Number Plate Recognition System

INTRODUCTION

The system developed here is a technology used to extract the license number of vehicles from images. It is an embedded system which has numerous applications and challenges. The sole intention of this project is to find the most efficient way to recognize the registration information from the digital image (obtained from the camera). This process usually comprises of three steps. First step is the license plate localisation, regardless of the license plate size and orientation. The second step is the segmentation of the characters and the last step is the recognition of the characters from the license plate. It also helps us in noting the date and time the vehicle arrived. Thus this project uncovers the fundamental idea of various algorithms required to recognise the characters from the license plate using template matching.

SOFTWARE USED

We have used Image Processing Toolbox with MATLAB to develop this system. The Image Processing Toolbox provides a comprehensive set of reference-standard algorithms and workflow apps for image processing, analysis, visualization, and algorithm development. We can perform image segmentation, image enhancement, noise reduction, geometric transformations, image registration, and 3D image processing using this toolbox.

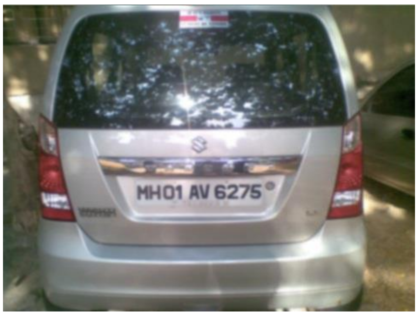
PROPOSED METHODOLOGY

In India, basically there are two kinds of license plates, black characters in white plate and black characters in yellow plate. The process of the proposed system is as follows.

LOAD RGB IMAGE
↓
GRAYSCALE CONVERSION
↓
BINARIZATION
↓
DILATION
↓
EDGE DETECTION
↓
PLATE REGION EXTRACTION
↓
CHARACTER SEGMENTATION
↓
TEMPLATE MATCHING
↓
OUTPUT STORED IN FILE

1. LOADING RGB IMAGE

The image whose number plate is to be recognised is loaded using the imread() function.

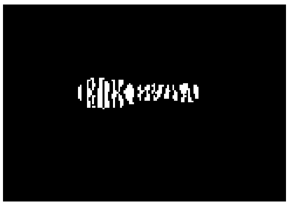


2. GRAYSCALE CONVERSION

This RGB image is converted to grayscale image using rgb2gray() function.

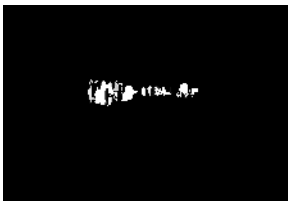
3. BINARIZATION

This image is then converted to binary using the imbinarize() function.



4. DILATION

The method by which image is processed on the basis of shapes is known as dilation. The image is dilated using imdilate() function.

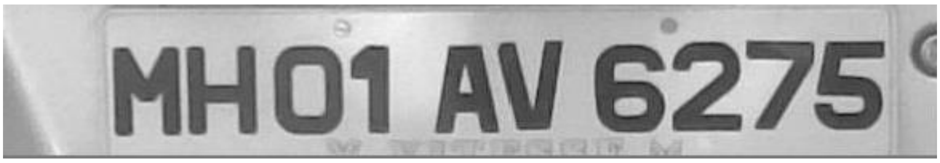


5. EDGE DETECTION

In the process of edge detection, the dilated image is subtracted from the original image to detect the corners and edges of the image. This process is accomplished using the imfill() function.

6. PLATE REGION EXTRACTION

Plate region is found out by passing a rectangular image over the previous using the regionprops() function.



7. CHARACTER SEGMENTATION

Characters are segmented from the number plate image which is then used for template matching.

Fig: Segmented character

8. TEMPLATE MATCHING

Segmented characters are template matched with templates of each character and the number plate is identified as string.

9. STORING OUTPUT AS A FILE

After matching with the templates, the output is stored as a text file along with the date and time. This will help to in making the security more efficient.

- DATE AND TIME – The date and time of the same is collected using the datetime function. It is stored as a string and it is copied to the same file where the number plate data is stored.
- STORING MULTIPLE NUMBER PLATES – The number plates and their respective times and dates are stored in the text file using the fprintf() function. Successive number plates and their respective times and dates are stored using the append ‘a’ operation of the fopen() function.

ABOUT MATLAB

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems.

As of 2018, MATLAB has more than 3 million users worldwide. MATLAB users come from various backgrounds of engineering, science, and economics.

ASSUMPTIONS AND LIMITATIONS

1. ASSUMPTIONS

- Input is an image of a stationary Car.
- Only the most common type of license plates (single line) will be dealt with for best results.
- The license plate has a yellow or white background with text written in black.

2. LIMITATIONS

- If the image contains too much spoiled license plate or has designs on it, the program can fail to localize the license plate.
- If the license plate happens to be much tilted from horizontal, then again the result of segmentation of the license plate is very poor.

THE CODING

```
close all;
clear all;

im = imread('car3.jpg');
im = imresize(im, [480 NaN]);
imgray = rgb2gray(im);
imbin = imbinarize(imgray);
im = edge(imgray, 'sobel');
im = imdilate(im, strel('diamond', 2));
im = imfill(im, 'holes');
im = imerode(im, strel('diamond', 10));
Iprops=regionprops(im,'BoundingBox','Area', 'Image');

area = Iprops.Area;
count = numel(Iprops);
maxa= area;
boundingBox = Iprops.BoundingBox;

for i=1:count
 if maxa<Iprops(i).Area
       maxa=Iprops(i).Area;
       boundingBox=Iprops(i).BoundingBox;
      end
    end

%all above step are to find location of number plate

im = imcrop(imbin, boundingBox);

%resize number plate to 240 NaN
im = imresize(im, [240 NaN]);

%clear dust
im = imopen(im, strel('rectangle', [4 4]));

%remove some object if its width is too long or too small than 500
im = bwareaopen(~im, 500);

%%%get width
[h, w] = size(im);
% Iprops=regionprops(im,'BoundingBox','Area', 'Image');
% image = Iprops.Image;
% count = numel(Iprops);
% for i=1:count
% ow = length(Iprops(i).Image(1,:));
% if ow<(h/2) % im = im .* ~Iprops(i).Image;
% end
% end

imshow(im);

%read letter
Iprops=regionprops(im,'BoundingBox','Area', 'Image');
count = numel(Iprops);
noPlate=[];
% Initializing the variable of number plate string.
for i=1:count
       ow = length(Iprops(i).Image(1,:));
       ow = length(Iprops(i).Image(:,1));
       if ow<(h/2) & oh>(h/3)
              letter=readLetter(Iprops(i).Image);% Reading the letter corresponding the binary image
       'N'.
              figure; imshow(Iprops(i).Image);
              noPlate=[noPlate letter]; % Appending every subsequent character in noPlate variable.
       end
end

% Storing the date and time
t=datetime;
dst=datestr(t);
fileID=fopen('Plate Details2.txt','a');
fprintf(fileID,'%s',noPlate);
fprintf(fileID,'\t%s',dst);
fprintf(fileID,'\r\n');
fclose(fileID);
```

```
% Creating the templates
%Letter
A=imread('char/A.bmp');B=imread('char/B.bmp');
C=imread('char/C.bmp');D=imread('char/D.bmp');
E=imread('char/E.bmp');F=imread('char/F.bmp');
G=imread('char/G.bmp');H=imread('char/H.bmp');
I=imread('char/I.bmp');J=imread('char/J.bmp');
K=imread('char/K.bmp');L=imread('char/L.bmp');
M=imread('char/M.bmp');N=imread('char/N.bmp');
O=imread('char/O.bmp');P=imread('char/P.bmp');
Q=imread('char/Q.bmp');R=imread('char/R.bmp');
S=imread('char/S.bmp');T=imread('char/T.bmp');
U=imread('char/U.bmp');V=imread('char/V.bmp');
W=imread('char/W.bmp');X=imread('char/X.bmp');
Y=imread('char/Y.bmp');Z=imread('char/Z.bmp');
Afill=imread('char/fillA.bmp');
Bfill=imread('char/fillB.bmp');
Dfill=imread('char/fillD.bmp');
Ofill=imread('char/fillO.bmp');
Pfill=imread('char/fillP.bmp');
Qfill=imread('char/fillQ.bmp');
Rfill=imread('char/fillR.bmp');




%Number
one=imread('char/1.bmp');  two=imread('char/2.bmp');
three=imread('char/3.bmp');four=imread('char/4.bmp');
five=imread('char/5.bmp'); six=imread('char/6.bmp');
seven=imread('char/7.bmp');eight=imread('char/8.bmp');
nine=imread('char/9.bmp'); zero=imread('char/0.bmp');
zerofill=imread('char/fill0.bmp');
fourfill=imread('char/fill4.bmp');
sixfill=imread('char/fill6.bmp');
sixfill2=imread('char/fill6_2.bmp');
eightfill=imread('char/fill8.bmp');
ninefill=imread('char/fill9.bmp');
ninefill2=imread('char/fill9_2.bmp');

function letter=readLetter(snap)
%READLETTER reads the character fromthe character's binary image.
%   LETTER=READLETTER(SNAP) outputs the character in class 'char' from the
%   input binary image SNAP.

load NewTemplates % Loads the templates of characters in the memory.
snap=imresize(snap,[42 24]); % Resize the input image so it can be compared with the template's
images.
comp=[ ];
for n=1:length(NewTemplates)
    sem=corr2(NewTemplates{1,n},snap); % Correlation the input image with every image in the template
for best matching.
    comp=[comp sem]; % Record the value of correlation for each template's character.
    %display(sem);

end
vd=find(comp==max(comp)); % Find the index which correspond to the highest matched character.
%display(max(comp));
%*-*-*-*-*-*-*-*-*-*-*-*-*-*-
% Accodrding to the index assign to 'letter'.
% Alphabets listings.
if vd==1 || vd==2
    letter='A';
elseif vd==3 || vd==4
    letter='B';
elseif vd==5
    letter='C';
elseif vd==6 || vd==7
    letter='D';
elseif vd==8
    letter='E';
elseif vd==9
    letter='F';
elseif vd==10
    letter='G';
elseif vd==11
    letter='H';
elseif vd==12
    letter='I';
```

```
elseif vd==13
    letter='J';
elseif vd==14
    letter='K';
elseif vd==15
    letter='L';
elseif vd==16
    letter='M';
elseif vd==17
    letter='N';
elseif vd==18 || vd==19
    letter='O';
elseif vd==20 || vd==21
    letter='P';
elseif vd==22 || vd==23
    letter='Q';
elseif vd==24 || vd==25
    letter='R';
elseif vd==26
    letter='S';
elseif vd==27
    letter='T';
elseif vd==28
    letter='U';
elseif vd==29
    letter='V';
elseif vd==30
    letter='W';
elseif vd==31
    letter='X';
elseif vd==32
    letter='Y';
elseif vd==33
    letter='Z';
    %*-*-*-*-*
% Numerals listings.
elseif vd==34
    letter='1';
elseif vd==35
    letter='2';
elseif vd==36
    letter='3';
elseif vd==37 || vd==38
    letter='4';
elseif vd==39
    letter='5';
elseif vd==40 || vd==41 || vd==42
    letter='6';
elseif vd==43
    letter='7';
elseif vd==44 || vd==45
    letter='8';
elseif vd==46 || vd==47 || vd==48
    letter='9';
else
    letter='0';
end
end
```

BIBLIOGRAPHY

Books:
International Journal of Computer Theory and Engineering, Vol. No. 2, 2 April, 2010.

Websites:
https://en.wikipedia.org/wiki/MATLAB
https://in.mathworks.com/products/matlab.html?s_tid=hp_products_matlab
(Matlab tutorial) https://