Automatic License Plate Recognition

Made By:-
Abhishek Goyal (2016csb1027)
Komal Chugh (2016csb1124)

Introduction

Automatic license-plate recognition (ANPR) is a technology that uses optical character recognition on images to read vehicle registration plates to create vehicle location data.

It can be used for many practical applications such as automatic toll collection, traffic law enforcement and road traffic monitoring.

Algorithm

Components Involved

Our ALPR system can be broken down into 3 major components:

- License plate extraction
- Character Segmentation
- Character Recognition

Component 1: License plate extraction

Extracting only the part of image displaying the license plate from a rear or front-view image of the vehicle.

Component 1: License plate extraction

1. First the colored image is **converted to grayscale** for easier processing.

   Grayscale intensity value=0.2989 * R + 0.5870 * G + 0.1140 * B

2. Then the **Gaussian low-pass blur filter** with parameter σ = 2.5 is applied to remove image noise.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Gaussian Filter Equation

Component 1 (continued)

3. **Vertical Sobel operator** is then used to detect vertical edges present in the image.

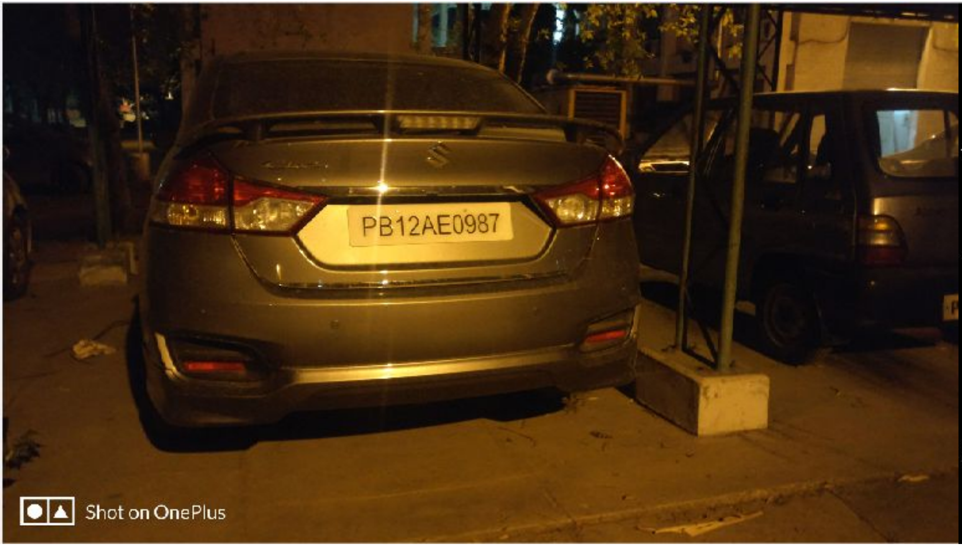
$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

3x3 Vertical Sobel Operator Mask

The given mask is convoluted over the whole image.

4. Resultant image is thresholded using **Otsu’s binarization**.

Component 1 (continued)



Initial Input Image

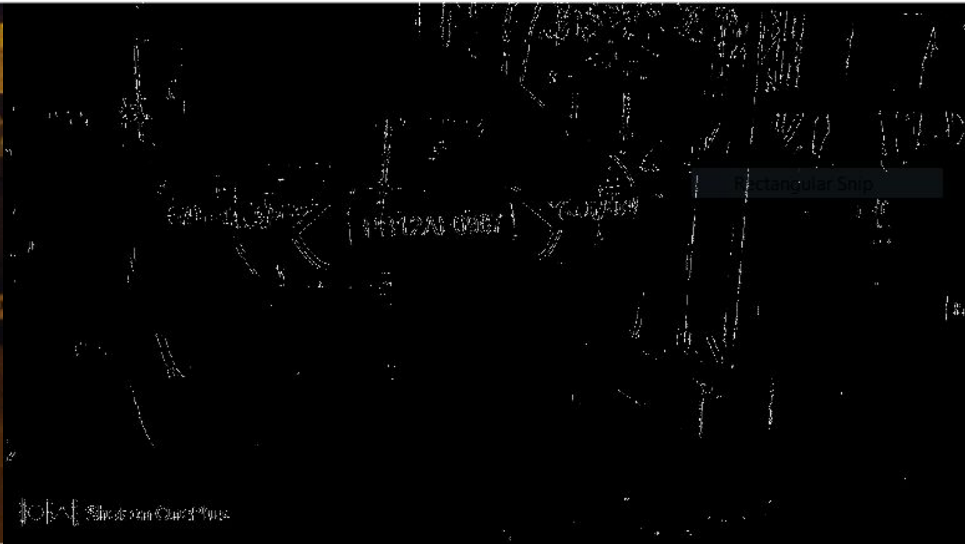


Image obtained after performing first 3 steps

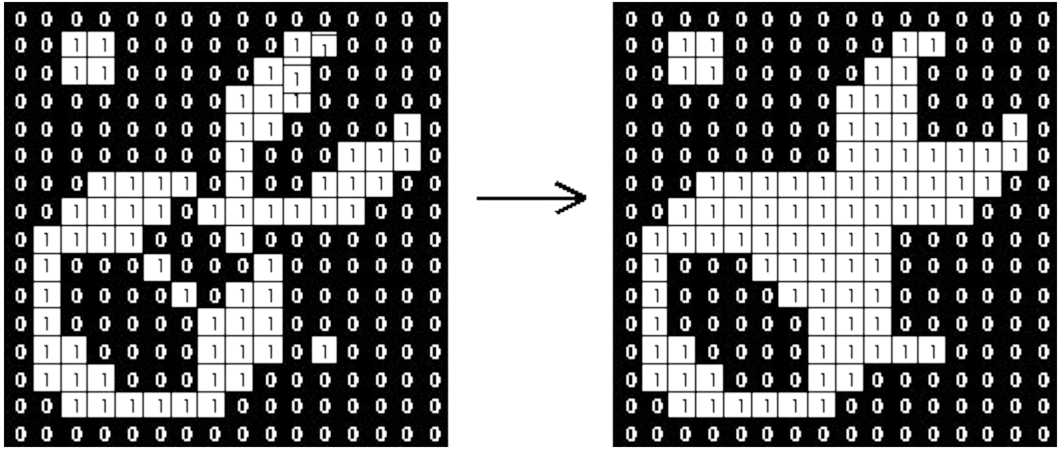
Applying vertical sobel operator on the grayscale noise-minimized image, displays out the vertical edges with the characters inside the plate outputting multiple close-to-each-other foreground lines.

Component 1 (continued)

5. **Morphological closing operation** is then applied on the resultant image using a rectangular structuring element. The closing of a binary image A by a structuring element B is the erosion of the dilation of that set,

$$A \bullet B = (A \oplus B) \ominus B,$$

where $\oplus$ and $\ominus$ denote the dilation and erosion, respectively.

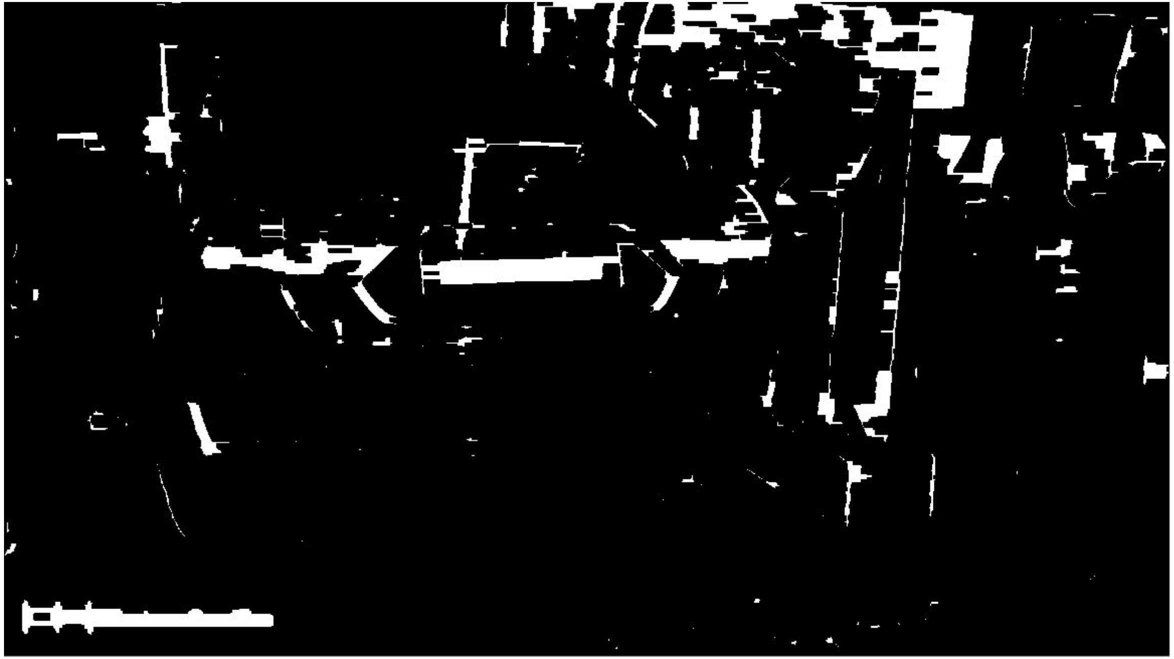


0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0
0	0	0	1	1	1	1	0	1	0	0	1	1	1	0	0
0	0	1	1	1	1	0	1	1	1	1	1	1	0	0	0
0	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	1	0	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	1	0	1	0	0	0	0
0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	1	1	1	0	0	0	0
0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The effect of a closing on a binary image using a 3×3 square structuring element

Component 1 (continued)

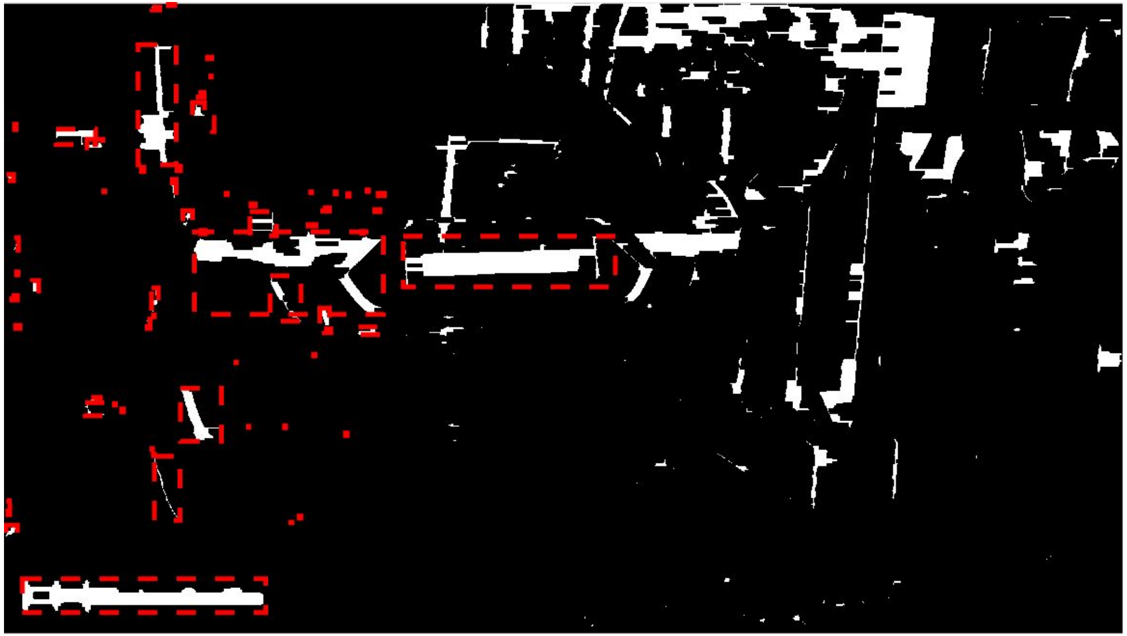


Closing applied on the previously obtained image

Applying closing using the rectangular element, the nearby vertical lines detected earlier corresponding to the license plate digits get combined into a single rectangle-like shape.

Component 1 (continued)

6. **Each connected component** in the image obtained is checked if it resembles a rectangle and **a rectangular bounding box is created** by using the max and min x&y coordinates present in that component.



Rectangular bounding boxes found, highlighted by red dash lines.

Component 1 (continued)

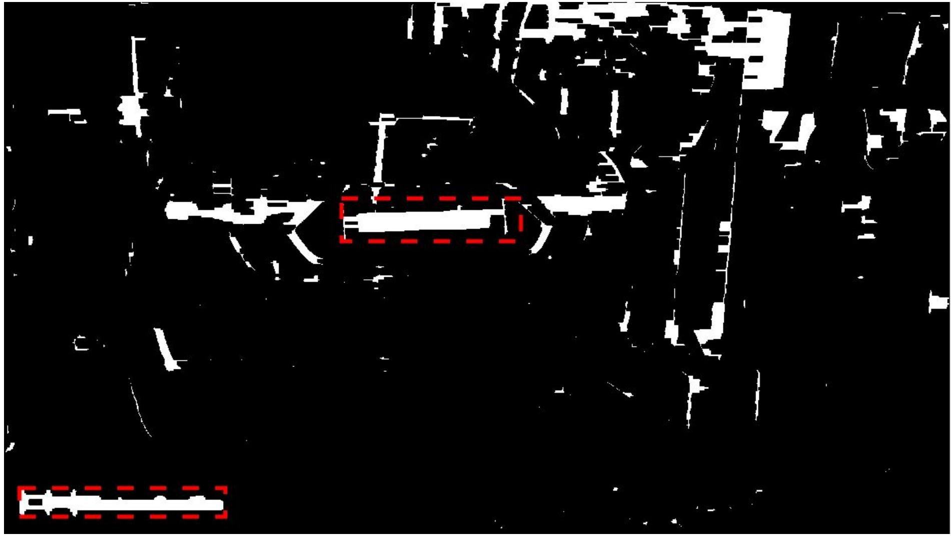
7. **Out of all these rectangular bounding boxes** found, only **a specific few are selected** using the following criteria:-

    - Aspect ratio of height and width between 2.5-8 value.
    - Area of bounding box (pixel^2) between 5000-500000
    - Edge Density (number of white pixels/total pixels) greater than 0.1
    - The box is present in the bottom 4/5th part of the image.

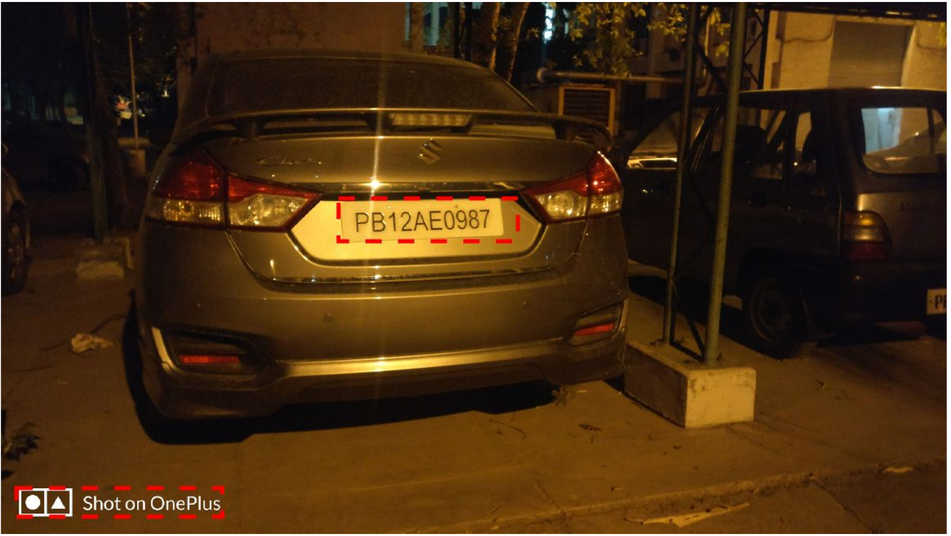
This filters out almost all the other bounding boxes with only the license plate one and some others left.

8. The coordinates corresponding to the remaining bounding boxes are subsequently extracted from the grayscale image.

Component 1 (continued)



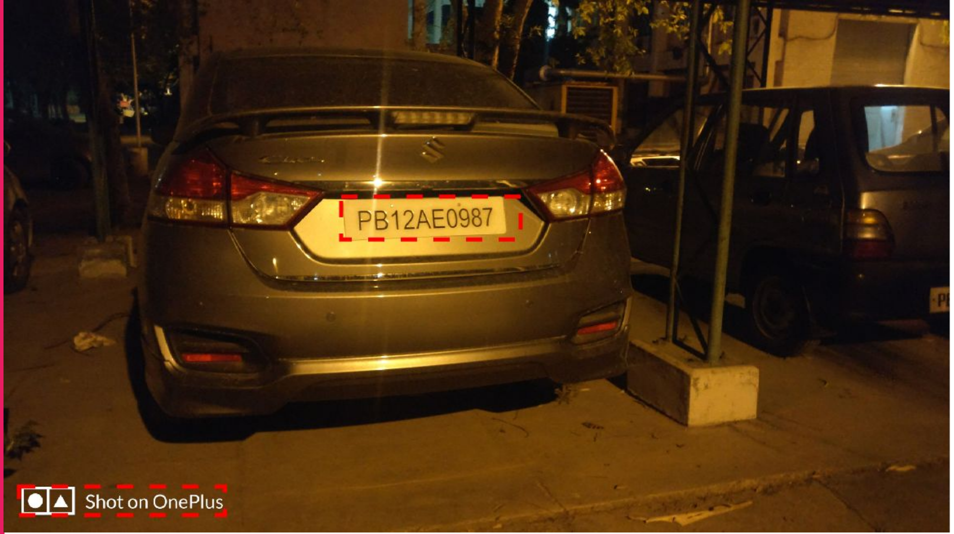
Selected bounding boxes on the previously obtained image

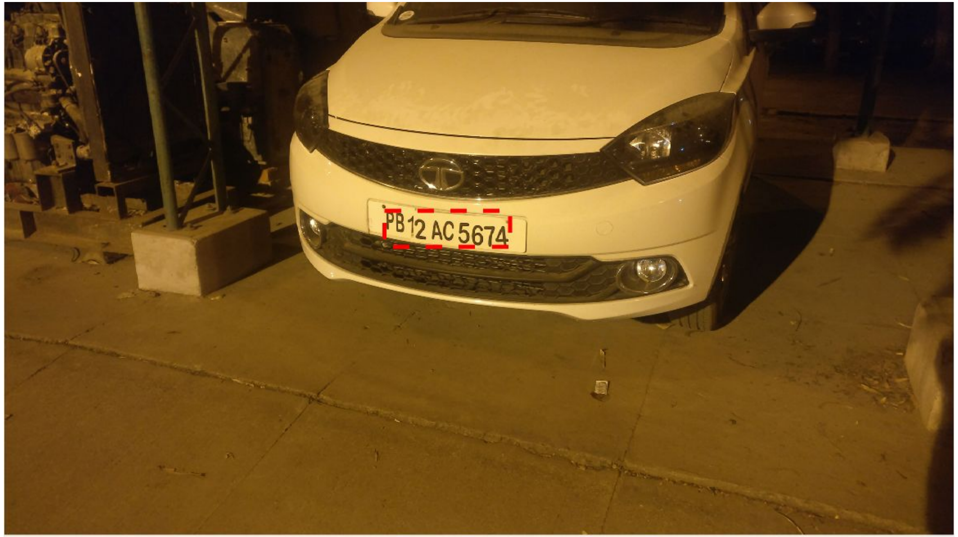


Same bounding boxes displayed on the original image

After filtering out the other bounding boxes, only the box corresponding to the actual plate and one more remains.

Component 1 Results




PB 12 AC 5674

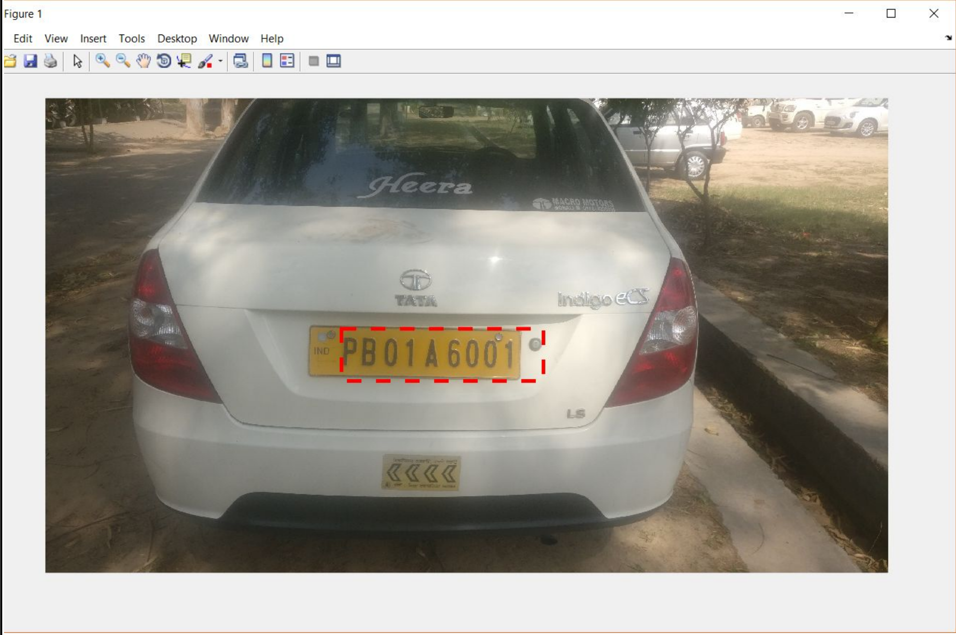
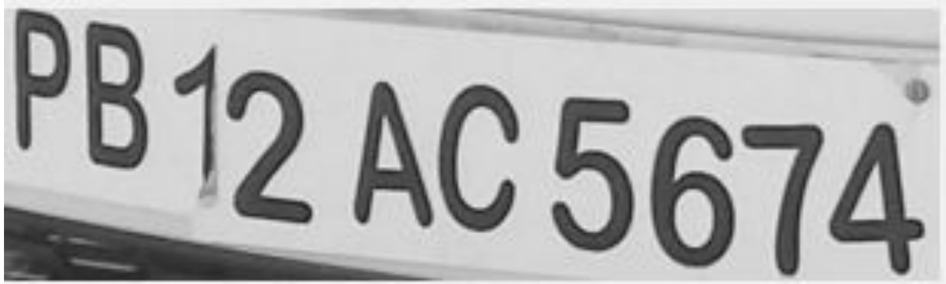


Figure 1
Edit View Insert Tools Desktop Window Help
Heera
MACRO MOTORS
TATA
Indigo eCS
IND
PB 01 A 6001
LS


PB 12 AC 5674

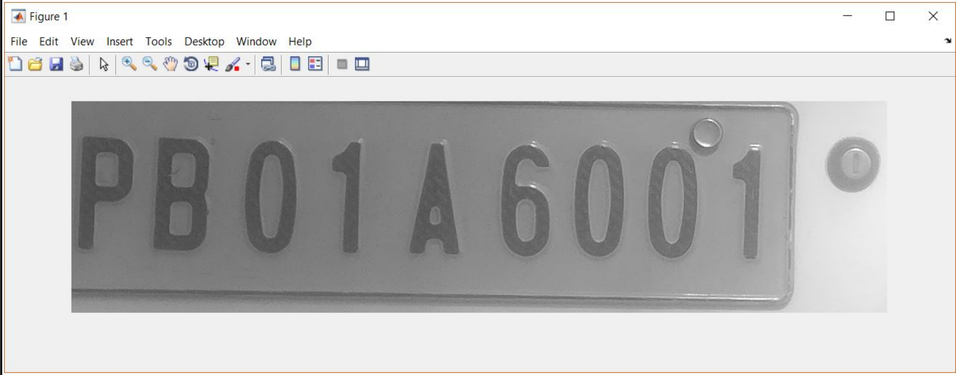
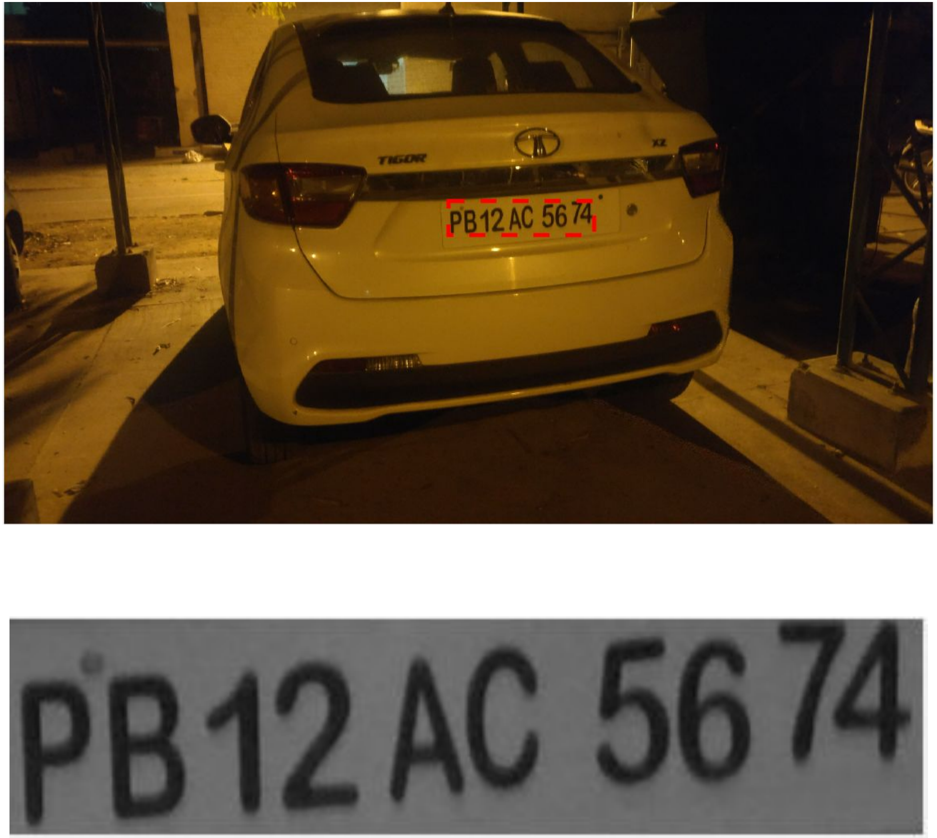


Figure 1
File Edit View Insert Tools Desktop Window Help
PB 01 A 6001

UP 16 U 4479
Shot on OnePlus
UP 16 U 4479
Shot on OnePlus


TIGOR
XZ
PB 12 AC 56 74
PB 12 AC 56 74

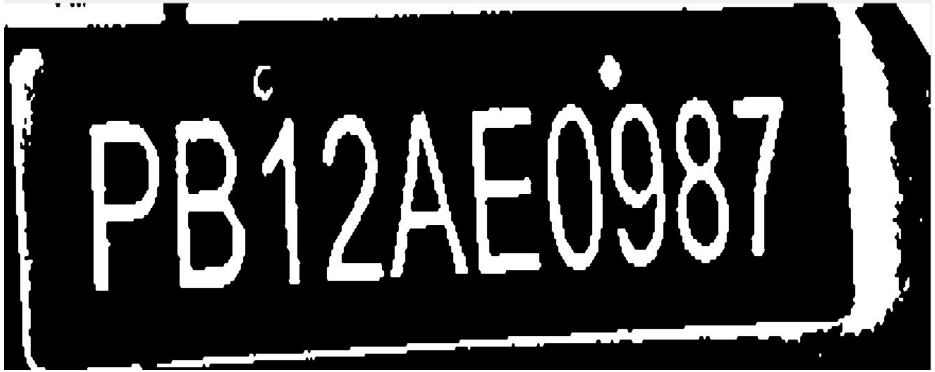
Component 2: Character Segmentation

Extracting the individual characters from the cropped image of the license plate.

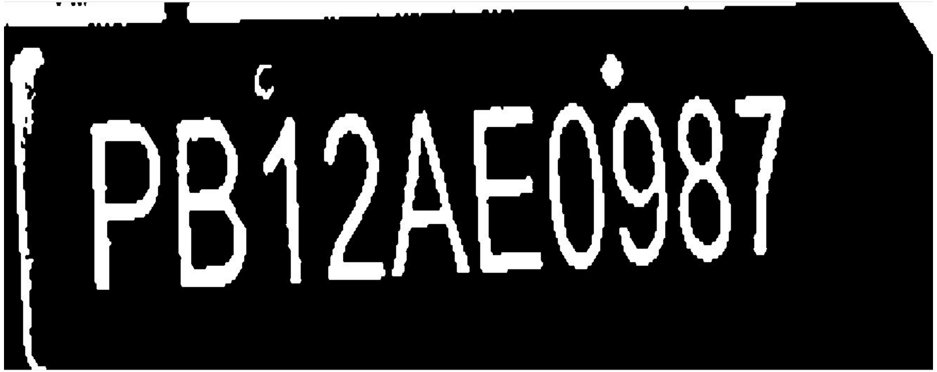
Component 2: Character Segmentation

1. The grayscale image(s) obtained from Component 1 are first resized to a set size so that set values can be used for future filtering to be done. The image is inverted so character pixels become foreground pixels.
2. The image is then binarized through **adaptive Otsu thresholding**. The algorithm splits the image into two classes of pixels in such a way that their inter-class variance is maximum possible.
3. Then through connected component analysis, only the connected components with number of pixels between 500 and 50000 are kept.

Component 2 (continued)



Cropped license plate image resized, inverted and thresholded using Otsu's method



Keeping the connected components only between the specified range of pixel count

Using connected component analysis, the extra foreground pixels present in the extracted image not corresponding to license plate characters are removed to an extent.

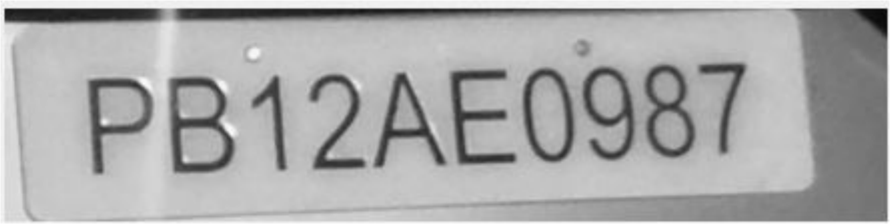
Component 2 (continued)

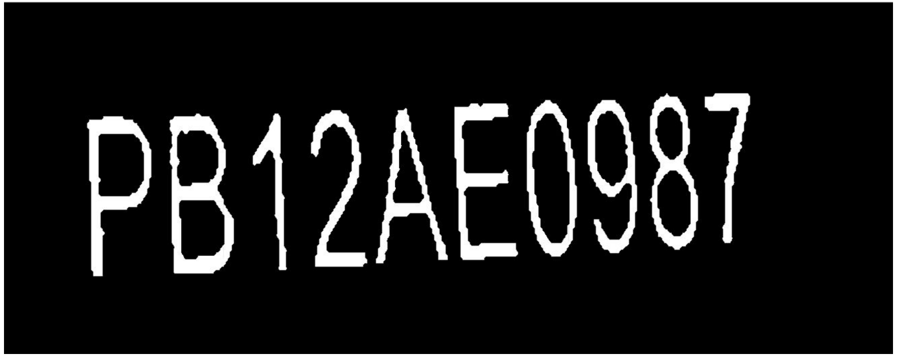
---

4. Then the rectangular bounding box around each connected component is found using the same method as before and only those matching a certain criteria are kept. The criteria followed is:-
    - Aspect ratio less than 0.95.
    - Area between 100-1000000 pixels.
    - Rectangle width>20 pixels and height>80 pixels.

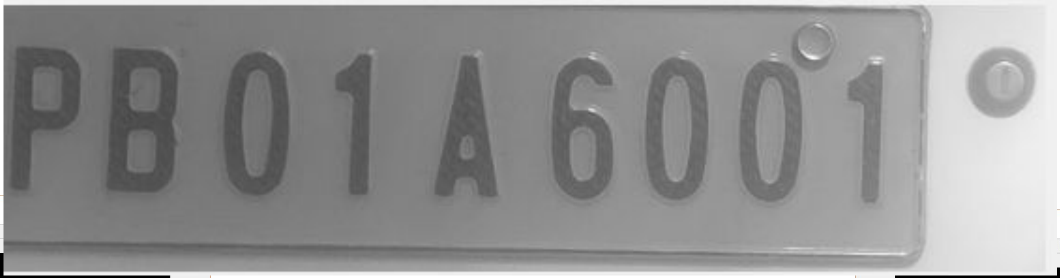
5. Only the characters required are left after this, and these characters are extracted one-by-one by connected component analysis.

Component 2 Results

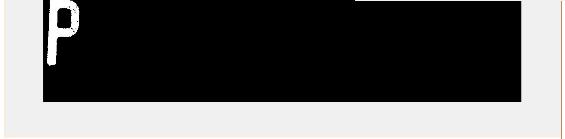


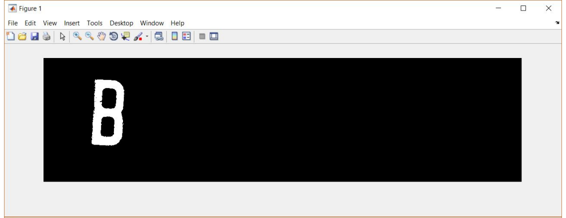


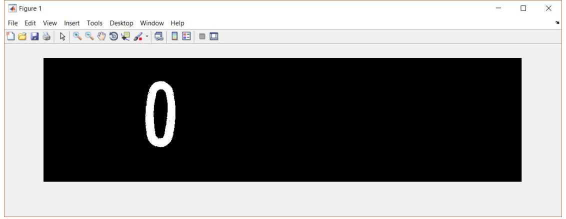
(The individual segmented characters displayed together as a whole plate).

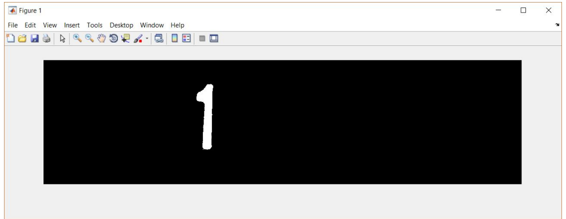


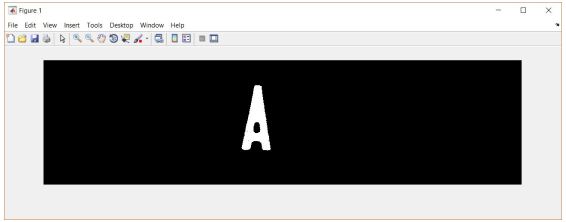
Input extracted plate image

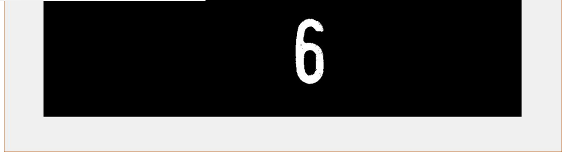


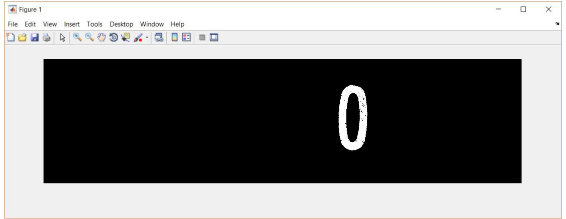


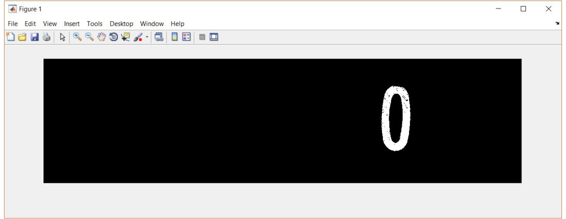


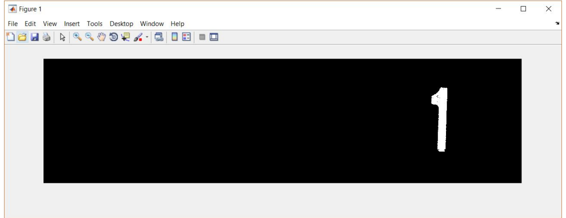












Component 3: Character Recognition

Recognizing the individual characters segmented earlier in Component 2 and keeping them in pure text form.

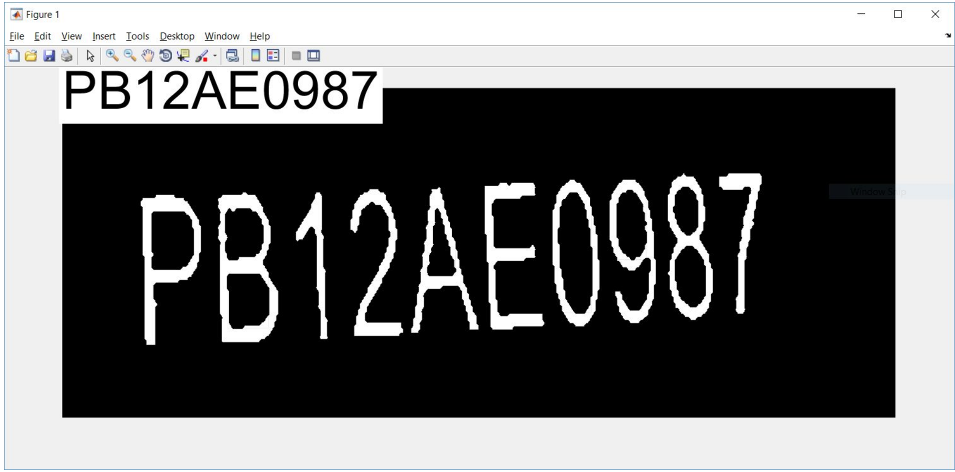
Component 3: Character Recognition

The letters and digits extracted are recognized using the in-built Optical Character Recognition function found in the Image Processing Matlab Toolbox.

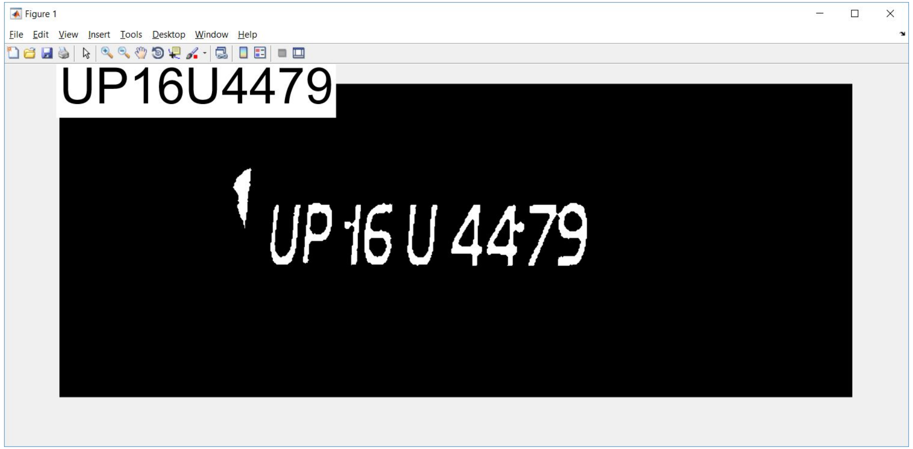
This Recognition has been improved from the default accuracy using the OCR trainer application present in the same toolbox which allows the user to train recognition based on some dataset character images and their corresponding character values.

Since the component 1 resulted in some more cropped images than just the license plates, the results after component 3 are compared and only the values containing more than 7 characters are returned.

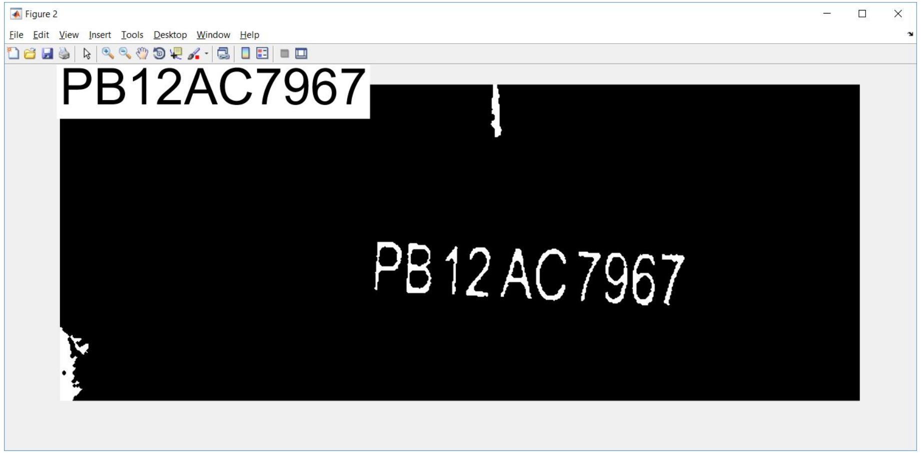
Component 3 Results



License plate detected characters displayed in upper left corner

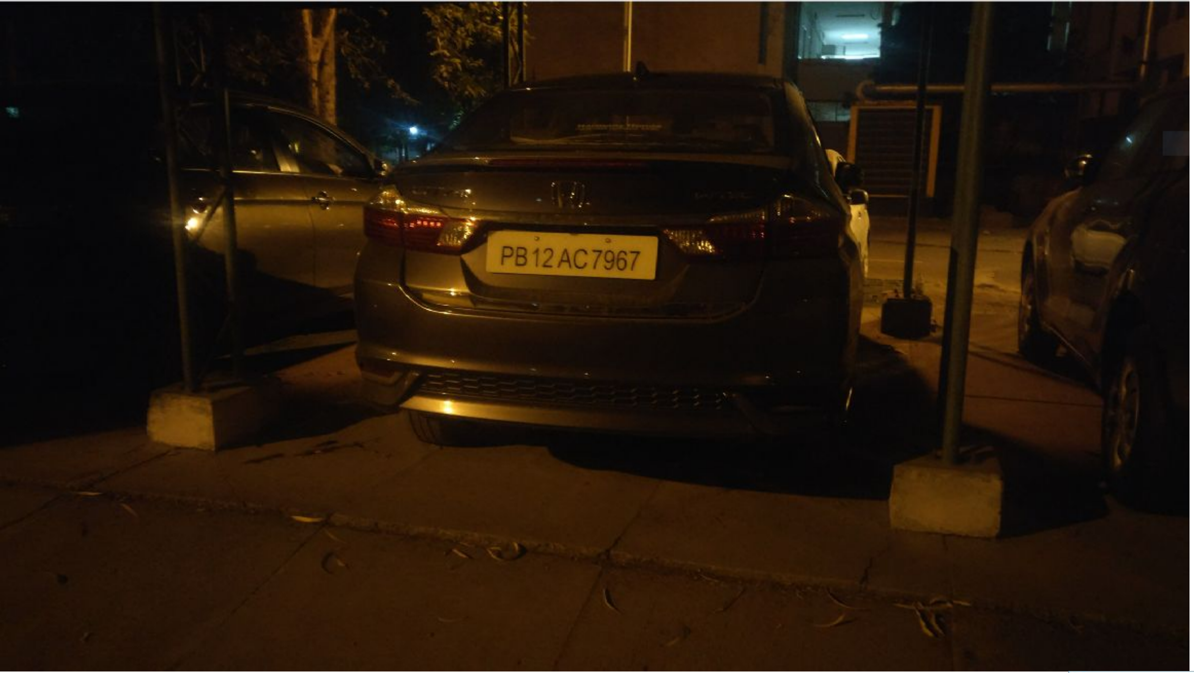


UP16U4479



PB12AC7967

Overall Results


PB12AC7967

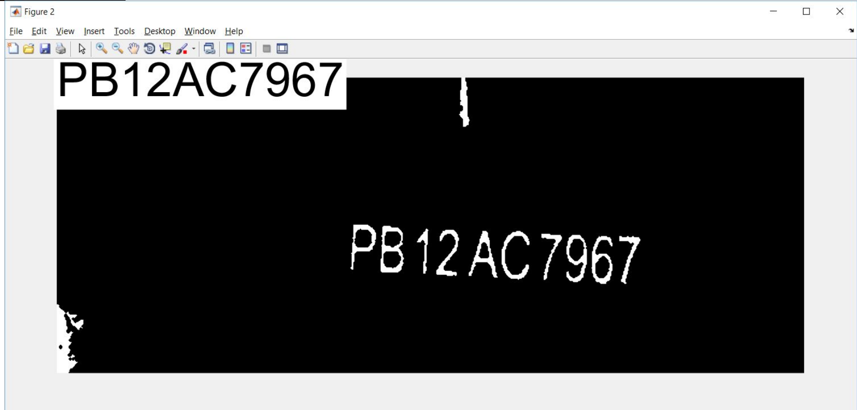
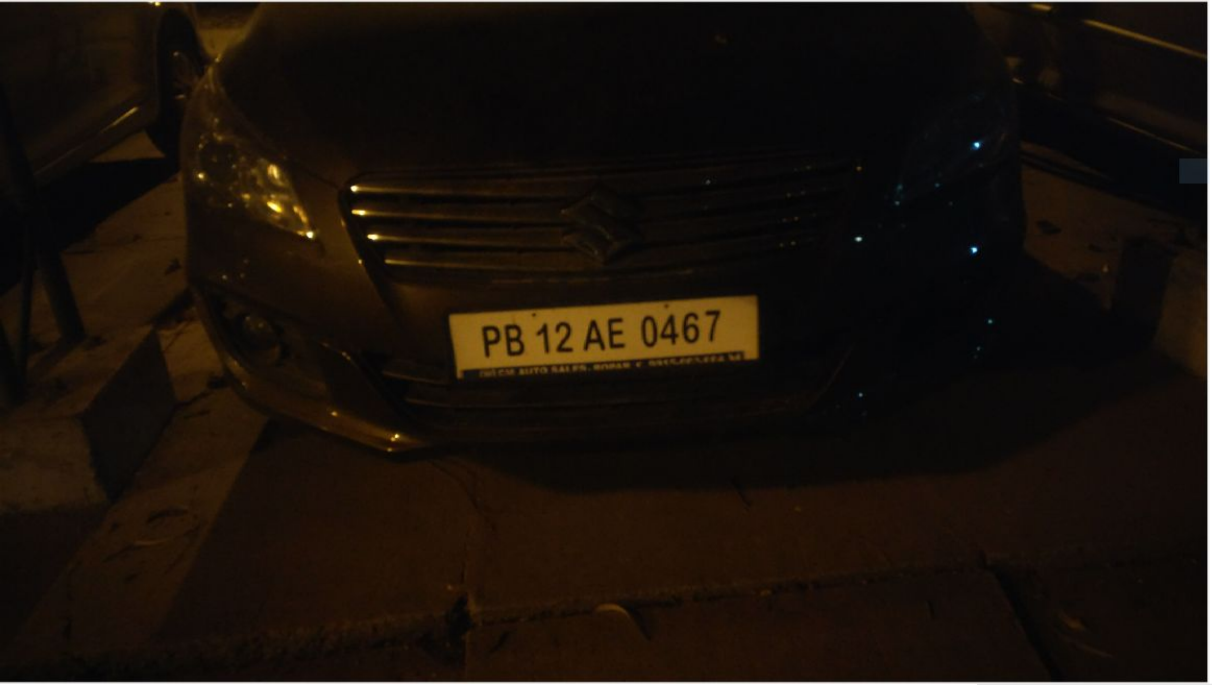
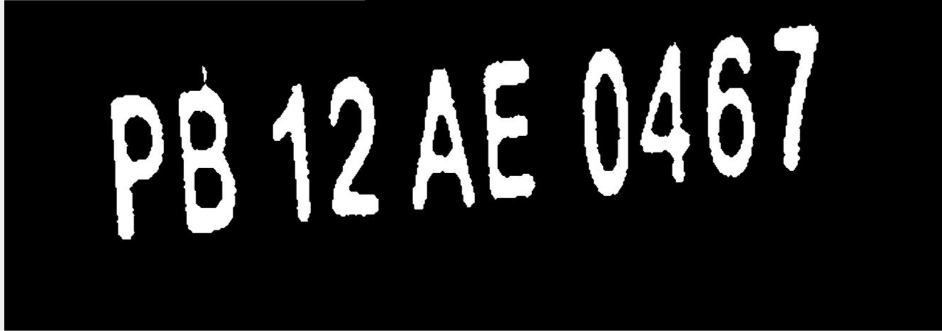


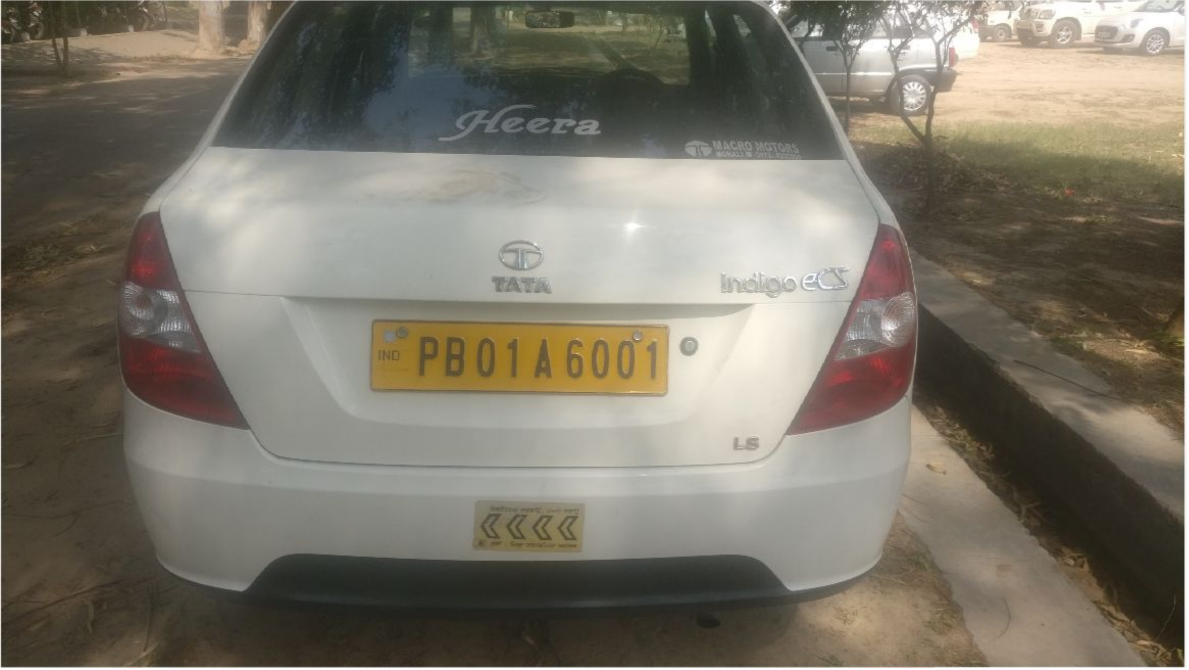
Figure 2
File
Edit
View
Insert
Tools
Desktop
Window
Help
PB 12 AC 7967

PB12AC7967

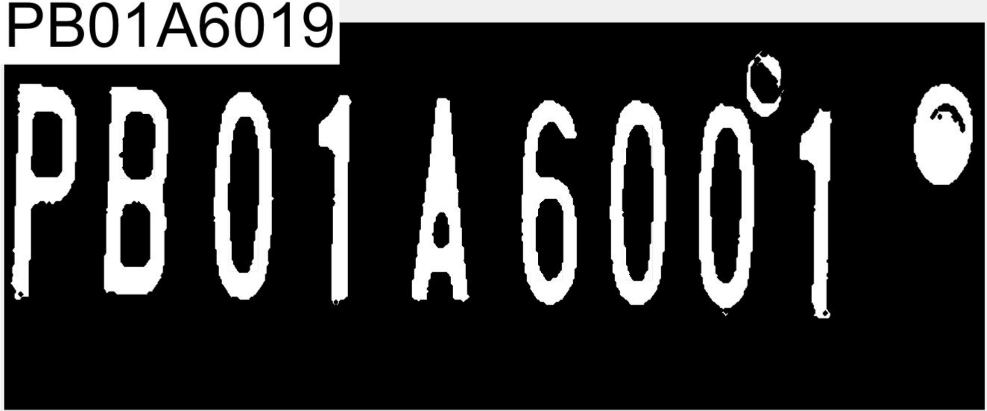


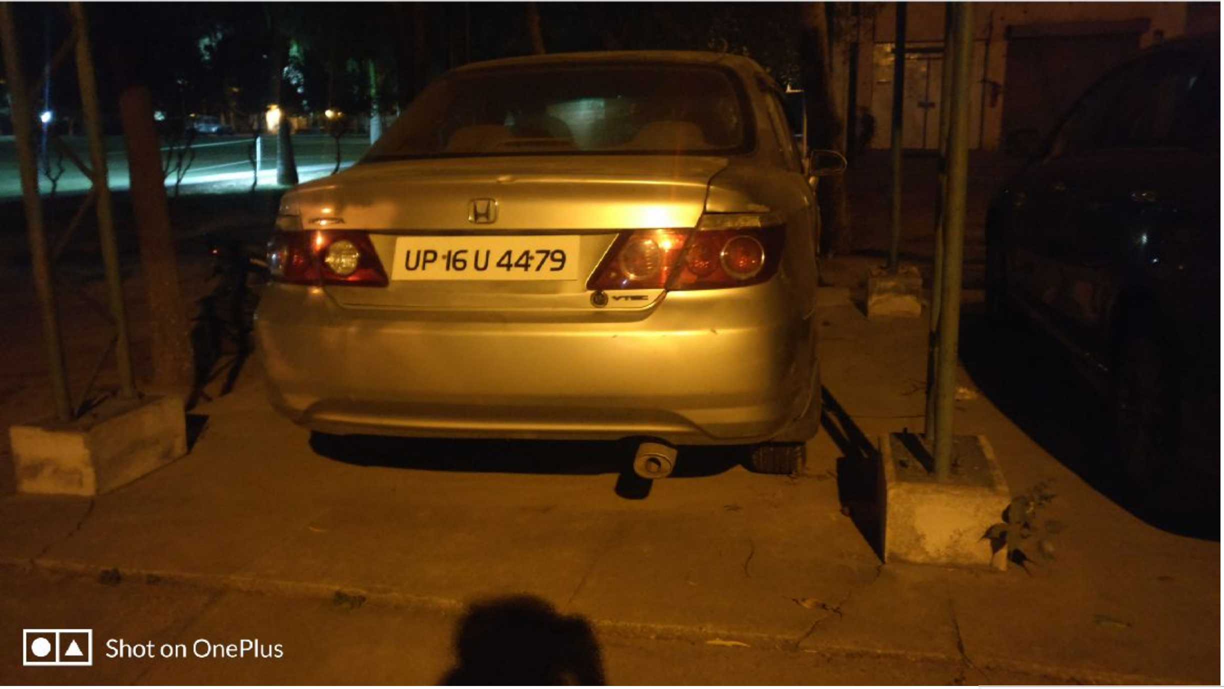
PB12AE0467

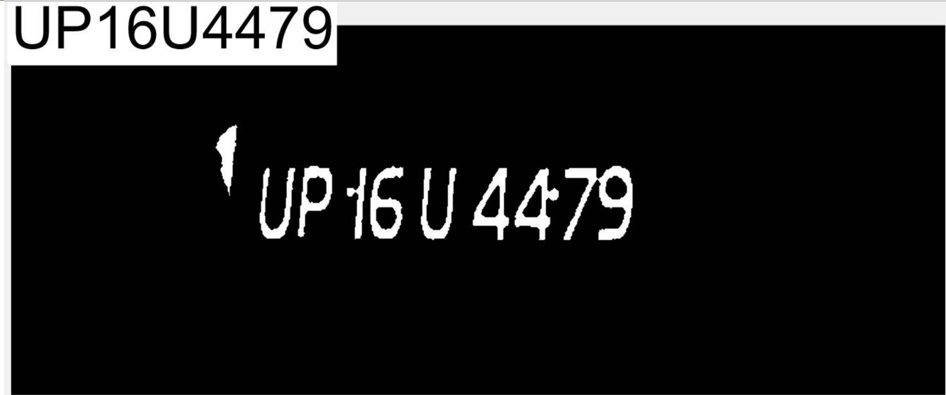


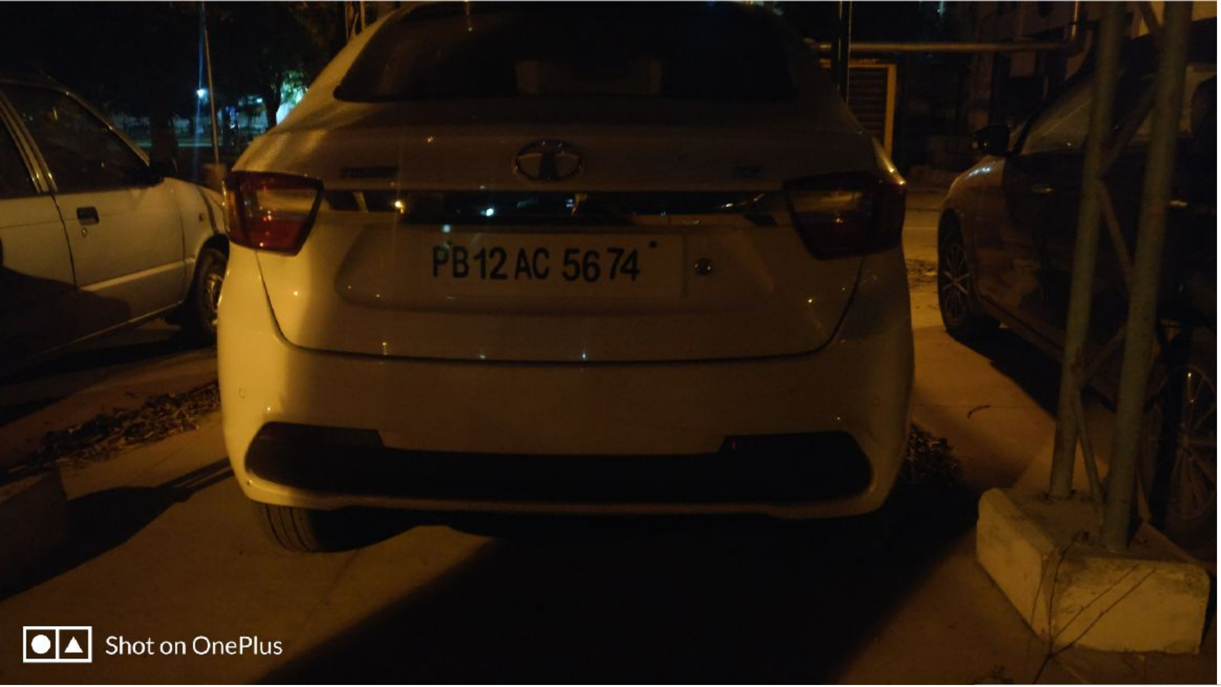

Heera
MACRO MOTORS
TATA
Indigo eCS
IND
PB01A6001
LS

PB01A6019

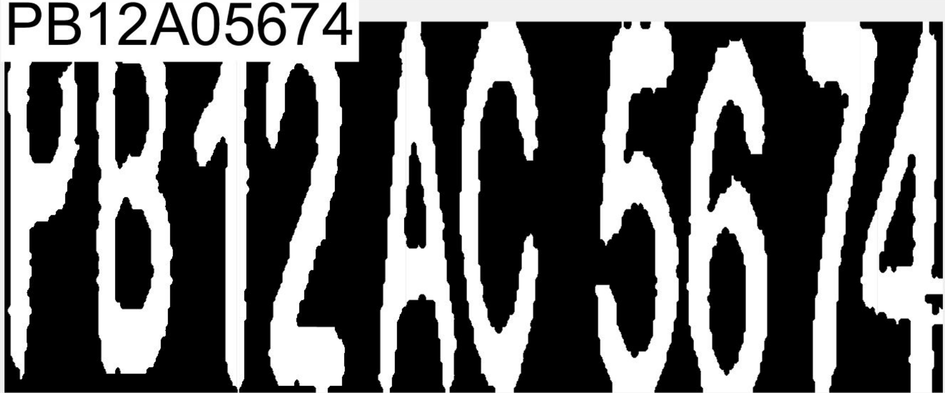

PB01A6001

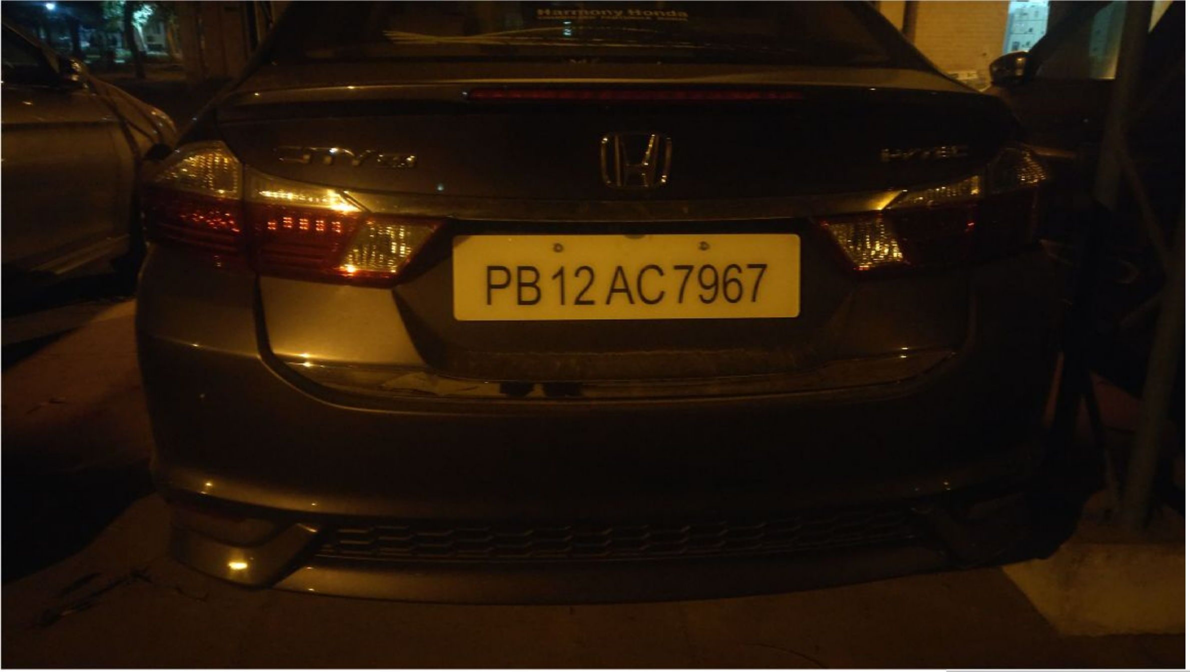


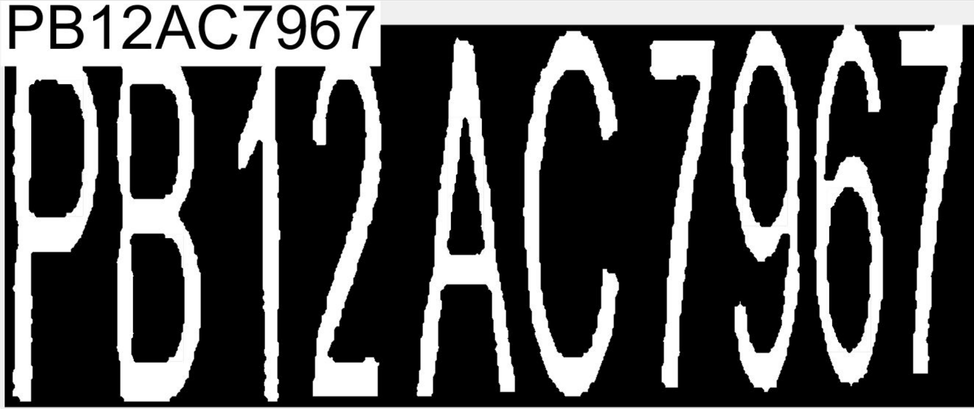

UP16U4479



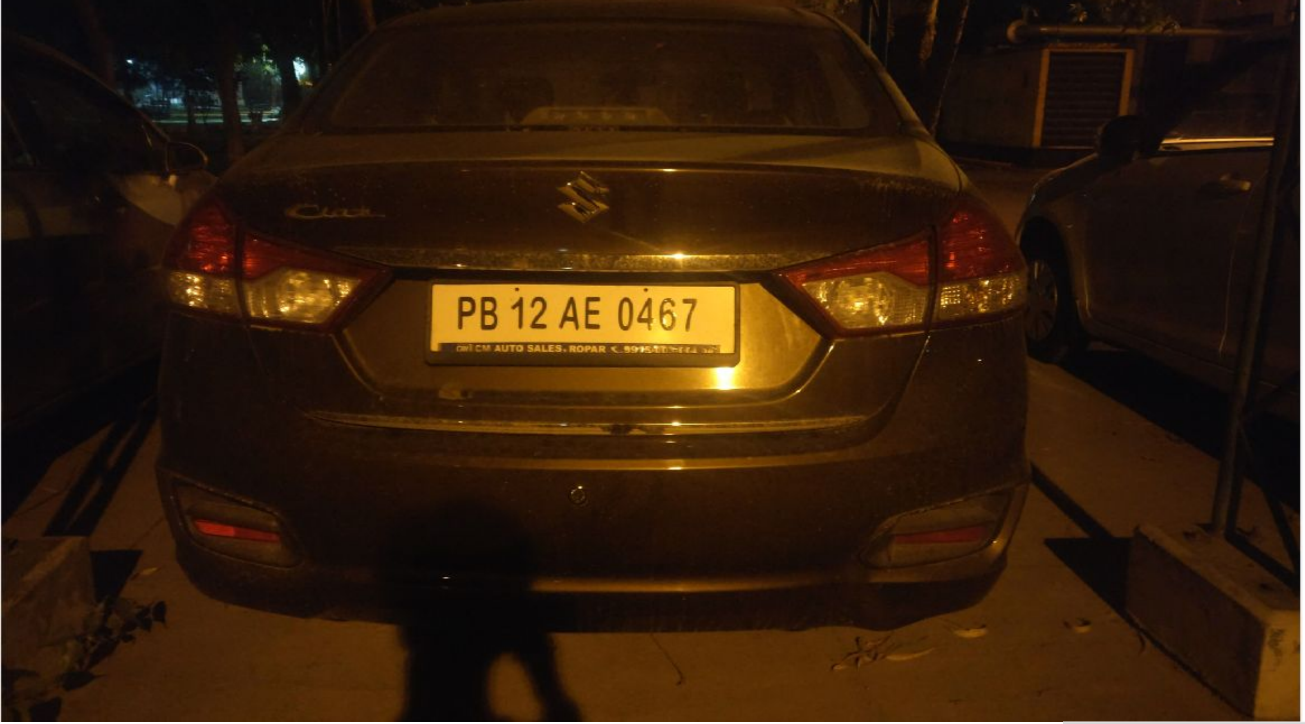
PB12A05674



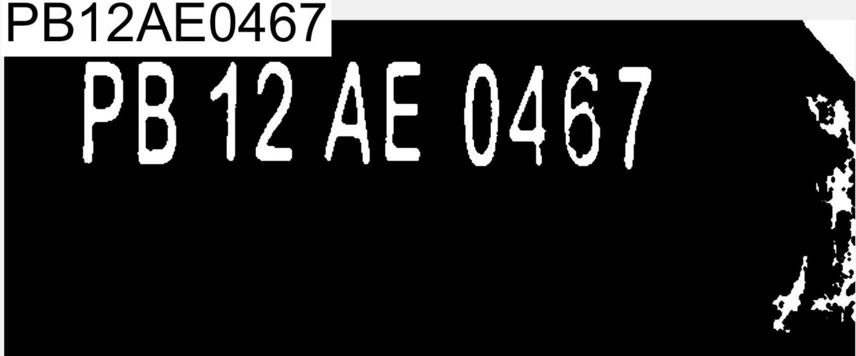




PB12AC7967



PB12AE0467



Accuracy Achieved

Accuracy

With the program tested on 30 images under different illumination conditions and at different angles, the accuracy results obtained are:-

1. For **component 1**, 28/30 plates extracted perfectly. Thus, accuracy = **93.33%**.
2. For **component 2**, 2 individual characters were lost only in 2 images (out of 280 total characters), thus accuracy = 276/280 or accuracy = **98.57%**

Accuracy (continued)

3. **Component 3**, the in-built OCR, produced comparatively the most inaccurate results with the following break-down in the 27 images obtained:-

- 14 plates with each character recognized correctly.
- 5 plates with one character wrongly recognized.
- 5 plates with two characters wrongly recognized.
- 3 plates with three characters wrongly recognized.
- 1 plate with 4 characters wrongly recognized.

Thus accuracy in this component= 248/276 or **89.85%**

Accuracy (continued)

Overall Accuracy obtained= 93.33% x 98.57% x 89.85% = **82.65%.**

Since we were largely unfamiliar with the machine learning component involved in OCR, improving the component-3 accuracy would have increases the total accuracy much as the accuracy of component 1 and component 2 together is **92%**.

Constraints

The accuracy of ALPR system depends on certain factors like the image resolution, image orientation, illumination conditions and the image quality.

The images used to test the above algorithm were taken by a specific 16MP camera and most of the images were captured under appropriate illumination and other conditions. Since the parameter values have to be appropriately defined, the ALPR works best only on images taken from the same camera and for different cameras, the parameters will have to be changed.

Other Algorithm Experimented

Plate detection using YOLO v2

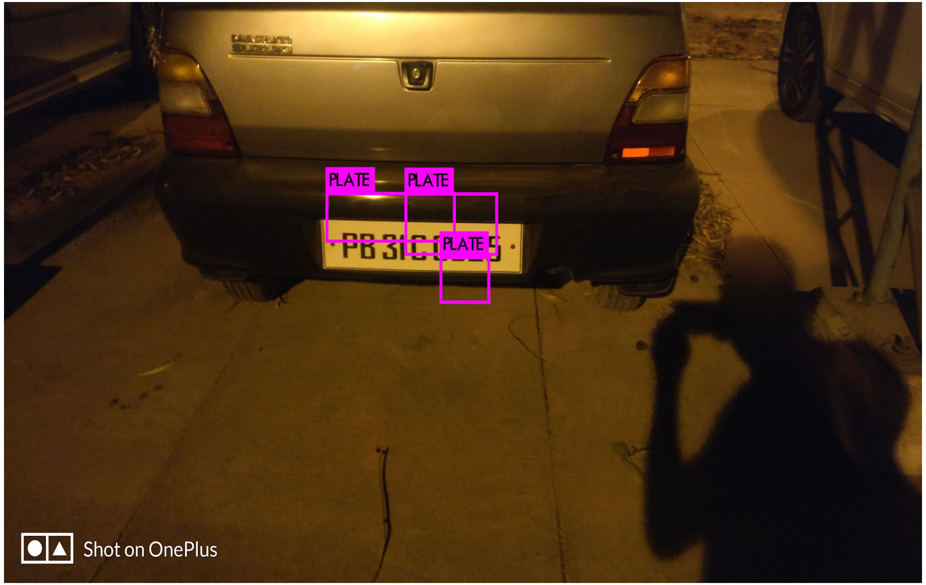
Reference: https://medium.com/@ravidesetty/number-plate-detection-on-indian-car-vehicles-using-yolov2-8c99e1a259f5

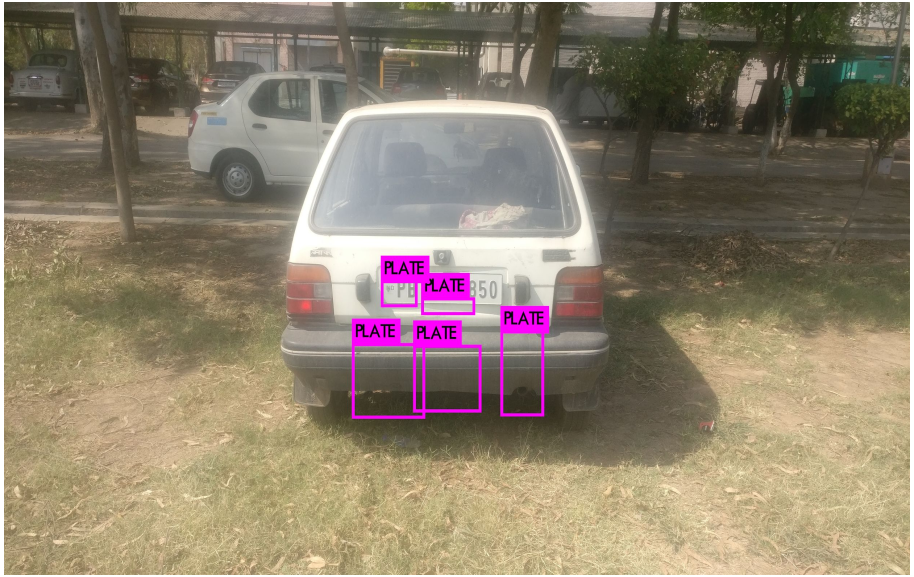
YOLO v2, an open-source object detection software employs CNN to detect certain objects in images according to the trained weights.

We tried using this to localize the license plates instead of the image processing operations we finally implemented.

After much time-consuming, Neural Network training done on 300 images with the license plate coordinates specified for each image, the YOLOv2 was tried upon sample images but the results obtained were much poorer than the algorithm we finally used.

YOLO v2 Results Obtained





References Used

1) https://ieeexplore.ieee.org/document/6213519/?part=1

2) https://waset.org/publications/10003561/indian-license-plate-detection-and-recognition-using-morphological-operation-and-template-matching

3) https://stackoverflow.com/questions/981378/how-to-recognize-vehicle-license-number-plate-anpr-from-an-image/37523538#37523538