATM



DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
(Affiliated to VTU ,Belagavi& Approved by AICTE,New Delhi)
(All B.E Branches are accredited by NBA, New Delhi.)
Bangalore-560082

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
(Accredited for 3 years by NBA, New Delhi (Validity: 26-07-2018 to 30-06-2021))

PROJECT SYMPOSIUM ON LITERATURE SURVEY
December 19th, 2020.

Certificate

This is to certify that Mr./Ms ***Sai Karthik P K*** USN ***1DT17IS072*** has presented a Survey Paper titled ***Image Segmentation and Lane Detection using Convolution Neural Network*** as a Part of 17ISP78 - Project work Phase I in "PROJECT SYMPOSIUM ON LITERATURE SURVEY" Conducted by the Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD/ISE
DSATM

Dr.B. R. Lakshmikantha
Principal
DSATM

7.1 NCCSTM Certificates

Dayananda Sagar Institutions
DAYANANDA SAGAR ACADEMY OF TECHNOLOGY & MANAGEMENT
Udayapura, Opp. Art of living, Kanakapura road, Bangalore-082
(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)
DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING
Accredited 3 years by NBA, New Delhi
Seventh National Conference on Convergence of Science, Technology and Management NCCSTM-2021 (E-Conference)

Certificate

This is to certify that **Mr./ Ms R Shreyas** of **Dayananda Sagar Academy of Technology & Management, Bangalore** has participated in Seventh National Conference On Convergence Of Science, Technology & Management NCCSTM-2021 (E-Conference) on 10/07/2021 and presented a paper titled **Road Lane Detection Using Deep Convolutional Neural Networks** at Dayananda Sagar Academy of Technology & Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD
Dept. of ISE, DSATM

Dr. B R Lakshmikantha
Principal
DSATM

Dayananda Sagar Institutions

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY & MANAGEMENT

Udayapura, Opp. Art of living, Kanakapura road, Bangalore-082

(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Accredited 3 years by NBA, New Delhi

Seventh National Conference on Convergence of Science, Technology and Management NCCSTM-2021 (E-Conference)

IEEE BANGALORE SECTION

Certificate

This is to certify that **Mr./ Ms R Ajay** of **Dayananda Sagar Academy of Technology & Management, Bangalore** has participated in Seventh National Conference On Convergence Of Science, Technology & Management NCCSTM-2021 (E-Conference) on 10/07/2021 and presented a paper titled **Road Lane Detection Using Deep Convolutional Neural Networks** at Dayananda Sagar Academy of Technology & Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD
Dept. of ISE, DSATM

Dr. B R Lakshmikantha
Principal
DSATM

Dayananda Sagar Institutions

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY & MANAGEMENT

Udayapura, Opp. Art of living, Kanakapura road, Bangalore-082

(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Accredited 3 years by NBA, New Delhi

Seventh National Conference on Convergence of Science, Technology and Management NCCSTM-2021 (E-Conference)

IEEE BANGALORE SECTION

Certificate

This is to certify that **Mr./ Ms Sai Karthik P K** of **Dayananda Sagar Academy of Technology & Management, Bangalore** has participated in Seventh National Conference On Convergence Of Science, Technology & Management NCCSTM-2021 (E-Conference) on 10/07/2021 and presented a paper titled **Road Lane Detection Using Deep Convolutional Neural Networks** at Dayananda Sagar Academy of Technology & Management, Bangalore-560082.

Dr. Sumithra Devi K A
Dean Academics & HOD
Dept. of ISE, DSATM

Dr. B R Lakshmikantha
Principal
DSATM

A SurveyonRoad Lane Detection UsingDeepConvolutional Neural Networks

[1]R Shreyas,[2]Sai Karthik P K, [3]R Ajay,
[4] Prof.KarthikS A
[1][2][3]*BE Students, Department of Information Science and Engineering*
[4]*Asst.Professor, Departmentof Information Science and Engineering*
[1][2][3][4]*DayanandaSagar Academy of Technology and Management, Bangalore, Karnataka, India*

Abstract:*detection of lanes is an essential module for autonomous vehicles and advanced driver assistance systems (ADAS). Many state of the art methods for lane detection have been suggested in recent years. Although, these techniques focus on identifying the lane from a single frame, and they usually provide arguably dissatisfying performance in dealing with certain extreme situations such as degradation of the lane line, large shadows, significant occlusion of vehicles, noisy inputs of images, etc.Practically, lanes are supposed to be on-road continuous line structures. Hence, a lane that cannot be precisely detected in thelive frame can be extrapolated fromthe information of previous frames. Therefore, we expect to use multiple frames from a continuous driving scenario to approach lane detection, and for this reason, a hybrid architecture- combination of a convolution neural networks (CNN) and a recurrent neural networks (RNN). This method is a partial re-implementation of thework referenced [19], keeping in mind the general poor conditions of roads across the Indian subcontinent. In an attempt to train the model for optimum robustness, we intend to perform comprehensive experiments on two massive datasets and then our own dataset containing driving scenes across multiple Indian roads.*

Key Terms—CNN, RNN, DCNN, LSTM, lane detection, autonomous driving.

1. INTRODUCTION

Our understanding of real-time driving scenarios has become increasingly realistic, given the substantial developments in high-precision optical sensors and electronic sensors, high-precision computer vision and effective machine learning algorithms. Amidst various otherfeatures of autonomous driving vehicles, road lane detection is the principal and most significant one. The vehicle will realise where to movewhen the position of the lanes are obtained, thereby avoiding the risks of overstepping into other lanes.As reported in the relevant works, there are a number of modern approaches presented with smooth and sophisticated performance. They include lane detection with geometric models[16],[18], some of which include such techniques focused on deep machine learning[5],[12]. Some also map out issues related to energy minimization[17] and some use certain supervised learning strategies[19] and so on to segment the road lane. Most of the above mentioned methodologies restrict their results by detecting road lanes from a single, current driving scenario frame and result in poor performance while handling extreme driving scenes such as large shadows, substantial road lane linedegradation, and significant vehicle occlusion, as depicted in top three images in Figure 1.

Considering these cases, the lane could possibly be predicted with inaccuracy or projected in the wrong direction, it can be partly detected, or it cannot even be detected. The important reason for thisbeing, the knowledgepresented by the current image frame is not nearly sufficient [19]. Considering that driving scenarios are continuous and usually identical or nearly identical between two to three immediate frames, the position of the lane lines in the next few immediate frames is nearly identical and related. By using several previous frames, it is possible to predict the location of the lane in the current frame, although the lane lines can suffer deterioration or degradation due to weathering effects and weather conditions, shadows, and occlusions due to inadequate lighting.This is the principal motivation for our team to use a series of continuous images from a driving scenario to approach lane detection.

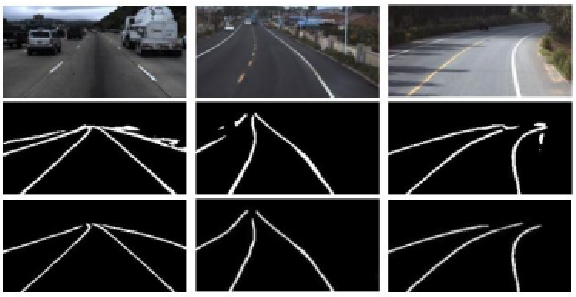


Figure 1. Lane detection in challenging situations. Top row: three example images of different driving scenes. Middle row: lane detection using only the current frame. Bottom row: lane detection using four previous frames and the current frame with the proposed method[19].

2. SURVEY STUDY

Muhammad Ali Aydin et ceteraall[8], presented a method where image processing techniques are used for determining lanes. The following methods are used: Image processing technique- Here we convert each video frame into HSV colour-space which differentiates various colours more accurately. Lane line determination- we use Sobel filter, Hough transform and canny edge detection for finding lanes on the roads. Find Region Of Interest(ROI): we determine ROI using skyline and various other methods. Horizon line is used to determine the required region of interest. This the line where sky and earth intersect when we see it from our naked eyes. The intersection lines and lane lines are merged to form ROI.

ThandaAunget ceteraall[6], proposed the following methods to warm drivers when they are going out of their lane. The methods used are: Smoothing: It is a process where the noise is eliminated using filters like 2d FRI filter to obtain a binary edge map and finally hough transform is used to detect edges and boundaries. Detecting line: we use hough transform for detecting lane lines which are subject to scenarios like short brakes in the lane which is caused due to noise and various other parameters. Tracking lane: we calculate distance between the lanes to find out the road width that a vehicle follows through its journey.

Fahmizalet ceteraall[7], proposed a method where road lanes are detected using CNN. Here, yolo is used to implement DCNN which has two connected layers and 23 convolutional layers for detecting objects. Road-Lane Detecting: four methods are used- Warping: here, images are handled by changing perspective of input. Filtering: we filter the lane colours with non-lane lines and we pick only the range of yellow and white colours using LUV and LAB format. Detection: The non-zero values are used from the process which was done earlier to detect lanes. We also crop images to 15 sub-images and aggregate them to left and right images. De-warping: we do exactly in opposite way as we did in warping resulting in clear images.

Rama Sai Mamidalaet cetera all[9], proposed “Dyanamic Approach for Lane Detection Using Google Street view and CNN” where a novel approach is being proposed for lane detection using CNN which is on the basis of SegNet decoder architecture.Thisarchitectures’s main feature is the usage of max-pooling indices in the decoders for up-sampling the low resolution feature maps which helps in retaining the frequency details of the segmented images and decreases the total number of training parameters in the decoders. To enable real time navigation, the author integrates Google APIs with the SegNet architecture. This interface helps in providing assistance for the Robotic systems.

VGPNet approach by Robotics and Computer Vision Lab(KAIST),[10]: Here, the scientists proposed a unified end to end trainable multitask network which was able to handle lane and road marking detection during extreme weather conditions and even the vanishing points of the lane. The network consists of four task modules in which each task performs a separate task: regression of the grid-box, object tracking, multi-label identification, vanishing point estimation. This technique indicated that it could detect generic marking shapes and attempt to match lines or splines to locate lanes.Classification is solved under a set of scenarios to check the effectiveness of the approach with advanced deep learning of tasks such as segmentation and classification. The accuracy is high in real-time at 20 fps.

Chanho Lee ceteraall[3], proposed this method for lane detection: Lane marking detection: Triangular ROI is determined and markings of lanes are searched and then the colour of the markings are changed to grey scale, various other edge detection are performed to obtain performance and efficiency. Here the segments of the lane are used to filter using various slope filters to find the segments of line and slope values for the marked points. Lane marking Tracking: In subsequent frames we use Hough transform and canny edge detector for surviving the line components in noisy environment. The proposed method helps in both efficiency and performance in various factors and scenarios. This method removes various edges and noisy elements decreasing horizontal lines and depth of the road which are mainly affected by speeds of the car.

3.METHODOLOGY

In this method, using a sequence of continuous driving scenario images, combining Deep Convolution Neural Networks (DCNN) and Recurrent Neural Network (RNN), we present a hybrid deep neural network for road lane detection (RNN).The proposed model comprises of a DCNN from a wider perspective that incorporates different sequential images as a feedback and predicts the lane path in a svm classification manner in the current frame. In order to attain this segmentation objective, a fully deep convolution (DCNN) approach is used.It consists of a network of encoders and also a network of decoders, assuring that the final feature map is exactly the same size as the source images.From a local point of view, the encoder network's summarised features of a Deep CNN are further explicated by a RNN. Following this, to handle the time series of encoded features, a long short-term memory (LSTM) network is used. The DRNN output should fuse the continuous input frame information and then be loaded into the network of DCNN decoders to assist in forecasting road lane routes.
We plan to build the network in the form of an encoder network - decoder network model in an effort to integrate CNN and RNN as an complete, end-to-end training network.The network architecture that is proposed is shown in Fig. 2. Both the encoder and decoder CNN are fully convolution networks. The encoder-CNN processes each of the frames with a sequence of continuous frames as input, and derives a time series of feature vectors.The feature vectors are then given to the LSTM network as sources for the lane-information prediction. In order to generate a probability distribution for the lane prediction, the LSTM output is then loaded into the decoder-CNN. The probability vector for the lane is the same scale as the source images.

LSTM Network:The RNN unit in the presented network architecture considers feature vectors produced by the encoder-CNN over each image as the feedback for modelling the sequence of continuous images of driving scenarios as a time series. Various types of RNN models, such as LSTM and GRU, have been suggested to tackle the variable time-series results. An LSTM network is used in this model, which typically outclasses the conventional RNN architecture with its ability to forget insignificant details and recall only the critical characteristics by using network cells to determine whether or not a segment of information is essential.With the first unit for simultaneous feature extraction and the second for assimilation, a dual-layer LSTM is implemented.

Dataset:The dataset we plan to use the lane dataset provided by TuSimple. TheTuSimple lane data set comprises of3,626 image sequences. These images are the front of the expressways. There are 20 continuous image sequences captured in one second in each series. The final image, that is, the 20th picture, is labelled with lane ground truth for each sequence. Additionally, keeping in mind the previously discussed challenges posed by Indian road lanes, we plan to create our own data set consisting of at least 20-30 image sequences, captured from a mounted camera on a car that is driven along various Indian road scenes covering a wide range of road conditions.

Deep Learning:is composed of several layers of nonlinear nodes, combine computer file with a collection of weights so that assigning significance to inputs for the corresponding task the algorithm is attempting to be told in supervised and/or unsupervised behaviour. The sum of the product of that input and weights is passed through the activation function of nodes. Each layer's output is fed synchronously as a feedback to the successive layer ranging from its input nodes. Major categories of descriptions corresponding to multiple points of abstraction are often used for learning.

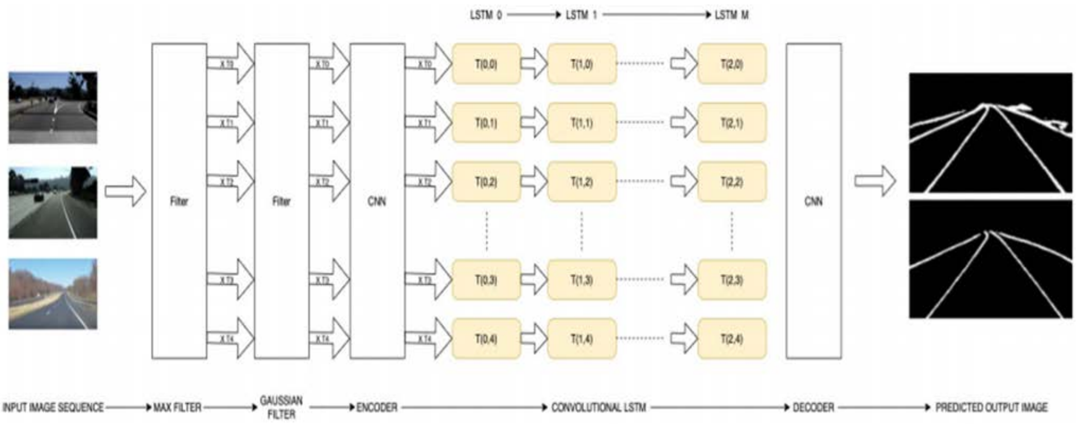


Figure 2. Architecture of the proposed network [19].

Flow Diagram:

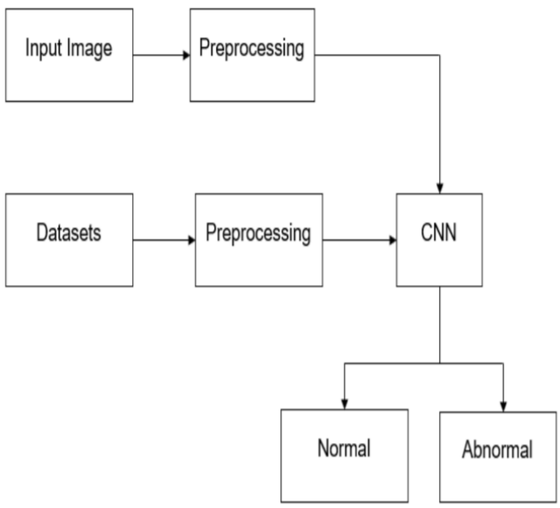


Figure 3: Flow diagram for the CNN architecture.

Convolutional Neural Networks (CNN): CNN is a kind of Deep Neural Network composed of several hidden layers, such as the RELU layer, the convolutionary layer, a layer for pooling and a fully related optimized layer. The weights within the convolutionary layer are primarily shared by CNN thereby reducing its memory requirements and increasing network performance.With the 3D capacities of its neurons, localized connections, and relative weights, the significant features of CNN contemplated. The convolutionl layer generates a feature vector via the convolution of various sub regions of the source images with a learned kernel.Then, via the ReLulayer, a non-linear activation function is applied to boost the convergence rate when the error is minimal. A section of the frame or feature network is selected in the pooling layer, and the pixels with the highest value between them or mean value is also preferred because the reference pixels are reduced to one scalar value by a 3x3 or 2x2 grid. This leads to an outsized decline in the sample size. In accordance with the convolution layers towards the output level, the conventional Fully-Connected (FC) layer is often used. Usually, two kinds of processes are done by the pooling layer: max pooling and mean pooling.The typical neighbourhood is calculated within the features extracted in the case of mean pooling, and within a maximum of features extracted in the case of max pooling. Mean pooling limits errors caused by the limited size of the neighbourhood and retains background info.Max pooling decreases the calculable error of the convolutional layer parameter induced by deviation from the mean and therefore holds a lot of image texture information. Softmax layer is applied to get the result.

4. RESULT COMPARISION

Recent Studies on Road Lane Detection using CNN.

Authors	**Model**	**Result**
Shivakumar et al	D-CNN	Accuracy=91.84 precision=0.4262 recall = 0.8085
Jiyong et al	SCNN and ConvGRUs	Accuracy=97.98 precision=0.8674 recall = 0.9562
Xianpeng et al	Dual View – CNN (DVCNN)	Precision=95.49 Recall=92.80
Rama Sai et al	CNN with Google Street View	Accuracy=0.9610 precision=0.9889r ecall = 0.8732
Ze Wang et al	CNN with LSTM	TPR = 87.3% FPR = 7.7%
Wei Wang et al	CNN with instance segmentation	ACU = 0.9785
Seokju Lee et al	VPGNet	Detection Score = 0.77

Table 1. Comparison of Results of the Papers.

The above table provides a comparison between the models proposed by various authors. It describes the accuracy, precision and recall values achieved by each of these proposed models. Thus, giving us a better overview of the results obtained from the mentioned methodologies.

5.CONCLUSION

From the above all paper, it is evident that recent advances have been made in the detection of road lanes on driving scenes.Even though there are several methods that achieved significant advancements using high precisionand efficient methodologies, there are still many challenges pertaining to extreme driving conditions that have to be addressed.To overcome all these challenges we are proposing an optimizedand hybrid combination of DeepCNN and RNN [19]. The reason we choose CNN is that it can extract the spatial from the data using kernels, which other networks are not capable of. The proposed method uses a combination of DCNN and RNN to predict road lanes using a continuous sequence of frames as an input.The DRNN output should fuse the concurrent input frame information and then be loaded into the network of DCNN decoders to assist in forecasting road lane routes. A double layer LSTM is also applied, with the initial layer for the extraction of concurrent features and the second for assimilation.

6. REFERENCES

[1] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W. chun Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” in Advances in neural information processing systems (NIPS), 2015, pp. 802–810.

[2]] M. Aly. Real time detection of lane markers in urban streets. 2008 IEEE Intelligent Vehicles Symposium, Jun 2008.

[3] Lee, C. and Moon, J.H., 2018. Robust lane detection and tracking for real-time applications. *IEEE Transactions on Intelligent Transportation Systems*, *19*(12), pp.4043-4048.

[4] Li, J., Mei, X., Prokhorov, D. and Tao, D., 2016. Deep neural network for structural prediction and lane detection in traffic scene. *IEEE transactions on neural networks and learning systems*, *28*(3), pp.690-703.

[5] Zhang, J., Deng, T., Yan, F. and Liu, W., 2020. Lane Detection Model Based on Spatio-Temporal Network with Double ConvGRUs. *arXiv preprint arXiv:2008.03922*.

[6] Aung, T. and Zaw, M.H., 2014, March. Video based lane departure warning system using Hough transform. In *International Conference on Advances in Engineering and Technology* (pp. 29-30).

[7] Nugraha, B.T. and Su, S.F., 2017, October. Towards self-driving car using convolutional neural network and road lane detector. In *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT)* (pp. 65-69). IEEE.

[8] Kamçı, S., Aksu, D. and Aydin, M.A., 2019. Lane Detection For Prototype Autonomous Vehicle. *arXiv preprint arXiv:1912.05220*.

[9] Mamidala, R.S., Uthkota, U., Shankar, M.B., Antony, A.J. and Narasimhadhan, A.V., 2019, October. Dynamic approach for lane detection using Google street view and CNN. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)* (pp. 2454-2459). IEEE.

[10] Lee, S., Kim, J., Shin Yoon, J., Shin, S., Bailo, O., Kim, N., Lee, T.H., Seok Hong, H., Han, S.H. and So Kweon, I., 2017. Vpgnet: Vanishing point guided network for lane and road marking detection and recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 1947-1955).

[11] 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, 7(1):20–37, March 2006.

[12] F. Pizzati, M. Allodi, A. Barrera, and F. Garc´ıa. Lane detection and classification using cascaded cnns, 2019.

[13] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 802–810. Curran Associates, Inc., 2015

[14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[15] W. Wang, D. Zhao, J. Xi, and W. Han. A learning-based approach for lane departure warning systems with a personalized driver model. IEEE Transactions on Vehicular Technology, PP, 07 2018.

[16]] Y. Wang, E. Teoh, and D. Shen. Lane detection and tracking using b-snake. Image and Vision Computing, 22:269–280, 04 2004.

[17] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In D. Forsyth, P. Torr, and A. Zisserman, editors, Computer Vision – ECCV 2008, pages 733–747, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[18] S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen. A novel lane detection based on geometrical model and gabor filter. 2010 IEEE Intelligent Vehicles Symposium, pages 59– 64, 2010.

[19] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang. Robust lane detection from continuous driving scenes using deep neural networks, 2019.

REAL TIME ROAD LANE DETECTION USING DEEP CONVOLUTIONAL NEURAL NETWORK

[1] R Shreyas, [2] Sai Karthik P K, [3] R Ajay, [4] Prof. Karthik S A
[1][2][3]*BE Students, Department of Information Science and Engineering*
[4] *Asst. Professor, Department of Information Science and Engineering*
[1][2][3][4] *Dayananda Sagar Academy of Technology and Management, Bangalore, Karnataka, India*

ABSTRACT

Detection of lanes is an essential module for autonomous vehicles and advanced driver assistance systems (ADAS). Many state of the art methods for lane detection have been suggested in recent years. Although, these techniques focus on identifying the lane from a single frame, and they usually provide arguably dissatisfying performance in dealing with certain extreme situations such as degradation of the lane line, large shadows, significant occlusion of vehicles, noisy inputs of images, etc. Practically, lanes are supposed to be on-road continuous line structures. Hence, a lane that cannot be precisely detected in the live frame can be extrapolated from the information of previous frames. Therefore, we have used multiple frames from a continuous driving scenario to approach lane detection, and for this reason, a hybrid architecture- combination of a convolution neural network (CNN) and a recurrent neural networks (RNN). In an attempt to train the model for optimum robustness, we intend to perform comprehensive experiments on two massive datasets.

Key Terms: *DCNN, LSTM, lane detection, autonomous driving.*

1. INTRODUCTION

Our understanding of real-time driving scenarios has become increasingly realistic, given the substantial developments in high-precision optical sensors and electronic sensors, high-precision computer vision and effective machine learning algorithms. Amidst various other features of autonomous driving vehicles, road lane detection is the principal and most significant one. The vehicle will realise where to move when the position of the lanes are obtained, thereby avoiding the risks of overstepping into other lanes. As reported in the relevant works, there are a number of modern approaches presented with smooth and sophisticated performance. They include lane detection with geometric models [16],[18], some of which include such techniques focused on deep machine learning[5],[12]. Some also map out issues related to energy minimization [17] and some use certain supervised learning strategies [19] and so on to segment the road lane. Most of the above mentioned methodologies restrict their results by detecting road lanes from a single, current driving scenario frame and result in poor performance while handling extreme driving scenes such as large shadows, substantial road lane line degradation, and significant vehicle occlusion, as depicted in top three images in Figure 1.
Considering these cases, the lane could possibly be predicted with inaccuracy or projected in the wrong direction, it can be partly detected, or it cannot even be detected. The important reason for this being, the knowledge presented by the current image frame is not nearly sufficient [19]. Considering that driving scenarios are continuous and usually identical or nearly identical between two to three immediate frames, the position of the lane lines in the next few immediate frames is nearly identical and related. By using several previous frames, it is possible to predict the location of the lane in the current frame, although the lane lines can suffer deterioration or degradation due to weathering effects and weather conditions, shadows, and occlusions due to inadequate lighting. This is the principal motivation for our team to use a series of continuous images from a driving scenario to approach lane detection.

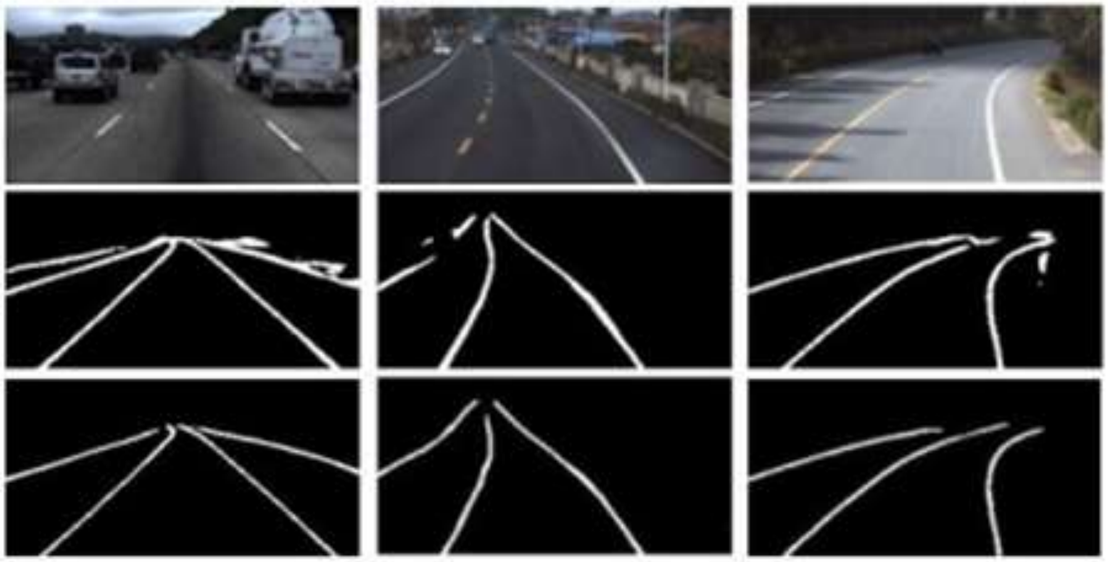


Fig -1: Lane detection in challenging situations. Middle row: Lane detection using only current frames Bottom row: lane detection using current and four previous frames as proposed by our method

2. SURVEY STUDY

Muhammad Ali Aydin et cetera all[8], presented a method where image processing techniques are used for determining lanes. The following methods are used: Image processing technique- Here we convert each video frame into HSV colour-space which differentiates various colours more accurately. Lane line determination- we use Sobel filter, Hough transform and canny edge detection for finding lanes on the roads. Find Region Of Interest (ROI): we determine ROI using skyline and various other methods. Horizon line is used to determine the required region of interest. This the line where sky and earth intersect when we see it from our naked eyes. The intersection lines and lane lines are merged to form ROI.

Thanda Aung et cetera all[6], proposed the following methods to warm drivers when they are going out of their lane. The methods used are: Smoothing: It is a process where the noise is eliminated using filters like 2d FRI filter to obtain a binary edge map and finally hough transform is used to detect edges and boundaries. Detecting line: we use hough transform for detecting lane lines which are subject to scenarios like short brakes in the lane which is caused due to noise and various other parameters. Tracking lane: we calculate distance between the lanes to find out the road width that a vehicle follows through its journey.

Fahmizal et cetera all[7], proposed a method where road lanes are detected using CNN. Here, yolo is used to implement DCNN which has two connected layers and 23 convolutional layers for detecting objects. Road-Lane Detecting: four methods are used- Warping: here, images are handled by changing perspective of input. Filtering: we filter the lane colours with non-lane lines and we pick only the range of yellow and white colours using LUV and LAB format. Detection: The non-zero values are used from the process which was done earlier to detect lanes. We also crop images to 15 sub-images and aggregate them to left and right images. De-warping: we do exactly in opposite way as we did in warping resulting in clear images.

Rama Sai Mamidala et cetera all[9], proposed “Dyanamic Approach for Lane Detection Using Google Street view and CNN” where a novel approach is being proposed for lane detection using CNN which is on the basis of SegNet decoder architecture.This architectures’s main feature is the usage of max-pooling indices in the decoders for up-sampling the low resolution feature maps which helps in retaining the frequency details of the segmented images and decreases the total number of training parameters in the decoders. To enable real time navigation, the author integrates Google APIs with the SegNet architecture. This interface helps in providing assistance for the Robotic systems.

VGPNet approach by Robotics and Computer Vision Lab(KAIST),[10]: Here, the scientists proposed a unified end to end trainable multitask network which was able to handle lane and road marking detection during extreme weather conditions and even the vanishing points of the lane. The network consists of four task modules in which each task performs a separate task: regression of the grid-box, object tracking, multi-label identification, vanishing point estimation. This technique indicated that it could detect generic marking shapes and attempt to match lines or splines to locate lanes. Classification is solved under a set of scenarios to check the effectiveness of the approach with advanced deep learning of tasks such as segmentation and classification. The accuracy is high in real-time at 20 fps.

Chanho Lee cetera all[3], proposed this method for lane detection: Lane marking detection: Triangular ROI is determined and markings of lanes are searched and then the colour of the markings are changed to grey scale, various other edge detection are performed to obtain performance and efficiency. Here the segments of the lane are used to filter using various slope filters to find the segments of line and slope values for the marked points. Lane marking Tracking: In subsequent frames we use Hough transform and canny edge detector for surviving the line components in noisy environment. The proposed method helps in both efficiency and performance in various factors and scenarios. This method removes various edges and noisy elements decreasing horizontal lines and depth of the road which are mainly affected by speeds of the car.

3. METHODOLOGY

In this method, using a sequence of continuous driving scenario images, combining Deep Convolution Neural Networks (DCNN) and Recurrent Neural Network (RNN), we present a hybrid deep neural network for road lane detection (RNN). The proposed model comprises of a DCNN from a wider perspective that incorporates different sequential images as a feedback and predicts the lane path in a svm classification manner in the current frame. In order to attain this segmentation objective, a fully deep convolution (DCNN) approach is used. It consists of a network of encoders and also a network of decoders, assuring that the final feature map is exactly the same size as the source images. From a local point of view, the encoder network's summarised features of a Deep CNN are further explicated by a RNN. Following this, to handle the time series of encoded features, a long short-term memory (LSTM) network is used. The DRNN output should fuse the continuous input frame information and then be loaded into the network of DCNN decoders to assist in forecasting road lane routes.

We plan to build the network in the form of an encoder network -decoder network model in an effort to integrate CNN and RNN as an complete, end-to-end training network. The network architecture that is proposed is shown in Fig. 2. Both the encoder and decoder CNN are fully convolution networks. The encoder-CNN processes each of the frames with a sequence of continuous frames as input, and derives a time series of feature vectors. The feature vectors are then given to the LSTM network as sources for the lane-information prediction. In order to generate a probability distribution for the lane prediction, the LSTM output is then loaded into the decoder-CNN. The probability vector for the lane is the same scale as the source images.

3.1 Convolutional Neural Networks (CNN)

CNN is a kind of Deep Neural Network composed of several hidden layers, such as the RELU layer, the convolutionary layer, a layer for pooling and a fully related optimized layer. The weights within the convolutionary layer are primarily shared by CNN thereby reducing its memory requirements and increasing network performance. With the 3D capacities of its neurons, localized connections, and relative weights, the significant features of CNN contemplated. The convolutionl layer generates a feature vector via the convolution of various sub regions of the source images with a learned kernel. Then, via the ReLulayer, a non-linear activation function is applied to boost the convergence rate when the error is minimal. A section of the frame or feature network is selected in the pooling layer, and the pixels with the highest value between them or mean value is also preferred because the reference pixels are reduced to one scalar value by a 3x3 or 2x2 grid. This leads to an outsized decline in the sample size. In accordance with the convolution layers towards the output level, the conventional Fully-Connected (FC) layer is often used. Usually, two kinds of processes are done by the pooling layer: max pooling and mean pooling. The typical neighbourhood is calculated within the features extracted in the case of mean pooling, and within a maximum of features extracted in the case of max pooling. Mean pooling limits errors caused by the limited size of the neighbourhood and retains background info.

3.2 LSTM Network

The RNN unit in the presented network architecture considers feature vectors produced by the encoder-CNN over each image as the feedback for modelling the sequence of continuous images of driving scenarios as a time series. Various types of RNN models, such as LSTM and GRU, have been suggested to tackle the variable time-series results. An LSTM network is used in this model, which typically outclasses the conventional RNN architecture with its ability to forget insignificant details and recall only the critical characteristics by using network cells to determine whether or not a segment of information is essential. With the first unit for simultaneous feature extraction and the second for assimilation, a dual-layer LSTM is implemented.
The activations of a general ConvLSTM cell at time t can be formulated as:

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c)$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * W_{t-1} + W_{cf} \circ C_{t-1} + b_f)$$

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * W_{t-1} + W_{co} \circ C_{t-1} + b_o)$$

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * W_{t-1} + W_{ci} \circ C_{t-1} + b_i)$$

$$H_t = o_t \circ \tanh(C_t)$$

where X_t denotes the input feature maps extracted by the encoder CNN at time t. C_t, H_t and C_{t1} , H_{t1} denote the memory and output activations at time t and $t - 1$, respetively. C_t, i_t, f_t and o_t denote the cell, input, forget and output gates, respectively. W_{xi} is the weight matrix of the input X_t to the input gate, b_i is the bias of the input gate. The meaning of other W and b can be inferred from the above rule. $\sigma(\cdot)$ represents the sigmoid operation and $\tanh(\cdot)$ represents the hyperbolic tangent non-linearities. '$*$' and '$\circ$' denote the convolutio operation and the Hadamard product, respectively.

3.3 Deep Learning

It is composed of several layers of nonlinear nodes, combine computer file with a collection of weights so that assigning significance to inputs for the corresponding task the algorithm is attempting to be told in supervised and/or unsupervised behavior. The sum of the product of that input and weights is passed through the activation function of nodes. Each layer's output is fed synchronously as a feedback to the successive layer ranging from its input nodes. Major categories of descriptions corresponding to multiple points of abstraction are often used for learning.

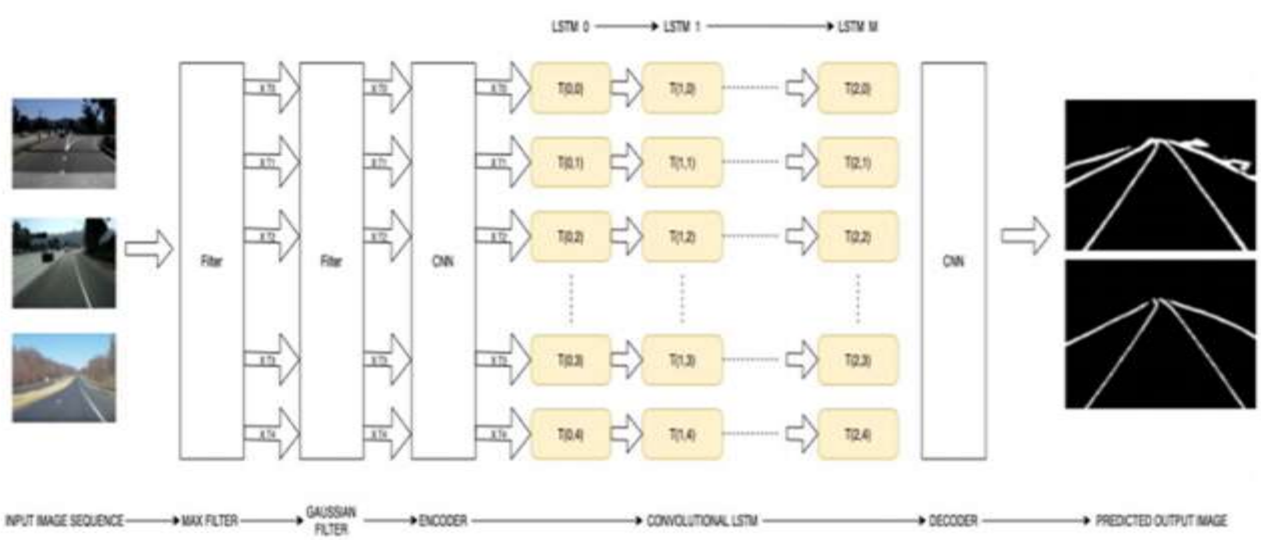


Fig -2: Architecture of the proposed network. [19]

3.4 Dataset

The dataset we plan to use the lane scenes provided by TuSimple and CULane Datasets. The TuSimple lane dataset comprises of 6,408 image sequences. These images are the front of the expressways in the United States of America. There are 20 continuous image sequences captured in one second in each series. The final image, that is, the 20th picture, is labelled with lane ground truth for each sequence. Additionally, keeping in mind the previously discussed challenges posed by Indian road lanes, we plan to create our own data set consisting of at least 20-30 image sequences, captured from a mounted camera on a car that is driven along various Indian road scenes covering a wide range of road conditions.

3.5 Flow Diagram

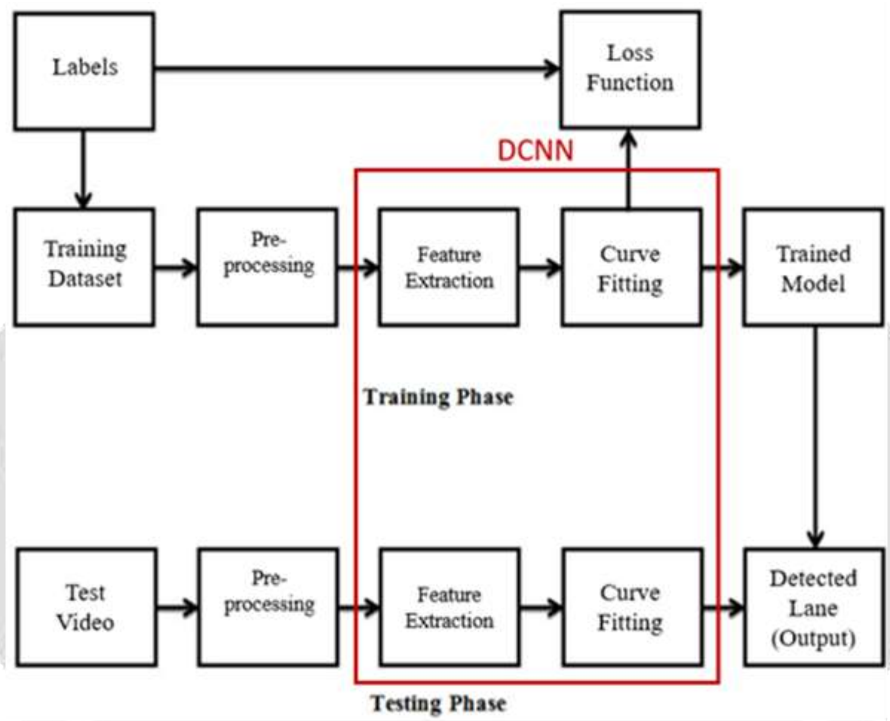


Fig -3: Flow Diagram for the proposed model.

4. RESULT COMPARISON

The below tables provide a comparison between the models proposed by various authors. It describes the accuracy, precision and recall values achieved by each of these proposed models. Thus, giving us a better overview of the results obtained from the mentioned methodologies.
Recent Studies on Road Lane Detection using CNN.

Authors	Model	Result
Shivakumar et al	D-CNN	Accuracy = 91.84 | Precision = 0.4262 | Recall = 0.8085
Jiyong et al	SCNN and ConvGRUs	Accuracy = 97.98 | Precision = 0.8674 | Recall = 0.9562
Xianpeng et al	Dual View –CNN (DVCNN)	Precision = 95.49 | Recall = 92.80
Rama Sai et al	CNN with Google Street View	Accuracy = 0.9610 | Precision = 0.9889 | Recall = 0.8732
Ze Wang et al	CNN with LSTM	TPR = 87.3% | FPR = 7.7%

Wei Wang et al	CNN with instance segmentation	ACU = 0.9785
Seokju Lee et al	VPGNet	Detection Score = 0.77

Table 1: Comparison of the results of the papers.

5. RESULTS

The CNN model is trained with both assorted driving scenes from TuSimple dataset and from Tvt dataset. The CNN model is trained with 3000 various driving scene images at 640x480 Resolution and around 133,000 classified driving scenes under shadow and occlusion conditions from CULane Dataset.

Basic measures derived from the confusion matrix:

Error rate (ERR) is calculated as the number of all incorrect predictions divided by the total number of the dataset. The best error rate is 0.0, whereas the worst is 1.0.

Error Rate=(FP+FN)/(P+N)

Accuracy (ACC) is calculated as the number of all correct predictions divided by the total number of the dataset. The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – ERR.

Accuracy = (TP+TN)/(P+N)

Precision (PREC) is calculated as the number of correct positive predictions divided by the total number of positive predictions. It is also called positive predictive value (PPV). The best precision is 1.0, whereas the worst is 0.0.

Precision = TP/TP+FP

False positive rate (FPR) is calculated as the number of incorrect positive predictions divided by the total number of negatives. The best false positive rate is 0.0 whereas the worst is 1.0. It can also be calculated as 1 – specificity.

FP= FP/TN+FP

The performance analysis results are as follows:

Sl. No	1
Dataset	Driving Scenes from TuSimple Dataset
Data Set Split	Total=6,408 Training=3,626 Testing=2,782
Error Rate(ERR) =FP+FN/P+N	4%
Precision =TP/FP+FN	0.964
Accuracy =(1-ERR)	±96%

Table 2: Performance analysis of DCNN model for TuSimple Dataset.

Sl. No	2
Dataset	Driving Scenes from CULane Dataset
Data Set Split	Total= 133,235 Training= 88,880 Testing= 34,680
Error Rate (ERR) =FP+FN/P+N	18%
Precision =TP/FP+FN	0.827
Accuracy =(1-ERR)	±82%

Table 3: Performance analysis of DCNN model for CULane Dataset.

The output window consists of three different windows as shown in figure 3. The first window depicts the final output video of the detected lane as a live video pipeline, another window shows the corresponding live video pipeline for all the processes that the test video is subjected to. This is done to increase transparency on the process of lane detection. The third window shows the live detected lane curvature parameters. Also, the predicted position of the vehicle and the supposed directed of steering according to the detected lane is printed live every 0.5 seconds on the terminal. This can be used as an input for an external automated vehicle steering module. The model is also tested with multiple test videos with varying degrees of complexity and is observed to be robust in most cases.

Fig -4: Output visual for a test video showing all the steps in lane detection procedure.

6. CONCLUSION

It is evident that recent advances have been made in the detection of road lanes on driving scenes. Even though there are several methods that achieved significant advancements using high precision and efficient methodologies, there are still many challenges pertaining to extreme driving conditions that have to be addressed. To overcome all these challenges we are proposed an optimized and hybrid combination of Deep CNN and RNN. The reason we choose CNN is that it can extract the spatial from the data using kernels, which other networks are not capable of. The proposed method uses a combination of DCNN and RNN to predict road lanes using a continuous sequence of frames as an input. Through vigorous experimentation and testing, our model has achieved a significant accuracy of 96% and a precision of 0.964. We have also demonstrated the step by step inner workings of our model in our output, for a better visual understanding of the lane detection process by our model.

7. REFERENCES

[1] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W. chun Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” in Advances in neural information processing systems (NIPS), 2015, pp. 802–810.

[2]] M. Aly. Real time detection of lane markers in urban streets. 2008 IEEE Intelligent Vehicles Symposium, Jun 2008.

[3] Lee, C. and Moon, J.H., 2018. Robust lane detection and tracking for real-time applications. *IEEE Transactions on Intelligent Transportation Systems*, *19*(12), pp.4043-4048.

[4] Li, J., Mei, X., Prokhorov, D. and Tao, D., 2016. Deep neural network for structural prediction and lane detection in traffic scene. *IEEE transactions on neural networks and learning systems*, *28*(3), pp.690-703.

[5] Zhang, J., Deng, T., Yan, F. and Liu, W., 2020. Lane Detection Model Based on Spatio-Temporal Network with Double ConvGRUs. *arXiv preprint arXiv:2008.03922*.

[6] Aung, T. and Zaw, M.H., 2014, March. Video based lane departure warning system using Hough transform. In *International Conference on Advances in Engineering and Technology* (pp. 29-30).

[7] Nugraha, B.T. and Su, S.F., 2017, October. Towards self-driving car using convolutional neural network and road lane detector. In *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT)* (pp. 65-69). IEEE.

[8] Kamçı, S., Aksu, D. and Aydin, M.A., 2019. Lane Detection For Prototype Autonomous Vehicle. *arXiv preprint arXiv:1912.05220*.

[9] Mamidala, R.S., Uthkota, U., Shankar, M.B., Antony, A.J. and Narasimhadhan, A.V., 2019, October. Dynamic approach for lane detection using Google street view and CNN. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)* (pp. 2454-2459). IEEE.

[10] Lee, S., Kim, J., Shin Yoon, J., Shin, S., Bailo, O., Kim, N., Lee, T.H., Seok Hong, H., Han, S.H. and So Kweon, I., 2017. Vpgnet: Vanishing point guided network for lane and road marking detection and recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 1947-1955).

[11] 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, 7(1):20–37, March 2006.

[12] F. Pizzati, M. Allodi, A. Barrera, and F. Garc´ıa. Lane detection and classification using cascaded cnns, 2019.

[13] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 802–810. Curran Associates, Inc., 2015

[14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[15] W. Wang, D. Zhao, J. Xi, and W. Han. A learning-based approach for lane departure warning systems with a personalized driver model. IEEE Transactions on Vehicular Technology, PP, 07 2018.

[16]] Y. Wang, E. Teoh, and D. Shen. Lane detection and tracking using b-snake. Image

[17] and Vision Computing, 22:269–280, 04 2004.

[18] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In D. Forsyth, P. Torr, and A. Zisserman, editors, Computer Vision – ECCV 2008, pages 733–747, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[19] S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen. A novel lane detection based on geometrical model and gabor filter. 2010 IEEE Intelligent Vehicles Symposium, pages 59– 64, 2010.

[20] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang. Robust lane detection from continuous driving scenes using deep neural networks, 2019.

REFERENCES

[1] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W. chun Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in Advances in neural information processing systems (NIPS), 2015, pp. 802–810.

[2]] M. Aly. Real time detection of lane markers in urban streets. 2008 IEEE Intelligent Vehicles Symposium, Jun 2008.

[3] Lee, C. and Moon, J.H., 2018. Robust lane detection and tracking for real-time applications. *IEEE Transactions on Intelligent Transportation Systems*, *19*(12), pp.4043-4048.

[4] Li, J., Mei, X., Prokhorov, D. and Tao, D., 2016. Deep neural network for structural prediction and lane detection in traffic scene. *IEEE transactions on neural networks and learning systems*, *28*(3), pp.690-703.

[5] Zhang, J., Deng, T., Yan, F. and Liu, W., 2020. Lane Detection Model Based on Spatio-Temporal Network with Double ConvGRUs. *arXiv preprint arXiv:2008.03922*.

[6] Aung, T. and Zaw, M.H., 2014, March. Video based lane departure warning system using Hough transform. In *International Conference on Advances in Engineering and Technology* (pp. 29-30).

[7] Nugraha, B.T. and Su, S.F., 2017, October. Towards self-driving car using convolutional neural network and road lane detector. In *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT)* (pp. 65-69). IEEE.

[8] Kamçı, S., Aksu, D. and Aydin, M.A., 2019. Lane Detection For Prototype Autonomous Vehicle. *arXiv preprint arXiv:1912.05220*.

[9] Mamidala, R.S., Uthkota, U., Shankar, M.B., Antony, A.J. and Narasimhadhan, A.V., 2019, October. Dynamic approach for lane detection using Google street view and CNN. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)* (pp. 2454-2459). IEEE.

[10] Lee, S., Kim, J., Shin Yoon, J., Shin, S., Bailo, O., Kim, N., Lee, T.H., Seok Hong, H., Han, S.H. and So Kweon, I., 2017. Vpgnet: Vanishing point guided network for lane and road marking detection and recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 1947-1955).

[11] 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, 7(1):20–37, March 2006.

[12] F. Pizzati, M. Allodi, A. Barrera, and F. Garc´ıa. Lane detection and classification using cascaded cnns, 2019.

[13] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 802–810. Curran Associates, Inc., 2015

[14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[15] W. Wang, D. Zhao, J. Xi, and W. Han. A learning-based approach for lane departure warning systems with a personalized driver model. IEEE Transactions on Vehicular Technology, PP, 07 2018.

[16]] Y. Wang, E. Teoh, and D. Shen. Lane detection and tracking using b-snake. Image and Vision Computing, 22:269–280, 04 2004.

[17] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In D. Forsyth, P. Torr, and A. Zisserman, editors, Computer Vision – ECCV 2008, pages 733–747, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[18] S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen. A novel lane detection based on geometrical model and gabor filter. 2010 IEEE Intelligent Vehicles Symposium, pages 59– 64, 2010.

[19] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang. Robust lane detection from continuous driving scenes using deep neural networks, 2019.