DIGITAL IMAGE PROCESSING PROJECT
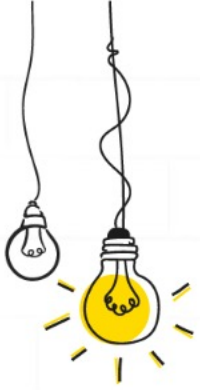
CAR NUMBER PLATE DETECTION USING MATLAB

Members:
Tejus D (PES2201800148)
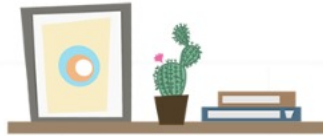Waris K R (PES2201800315)
Tushar Dixit (PES2201800138)

PROBLEM STATEMENT

To develop a system which would automatically detect the number plate and store it in its database. Later on when the information is required one can get it and use it. This process also helps to get the correct result compared to the manual one.
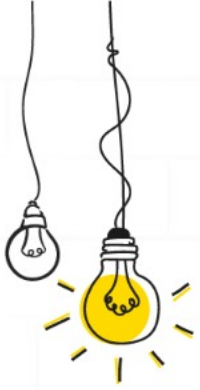
LITERATURE SURVEY

- Paper 1:

"Automatic Vehicle Identification By Plate Recognition"

Presented By:- Serkan Ozbay and Ergun Ercelebi

In this paper, a smart and simple algorithm is presented for vehicle's license plate recognition system. The proposed algorithm consists of three major parts: Extraction of plate region, segmentation of characters and recognition of plate characters. For extracting the plate region, edge detection algorithms and smearing algorithms are used. In segmentation part, smearing algorithms, filtering and some morphological algorithms are used.
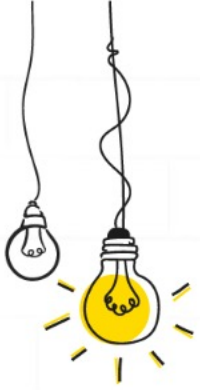
LITERATURE SURVEY

- **Paper 2**:

"Automobile Number Plate Recognition And Extraction Using Optical Character Recognition"

International Journal of Scientific & Technology Research Volume 3, Issue 10, October 2014

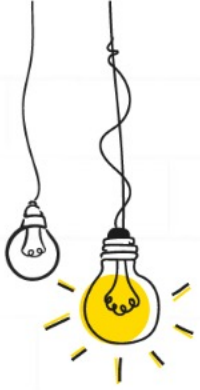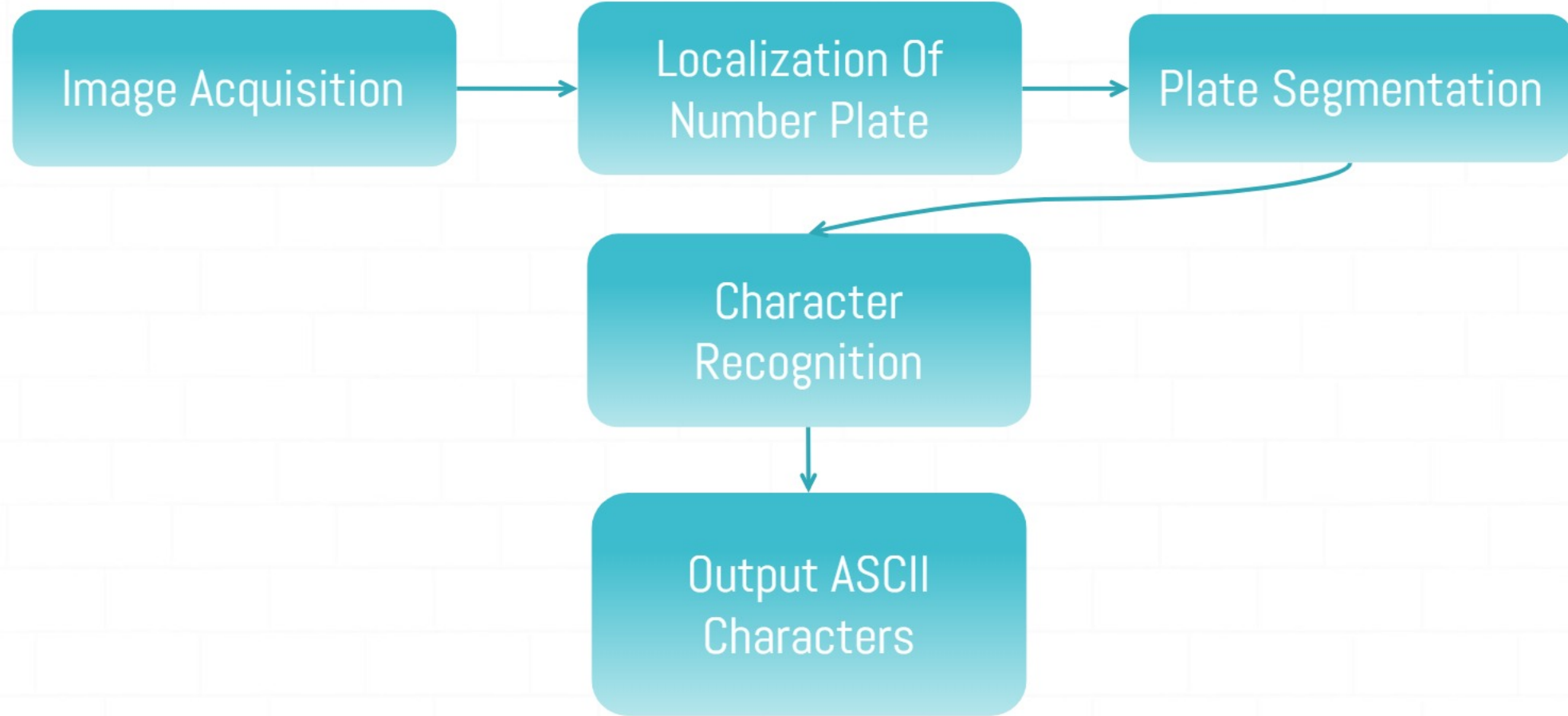*Presented By: Raja Vikramdeep Singh, Navneet Randhawa*

The purpose of this research is to develop an application which recognizes lincese plates from vehicles. The system takes images of automobiles as input and processes them. Once a license plate is detected, its digits are recognized and displayed on the User Interface. This work focuses on the design of a simple algorithm used for extracting the license plate from a single image, isolating the characters of the plate and identifying the individual characters.
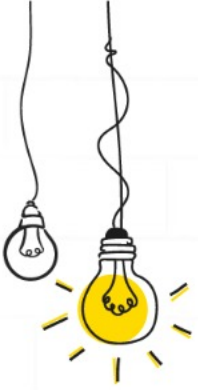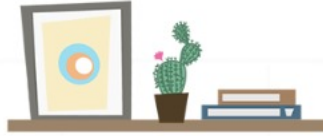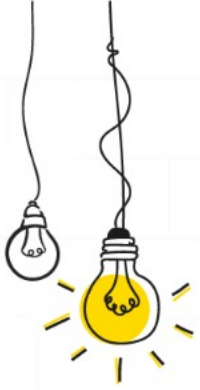
ABSTRACT

It is a vehicle's number/license plate recognition algorithm based on the very elementary technique of Templates matching. The algorithm takes an input image of the number plate (number plate should be dominant in the image) and after filtering the image, it performs region based operations. Then it tries to capture the characters regions in a processed binary image and with the aid of template matching outputs the string of number plate characters

```mermaid
Image Acquisition → Localization Of Number Plate → Plate Segmentation → Character Recognition → Output ASCII Characters
```

Image Acquisition → **Localization Of Number Plate** → **Plate Segmentation** → **Character Recognition** → **Output ASCII Characters**

CODE EXPLANATION
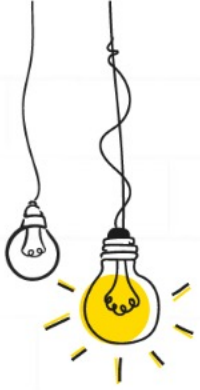
1. CREATING TEMPLATES OF CHARACTERS

```
%Creating Array for Alphabets
letter=[A Afill B Bfill C D Dfill E F G H I J K L M...
    N O Ofill P Pfill Q Qfill R Rfill S T U V W X Y Z];
%Creating Array for Numbers
number=[one two three four fourfill five...
    six sixfill sixfill2 seven eight eightfill nine ninefill ninefill2 zero zerofill];

%NewTemplates=[letter number];
character=[letter number];

NewTemplates=mat2cell(character,42,[24 24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24 ...
    24 24 24 24 24 24 24]);



save ('NewTemplates','NewTemplates')
clear all
```

Saved as
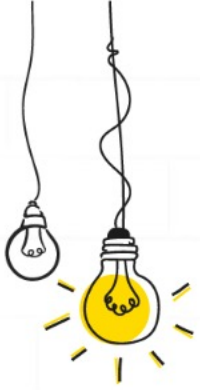NewTemplates.mat

2. IMAGE ACQUISITION AND CONVERSION

```matlab
im = imread('Number Plate Images/image1.png');
figure, imshow(im), title('Original Image');


imgray = rgb2gray(im);
figure, imshow(imgray), title('Grayscale Image');


imbin = imbinarize(imgray);
figure, imshow(imbin), title('Binary Image');

%This snippet displays the original, grayscale and binary image.
```

Original Image

**Grayscale Image**

Binary Image

3. LOCALIZATION OF PLATE

```
im = edge(imgray, 'prewitt');
figure, imshow(im), title('Detected Edges Using Prewitt Operator');


figure, imshow(im), title('Object Detection');
%Below steps are to find location of number plate
Iprops=regionprops(im,'BoundingBox','Area', 'Image');
count = numel(Iprops);
for i = 1:count
    rectangle('Position', Iprops(i).BoundingBox, 'EdgeColor', 'r', 'LineWidth', 2);
end
%Finding Object With Maximum Area
maxa = Iprops.Area;
boundingBox = Iprops.BoundingBox;
for i=1:count
  if maxa<Iprops(i).Area
      maxa=Iprops(i).Area;
      boundingBox=Iprops(i).BoundingBox;
  end
end
```

Detected Edges Using Prewitt Operator
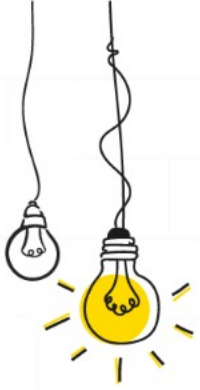
Object Detection
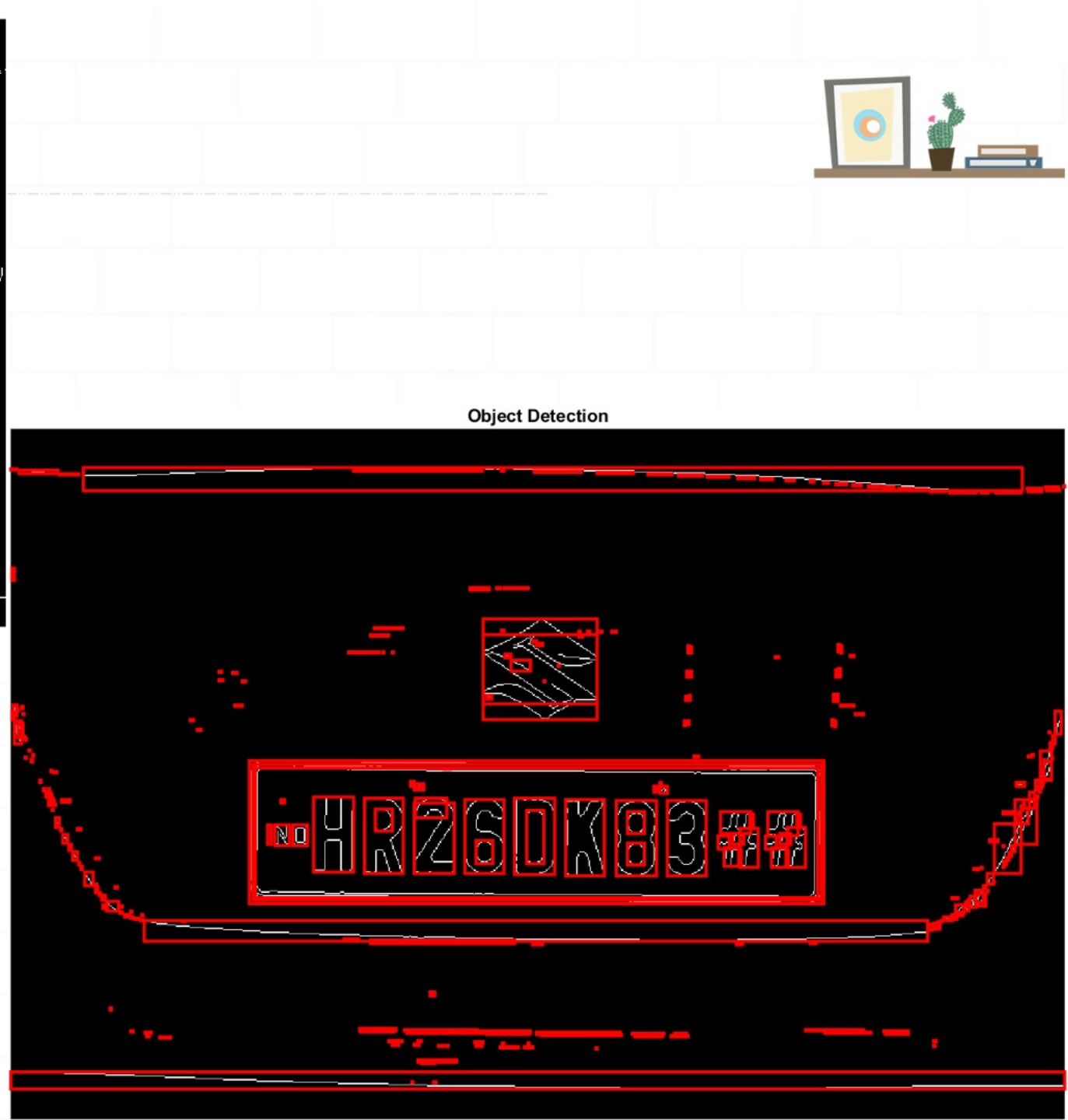
4. CROPPING AND REMOVAL OF SMALL OBJECTS
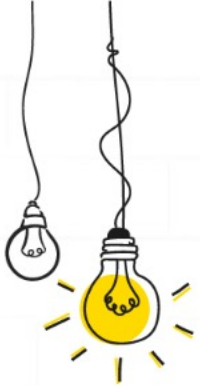
```
im = imcrop(imbin, boundingBox);%crop the number plate area
figure, imshow(im), title('Cropped image');
im = bwareaopen(~im, 500); %remove some object if it width is too long or too
small than 500
figure, imshow(im), title('After complementing and removing smaller objects');

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

Cropped image

IND **HR26DK83##**

After complementing and removing smaller objects

HR26DK83##
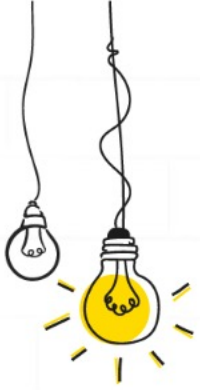
5. LETTER DETECTION WITH TEMPLATE MATCHING

```matlab
figure, imshow(im), title('Character Detection');

Iprops=regionprops(im,'BoundingBox','Area', 'Image'); %read letter
count = numel(Iprops);

for i = 1:count
    rectangle('Position', Iprops(i).BoundingBox, 'EdgeColor', 'r', 'LineWidth', 2);
end

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
```
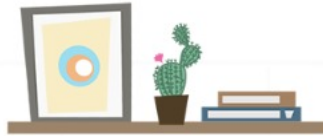
LETTER DETECTION FUNCTION

```
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);

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';
```

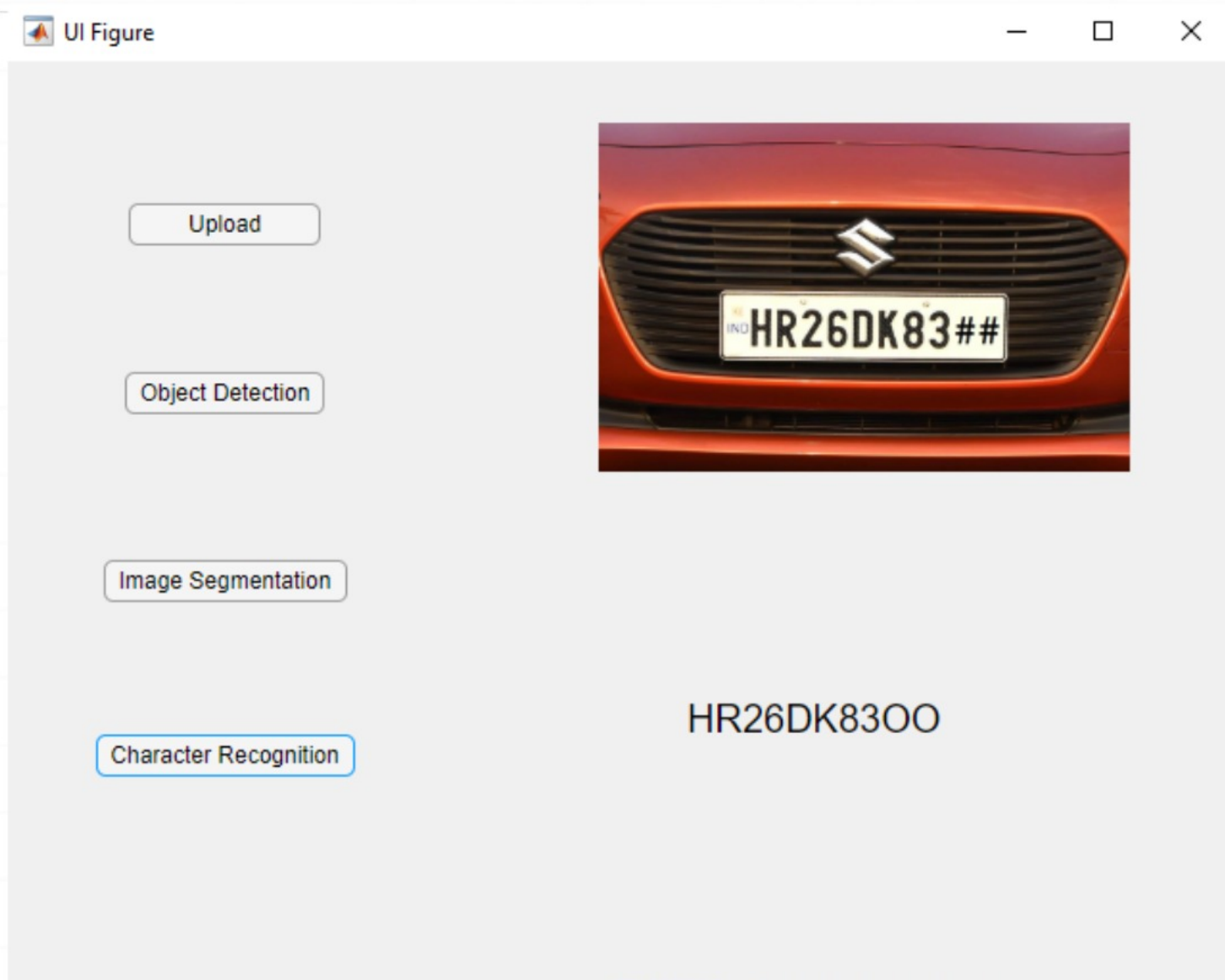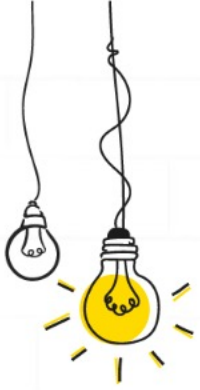Rec keeps recording all the correlation values
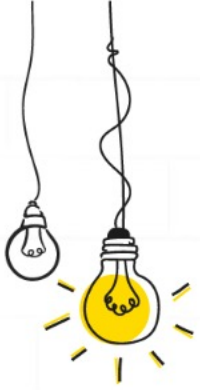
Character Detection


```
ind =

    19


noPlate =

    'HR26DK83OO'

fx >>
```

Final output on matlab console

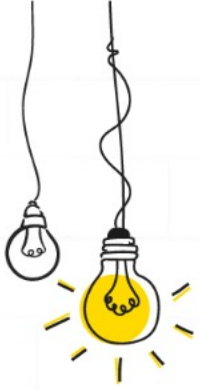# CREATING A MATLAB APPLICATION USING APP DESIGNER

CONCLUSION

- We presented application software designed for the recognition of car license plate. Firstly we extracted the plate location, then we separated the plate characters individually by segmentation and finally applied template matching with the use of correlation for recognition of plate characters.

- The system is tested over a large number of images. Finally it is proved to be %97.6 for the extraction of plate region, %96 for the segmentation of the characters and %98.8 for the recognition unit accurate, giving the overall system performance %92.57 recognition rates. This system can be redesigned for multinational car license plates in future studies.

THANK YOU