MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
A Constituent Institution of Manipal University

Car Number Plate Detection and Owner Details Retrieval Using MATLAB

SHREYASI PRADHAN
Department of Electrical &Electronics Engineering

***Abstract*—The goal of the project is to identify a car's license plate from a photograph of the vehicle or while it is stationary. The goal is to identify the vehicle's owner by using MATLAB software to identify specific characters on the license plate, which can then be compared to the licensing database. Nevertheless, the research is restricted to image processing and number plate character recognition.**

I. Introduction

In every nation, license plates are used to identify cars and their owners. There are two types of number plate identification systems: automated and manual (operator-controlled). License plate recognition and identification systems are used for traffic management and security functions like vehicle tracking and access control to restricted locations. Depending on their requirements, each nation has a unique system for identifying license plates. Because there is no uniform regulation governing the aspect ratio and characters of Indian license plates, for instance, they are challenging to identify. Generally speaking, the identification task is difficult for a number of reasons, including a broken number plate, colour, inadequate upkeep, writing in a foreign language, etc. To find and identify the number plate in an image, the number plate detection method scans the input image. A plate can appear anywhere in an image of different sizes, therefore it is impossible to find it by looking at every pixel in the image. When a car reaches a gate or a toll stop, its number plate is instantly recognised and identified by comparing it with a database. The colour of the characters and background on the number plate make it challenging to distinguish the number plate's border from an image of the car in an outdoor setting. Some techniques combine segmentation, Canny edge detection, and morphological operation. Edge detection, morphological procedures, noise filtration, character segmentation, and template matching are some of the steps involved in finding the license plate.

II. Objective & Application

A. Objective

The characters from the license plate should be recognisable and visible at the project's conclusion. The objective is to keep the procedure as straightforward as feasible, and the recognised characters should be accurate.

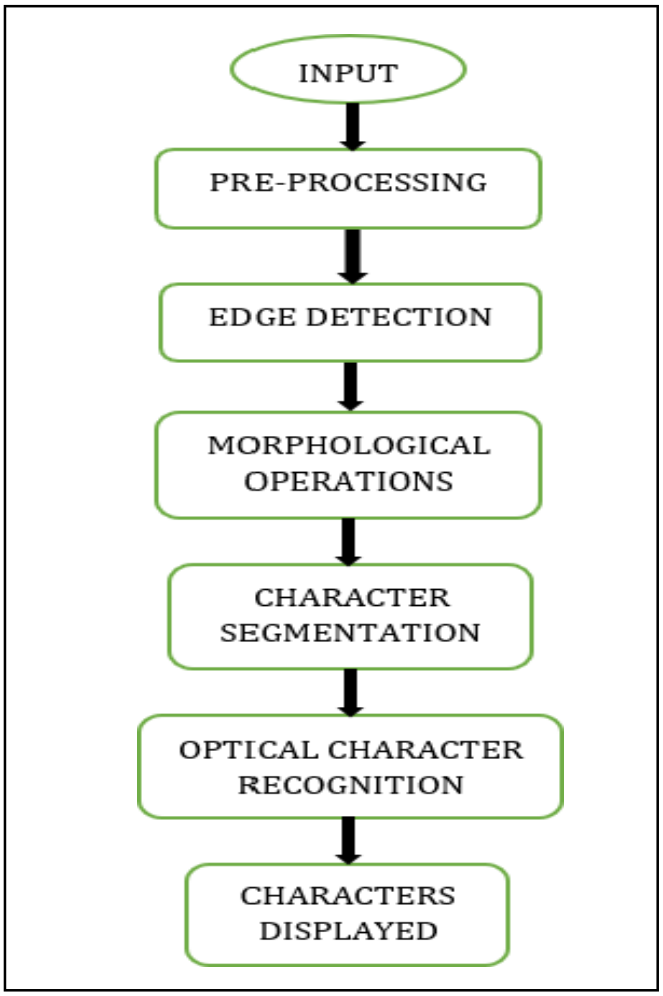
B. Application

- Parking: To allow people whose license plates are recognised to park
- Safety of the Community: Only cars with registered license plates are permitted inside.
- Stops at Tolls: Stops fraud and theft
- Border Security: When a car enters or exits a nation, it is registered.
- Traffic Management: Traffic is managed more effectively.

III. Methodology

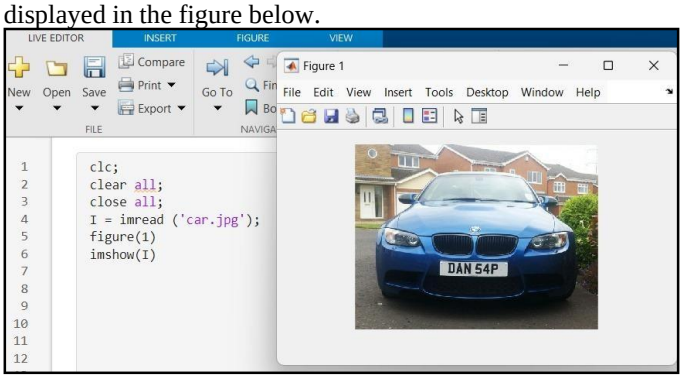
A. Background

MATLAB software is used for image processing and character recognition. A step-by-step process is followed to get the characters of the number plate recognized and displayed. Processes for image enhancement will be done before character segmentation and recognition. Figure below shows how the process steps will be executed.



B. Image Input

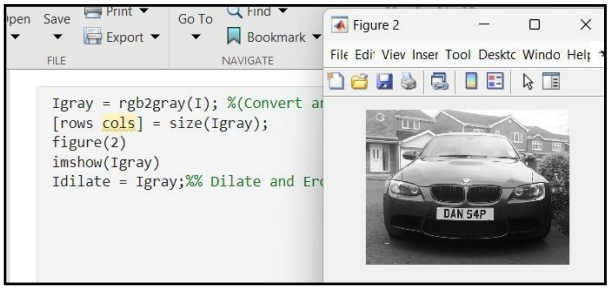
The picture of the license plate that serves as the project's sample was obtained online. Using ‘imread’ the image must first be written into MATLAB before it can be displayed, that is, using ‘imshow’. The code and the outcome are displayed in the figure below.



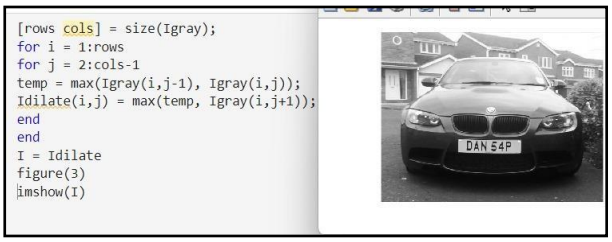
C. Pre-processing

In Pre-processing, we use various methods to improve the image. Filter out noise, improve contrast, convert the image to a more computable form.

- **Converting from RGB to Grayscale:** We can improve the image's features and lessen the brightness effect by using greyscale conversion, which makes the image's darker values stand out more, using the function ‘rgb2gray’. To remove the noise we have used ‘Idilate = Igray’ The code and the converted grayscale image are displayed in the figure given below.

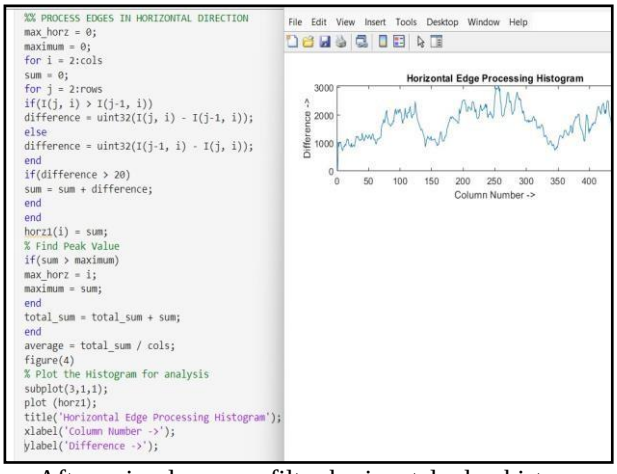


- **Edge Detection:** The Edge Detection technique preserves the image's characteristics for further processing stages while removing a significant quantity of data from the image. to identify and pinpoint abrupt changes, highlighting areas of interest like text or visual patterns.

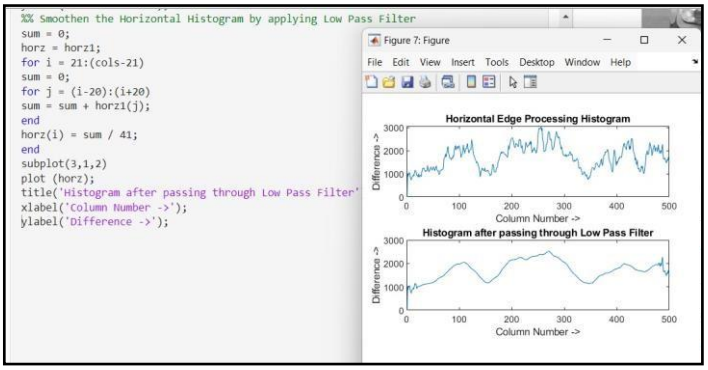


(a) Horizontal Edge Detection:

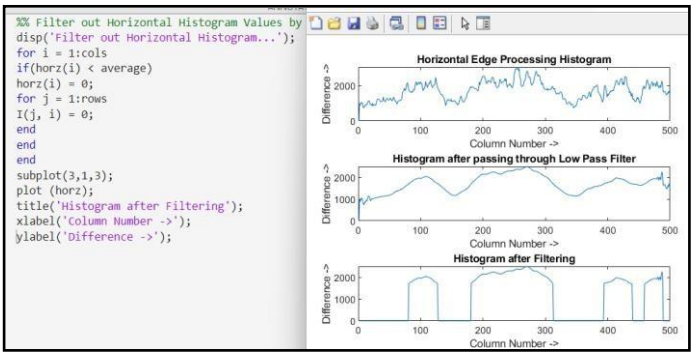
The below code performs horizontal edge detection across columns and identifies the column with the strongest horizontal edge activity, which is likely where a number plate or another high-contrast feature is located.



After using low pass filter horizontal edge histogram is smoothened out. The code and figure is given below.

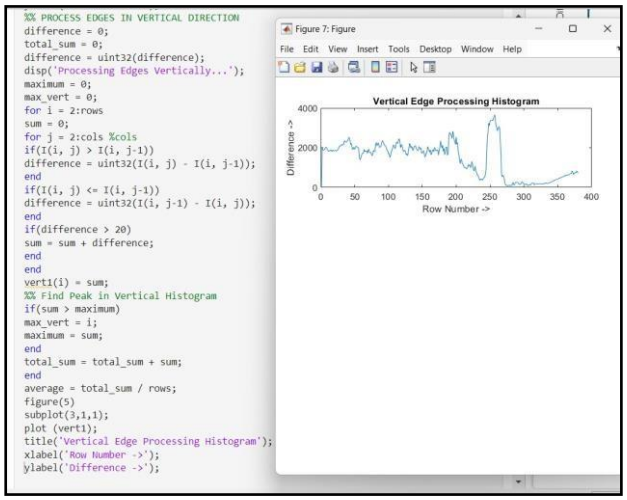


Next we filter out horizontal histogram values by applying dynamic threshold. The code and figure are shown.



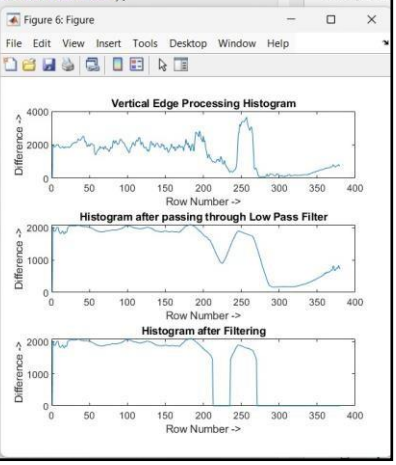
(b) Vertical Edge Detection:

The below code performs vertical edge detection across columns and identifies the column with the strongest horizontal edge activity, which is likely where a number plate or another high-contrast feature is located.

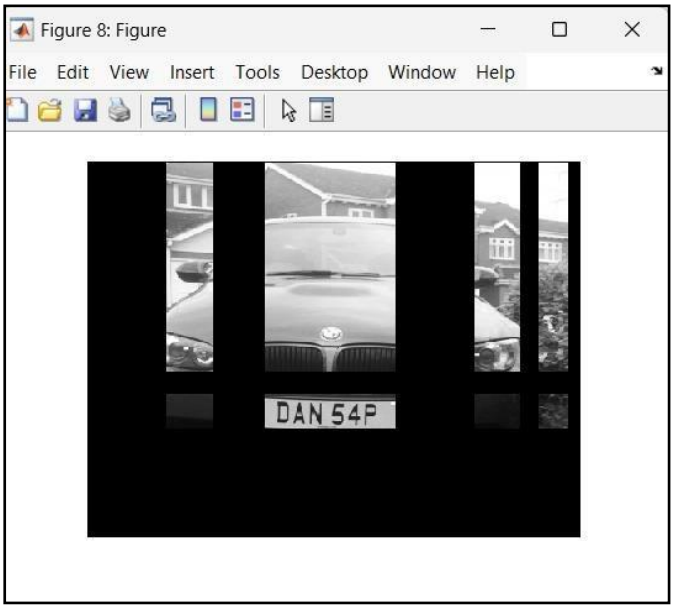


After using low pass filter vertical edge histogram is smoothened out. Next we filter out vertical histogram values by applying dynamic threshold. The code and figure are shown.

```
%% Smoothen the Vertical Histogram by applying Low Pass Filter
disp('Passing Vertical Histogram through Low Pass Filter...');
sum = 0;
vert = vert1;
for i = 21:(rows-21)
sum = 0;
for j = (i-20):(i+20)
sum = sum + vert1(j);
end
vert(i) = sum / 41;
end
subplot(3,1,2);
plot (vert);
title('Histogram after passing through Low
xlabel('Row Number ->');
ylabel('Difference ->');
%% Filter out Vertical Histogram Values by
disp('Filter out Vertical Histogram...');
for i = 1:rows
if(vert(i) < average)
vert(i) = 0;
for j = 1:cols
I(i, j) = 0;
end
end
end
subplot(3,1,3);
plot (vert);
title('Histogram after Filtering');
xlabel('Row Number ->');
ylabel('Difference ->');
```



After edge- detection is completed, this is how the processed image of the car would look.

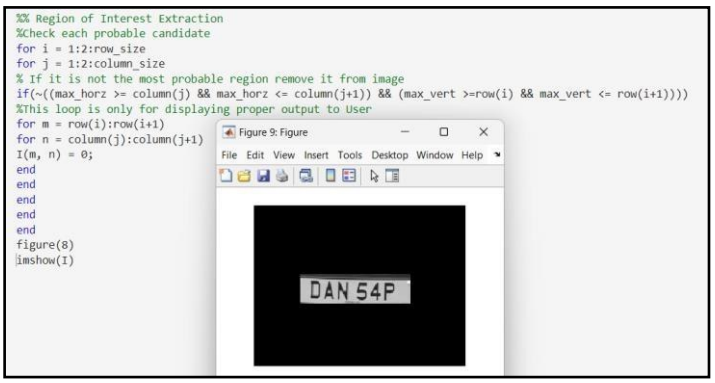


- **Character Segmentation:** This code segment identifies character regions within an image where the license plate may be located by detecting transitions or isolated peaks in horizontal and vertical projections, then storing these boundary indices in column and row.

```
%% Find Probable candidates for Number Plate
j = 1;
for i = 2:cols-2
if(horz(i) ~= 0 && horz(i-1) == 0 && horz(i+1) == 0)
column(j) = i;
column(j+1) = i;
j = j + 2;
elseif((horz(i) ~= 0 && horz(i-1) == 0) || (horz(i) ~= 0 && horz(i+1) == 0))
column(j) = i;
j = j+1;
end
end
j = 1;
for i = 2:rows-2
if(vert(i) ~= 0 && vert(i-1) == 0 && vert(i+1) == 0)
row(j) = i;
row(j+1) = i;
j = j + 2;
elseif((vert(i) ~= 0 && vert(i-1) == 0) || (vert(i) ~= 0 && vert(i+1) == 0))
row(j) = i;
j = j+1;
end
end
[temp column_size] = size (column);
if(mod(column_size, 2))
column(column_size+1) = cols;
end
[temp row_size] = size (row);
if(mod(row_size, 2))
row(row_size+1) = rows;
end
```

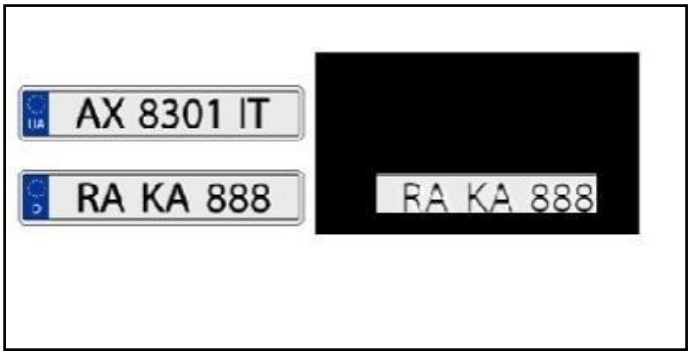
Ensuring the boundary arrays have even numbers of entries, which would be useful for marking complete regions of interest (ROIs) in subsequent processing steps.

```
%% Region of Interest Extraction
%Check each probable candidate
for i = 1:2:row_size
for j = 1:2:column_size
% If it is not the most probable region remove it from image
if(~((max_horz >= column(j) && max_horz <= column(j+1)) && (max_vert >=row(i) && max_vert <= row(i+1))))
%This loop is only for displaying proper output to User
for m = row(i):row(i+1)
for n = column(j):column(j+1)
I(m, n) = 0;
end
end
end
end
end
figure(8)
imshow(I)
```



IV. Experimentation & Results

We also experimented on our system, with multiple types of car number plates as image input, in order to check its capability to cope with practical examples and determine our system's limitations. Below are the results with image input on the left and image output on the right –



From the above experimentation we can determine the following limitations of our system -

- It is impossible to identify a cracked license plate.
- Number plates with different symbols or in various languages or in abnormal fonts cannot be identified.
- Character similarities may result from poorly maintained license plates. For instance, 0 & D, 5 & 8.
- Hazy or low-resolution license plate.
- When two number plates are given in one image, the one with better resolution is detected.
- If the number plate is at any angle other than 180° it cannot be detected.

V. Conclusion

Tracking or identifying a car by its number plate can be made much easier with the use of a vehicle number plate recognition system. For a fixed visual style, the system functions well. There is a great deal of room for refinement and precision.

References

[1] Ragini Bhat, Bijender Mehandia, “Recognition of Vehicle Number Plate using MATLAB”, IJIREEICE, Vol. 2, Issue 8, August 2014

[2] https://www.ripublication.com/ijcir17/ijcirv13n9_10.pdf

[3] https://www.irjet.net/archives/V3/i5/IRJET-V3I5177.pdf

[4] Automatic Vehicle Number Plate Recognition System using MATLAB, IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834, p- ISSN: 2278-8735.Volume 11, Issue 4, Ver. II (Jul.-Aug .2016)

[5] Number plate recognition using template comparison for various fonts in MATLAB, DOI: 10.1109/ICCIC.2016.7919542

[6] MATLAB Based Vehicle Number Plate Recognition, International Journal of Computational Intelligence Research ISSN 0973-1873 Volume 13, Number 9 (2017)

[7] T.Pratheeba, “Morphology Based Text Detection and Extraction from Complex Video Scene," International Journal of Engineering and Technology Vol.2 (3), 200-206, 2010.

[8] Saeed Rastegar, Reza Ghaderi, Gholamreza Ardeshipr & Nima Asadi, " An intelligent control system using an efficient License Plate Location and Recognition Approach", International Journal of Image Processing (IJIP) Volume(3), Issue(5) 252, 2009.

[9] Wisam Al Faqheri and Syamsiah Mashohor, "A Real-Time Malaysian Automatic License Plate Recognition (M-ALPR) using Hybrid Fuzzy" ,IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.2, February 2009.

[10] Satadal Saha1, Subhadip Basu, Mita Nasipuri, Dipak Kumar Basu,” License Plate Localization from Vehicle Images:An Edge Based Multistage Approach”, International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.

[11] Loumos, V.; Kayafas, E.,”License plate recognition from still images and video sequences: A survey” IEEE Transactions on Intelligent Transportation Systems, volume 9, issue 3, pages 377-391, September 2008.

[12] Ganapathy and W.L.D. Lui, "A Malaysian Vehicle License Plate Localization and Recognition System", Journal of Systemic, Cybernetics and Informatics, Vol. 6, No. 1, 2008.

[13] Roushdy M., "Comparative Study of Edge detection Algorithms Applying on the Grayscale Noisy Image Using Morphological filter", ICGST, International Journal of Graphics, Vision, and Image Processing GVIP, Vol. 6, Issue 4, pp. 17-23, , Dec. 2006.

[14] Chirag N. Paunwala, Suprava Patnaik, “A Novel Multiple License Plate Extraction Technique for Complex Background in Indian Traffic Conditions”, In Proceedings of International Journal of Image Processing, vol.4**,** issue2, 2007

[15] D. Zheng, Y. Zhao, and J. Wang, “An efficient method of License Plate location,” Pattern Recognit. Lett. vol. 26, no. 15, pp. 2431–2438, 2005

[16] Feng Yang and Zheng Ma. “Vehicle License Plate Location Based on Histogramming and Mathematical Morphology”, Automatic Identification Advanced Technologies, pp: 89 – 94, 2005.