License Plate Recognition

by

Darshan Chandbodhle, IEN No 11812006

Vikas Chaudhary, IEN No 11812007

Apoorva Gondane, IEN No 11812013

Guide:

Prof. Rushikesh R. Nikam

Department of Computer Engineering

New Horizon Institute of Technology and Management University of Mumbai

(2020-2021)

NEW HORIZON INSTITUTE OF TECHNOLOGY AND MANAGEMENT

CERTIFICATE

This is to certify that the Mini Project entitled "**License Plate Recognition**" is a bonafide work of **"Darshan chandbodhle" (IEN No. 11812006), "Vikas Chaudhary" (IEN No. 11812007)** and **"Apoorva Gondane" (IEN No. 11812013)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of "Bachelor of Engineering" in "Computer Engineering".

____

(Prof. Rushikesh R. Nikam)

Guide

_______________________ ________________________

(Dr. Sanjay N. Sharma)
Head of Department

(Dr. Prashant D. Deshmukh)
Principal

NEW HORIZON INSTITUTE OF TECHNOLOGY AND MANAGEMENT

Project Report Approval for T.E.

This project report entitled "***License Plate Recognition***" by **Darshan Chandbodhle, Vikas Chaudhary,** and **Apoorva Gondane** is approved for the ***Mini Project*** in ***Computer Engineering***, ***2020-21***.

Examiner Name Signature

1. ___________________ _________________

2. ___________________ _________________

Date:

Place:

Abstract

A License Plate Recognition (LPR) System is one kind of automatic inspection of transport systems and is of considerable interest because of its potential applications to area such as automatic toll collection, traffic law enforcement and security control of restricted areas. The license plate recognition is done in the several stages. First existence of vehicle detected from desirable windows by image processing algorithms. In the last stages locating of plate in the image is recognize and characters of plate area are extracted by region growing and connectivity algorithms. The end of system are recognized all of characters set of license plate by properly neural network pattern recognition and result stored in database for uses in traffic application. This system work under variable illumination ,variable size of plate and dynamic backgrounds. The performance of the LPR system has been tested on 300 vehicles images which captured under various sizes of plate and variable illumination conditions. The rate of success recognition is 80.

Index

Chapter 1

Introduction

License plate recognition(LPR)is a tool for automatic vehicle and traffic monitoring by using digital image processing. For implementing LPR system we have used digital image processing technique and artificial neural network. This application provides access for entering car parks for register specification and time, depots and restricted areas which is necessary security, but can also applied to automating fuel supplies in filling station and traffic law enforcement, congestion pricing, and automatic toll collection.

The LPR system can be used to traffic control management for recognize vehicles that commit traffic violation, such as entering restricted area without permission , occupying lanes reserved for public transport, crossing red light, breaking speed limits ; etc.
The purpose for which this system is implemented real-time applications, this system is using advance and new techniques of digital image processing such as pattern recognition for recognize characters of license plate. Most previous applications and researches have in some way restricted their working conditions, such as limiting them to indoor scenes, stationary backgrounds, fixed illumination, fixed type of license plate, prescribed driveways, limited vehicle speeds, or designated ranges of the distance between camera and vehicle. The aim of this application is to remove many of these restrictions.

1.1 Objective

1.Analysis of Vehicular Traffic at Peak Conditions.
2.Flexible Toll Collection.
3.Effective Law Enforcement.
4.Vehicle Theft Prevention.
5.Effective Enforcement of Traffic Rules.

1.2 Scope

- ✓ A License Plate Recognition (LPR) System is one kind of automatic inspection of transport systems and is of considerable interest because of its potential applications to areas such as automatic toll collection, traffic law enforcement and security control of restricted areas.
- ✓ This project proposes an automatic license plate recognition system for license plates.
- ✓ We have different type of license plate with different shape, background, font size and structures.
- ✓ The license plate recognition is done in the several stages.
- ✓ This system worked under variable illumination, variable size of plate and dynamic backgrounds.
- ✓ This system implementation of the LPR system has been tested on 300 vehicles images which captured under various sizes of plate and variable illumination conditions. The rate of success recognition is 80 **.**

Chapter 2

Present Investigation

2.1 Problem Definition

To identify the number plate of a car using Optical Character Recognition (OCR) with the help to OpenCV and Python pytesseract for using it in use cases like vehicle surveillance.

2.2 Feasibility Analysis

Feasibility is the determination of whether or not a project is worth doing. The process followed in making this determination is called Feasibility Study. We have conducted three types of Feasibility Studies namely Technical , Operational and Economic Feasibility.

Technical Feasibility:

This is considered with specifying equipment and software that will successful satisfy the user requirement. Our project is technically feasible with less amount of technical expertise required and all other technical requirements such as hardware are easily available. The technical needs of the system may vary considerably but might include:

1. The facility to produce outputs in a given time.
2. Response time under certain conditions.
3. Ability to process a certain column of transaction at a particular speed

Operational Feasibility:

The system is providing option of input that is through keyboard / mouse and touchscreen. The users who are comfortable with using mobile user interface with touchscreen find this application operationally much more feasible.

Economic Feasibility:

The resources required by the system are available easily in the market at reasonable cost. This procedure is to determine the benefits and savings that are expected from a candidate system and compare with cost. Otherwise further justification or alterations in proposed system that have to be made if it is having a change of being approved .This is an ongoing effort that improves in accuracy of each phase of the system lifecycle. For our project we are not expecting any feasibility costs spent on this project because here we are using open source environments.

Chapter 3

System Design

3.1 Proposed Architecture

In this python project, to identify the number plate in the input image, we will use fol- lowing features of openCV:

GaussianBlur: Here we use a Gaussian kernel to smoothen the image. This technique is highly effective to remove Gaussian noise. OpenCV provides a cv2.GaussianBlur() function for this task.

Sobel: Here we calculate the derivatives from the image. This feature is important for many computer vision tasks. Using derivatives we calculate the gradients, and a high change in gradient indicates a major change in the image. OpenCV provides a cv2.Sobel() function to calculate Sobeloperators.

Morphological Transformation: These are the operations based on image shapes and are performed on binary images. The basic morphological operations are Erosion, Dila-tion, Opening, Closing. The different functions provided in OpenCV are: cv2.erode() cv2.dilate() cv2.morphologyEx()

Contours: Contours are the curves containing all the continuous points of same intensity. These are very useful tools for object recognition. OpenCV provides cv2.find Contours() functions for this feature.

License plate of the vehicle is detected using various features of image processing li- brary open CV and recognizing the text on the license plate using python tool named as tesseract. To recognize the license plate we are usingthe fact that License plate of any vehicle has rectangular shape. So, after all the processing of an image we will find the contour having four points inside the stand consider it as the License Plate of the vehicle.

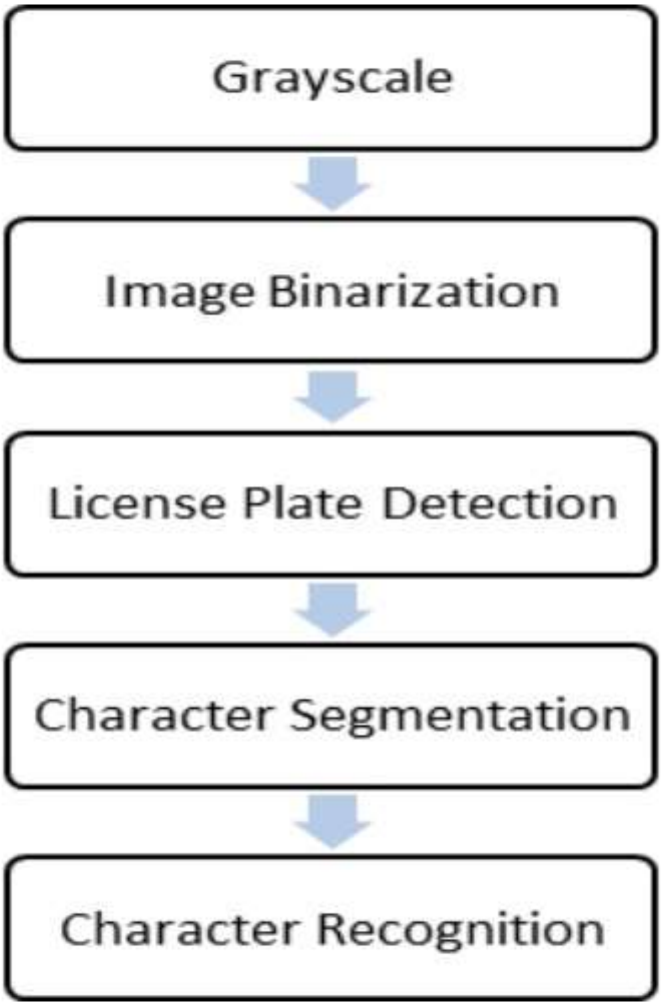


Figure I :Diagrammatic Representation of Workflow

Chapter 4

Implementation Details

4.1 Algorithm

LPR algorithms
For each new detected object in real time, a LPR algorithm is invoked.
This algorithm has three main steps:
License plate detection
Character segmentation
Character recognition

The most computational expensive task is the first one, where a white rectangle is search in the image. For this task, cascade classifiers are commonly used, and this could run both in a CPU or in a GPU based hardware platform. The input image is the already detected object, coming from the real- time pipeline. If the object size is too small, the image is discarded. Then, the plate detection is applied. If it is found a match, the algorithm continues, until trying to detect the whole plate. This algorithm could be running locally in the camera or in a centralized server that is receiving the objects detected.

As the images are coming from video, there are several pictures from the same object, and even several objects could be observed. By now, all the object cropped- images are evaluated, but in the future a prioritization rule must be applied.

4.2 Code

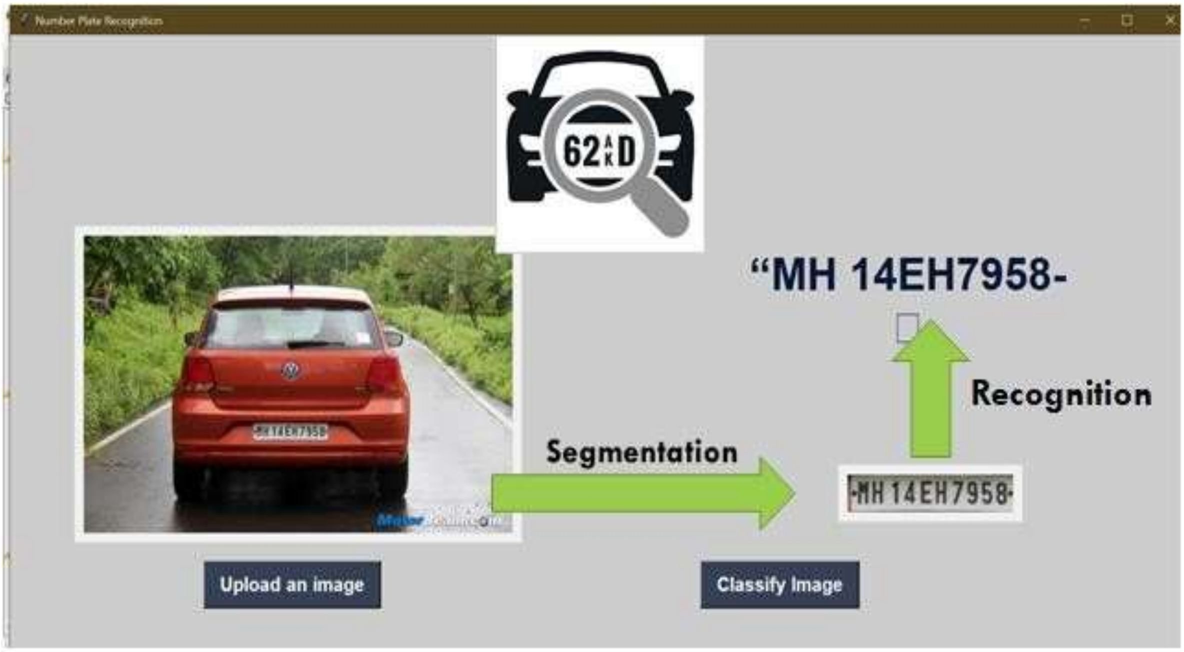
```
importtkinter as tk
fromtkinter import filedialog
fromtkinter import *
from PIL import ImageTk, Image
fromtkinter import PhotoImage
importnumpy as np
import cv2
importpytesseract as tess
def clean2_plate(plate):
gray_img = cv2.cvtColor(plate, cv2.COLOR_BGR2GRAY)

    _, thresh = cv2.threshold(gray_img, 110, 255, cv2.THRESH_BINARY)
num_contours,hierarchy = cv2.findContours(thresh.copy(),cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_NONE)

ifnum_contours:
contour_area = [cv2.contourArea(c) for c in num_contours]
max_cntr_index = np.argmax(contour_area)

max_cnt = num_contours[max_cntr_index]
max_cntArea = contour_area[max_cntr_index]
x,y,w,h = cv2.boundingRect(max_cnt)

if not ratioCheck(max_cntArea,w,h):
returnplate,None

final_img = thresh[y:y+h, x:x+w]
returnfinal_img,[x,y,w,h]

else:
returnplate,None

defratioCheck(area, width, height):
ratio = float(width) / float(height)
if ratio < 1:
ratio = 1 / ratio
if (area < 1063.62 or area > 73862.5) or (ratio < 3 or ratio > 6):
return False
return True

defisMaxWhite(plate):
```

```
avg = np.mean(plate)
if(avg>=115):
return True
else:
return False

defratio_and_rotation(rect):
    (x, y), (width, height), rect_angle = rect

if(width>height):
angle = -rect_angle
else:
angle = 90 + rect_angle

if angle>15:
return False

if height == 0 or width == 0:
return False

area = height*width
if not ratioCheck(area,width,height):
return False
else:
return True

top=tk.Tk()
top.geometry('900x700')
top.title('Number Plate Recognition')
# top.wm_iconbitmap
top.iconphoto(True, PhotoImage(file="tesla.png"))
img = ImageTk.PhotoImage(Image.open("tesla.png"))
top.configure(background='#CDCDCD')
label=Label(top,background='#CDCDCD', font=('arial',35,'bold'))
# label.grid(row=0,column=1)
sign_image = Label(top,bd=10)
plate_image=Label(top,bd=10)
def classify(file_path):

    ###########################################################

res_text=[0]
res_img=[0]
```

```
img = cv2.imread(file_path)
    # cv2.imshow("input",img)

    # if cv2.waitKey(0) & 0xff == ord('q'):
    #     pass
    img2 = cv2.GaussianBlur(img, (3,3), 0)
    img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

    img2 = cv2.Sobel(img2,cv2.CV_8U,1,0,ksize=3)
    _,img2 = cv2.threshold(img2,0,255,cv2.THRESH_BINARY+cv2.THRESH_OTSU)

element = cv2.getStructuringElement(shape=cv2.MORPH_RECT, ksize=(17, 3))
morph_img_threshold = img2.copy()
cv2.morphologyEx(src=img2, op=cv2.MORPH_CLOSE, kernel=element,
dst=morph_img_threshold)
num_contours, hierarchy=
cv2.findContours(morph_img_threshold,mode=cv2.RETR_EXTERNAL,method=cv2.
CHAIN_APPROX_NONE)
cv2.drawContours(img2, num_contours, -1, (0,255,0), 1)


fori,cnt in enumerate(num_contours):

min_rect = cv2.minAreaRect(cnt)

ifratio_and_rotation(min_rect):

x,y,w,h = cv2.boundingRect(cnt)
plate_img = img[y:y+h,x:x+w]
print("Number identified number plate...")
            # cv2.imshow("num plate image",plate_img)
            # if cv2.waitKey(0) & 0xff == ord('q'):
            #     pass
res_img[0]=plate_img
cv2.imwrite("result.png",plate_img)
if(isMaxWhite(plate_img)):
clean_plate, rect = clean2_plate(plate_img)

ifrect:
fg=0
x1,y1,w1,h1 = rect
x,y,w,h = x+x1,y+y1,w1,h1
plate_im = Image.fromarray(clean_plate)
text = tess.image_to_string(plate_im, lang='eng')
```

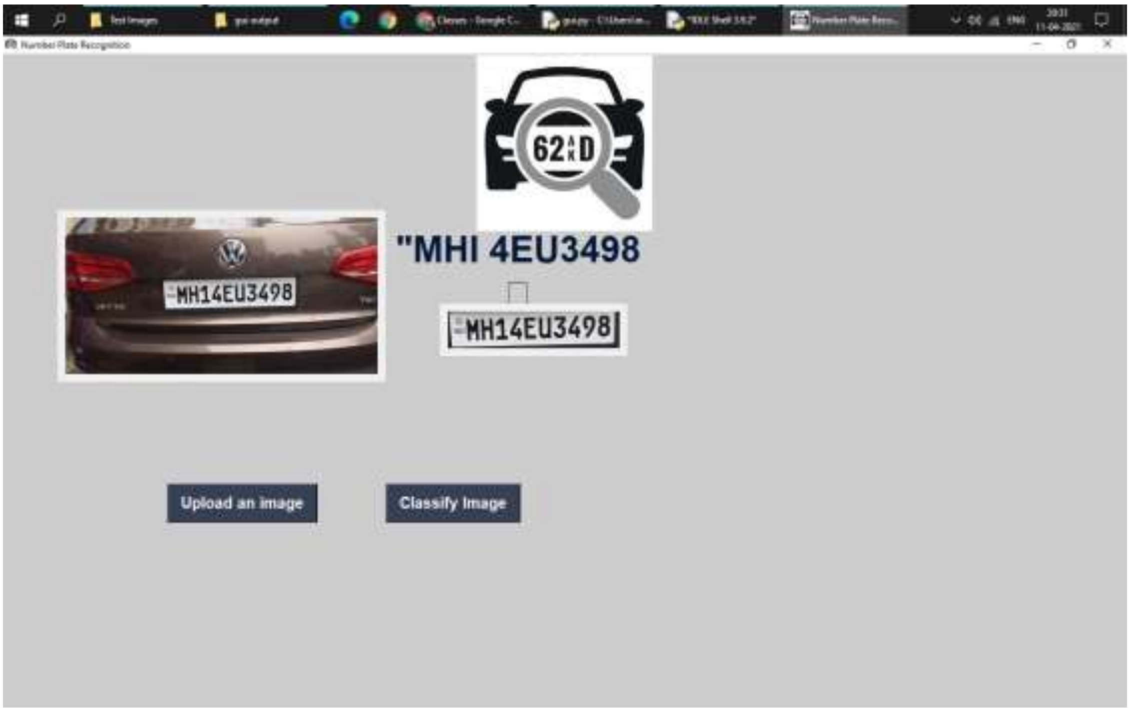
```
res_text[0]=text
if text:
break
                    # print("Number Detected Plate Text : ",text)

    ########################################################
label.configure(foreground='#011638', text=res_text[0])
    # plate_img.configure()
uploaded=Image.open("result.png")
im=ImageTk.PhotoImage(uploaded)
plate_image.configure(image=im)
plate_image.image=im
plate_image.pack()
plate_image.place(x=560,y=320)
defshow_classify_button(file_path):
classify_b=Button(top,text="Classify Image",command=lambda:
classify(file_path),padx=10,pady=5)
classify_b.configure(background='#364156', foreground='white',font=('arial',15,'bold'))
classify_b.place(x=490,y=550)
    # classify_b.pack(side=,pady=60)
defupload_image():
try:
file_path=filedialog.askopenfilename()
uploaded=Image.open(file_path)
uploaded.thumbnail(((top.winfo_width()/2.25),(top.winfo_height()/2.25)))
im=ImageTk.PhotoImage(uploaded)
sign_image.configure(image=im)
sign_image.image=im
label.configure(text=")
show_classify_button(file_path)
except:
pass
upload=Button(top,text="Upload an
image",command=upload_image,padx=10,pady=5)
upload.configure(background='#364156', foreground='white',font=('arial',15,'bold'))
upload.pack()
upload.place(x=210,y=550)
# sign_image.pack(side=BOTTOM,expand=True)
sign_image.pack()
sign_image.place(x=70,y=200)

# label.pack(side=BOTTOM,expand=True)
label.pack()
label.place(x=500,y=220)
heading = Label(top,image=img)
heading.configure(background='#CDCDCD',foreground='#364156')
heading.pack()
top.mainloop()
```

4.3 Screenshot



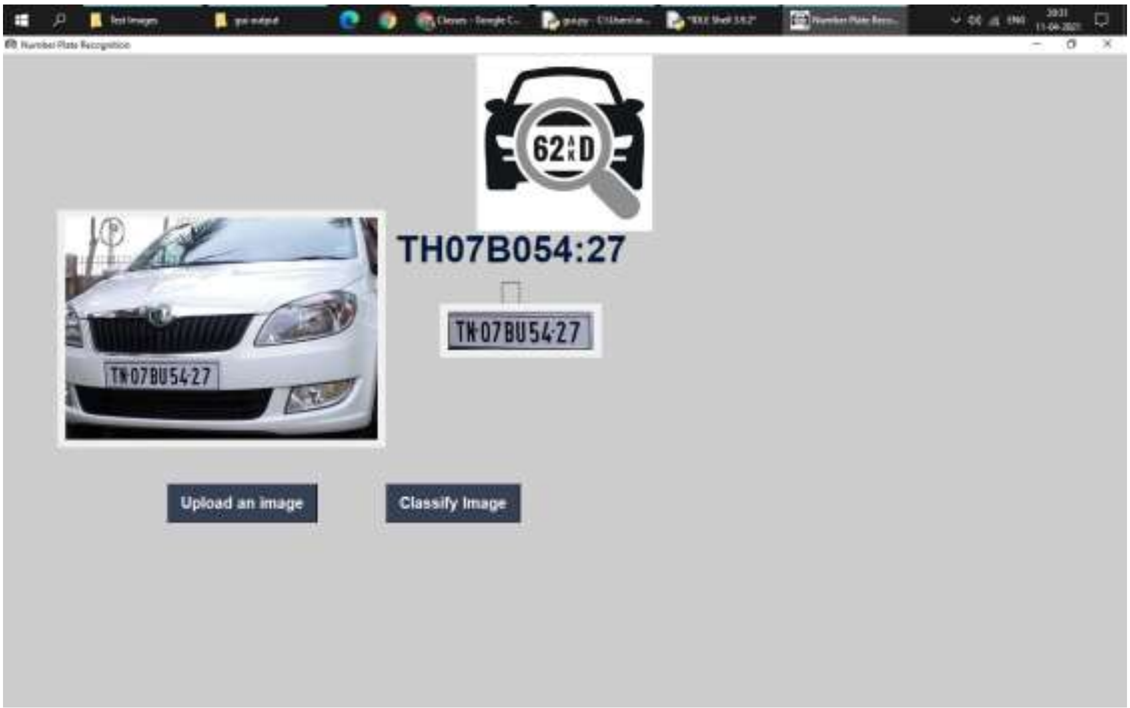
“MH 14EH7958-

Image 1



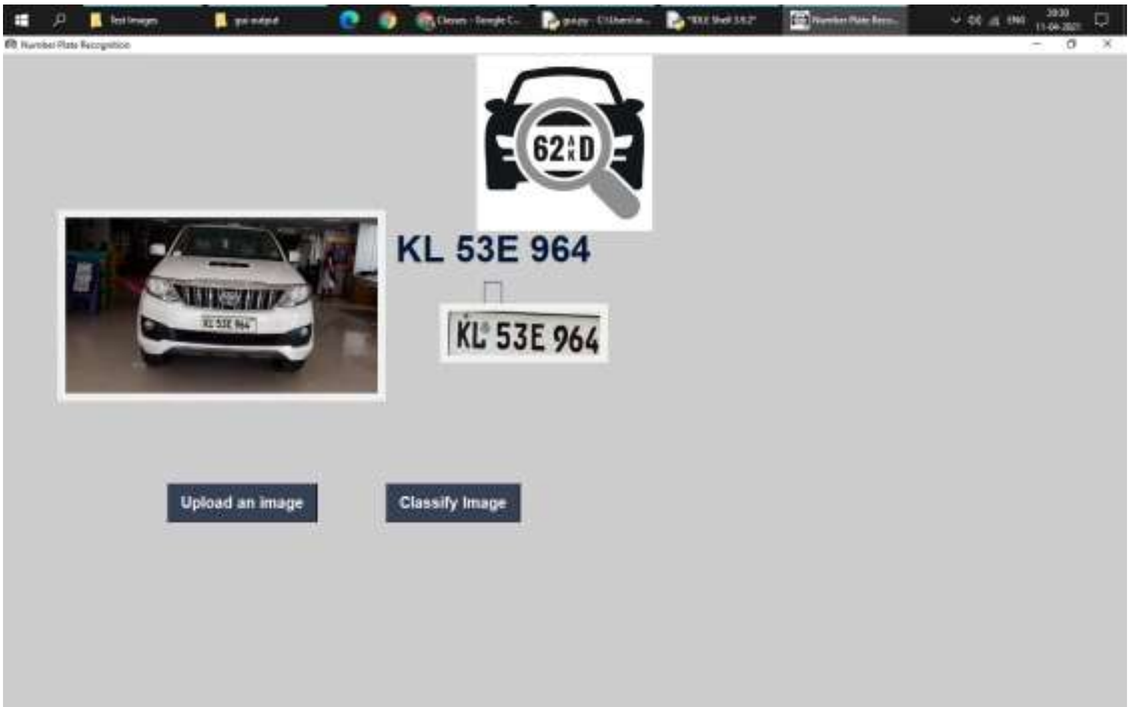
"MHI 4EU3498

Image 2



```
TH07B054:27
```

Image 3



```
KL 53E 964
```

Image 4

Chapter 5

Conclusion and Future Scope

A camera based algorithm for real time number plate detection and recognition was proposed. This algorithm is designed mainly for autonomous vehicles .The contour of moving vehicles can be detected accurate by the improved same difference algorithm .In the presented work we have design the Number Plate Recognition System. In this system we have developed a new method for detecting and recognizing car license plates .Future work is in-tended to be done in improving and testing the system on a larger number of images. It is found that it is working robustly and finely. The implementation language java has given wide portability and simplicity. It is very simple to install and operate. Not more manual energy is required. Security is provided to system itself by maintaining valid usernames and passwords.

This paper describes a study of vehicle number plate identification in traffic surveillance. LPR is very helpful and reliable for efficient traffic monitoring. Device with powerful image processing technique can easily detect interested vehicles from various angles and view plate information as output. LPR systems play an important role in the growth of the smart transport network. Recognition may use the image processing technique combined with neural networks to identify the number plates where the angled or side view images, moving distance images, numbering scheme, and number plate type (background) can be further improved. Detection of objects and neural networks is useful for detecting side views or tilted images and moving images from distance. For potential recognition systems, the choice is to use high-resolution cameras with an increased number of frames for good accuracies and improved accuracy for recognition in future.

Acknowledgement

We would like to take this opportunity to thank one and all.

It is our immense pleasure to express our gratitude to our Guide, Prof. Rushikesh Nikam for providing us with constructive and positive feedback during the preparation of this project.

We express gratitude to Prof. Megha V Gupta for constantly guiding us and giving us valuable insights.

We would like to express our thanks to the Head of Computer Department, Dr. Sanjay N. Sharma and all other staff members for their encouragement and suggestions.

Last but not the least, we are thankful to our friends for their support and coordination. We are also thankful to our parents for their constant support and best wishes.

Mr. Darshan Chandbodhle ____________________

Mr. Vikas Chaudhary ____________________

____

Mr .Apoorva Gondane

Date: