

# Object Detection: Automatic License Plate Detection using Deep Learning and OpenCV



R Mario Vishal, Deekshith Maram, P Krishna Chaitanya, R Angeline

**Abstract:** Object Detection is one of the most important concepts of Computer Vision which is used in various areas like Medical Field, Security, Self Driving cars, Automated vehicle systems etc. We choose the application of Automatic License plate Recognition. Automatic License Plate Recognition is an emerging technology which is helpful in many fields and at the same time is challenging. It's challenging because we need to get the accurate recognition of the characters in a number plate. In practical applications where sometimes the images are captured in the worst weather condition, bad lighting, wind. And to the addition, license plates are often dirty or blackened due to the smoke, half broken, or having scratches on certain characters and detection of too many license plates in a single frame. All these will act as the obstacles in developing an effective ALPR system. So basically, this is a system where recognition of characters from images using Computer Vision techniques are performed. This system is implemented in many fields like parking lots, private and public entrances, toll gates, theft control, checking the authentication of a vehicle. Procedure followed in this paper are, first capturing images from camera then loading that into system, preprocessing done using OpenCV library. Then we use Attention OCR a deep learning model to recognize the characters from an image. And later display that in the GUI and store them in the databases for different operations later.

**Keywords:** ALPR, Computer Vision, Deep Learning, OCR.

## I. INTRODUCTION

Automatic License Plate Recognition (ALPR) has been an interest in research due to many practical applications, such as automated toll collection, traffic control, checking the authenticity. It is a emerging technology which uses many methods like Optical Character recognition, deep learning to capture license plates and interpret on those captured images to extract the characters and converting digital image

characters to text format [5]. And with these plate numbers they can perform various operations like tracking of stolen vehicles, keeping records of toll collection, automatic pay-per-usage parking lot system where it calculates a fare based on the vehicle in and out time etc., Currently Technologies like closed-circuit-television(cctv), road-rule enforcement camera's, or camera's which are specifically designed for this task, are used for ALPR[2]. ALPR systems have three stages: License Plate region detection, segmentation of characters and recognition of characters. The earlier stages of the process require higher accuracy or in other words at most perfection in identifying the location of the license plate, since failing to detect the license plate would probably lead to a failure in the remaining next stages either[3]. Many ALPR systems approach first search for the vehicle and then its license plate in order to reduce the processing period and remove inaccurate identification[1]. Although ALPR systems are being studied in the literature over the time, it's still not capable enough to meet the current world conditions. Some constraints like camera viewing angles, backgrounds, lighting conditions, types of vehicles still hold the system back to make accurate prediction[5].



**Fig.1. License Plate detection**

In order to recognize characters from images we are going to implement Attention OCR. Attention OCR is a TensorFlow project which is a combination of CRN Network(CRNN/CNN+RNN) and an Attention Decoder. It first uses layers of Convolutional networks to extract encoded image features[4][5][8]. These extracted features of these images are then converted into strings and then proceeded through a recurrent networks for the implementation of the attention mechanism on them. The mechanism used in the implementation takes the concept from TensorFlow model called seq2seq machine. Then it finally uses the Attention Decoder to predict the text in our image.

Revised Manuscript Received on October 30, 2019.

\* Correspondence Author

**R Mario Vishal\***, Computer Science and Engineering, SRM institute of science and technology, Chennai, India. Email: mariovishal\_r17@srmuniv.edu.in

**Deekshith Maram**, Computer Science and Engineering, SRM institute of science and technology, Chennai, India. Email: maram\_deekshith@srmuniv.edu.in

**P Krishna Chaitanya**, Computer Science and Engineering, SRM institute of science and technology, Chennai, India. Email: poladi\_krishnachaitanya17@srmuniv.edu.in

**R Angeline**, Department of Computer Science Engineering in SRM Institute of science and technology, Chennai, India Email :angelinr1@srmist.edu.in

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)



## II. EXISTING SYSTEM

In the existing system there are certain types of License plate recognition systems(LPR). Some of them are sensor detection by using infrared sensors, image processing methods, loops techniques[7][9]. In this paper, we are implementing it with image processing. All the current ALPR systems almost uses neural networks for character recognition and some of the image processing tools like OpenCV[6][7]. Although being implementing these algorithms over the time, still ALPR is not so robust under the real word conditions[3]. And more over the neural network models in previous literature studies have been trained with foreign license plate datasets and makes it inaccurate for the recognition of Indian license plate numbers. So it will be difficult to recognize and extract the characters from the image. The important task is to convert the License plate image characters to digital text. Many of the algorithms are accurate enough to covert these but these algorithms are being improved over the time. So in this paper we have implemented the Attention OCR algorithm to convert the image characters to digital characters.

## III. PROPOSED SYSTEM

Deep Learning algorithms have been improved over the time. But some of these algorithms have some limitations. In this paper we are going to implement Attention OCR which is a TensorFlow OCR project. It is a combination of CRNN and attention decoder. Attention Decoder follows the principles of attention mechanics[10]. Let us know about the things which we are going to implement in this paper.

### A. Attention Mechanics

Neural network architectures like RNNs or LSTMs which predict the output at each time step, providing us with sequence generation as we require for language[8][10]. This type Neural networks designed to learn patterns in sequential data by modifying or changing their current state based on current input and previous states iteratively.

But due to limitations on memory and issues with vanishing gradients, we find that RNNs and LSTMs not able to really capture the influence of words further away. Attention mechanics tries to fix this problem[10]. It is a way to get the model learn long range dependencies in a sequence. This mechanics are used in Natural Language Processing to get the accurate word sequence generator.

So to say in simple terms, attention is a feed-forward layer with trainable weights that help us capture the relationships between different elements of sequences[9][10]. It works by using the query, key and value matrices, passing the input embeddings through a series of operations and getting an encoded representation of our original input sequence.

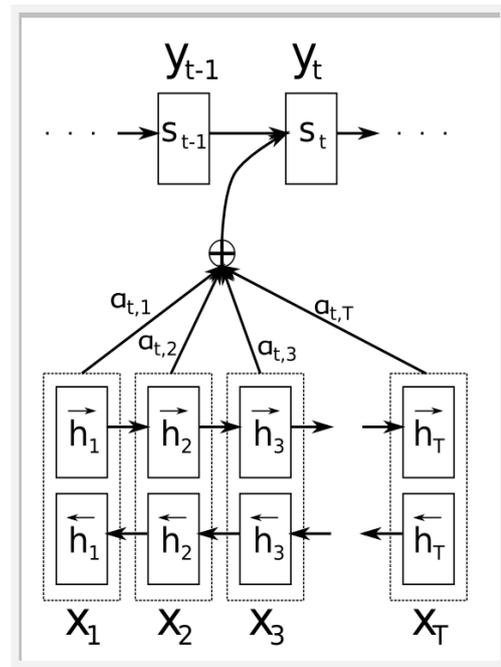


Fig.2.Attention Mechanics

The encoder in the attention mechanics Generates  $h_1, h_2, h_3, h_4 \dots h_T$  from the inputs  $x_1, x_2, x_3 \dots x_T$  Then we have to find the context vector  $c_i$  for each output time step[10].

$$e(ij) = a(s(i) - 1, h(j))$$

Where  $a$  is alignment model which is a feed forward neural network that is trained with the data sets provided to the system.

The alignment model scores which is denoted by 'e'. Calculates how well each encoded input ( $h$ ) matches the present output of the decoder(s)[9][10].

$$at, i = align(yt, xi) = \frac{\exp(score(st - 1, hi))}{\sum_n i' = 1 \exp(score(st - 1, hi'))}$$

The alignment scores are normalized with a softmax function.

$$ct = \sum_n i = 1 nat, ihi$$

The context vector is the sum of the annotations( $h_j$ ) and normalized alignment scores.

### B. CRNN

Convolutional Recurrent Neural Networks is a type of neural network which has three types of layers embedded into one architecture as shown in Fig(2)[5][7][9]. They are Convolutional layers, Recurrent layers, Transcription Layers. CRNNs are a combination of Recurrent Neural Network(RNN), Convolutional Neural Network(CNN) and CTC (Connectionist Temporal Classification).They are used in diverse fields of deep learning like Natural Language Processing, scene text recognition, optical character recognition.

They don't treat OCR as a reinforcement problem but as a machine learning problem where we load a data set and train the model with our preferred datasets. Using machine learning we get some CTC(Connexionist Temporal Classification) loss[10][9]. The neural network extracts features from the input image. The recurrent network predicts the label sequence + with some relation between the characters. The transcription layer predicts each frame and then finally we get the predicted text.

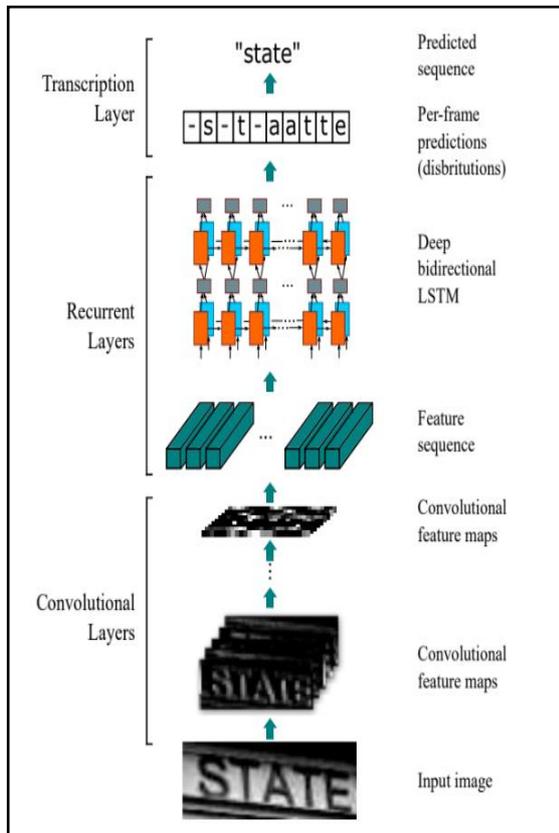


Fig.3.CRNN

**C. Attention OCR**

It is a project available on TensorFlow library. It came into being as a way to solve the problem of the image captioning problem. It can be thought of as CRNN followed by an attention decoder[10]. First it uses Convolutional networks to extract the encoded image features, then these are encoded to strings and passed through a Recurrent network for the attention mechanism to process[7][10]. The attention mechanism used in this implementation is borrowed from seq2seq machine translation model. We use this attention based decoder to finally predict the text in our image.

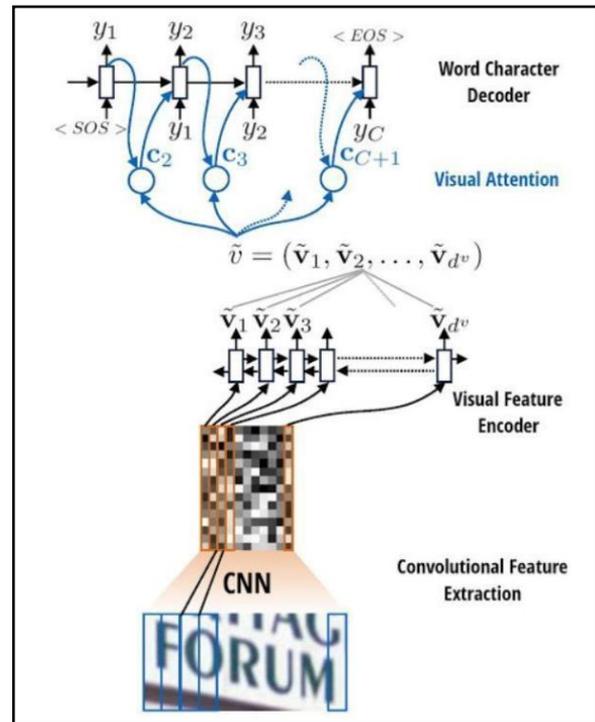


Fig.4.Attention OCR

**IV. TOOLS**

**A. Open CV**

OpenCV(Open Source Computer Vision Library) is an open source AI and Computer vision library. OpenCV is a library of capacities which are utilized in Computer vision initially and created by Intel and now upheld by Willogarage[9]. It is free for use under the open source BSD permit. The library has in excess of 500 upgraded functions. It is being used all over the world with forty thousand individuals in the client gathering. The library is primarily written in C which makes it versatile to some particular stages. For example, C, Python, Ruby and Java (utilizing Java CV) have been created to energize selection by a more extensive group of spectators[1][4][9].

**B. Data Set**

The algorithm which we are going to implement in this paper uses a certain type of dataset called French Street Name Signs(FSNS) dataset[10]. This is Google's dataset which contains more than one million of street name signs, cropped from Google Street View images of France. Each image contains four views of the same street sign. The Attention OCR was trained by this dataset. So to implement or to train this OCR we have to make our dataset in the form of FSNS dataset. It uses two types of data forms for storing test and training data and TFrecords[10]. We prepared dataset by obtaining Indian license plate numbers and annotating it with a annotation tool. We placed those images in the FSNS format.



files	text	xmin	ymin	xmax	ymax
0.png	MH15-TC-5	589	280	694	311
1.png	MH 02 CB 4	89	397	254	448
3.png	KA03 1351	232	318	346	357
4.png	TN 06 0 000	173	31	959	168

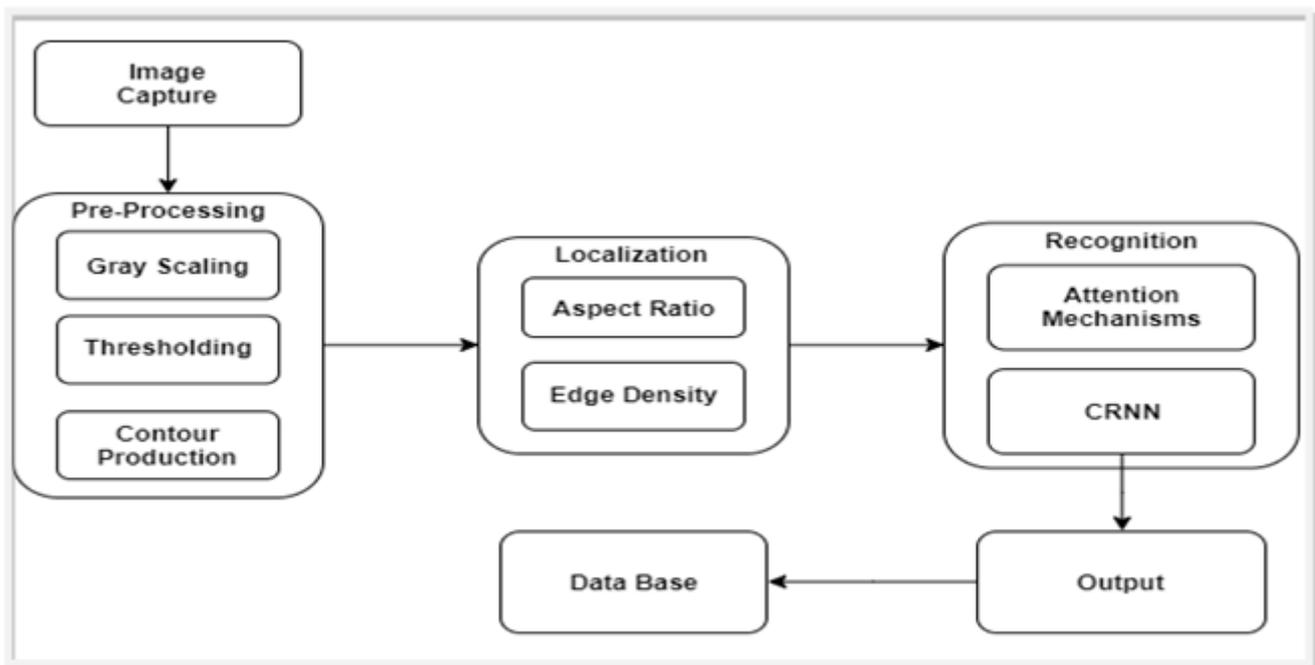
**Fig .4.Dataset**

Which has the attributes files,text, xmin, ymin, xmax, ymax as shown in fig (4).These attributes are related to the bounding box around the plate,text on the plate and the name of the image file.

## V. ARCHITECTURE DIAGRAM

### A. TENSOR-FLOW

TensorFlow is a Deep Learning Library created by Google. It is free and open source. It works with Python. It was released in 2015, to make it easier for developers to build, design and train Deep Learning models. It allows users to describe arbitrary computations as a graph of data flows. Data is stored as Tensors which are multidimensional arrays. It also makes GPU processing system as the data is stored as binary data, it will be easy to work with the data.



**Fig.5.Architecture Diagram of Proposed System**

## VI. IMPLEMENTATION

### A. Getting Images

The images of vehicles are captured through a camera and are loaded into the system for pre -processing.

### B. Preprocessing

Pre-processing is a group of functions applied to the image before proceeding to next methods. It involves change of

color, change of background illumination, enhancing contrast, saturation, removing background noise.

The OpenCV supports every above mentioned operations. The images which are taken from the camera is imported into the OpenCV program, and its format will be in BGR(Blue ,Green, Red) instead of being in RGB(Red, Green, Blue). The color of the image is changed into gray, because extracting features from a grayscale image

will be accurate and effective rather than a color image. It is changed by the following function.  
First importing cv2.

```
img = cv2.imread('141.jpg')
```



Fig.6.Original Image

```
img1= cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)  
plt.imshow(img1,cmap='gray')
```



Fig.7.Gray Scale image

**C. Threshold**

Image thresholding is a method of image segmentation. It is used in applications like extracting certain regions or objects from the image on which we want to analyse or work on them. Also used to differentiate pixels from those which we are interested in. It has five parameters (input\_img, output\_img, threshold\_value, max\_binary\_value, threshold\_type)

The basic idea here is the threshold value is applied to every pixel. And each pixel if lower than the specified value changes to that color completely and which is higher than that value changes into another color completely.

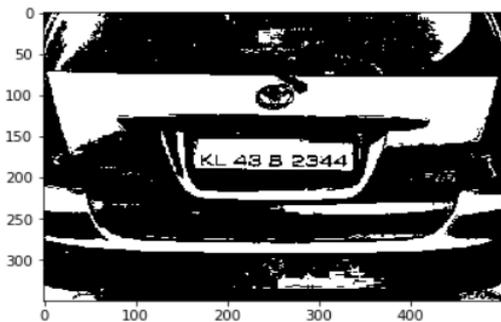


Fig.8.Threshold image from grayscale image

**D. Contours**

Contours is a simple curve joining all the continuous points which are along the boundary having the same pixel color.This can extract the exact object shapes from the images and are used as a tool for shape analysis and object detection.To get a better contours from a image we have to use binary images .Before finding contours we need apply threshold or canny edge detection to that image. It is like extracting white objects from a black background.

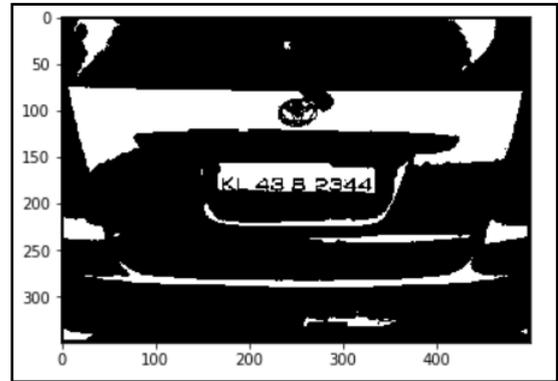


Fig.9.Contours of an image

**E. Number Plate Localization**

The number plate localization is the phase where we find the location of license plate in our scenario it is Region Of Interest(ROI) which are found out from the contours of an image.

To identify the region which contains the license plate, two features depend.

- (a) Aspect ratio (b)Edge Density

Aspect ratio

The aspect ratio is defined as the ratio of the width of a frame to the height of the frame.

$$Aspect\_ratio = width/height$$

In this case we are assuming the aspect ratio is 16/9.

Edge density

Edge density is defined as the average density i.e summation(W)/N where 'W' are all the white pixels 'N' is the total number of pixels which width\*height.

From the edge density we can get the rectangular area of license plate. And save this output and load it to the next process.



Fig.8.The extracted region from the image

**F. Recognition**

In this phase we get the input cropped image from the above methods done by OpenCv.As we are going to implement the Attention OCR which is a TensorFlow model.We need to install TensorFlow in our system.Then we need to import it in our python programme script. And all the necessary modules like numpy,pandas, matplotlib. The dataset contains a lot of cropped Indian license plate images which is cropped because it increases the prediction percentage of the training model.



First we need to annotate it with its height and width. This is because we need to make the dataset in the form of FSNS which we already discussed. These are required to generate TF records for training the Attention OCR model.

## G. TF Records

Tf records are the files which stores your data as a sequence of binary strings. This means you need to specify the structure of the data before you write it to the file. If we are working on a large dataset it may require a lot of time for your model to train or perform predictions. But converting it to a binary file format for storage have a significant impact on the performance of the model. And one of the reasons we use Tf records are, it is simply compatible with TensorFlow which also impacts the accuracy.

So the cropped image from the dataset is annotated with the boundary coordinates and stored in TF records in the form of FSNS dataset.

## H. Training the model

And finally training the model with the data we stored in the tf records.

## I. Prediction

Then we load the cropped license plate region image which we got from the OpenCV methods above. Then we give the input to the model and finally get the prediction as shown in Fig(9).

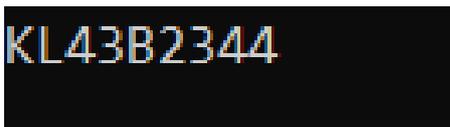


Fig.9. The final Recognized characters

## J. Database

The recognized license plate characters are stored in the database. Which are served for various purposes like extracting the first 2 letters, will help you understand the origins of that vehicle, like from which state and we can maintain a separate data base for different states. To track the stolen vehicles and the vehicles who fail to follow the traffic law.

## VII. LITERATURE SURVEY

### i. Rayson Laroca, Luiz A. Zanlorensi, Gabriel R. Gonc,alves, Eduardo Todt, William Robson Schwartz, David Menotti(2019)

This literature study presents a layout independent ALPR system which is based on state-of-the-art YOLO object detector. They have chosen a unified approach to detect and classify the layouts in post-processing methods. They used networks to train models using images from different datasets and with additional augmentation techniques.

### ii. Satadal Saha(2019)

This literature study focuses on feature extraction from images. They have used pre processing methods like binarization, localization, segmentation. They have implemented Multi layer perceptron (MLP) as a classifier, Quad Tree Based Longest Run (QLTR) is used to train network. And obtain the predicted characters.

### iii. Andrej Jokie, Nikola Vukovic(2018)

This literature paper focuses mainly on feature based extraction. Their main objective is to reduce the computation power used for the ALPR system and the data used for training should be less. It also implements segmentation for pre-processing and they proposed a dimensionality reduction method based on compressive sensing.

### iv. Mohamed Yousef, Khaled F. Hussain, and Usama S. Mohammed(2018)

In this literature study they focused on the prevention of accuracy loss. They have used the CTC loss function to train their model. And implemented Highway networks for classifying the images. They also implemented it successfully on the captcha recognition and on street images. They have used a lot of different datasets for different implementations.

### v. Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel Resende Gonc,alves, William Robson Schwartz and David Menotti(2018)

Their proposed model also done some pre-processing works like segmentation. They have implemented Convolutional Neural Networks for the identification of the image. They have designed a two way approach augmentation techniques like inverted license plates and flipped characters. They have used a dataset called UFPR-ALPR dataset which contains more than 150 videos with 4500 frames captured of vehicles from it. And also used SSIG dataset to both train the models separately.

## VIII. CONCLUSION

The ALPR systems in general have a long way to go and every year there are some ground breaking algorithms of deep learning been created, for better accuracy with low computation power. We have successfully implemented Attention OCR of the TensorFlow which is better in identifying street signs and in Natural Language Processing. Which identifies the characters according to the sentence formation technique. We have focused this paper mostly on recognizing the characters on a License Plate to increase the accuracy of the system. We have Obtained 93% accuracy on our own collected dataset, but it can be improved more if we obtain a larger dataset of Indian License plates. We wanted to make use of latest technologies which is currently available such as TensorFlow as it is rising eventually in Deep Learning sector and improving constantly. And also made sure to use OpenCV we had a choice to choose Matlab but OpenCV is much more simpler and usage of python is one of the important reasons we choose OpenCV for the image processing techniques. And finally got the inspiration from lots of web articles to research on ALPR system. Although many ALPR systems have been proposed, all the systems fail to work in real world scenarios some of them like climate conditions, lighting problems, damaged license plate or half broken LP, smoke. These are all the factors which act as an obstacle for an effective ALPR system. But from the look of Deep Learning, which is daily evolving, in future the algorithms may improve and

ALPR regarding hardware device may be developed for efficiency. The main drawback for all the ALPR systems are to some extent are hardware, which plays an important role in identifying the license plates all at once i.e multiple license plates in a single frame. As we have already discussed the perks of ALPR system which makes a difficult task which a human being cannot perform to a simpler task where a machine can perform and that too with lot of accuracy. The future of ALPR systems is yet to grow, and maybe a full pledged hardware device with a best fitting software could be introduced later in the near future. The exponential growth of Deep Learning and Computer Vision is sure happening. And with this research paper we want to be helpful, for those who want to continue this research and further improve the features of ALPR system.



**P Krishna Chaitanya** currently pursuing Bachelors Degree in Computer Sciences from SRM Institute of Science and Technology. His current interests are of the domain Web Development. He is meritorious student. He is planning to work in a product-based company.



**Mrs R Angeline** currently working as Assistant Professor in SRM Institute of Science and Technology. She completed her Masters in Information Technology. She continues to pursue her research in Deep Learning. Also she is the guide for this paper.

## REFERENCES

1. Sweta kumari, Leeza gupta ,Prena gupta."Automatic License Plate Recognition Using OpenCV and Neural Networks". IJCST – Volume 5 Issue 3, May – Jun 2017.
2. .Rayson Laroca, Luiz A. Zanlorens, Gabriel R. Gonc alves, Eduardo Todt, William Robson Schwartz, David Menotti1. "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO detector". arXiv:1909.01754v2 [cs.CV] 5 Sep 2019.
3. .Xuewen Yang, Xin Wang."Recognizing License Plates in Real-Time".arXiv:1906.04376v1 [cs.CV] 11 Jun 2019.
4. .Satadal Saha. "A Review on Automatic License Plate Recognition System".Students' Technical Article Competition: PRAYAS-2018,29th april 2018.
5. Andrej Jokic, Nikola Vukovic."License Plate Recognition with Compressive Sensing Based Feature Extraction".arXiv:1902.05386v1 [cs.CV] 7 Feb 2019.
6. .Pratiksha Jain, Neha Chopra, Vishali Gupta."Automatic License Plate Recognition Using Opencv".International Journal of Computer Applications Technology and Research Volume 3- Issue 12, 756 - 761, 2014, ISSN:- 2319–8656
7. Mrs. J. V. Bagade , MSukanya Kamble , Kushal Pardeshi , Bhushan Punjabi, Rajpratap Singh."Automatic Number Plate Recognition System - Machine Learning Approach". ISSN: 2278-0661, ISBN: 2278-8727, PP: 34-39.
8. N.Abirami, Dr. J.S.Leena Jasmine."ACCURATE VEHICLE NUMBER PLATE REGOGNITION AND REAL TIME IDENTIFICATION USING RASPBERRY PI".e-ISSN: 2395-0056.p-ISSN: 2395-0072.Volume: 05 Issue: 04 Apr-2018.
9. Denisa Lippaiova, Miroslav Michalko, Ondrej Kainz, Frantisek Jakab."AUTOMATIC LICENSE PLATE RECOGNITION SYSTEM USING OPENCV LIBRARY".ISSN: 2277-9655.Impact Factor: 5.164.CODEN: IJESS7,August 2018.
10. Zbigniew Wojna\* Alex Gorban† Dar-Shyang Lee† Kevin Murphy† Qian Yu† Yeqing Li† Julian Ibarz†. "Attention-based Extraction of Structured Information from Street View Imagery".arXiv: 1704.03549v4,August 2017

CODEN: IJESS7

## AUTHORS PROFILE



**R Mario Vishal** currently pursuing his Bachelors Degree in Computer Science from SRM institute of science and Technology. His current interests lie in the Domain of Machine Learning and Artificial Intelligence. He is meritorious student. He is planning to pursue his masters in Artificial Intelligence.



**Deekshith Maram** currently pursuing his Bachelors Degree in Computer Sciences from SRM Institute of Science and Technology. He is interested in the domain of cloud computing and data analysis. He is meritorious student. He later wants to pursue his masters in data science.