VEHICLE NUMBER PLATE RECOGNITION SYSTEM

A MINI-PROJECT REPORT

FOR THE COURSE

21CS212 – INTERNET PROGRAMMING LABORATORY

SUBMITTED BY

M.V.KSHEERABDINI (913118104054)

T.S.B.DHARSHINI (913118104023)

DEPARTMENT OF

COMPUTER SCIENCE AND

ENGINEERING

VELAMMAL COLLEGE OF ENGINEERING

AND TECHNOLOGY

MADURAI-625 009

MAY 2023

BONAFIDE CERTIFICATE

Certified that this project report "**TIME SERIES ANALYSIS OF ARIMA ALGORITM ON INDIA DEBT PREDICTION"** is the bonafide work of "**KSHEERABDINI.M.V(913118104054) & DHARSHINI.T.S.B(91311810402**" who carried out the project work under my supervision.

SIGNATURE

Mrs. S.PADMADEVI

Assistant Professor

Departtment Of Computer Science and Engineering
Velammal College of Engineering and Technology
Madurai-625009

OBJECTIVE

This project has been composed with the aim of designing a software on vehicle number plate recognition system. A lot of effort has been made to make this project report interesting and a learning experience for us. This report has been explained with the help of diagrams and figures. The running project has presented through a PowerPoint representation. The subject matter has been compiled in a simple, illustrative and lucid manner.

LIST OF ABBREVATIONS

OpenCV	Open source computer vision library
OCR	Optical character recognition
Pytesseract	Python tesseract
ANPR	Automatic number plate recognition system

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW:

ANPR or license plate recognition (LPR) has been one of the useful approaches for vehicle surveillance. It is can be applied at number of public places for fulfilling some of the purposes like traffic safety enforcement, automatic toll text collection , car park system and Automatic vehicle parking system .

ANPR is generally divided in four steps: (1) Vehicle image as input (2) Number plate detection (3) Character segmentation and (4) Character recognition. As it is shown in Fig.1, the first step i.e. the images is given as the input to check out the number plate then the detection takes place followed by character segmentation and recognition. Presently number plate detection and recognition processing time is less than 50 ms in many systems. The success of fourth step depends on how second and third step are able to locate vehicle number plate and separate each character.

These systems follow different approaches to locate vehicle number plate from vehicle and then to extract vehicle number from that image. Most of the ANPR systems are based on common approaches like artificial neural network (ANN) , Probabilistic neural network (PNN) , Optical Character Recognition (OCR), Feature salient , MATLAB, Configurable method, Sliding concentrating window (SCW), BP neural network, support vector machine(SVM), inductive learning , region based , colour segmentation , fuzzy based algorithm , scale invariant feature transform (SIFT) , trichromatic imaging, Least Square Method(LSM) ,online license plate matching based on weighted edit distance and colour-discrete characteristics . A case study of license plate reader (LPR) is well explained in . Some authors focus on improving resolution of the low-resolution image by using technique called super resolution .

CHAPTER 2

Module Descriptions

2.1 Detection and Capturing of Images

- Detection of the number plate is considered crucial to identify the presence of number plate at the entry barrier. One of the most difficult tasks in computer vision and image processing is to detect the object .
- To detect and capture the image, preprocessing is the technique used to process the image before extraction. For pre-processing, the input of RGB image is converted to grayscale and it is best suited with adaptive binarization methods such as Niblack's method .
- Using acquisition technique, the consideration of the quality of image retrieved by the computer has sufficient light from the camera to get a clear image of the license plate from the car.
- The accuracy of number plate detection may be challenging due to an open-air environment .
- A few approaches to be considered are criteria such as the edge of the license plate, texture of pixel intensity distribution and colour character which helps in license or number plates which are tilted or deformed .

2.2 Pre-processing of Image

- Once an image has been captured, the preprocessing can begin as it is critical in order to continue to the Edge detection phase.
- The preprocessing consists of converting from RGB to grayscale followed by contrast and intensity adjustments. Removal of RGB is done by eliminating the saturation and hue data of the image.
- The RGB image comprises 30% of red, 60% of green and 11% of blue
- The intensity and contrast are adjusted to reduce noise of image. Histogram Equalization technique is used to adjust the intensity and helps to improve the image contrast when the value of intensity image is transformed. Huge amount of data detected in grayscale image.
- The general way of edge detection of the image is through filtering, differentiation and detection which have higher chances of detecting the image.

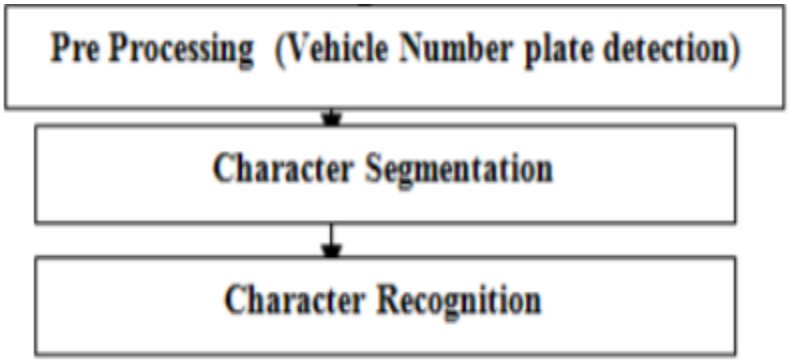

Figure 2.1 - FLOWCHART

2.3 CODE:

```
import cv2
import imutils
import numpy as np
import pytesseract
from PIL import Image
pytesseract.pytesseract.tesseract_cmd = 'C:\\Program Files\\Tesseract-OCR\\tesseract.exe'
img = cv2.imread('1.jpg',cv2.IMREAD_COLOR)
img = imutils.resize(img, width=500 )
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) #convert to grey scale
gray = cv2.bilateralFilter(gray, 11, 17, 17) #Blur to reduce noise
edged = cv2.Canny(gray, 30, 200) #Perform Edge detection
cnts,new = cv2.findContours(edged.copy(), cv2.RETR_LIST, cv2.CHAIN_APPROX_SIMPLE)
img1=img.copy()
cv2.drawContours(img1,cnts,-1,(0,255,0),3)
cv2.imshow("img1",img1)
cnts = sorted(cnts, key = cv2.contourArea, reverse = True)[:30]
screenCnt = None #will store the number plate contour
img2 = img.copy()
cv2.drawContours(img2,cnts,-1,(0,255,0),3)
cv2.imshow("img2",img2) #top 30 contours

count=0
idx=7
# loop over contours
```

```
for c in cnts:
  # approximate the contour
      peri = cv2.arcLength(c, True)
      approx = cv2.approxPolyDP(c, 0.018 * peri, True)



      if len(approx) == 4: #chooses contours with 4 corners
            screenCnt = approx
            x,y,w,h = cv2.boundingRect(c) #finds co-ordinates of the plate
            new_img=img[y:y+h,x:x+w]
```

CHAPTER 3

RESULTS

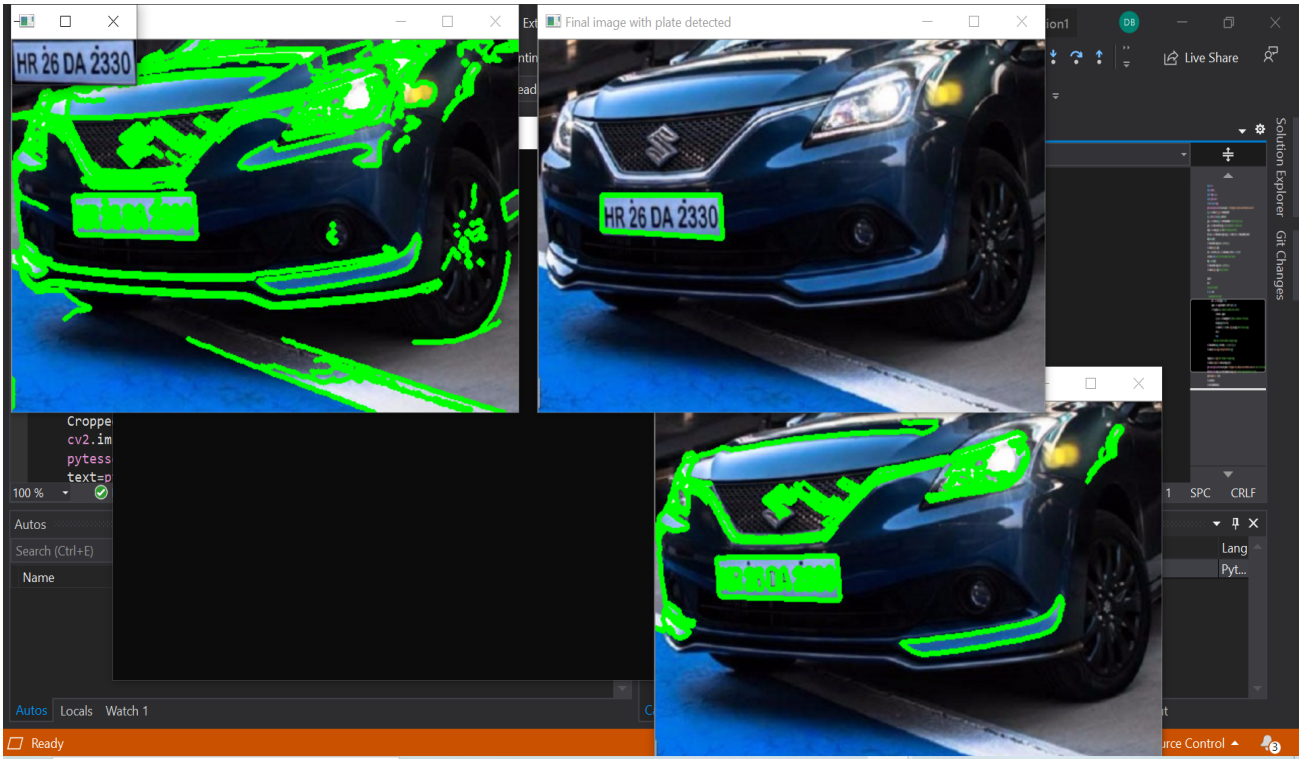

Figure 4.1-Edge Detection

Figure 4.2-cropped image

Figure 4.3-Contour detection

CHAPTER 4

CONCLUSION AND FUTURE SCOPE

This is used to design an efficient automatic authorized vehicle identification system by using the vehicle number plate in order to reduce crime rate. This system use image processing techniques for recognition of the vehicle from the database stored in the computer. The system works satisfactorily for wide variation of conditions and different types of number plates. The system is implemented and executed in OCR and performance is tested on genuine images.

FUTURE SCOPE:

- Today advances technology took Automatic Number Plate Recognition (ANPR) systems from hard to set up, limited expensive, fixed based applications to simple mobile ones in which “point to shoot” method can be used.
- This is possible because of the creation of software which ran on cheaper PC based and also non specialist hardware in which their no need to give pre- defined direction, angels, speed and size in which the plate would be passing
- The camera field of view.
- Also Smaller cameras which can read license plates at high speed, along with smaller, more durable processors that can fit in police vehicles, allowed law enforcement officers to patrol daily with the benefit of license plate recognition in real time.

ANNEXURE

REFERENCES

Websites:

- https://www.w3schools.com/PHP
- https://www.tutorialspoint.com/php
- https://www.codecademy.com/learn/php