NAME - VIKAS SINGH NEGI and SHASHWAT VERMA

BRANCH - ECE (BOTH)

Digital Image Processing Project

Progress Report on Car Number Plate Detector

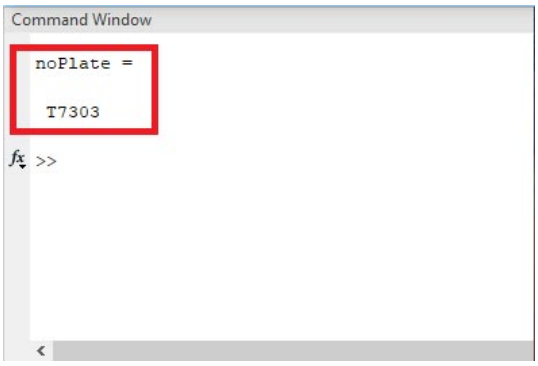
Overview/Problem statement - In real word, traffic is so much high that we cannot control and check all the vehicles passing by through high ways and roads. So, we need some advance system which can automatically record and check the passing vehicles and note down the number in digital form. So, it would help to maintain the perfect check on the vehicles passing by and the security would be easy and reliable.

Need - It's true that we can also take image by camera and store them, then what is the use of this project. Because images taken by camera would be of large size and storing thousands of image would waste lot of time and memory space. So we need such system which can reduce complexity. This project convert only no. plate area (i.e. useful information) into the digital text format, which takes size merely in kb. So, we can store lot of record in such small space and also it would be easy to match records of criminals using this digital information.

Technology/Methods used - We have used digital image processing for this purpose. First we have taken the image, then convert it into grayscale then binary and finally extracted the information. There are many image processing tools available for this Number plate detection, but here in this project we will use **MATLAB Image Processing to get the vehicle license plate number** into the text format.

Expected output –

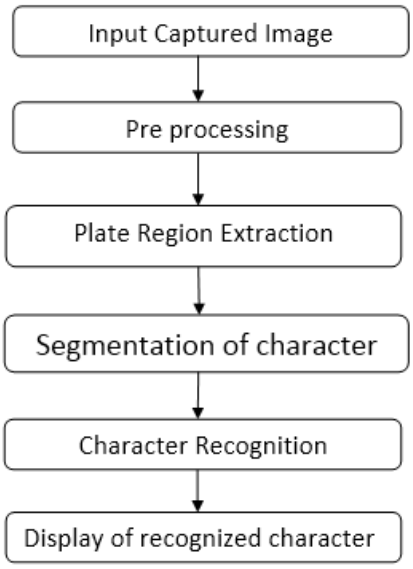
=>



METHODOLOGY -

The working of full NPR system can be divided in to two broad sections. The hardware part and the software part. The working mechanism of all the parts is described in details below.

Software Model: The first and the most important part in this process is the software model. The software model uses the image processing technology. The programs are implemented in MATLAB. The algorithm is divided into following parts: Capture image, Pre-processing, Plate region extraction, Segmentation of character in the extracted number plate, Character recognition, Comparison with database and Indicate result. The flow chart of license plate recognition system implementation in this work is shown in the following figure. There are various steps in this approach and these are implementation in MATLAB.



WORK FLOW PROCESS

Capture of Image: The first step is the capture of image. The image is captured by electronic device. Digital Camera or Webcam. The image captured is stored in JPEG format. Later on, it is converted in to Gray scale image in MATLAB.
Pre-processing: The next step after capturing the image is the pre-processing of the image. When the image is captured there is lot of disturbances and noises present in the image for which the image can't be used properly. So, in this step the noises from the image are required to be cleared to obtain an accurate result.

a. Gray Processing: this step involves the conversion of image in to Gray levels. Colour images are converted in to Gray image. According to the R, G, B value in the image, it calculates the value of Gray value, and obtains the Gray image at the same time.

b. Median Filtering: media filtering is the step to remove the noises from the image. Gray level cannot remove the noises. So, to make image free from noise media filtering is used.

Plate region extraction: The most important stage is the extraction of number plate from eroded image significantly. The extraction can be done by using image segmentation method. There are numerous image segmentation methods available in various literatures. In most of the methods image binarization is used.
Character segmentation: In this step get the o/p of extracted number plate using labelling components, and then separate each character and split the each and

every character in the number plate image by using split and also find the length of the number plate, then find the correlation and database if both the value is same means it will generate the value 0-9 and A - Z, and finally convert the value to string and display it in edit box, and also store the character in some text file in this code. Following figure shows the segmented characters.

The character recognition is now used to compare each individual character with the character stored in the database. OCR uses the correlation method to match the characters. And if both the character matches then it displays the authorized otherwise it will display the unauthorized.

(A) Hardware Model: The hardware model consists microcontroller for controlling the complete hardware of the ANPR system. The ANPR algorithm on a PC receives the image and performs the processing, which Yields the vehicle number. This Number is then compared to standard database and finally provides signal to microcontroller to control the system Hardware. If the inputted plate contains the authorized number then the green indication light will be switched on w, and if the inputted plate contains an unauthorized number then red indication will be switched-on.

Project details/ Procedure - Concept we are using for detecting number plates. There are three programs or '.m' files for this project.

- Template Creation (*template_creation.m*) – This is used to call the saved images of alphanumeric and then save them as a new template in MATLAB memory.
- Letter Detection (*Letter_detection.m*) – Reads the characters from the input image and find the highest matched corresponding alphanumeric.
- Plate Detection (*Plate_detection.m*) – Process the image and then call the above two m-files to detect the number.

Template Creation –

1. In the below code (see snapshot), we are saving the images into a variable by using command 'imread()'. This function is used to call the images from the folder or from any location of the PC into the MATLAB.
2. Then create a matrix of 'letter' and 'number' and save it in variable 'NewTemplates' by using command 'save(filename,variables)'.

Plate_detection.m | Letter_detection.m | template_creation.m

```
%CREATE TEMPLATES
%Alphabets
A=imread('alpha/A.bmp');B=imread('alpha/B.bmp');C=imread('alpha/C.bmp');
D=imread('alpha/D.bmp');E=imread('alpha/E.bmp');F=imread('alpha/F.bmp');
G=imread('alpha/G.bmp');H=imread('alpha/H.bmp');I=imread('alpha/I.bmp');
J=imread('alpha/J.bmp');K=imread('alpha/K.bmp');L=imread('alpha/L.bmp');
M=imread('alpha/M.bmp');N=imread('alpha/N.bmp');O=imread('alpha/O.bmp');
P=imread('alpha/P.bmp');Q=imread('alpha/Q.bmp');R=imread('alpha/R.bmp');
S=imread('alpha/S.bmp');T=imread('alpha/T.bmp');U=imread('alpha/U.bmp');
V=imread('alpha/V.bmp');W=imread('alpha/W.bmp');X=imread('alpha/X.bmp');
Y=imread('alpha/Y.bmp');Z=imread('alpha/Z.bmp');

%Natural Numbers
one=imread('alpha/1.bmp');two=imread('alpha/2.bmp');
three=imread('alpha/3.bmp');four=imread('alpha/4.bmp');
five=imread('alpha/5.bmp'); six=imread('alpha/6.bmp');
seven=imread('alpha/7.bmp');eight=imread('alpha/8.bmp');
nine=imread('alpha/9.bmp'); zero=imread('alpha/0.bmp');

%Creating Array for Alphabets
letter=[A B C D E F G H I J K L M N O P Q R S T U V W X Y Z];
%Creating Array for Numbers
number=[one two three four five six seven eight nine zero];

NewTemplates=[letter number];
save ('NewTemplates','NewTemplates')
clear all
```

Letter Detection –

1. We have created code file, *Letter_detection.m* .
2. In below code (see snapshot), we have created a function named letter which gives us the alphanumeric output of the input image from class ‘alpha’ by using command ‘readLetter()’. And then load the saved templates by using command load ‘NewTemplates’.
3. Then, we have resized the input image so it can be compared with the template’s images by using the command ‘imresize(filename,size)’. Then for loop is used to correlates the input image with every image in the template to get the best match.

Plate_detection.m | Letter_detection.m | template_creation.m

```
function letter=readLetter(snap)

load NewTemplates
snap=imresize(snap,[42 24]);
rec=[ ];

for n=1:length(NewTemplates)
    cor=corr2(NewTemplates{1,n},snap);
    rec=[rec cor];
end

ind=find(rec==max(rec));
display(ind);

% Alphabets listings.
if ind==1 || ind==2
    letter='A';
elseif ind==3 || ind==4
    letter='B';
elseif ind==5
    letter='C';
elseif ind==6 || ind==7
    letter='D';
elseif ind==8
    letter='E';
elseif ind==9
    letter='F';
elseif ind==10
    letter='G';
elseif ind==11
    letter='H';
elseif ind==12
    letter='I';
elseif ind==13
    letter='J';
elseif ind==14
    letter='K';
elseif ind==15
    letter='L';
elseif ind==16
    letter='M';
elseif ind==17
    letter='N';
elseif ind==18 || ind==19
    letter='O';
elseif ind==20 || ind==21
    letter='P';
elseif ind==22 || ind==23
    letter='Q';
elseif ind==24 || ind==25
    letter='R';
elseif ind==26
    letter='S';
elseif ind==27
    letter='T';
elseif ind==28
    letter='U';
elseif ind==29
    letter='V';
elseif ind==30
    letter='W';
elseif ind==31
    letter='X';
elseif ind==32
    letter='Y';
elseif ind==33
    letter='Z';
    %*-*-*-*-*
% Numerals listings.
elseif ind==34
    letter='1';
elseif ind==35
    letter='2';
elseif ind==36
    letter='3';
elseif ind==37 || ind==38
    letter='4';
elseif ind==39
    letter='5';
elseif ind==40 || ind==41 || ind==42
    letter='6';
elseif ind==43
    letter='7';
elseif ind==44 || ind==45
```

Number Plate Detection- This includes processing the image and then call the above two m-files to detect the number.

In this are calling the input image and converting it into the grayscale. Then the grayscale is converted into the binary image, and the edge of the binary images is detected by the Prewitt method. Then we will process that cropped license plate image and to display the detected number in the image and text format (in the command window).

Editor - F:\circuit digest\Number Plate Detection\Plate_detection.m

Plate_detection.m | Letter_detection.m | template_creation.m

```
close all;
clear all;

im = imread('Number Plate Images/image1.png');
imgray = rgb2gray(im);
imbin = imbinarize(imgray);
im = edge(imgray, 'prewitt');

%Below steps are to find location of number plate
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

im = imcrop(imbin, boundingBox);%crop the number plate area
im = bwareaopen(~im, 500); %remove some object if it width is too long or too small than 500

 [h, w] = size(im);%get width

imshow(im);

Iprops=regionprops(im,'BoundingBox','Area', 'Image'); %read letter
count = numel(Iprops);
noPlate=[]; % Initializing the variable of number plate string.

for i=1:count
   ow = length(Iprops(i).Image(1,:));
   oh = length(Iprops(i).Image(:,1));
   if ow<(h/2) & oh>(h/3)
       letter=Letter_detection(Iprops(i).Image); % Reading the letter corresponding the binary image 'N'.
       noPlate=[noPlate letter] % Appending every subsequent character in noPlate variable.
   end
end
```

Basic commands used in above code are-

imread() – This command is used to open the image into the MATLAB from the target folder.

rgb2gray() –This command is used to convert the RGB image into grayscale format.

im2bw() – It converts the input image to grayscale format (if it is not already an intensity image),and then uses thresholding to convert this grayscale image to binary.

edge() – This command is used to detect the edges in the image, by using various methods like Roberts, Sobel, Prewitt and many others.

regionprops() – This command is used to measure properties of image region.

numel() – This command is used to calculate the number of array elements.

imcrop() – This command is used to crop the image in the entered size.

bwareaopen() – This command is used to remove small objects from binary image.