Parking Assistance System using ANPR

Abhishek Kumar Shukla
Asst. Professor, Dept of CSE
ABES Engineering College
Ghaziabad, India
abhishek.shukla@abes.ac.in

Yash Kumar
Dept of CSE
ABES Engineering College
Ghaziabad, India
yash.19b101127@abes.ac.in

Shourya Nagpal
Dept of CSE
ABES Engineering College
Ghaziabad, India
shourya.19b101103@abes.ac.in

Vishal Singh Chaudhary
Dept of CSE
ABES Engineering College
Ghaziabad, India
vishal.19b101139@abes.ac.in

***Abstract*— The shortage of space results in people parking their vehicles almost anywhere without organisation. Usually what happens is that in unorganised spaces, people park their vehicles behind other already parked vehicles and this many times leads to chaos when the vehicle parked in the interior needs to get out of there as some vehicle is always blocking the exit. There needs to be a medium to contact the driver of the the vehicle blocking the exit so that the stuck vehicle owner can ask them to move their vehicle a bit so that they can bring out theirs. Tackling this issue is the first phase of our journey towards fixing the vehicle management system of India.**

How are we going to do that? Let's suppose A's car is stuck behind B's car in an unorganised parking lot. A and B will both have an app designed by us in their mobile phones. A will just need to open the app and scan the number plate of B from our app's scanner and our tech will read that number plate and give out the details B and the car. A will also have options to contact B via text, call, or just a simple notification alert to B so that he/she knows that they need to move their car.

Keywords — Automatic Number Plate Recognition (ANPR), Artificial Neural Network (ANN), Optical Character Recognition

I. Introduction

The size of a country like India is gauged not only in terms of its area. The population and number of resources required to run this kind of a great country is of great importance to the researchers. In direct proportion with the number of people in the nation, the number of vehicles being used for their locomotion is also growing rapidly. As the vehicles on road increase, they pose a number of problems along with their obvious benefits.

A. Status of vehicles on Indian roads

In the year 2019, 295.8 million vehicles were registered in india, making it the 3rd largest network of roads. More than sixty percent of Indian population travels by road. 200 crore metric tonnes of cargo was transported in 2017 alone.

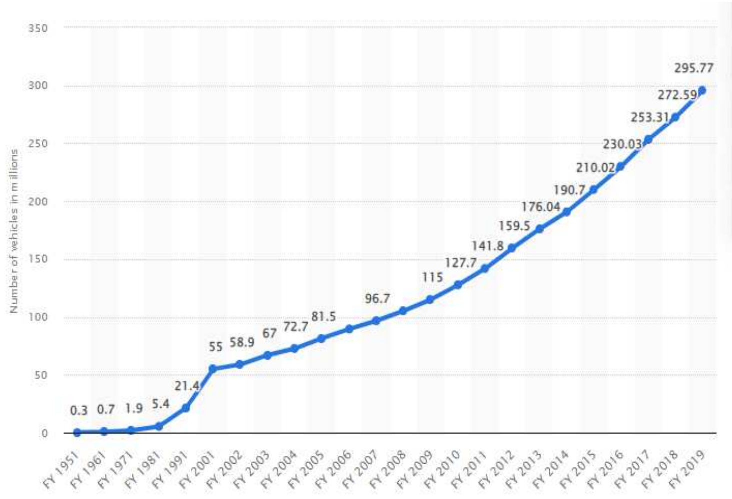

Fig. 1. Number of active vehicles in India FY 1951-2019

B. Vehicle related crimes in India

Delhi is at the top of the list of vehicles being stolen in any city across India. A vehicle is stolen every 12 minutes in capital city. Mumbai sees a theft every 4 hours. In Bengaluru every two hours a vehicle gets stolen. This data is according to 2019 data from the National Crime Records Bureau (NCRB).

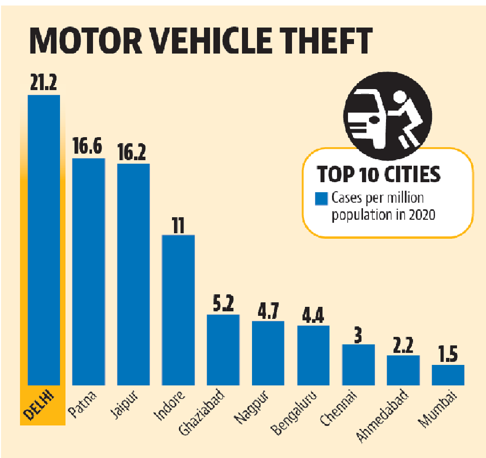

Fig. 2. Motor vehicles theft with respect to cities

18 Indian states and Union Territories have collected an approximate of Rs 2000 crore in fines via the e-challan system for violating traffic rules since September of 2019. This has been seen after the amendment of Motor Vehicles Act.

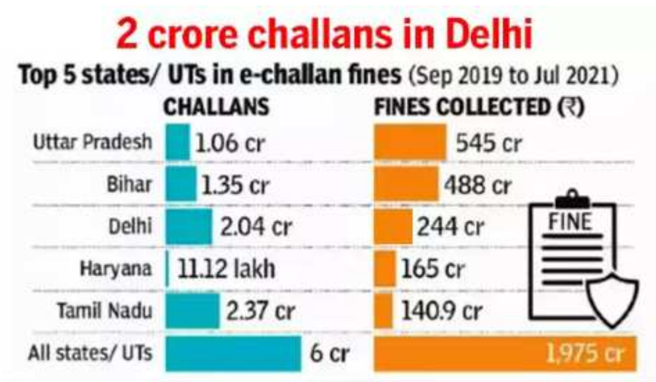

Fig. 3. Top 5 States and UTs in terms of Challans

C. Unorganized parking scenarios

Cars in India are standing over 95% of the times and are on the roads < 5% of the time. An average car is parked over 8,360 hours and only 400 hours being driven every year.

A study showed a lot of land area being wasted because of the alarmingly high ask for parking space in city areas. Delhi alone requires 471 additional parking spaces each year, Chennai 100, Chandigarh 58, and Gurgaon 179. A total of 85% of parking spaces are used by cars or two-wheelers, but only 4-5% of travel is provided by them. While buses account for only 4-5% of parking, they take 20x people than cars.

The result of this statistics is the plethora of unorganized parking in the country. And that results in chaos and vehicles getting stuck in the unauthorized parking plots. Ultimately a lot of time and resource is wasted over this situation.

D. Usage of ANPR to solve these problems

The problem of vehicle related crimes, e-challans, toll collection, unorganized parking etc., can be taken into control by the use of ANPR technology.
ANPR is a technology innovating the image processing of systems which is used to capture the details of vehicles just by using the photo of its number plate. This system is still in its early phases in the country.
It is an application of the PC vision technology.

A typical ANPR system follows these steps:

i. Capturing vehicle image(preferably number plate)
ii. Pre-processing
iii. Extraction of number plate
iv. Segmentation of characters
v. Recognition of characters

The ANPR system works by following these steps. First of the image of the vehicle is captured using a specialized camera. The requirement for a specialized camera getting eliminated is sort of a major objective of this research work. Post that, a bit of pre processing is done on the captured image. It eases the task of segmentation. After pre processing, the number plate is extracted from the image of the car. It is extracted using the concept of contours and contrasts. The segmentation of each character present in the number plate is done following the extraction. Finally each segmented character is recognized and the final number plate is detected. This involves the technology of Optical Character Recognition (OCR).

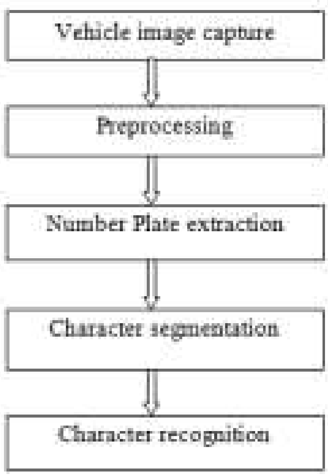

Fig. 4. Steps of ANPR

II. Literature Review

As a team we have gone through various articles, models, algorithms , and APIs for this project .
Some of the research papers which we surveyed are mentioned below. Their conclusions and learning is also discussed briefly.

[1] This illustration shows how the Otsu method and K nearest neighbor (KNN) are applied and how accurate they are. RGB images are converted into binary images using the Otsu method. The KNN algorithm is robust against noise and is used for classification. As part of pattern recognition, pixels are converted into binary data through feature extraction. Through the use of the Otsu method, the KNN classifies neighborhood test data based on its comparison with training data. A learning algorithm and classification

process are used to categorize testing data. In the Otsu method, threshold values are not affected by the binary vector used in pattern recognition. To obtain accurate results and better binary segmentation, it is necessary to adjust the distribution of the pixel values of an imagification. [1].

[2] A supervised K-means ML algorithm is applied to blurry license plate images to classify the characters into subgroups and then classify these subgroups further with a neural network. In character recognition, this method will distinguish obstacles based on the orientation of the camera, the velocity of the car, as well as the environmental conditions. Images of dull characters are captured by the camera. Due to its efficiency, SVMs are applied in here. SVM classifiers should be able to classify samples from different classes. SVM algorithms are affected by the high number of samples, which increases their workload. With supervised K-means, it is easy to classify difficult-to-recognize characters. In addition, SVMs can be used in classifying the character of subsets and reduce the number of character classes.[2].

[3] In this study, we focus on classifying characters from number plates using a KNN algorithm. In order to capture images of vehicles traveling on a highway, images are received from a special purpose camera fit on the roads. Number plates contours are detected. Each contour is classified using the KNN algorithm. In order to train the KNN, thirty six characters containing twenty six alphabets and ten digits are considered. Character recognition techniques such as artificial neural networks are compared with the algorithm on previously segmented characters. [3]

[4] Demonstrates the use of a system to identify characters of registration plate and uploads information on a server. This process is hindered by low quality images. It is then segmented for extraction of the image of the number plate. Characters are then compartmentalized and KNN is used to find the characters. [4].

[5] Deep learning methods are used. Machine learning classifies license plates. Two parts are involved in this system. The intial part preprocess and take out features from HOG. The next phase groups every character to study and differentiate each character on the plate. ELM is a quick supervised model running on one hidden layer feed forward networks and its efficiency is similar to SVM. ELM is needed to classify while HOG is used to fish out attributes from the number plate to identify elements on the plate. [5].

[6] Number plate detection in India is plagued by a variety of problems. Due to the many font sizes, different colors, and double-line number plates, etc., there is a lot of confusion. The results are also highly inaccurate. This site provides a real-life solution to all of these problems. Support Vector Machines are used to detect plate contours and Artificial Neural Networks are used for character recognition. Various algorithms are employed to reduce noise and improve plate recognition, as well as neural networks for optimal results with a reduction of camera constraints.[6].

[7] SVM, ANN, and KNN algorithms have been used in this paper for recognizing UK number plates. Machine learning algorithms are trained with a huge car image dataset created from scratch in order to develop efficient applications. The most powerful machine learning algorithms are combined with high-resolution digital images to create fast and reliable number plate applications. In today's world, computational sciences have made incredible advances. Several computer vision techniques are used to analyze the car image, including KNN, SVM, and several other approaches. In the final result, the number plate of the car is identified. [7].

[8] Among the challenges associated with the proposed system are blur number plates in tough weather, fast vehicles, and unique traffic conditions. These challenges further complicate the process of extracting relevant information from the vehicle number plate. Using real-time and intelligent algorithms to overcome the challenges of a hardware platform. This dataset includes images from various paths, including roads, streets and highways, daytime and nighttime, inclement weather, and other factors such as number plate clarity. There is no impact on system due to changes in the lighting, dimensions, and visibility of the number plates. In high-speed applications without language specificity, we recommend ANPR due to its industry-proven reliability. A defined set of answers to number-plate recognition challenges has been developed using these techniques and algorithms, along with their dataset. [8]

III. SETUP OF EXPERIMENT

A. Dataset

Fig. 5. Car Image dataset

The initial dataset of the research is a large number of car pictures consisting of their number plates.
These images will be used for training and testing purpose. This data has been collected over multiple days.

Next up, each character from these number plates is individually extracted to train the CNN model.

Before training, those characters are processed through a Deep neural network program which converts the images into binary images. It eliminates the noise from the images.

Proper contouring and contrasting is done using that program.

The initial character dataset is kept in train and validation folders.

Fig. 6. Character folder

This is the train folder which has multiple photos of each kind of character up for processing.
Each of these images will be processed and saved in place.

If we look inside these folders, this is how the characters are stored.

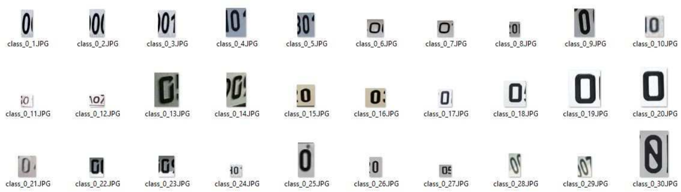

Fig. 7. Initial characters

Post processing, the characters look like this.

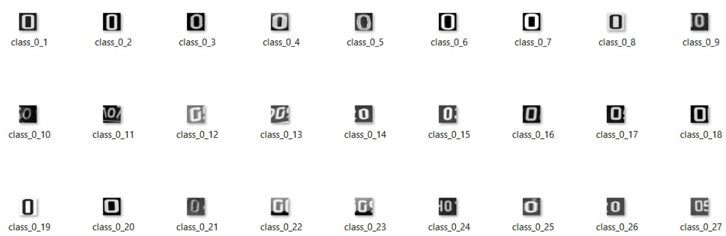

Fig. 8. Processed characters

Let us have a look at step by step phases of the conversion.

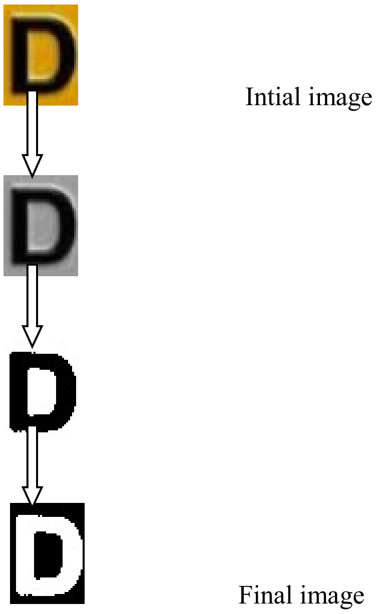

Fig. 9. Process of image modification

This is how the start to end process of image modification takes place.

IV. METHODOLOGY

A. Name of Algorithms used

- OpenCV + Deep neural networks for image modification/conversion(explained in Datasets).
- Contour detection using OpenCV for Number plate location and character segmentation.
- YOLO V3 for Number plate location.
- CNN for character recognition and output generation

B. Steps of Execution

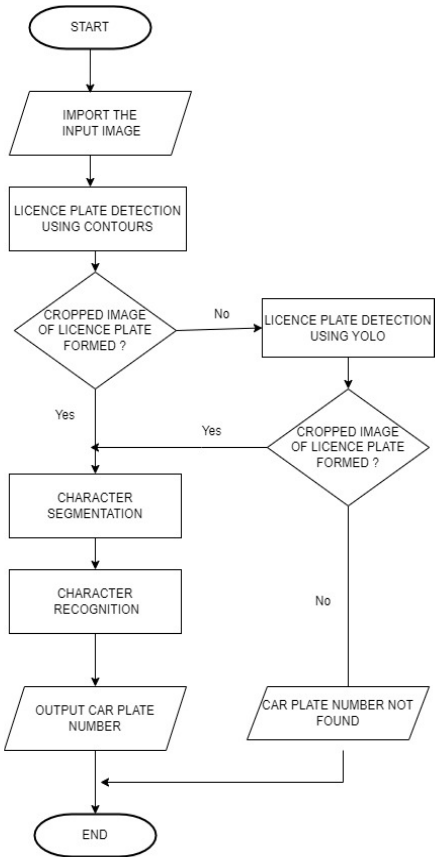

Fig. 10. OCR model algorithm

1) First of all, a car image is captured and input into the program.
2) Contour detection method is applied on the image to locate the number plate.
3) If cropped number plate is formed then continue to step 4 to step 6, otherwise step 7.
4) Character segmentation is done on the cropped plate image.
5) After that Character recognition is performed.
6) Finally the resultant text is given as output.
7) Plate detection using YOLO is done.
8) If cropped number plate is formed then continue to step 4 to step 6, otherwise step 9.
9) The user is prompted that the car number is not found.

Let us analyze these steps in detail.

Firstly let's have a look at the **Plate detection using contours algorithm**.

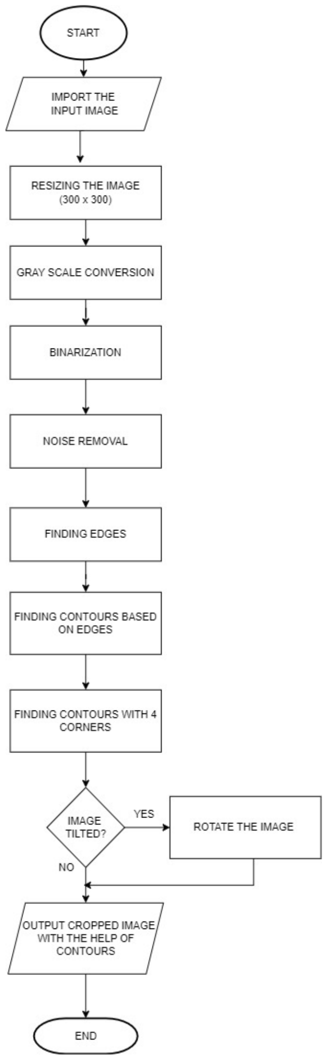

Fig. 11. Plate location using contours algorithm

In this algorithm, firstly the car image is taken as input.

Fig. 12. Image acquisition

The next stage is pre-processing of the image.

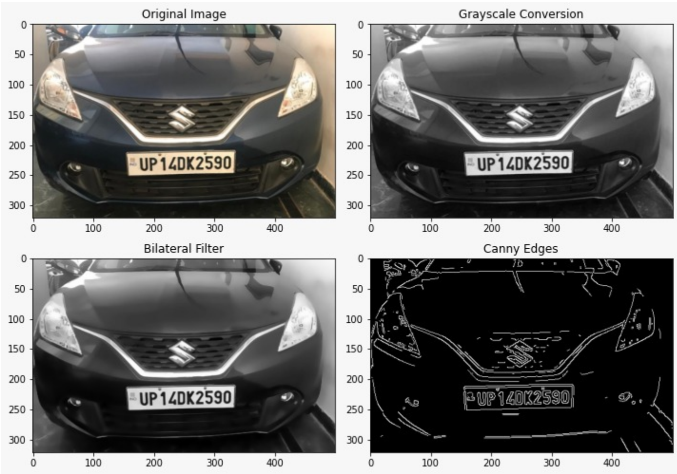

Fig. 13. Pre-processing stage

In this stage first the image is resized properly.
After that the images are converted into grayscale for proper visibility.
Then binarization post noise removal is done to find clear edges and detection of the plate.

In the next step the plate is finally located in the image by finding contours consisting of 4 corners.

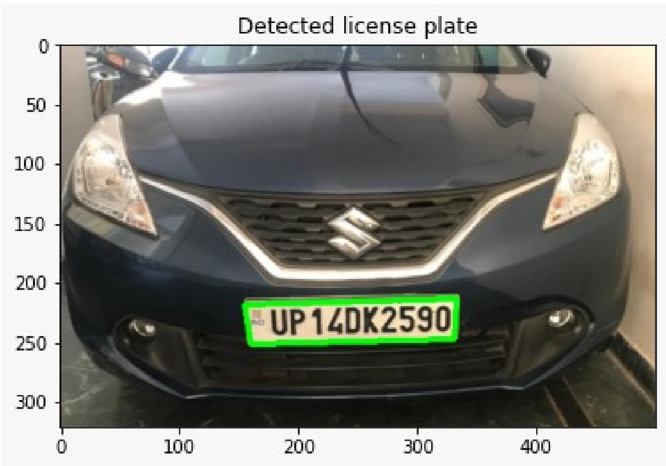

Fig. 14. Plate location

Now as the plate on the vehicle has been detected, it is cropped out as a singular image.

If the alignment of the image is not correct, then it is rotated appropriately to get the correct alignment.

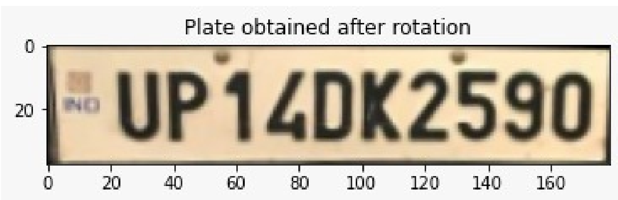

Fig. 15. Cropped number plate

Next up the cropped image is inverted so that contrast of the image is managed and converted into binary image.

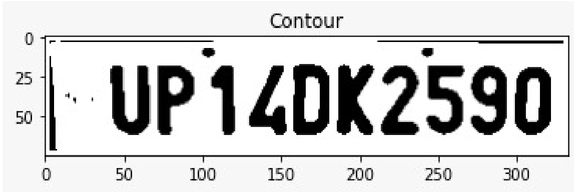

Figure. 16 . Binary image

Finally the characters from the cropped image are segmented into individual characters.

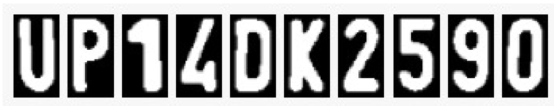

Fig. 17. Character segmentation

Now let's discuss the **YOLO V3 algorithm for plate detection :**

Firstly in this algorithm too the car image is taken as input.

Fig. 18. Image input

Next the parameters of the required image are initialized. The height, width, confidence threshold etc., of the image are initialized.
Then the class file named LP is accessed.
The YOLO configuration file and model weights are restored. This is done to restore the neural network.

The post production functions are defined in the next steps. It also involves drawing of the prediction box which will contain the number plate. Certain post process functions are used to remove the bounding boxes with low confidence level.

The next bit of code generates 4D contour from cropped image, initializes network input, fires up the forward pass to extract answer from output layer and inputs the picture to a post process method to remove reduced confidence level box and to draw the predictable box.

Finally the predicted box is located.

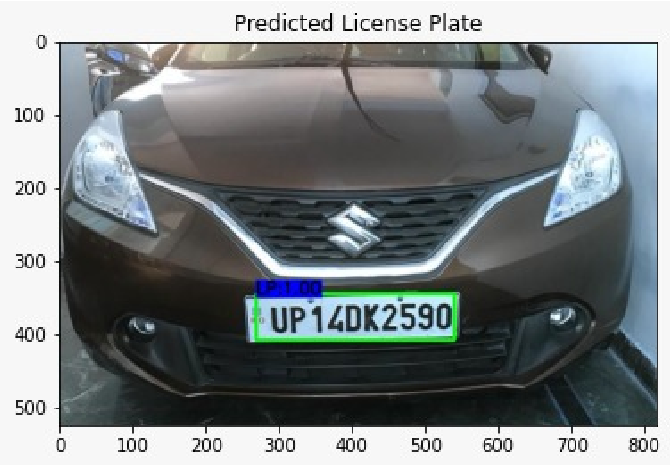

Fig. 19. Detected Number Plate

After this the number plate is cropped out from the image and the rest of segmentation and recognition procedure is performed as in the contour detection algorithm.

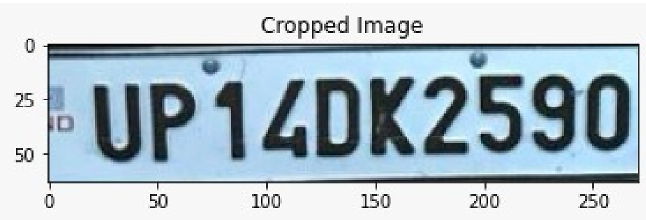

Fig. 20. Cropped Number Plate

Finally for **character recognition CNN** based model is used.A sequential object is created. 4 C layers are used with 'ReLu' being the activation function.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 28, 28, 16)        23248

 conv2d_1 (Conv2D)           (None, 28, 28, 32)        131104

 conv2d_2 (Conv2D)           (None, 28, 28, 64)        131136

 conv2d_3 (Conv2D)           (None, 28, 28, 64)        65600

 max_pooling2d (MaxPooling2D  (None, 7, 7, 64)         0
 )

 dropout (Dropout)           (None, 7, 7, 64)          0

 flatten (Flatten)           (None, 3136)              0

 dense (Dense)               (None, 128)               401536

 dense_1 (Dense)             (None, 36)                4644

=================================================================
Total params: 757,268
Trainable params: 757,268
Non-trainable params: 0
_________________________________________________________________
```

Fig. 21. CNN model

Next, a maxpooling layer is added having size of window as (4x4). Maxpooling is simply a sampling process of discretization. The aim of this task is to reduce the sample of representation of input, by reducing its dimensionality and making assumptions about attributes inside the sub regions.

To tackle overfitting, some dropout rate is added. Dropout is a hyper-parameter of standardization which is given initial values to inhibit Deep Networks from getting Overfitted. The dropout rate is kept at 0.4.

Flatten layer is added to flatten the node. It takes information from the last layer and portrays it in a 1D. The final layer having more density consists of 36 outputs as there are 26 alphabets (Z from A) and 10 digits (9 from)) up for recognition. 'Soft-max' activation is applied here being a multi-class classification issue.

Lastly, we will add two dense layers, one with 128, activation function 'ReLU' and the other for categorizing 26 alphabets (A to Z) plus ten digits (0 to 9) with activation function 'softmax'.

Grid Search has already optimized all the above parameters used in the model.

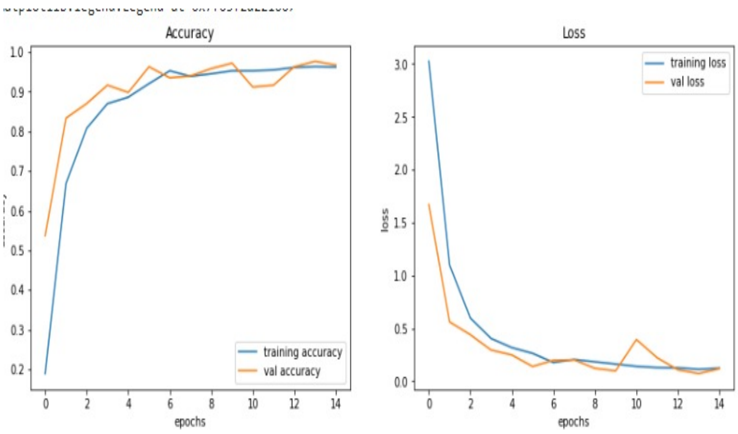

Fig. 22. Finding right number of epochs

By using image augmentation techniques such as width shifting and height shifting, we'll generate some more data to train the model using a generating class in Keras.

```
Epoch 1/15
932/932 [==============================] - 317s 339ms/step - loss: 3.0256 - accuracy: 0.1888 - val_loss: 1.6704 - val_accuracy: 0.5370
Epoch 2/15
932/932 [==============================] - 75s 80ms/step - loss: 1.0984 - accuracy: 0.6685 - val_loss: 0.5608 - val_accuracy: 0.8333
Epoch 3/15
932/932 [==============================] - 70s 75ms/step - loss: 0.5975 - accuracy: 0.8079 - val_loss: 0.4392 - val_accuracy: 0.8704
Epoch 4/15
932/932 [==============================] - 70s 75ms/step - loss: 0.4040 - accuracy: 0.8702 - val_loss: 0.2945 - val_accuracy: 0.9167
Epoch 5/15
932/932 [==============================] - 72s 77ms/step - loss: 0.3173 - accuracy: 0.8863 - val_loss: 0.2477 - val_accuracy: 0.8981
Epoch 6/15
932/932 [==============================] - 70s 75ms/step - loss: 0.2629 - accuracy: 0.9206 - val_loss: 0.1394 - val_accuracy: 0.9630
Epoch 7/15
932/932 [==============================] - 74s 80ms/step - loss: 0.1772 - accuracy: 0.9528 - val_loss: 0.1978 - val_accuracy: 0.9352
Epoch 8/15
932/932 [==============================] - 70s 76ms/step - loss: 0.2050 - accuracy: 0.9388 - val_loss: 0.2015 - val_accuracy: 0.9398
Epoch 9/15
932/932 [==============================] - 70s 76ms/step - loss: 0.1834 - accuracy: 0.9453 - val_loss: 0.1216 - val_accuracy: 0.9583
Epoch 10/15
932/932 [==============================] - 73s 78ms/step - loss: 0.1621 - accuracy: 0.9528 - val_loss: 0.0974 - val_accuracy: 0.9722
Epoch 11/15
932/932 [==============================] - 70s 75ms/step - loss: 0.1403 - accuracy: 0.9528 - val_loss: 0.3938 - val_accuracy: 0.9120
Epoch 12/15
932/932 [==============================] - 72s 78ms/step - loss: 0.1286 - accuracy: 0.9549 - val_loss: 0.2212 - val_accuracy: 0.9167
Epoch 13/15
932/932 [==============================] - 70s 76ms/step - loss: 0.1263 - accuracy: 0.9614 - val_loss: 0.1077 - val_accuracy: 0.9630
Epoch 14/15
932/932 [==============================] - 70s 76ms/step - loss: 0.1123 - accuracy: 0.9635 - val_loss: 0.0714 - val_accuracy: 0.9769
Epoch 15/15
932/932 [==============================] - 74s 80ms/step - loss: 0.1227 - accuracy: 0.9624 - val_loss: 0.1185 - val_accuracy: 0.9676
```

Fig. 23. Processing of detected number plates

V. RESULTS

A. Actual vs Predicted Number plates

	ID	NUMBER	prediction
0	1.png	DL7CQ1939	DL7CQ1939
1	2.png	KL65H4383	KL65H4383
2	3.png	HR26DK8337	HR26DK8337
3	4.png	MH20EJ0364	MH20EJ0364
4	5.png	DL8CAF5030	DL8CAF5030
5	6.png	CG07CA5144	6GQ7GA51
6	7.png	MH12DE1433	MH12DE1433
7	8.png	GJ01RK4770	GJ01RK4770
8	9.png	MH12BG7237	MH12BG7237
9	10.png	UP14CC6162	UP14CC6162
10	11.png	KL55R2473	KL55R2473
11	12.png	HR24DX6589	MR26DI65B9
12	13.png	UP14DK2590	UP14DK2590
13	14.jfif	RJ14WC9445	RJ14MC924
14	15.jpg	UP14EL7210	UP14EL7210
15	16.jpg	UP14FE8938	UP14FEB93B
16	17.jpg	HR05BE0683	HR05BE0683
17	18.jpg	UP14CF8331	UP14CF8331
18	19.jpg	HR05X8772	HR05X8772
19	20.jpg	UP14CU8535	UP14CU8535

Fig. 24. Actual vs predicted number plate characters

We see that 16 out of 20 number plates are correctly detected.

TABLE I RESULTS OF NUMBER PLATE DETECTION

B. Comparison between different models and their accuracy

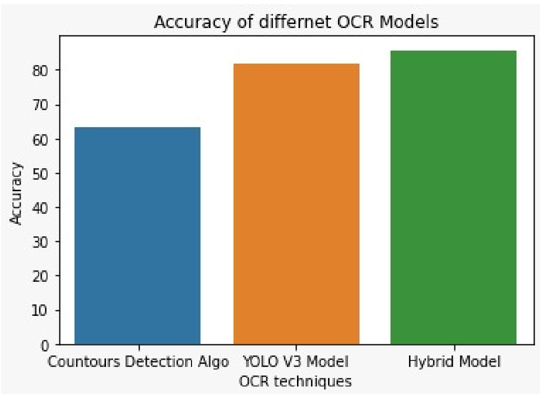

Fig. 25. Accuracy of different OCR models

It is visible from the graph that the Hybrid model has the highest accuracy percentage amongst other models.

C. Comparison between different models and their execution time

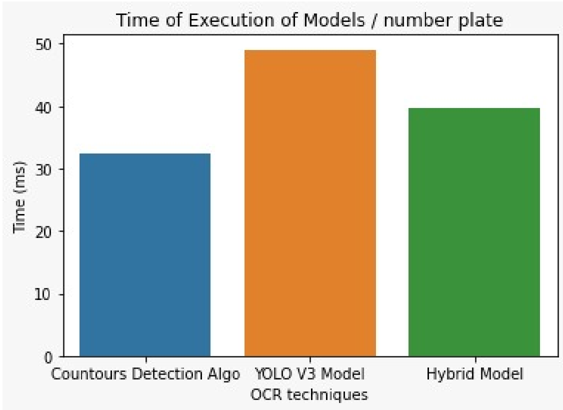

Fig. 26. Model execution time per number plate

It can be seen from the graph that the Contour detection model has the lowest execution time in milliseconds amongst other models.

VI. FUTURE SCOPE AND LIMITATIONS

Total number of images	**Number of accurate predictions**	**Success rate (Efficiency %)**
20	16	80

A. Limitations

- Model Accuracy : The Indian number plates are very complex and have many different styles which makes the model accuracy stay low.
- API : Building APIs for use in applications is challenging.
- Cameras : Usually specialized camera is required for the general ANPR purpose. And it is not very affordableFigures and Tables

B. Future Scope

The increasing adoption of ANPR in tolling booths will be great for the market. With improved road connectivity, toll booths have shot up in numbers around the globe. There is an increasing requirement for automatic solution to decrease crowding and monitoring vehicle systems at toll setups, which is encouraging the application of ANPR for tolls. Frauds and scams can be decreased using ANPR, and car charging stations can be improved.

VII. CONCLUSION

In this research we have combined several ANPR models and algorithms to create one single strong entity. We also created system for aligning the number plates according to the screen.

We saw improved accuracy and speed for the number plate detection task. Indian number plates had never seen such performance. There definitely is scope for improvement in the model which will be carried on in the future.

A large amount of dataset is needed in order to train these Deep Learning models to their full capacity. The requirement for better processing units was also observed during the execution of this project.

Government of India should also be bringing policies for making of uniform Number plates mandatory.

One major find of this research is that using contour based and CNN based models for plate detection and recognition together provide much better results than using any of them individually.

REFERENCES

[1] Maulidia R. Hidayap, Isa Akhlis2, Endang Sugiharti3 Recognition Number of The Vehicle Plate Using Otsu Method and K-Nearest Neighbour Classification, Scientific Journal of Informatics Vol. 4, No. 1, May2017.

[2] Liu, W.-C., &Lin, C.-H. (2017). A hierarchical license plate recognition system using supervised K-means and Support Vector Machine, 2017 International Conference on Applied System Innovation(ICASI).

[3] Quiros,A.R.F.,Bedruz,R.A.,Uy,A.C.,Abad,A.,Bandala,A.,Dadios,E.P.,Sa lle,D.L.(2017),A kNN-based approach for the machine vision of character recognition of license plate numbers, TENCON 2017 – 2017 IEEE Region 10 Conference.

[4] Thangallapally,S.K.,Maripeddi,R.,Banoth,V.K.,Naveen,C., &Satpute,V. R.(2018).E- Security System for Vehicle Number Tracking at Parking Lot,(Application for VNIT Gate Security). 2018 IEEE International Students Conference on Electrical, Electronics and Computer Science (SCEECS).

[5] Subhadhira,S., Juithonglang, U., Sakulkoo, P., &Horata, P. (2014). License plate recognition application using extreme learning machines,2014Third ICT International Student Project Conference (ICT-ISPC).doi:10.1109/ict-ispc.2014.6923228.

[6] Singh, A. K., & Roy, S. (2015). ANPR Indian system using surveillance cameras,2015 Eighth International Conference on Contemporary Computing(IC3).

[7] Leticia Fernandez sanchez, cranfield university, Automatic number plate recognition System using machine learning techniques, PHD Thesis, cranfielduniversity,2017-18.

[8] Rahim Panahi and Iman Gholampour Accurate Detection and Recognition of Dirty Vehicle Plate Numbers for High-Speed Applications,IEEE Transactions on intelligent transportation systems, vol. 18, no. 4, april2017.