VISVESVARAYA TECHNOLOGY UNIVERCITY

“Jnana Sangama”, Belgaum -590 018

A project report on

“TO MAKE AN INTELLIGENT GATE BY USING DIGITAL IMAGE PROCESSING”

A project report submitted in partial fulfilment of the requirement for the VIII semester degree of

Bachelor of engineering

In

Electrical & Electronics engineering

Submitted by

SACHINKUMAR ANJUTAGI	(C1R16EE063)
PRAVEENKUMAR D N	(1CR16EE054)
SHASHIKUMAR	(1CR17EE412)
BHARATH R	(1CR17EE402)

Under the guidance of

Ms. REBA KUNDU

Associate professor, Department of Electrical & Electronic engineering

CMR institute of technology

CMR institute of technology, Bengaluru- 560 037

Department of Electrical & Electronic engineering

2019-2020

CMR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS AND ENGINEERING

#132 AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI,

BANGALORE-560 037

Certificate

This is to certify that the project entitled **"TO MAKE AN INTELLIGENT GATE BY USING DIGITAL IMAGE PROCESSING"** has been successfully completed **Sachinkumar anjutagi (1CR16EE063), Praveen Kumar D N (1CR16EE054), Shashikumar (1CR17EE412), Bharath R (1CR17EE402)** bonafide students of **CMR institute of technology** in partial fulfilment of the requirement for the award of degree of bachelor of engineering in **Electrical & Electronics Engineering** of the Visvesvaraya technology university, Belgaum during the academic year 2018-2019. It is certified that all the correction indicated for internal Assessment have been incorporated in the project report. The project report has been approved as it satisfies the academic requirement in respect of project work prescribed for the said degree.

Signature of the Guide	*Signature of the HOD*	*Signature of the Principal*
Ms. Reba kundu Assistant professor EEE department CMRIT, Bengaluru	Dr. K. Chitra professor & HOD EEE department CMRIT, Bengaluru	Dr. Sanjaya jain Principal CMRIT, Bengaluru

External Viva

Name of Examiners	Signature & Date
1.	
2.	

CMR INSTITUTE OF TECHNOLOGY

Department of Electrical & Electronics Engineering

AECS, Layout, Bengaluru-560 037

DECLARATION

We, **Sachinkumar anjutagi (1CR16EE063), Praveen Kumar D N (1CR16EE054), Shashi Kumar (1CR17EE412), Bharath R (1CR17EE402)** bonafide students of CMR instate of technology, Bengaluru, here by declare that the dissertation entitled, **TO MAKE AN INTELLIGENT GATE BY USING DIGITAL IMAGE PROCESSING**" has been carried out by us under the guidance of **Ms. Reba Kundu, Associate** professor, CMRIT, Bengaluru, in partial fulfilment of the requirement for the award of degree of bachelor of engineering in, in **Electrical and Electronics Engineering of Visvesvaraya technology University,** Belgaum during the academic year 2019-2020. The work done in the dissertation report is original and it has not been submitted for any other degree in any university.

Place: Bengaluru

Date:

Sachinkumar Anjutagi (1CR16EE063)

Praveen Kumar (1CR16EE054)

Shashi Kumar (1CR17EE412)

Bharath R (1CR16EE402)

Acknowledgement

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without mentioning people who made it possible, successes the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

I take great pleasure in expressing our sincere thanks to **Dr. Sanjay Jain, Principal, CMR Institute of Technology,** Bengaluru for providing an excellent academic environment in the college and for his continuous motivation towards a dynamic career. I would like to profoundly thank **Dr. B Narasimha Murthy,** Vice-principal of CMR Institute of Technology and the whole Management for providing such a healthy environment for the successful completion of the project work.

I would like to convey my sincere gratitude to **Dr. K Chitra, Head of Electrical and Electronics Engineering Department, CMR Institute of Technology**, **Bengaluru** for herin valuable guidance and encouragement and for providing good facilities to carry out this project work.

I would like to express my deep sense of gratitude to **Ms. Reba Kundu, Assistant Professor, Electrical and Electronics Engineering, CMR Institute of Technology, Bengaluru** for his/her exemplary guidance, valuable suggestions, expert advice and encouragement to pursue this internship work.

Finally, I acknowledge the people who mean a lot to me, my parents, for their inspiration, unconditional love, support, and faith for carrying out this work to the finishing line.

Lastly, to the Almighty, for showering His Blessings and to many more, whom I didn't mention here.

Sachinkumar Anjutagi (1CR16EE063)

Praveen Kumar (1CR16EE054)

Shashi Kumar (1CR17EE412)

Bharath R (1CR16EE402)

Abstract

A designing of a system which captures the image of the number plate automatically of a Shovel-Dumper combination and these details were verified using Raspberry Pi4 processor for authentication. This system captures the number plate of shovel and dumper further processing for the character recognition. Automation is the most frequently spelled term in the field of electronics. The hunger for automation brought many revolutions in the existing technologies. This paper makes use of an onboard computer, which is commonly termed as Raspberry Pi4 processor. It acts as heart of the project. This onboard computer can efficiently communicate with the output and input modules which are being used. The device which is able to perform the task is a Raspberry Pi4 processor. When any vehicle passes by the system, the image of the number plate of every vehicle is captured using camera. The image of the number plate details is fed as input to the Raspberry Pi4 processor. Once the details are recognized then the processor operates it detects an unauthorized image of number plate was detected. To perform this task, Raspberry Pi 4 processor is programmed using embedded 'Raspbian'.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter - 1

INTRODUCTION

1. INTRODUCTION

1.1 Problem statement

To design an automated Licence Plate recognition system Using Raspberry Pi 4 in the application of the shovel-Dumper combination.

1.2 Challenges

Generally, an automatic license plate recognition (ALPR) system is made up of five modules: Plate detection, Segmentation, Classification, Plate recognition, OCR Segmentation modules (Figure 1)

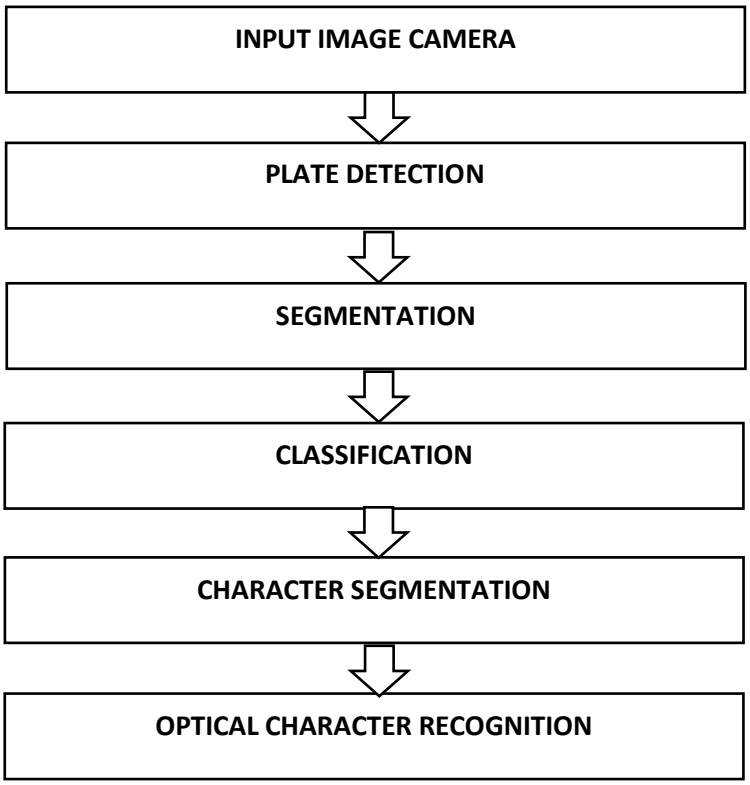

FIG 1. ALPR process

- Firstly, License Plate localization from Shovel-Dumper images.
- Secondly, Character segmentation from localized license plate.
- Finally, optical character recognition of extracted characters**.**

The most well-known answers for tag restriction in computerized pictures are through the execution of edge extraction, morphological administrators, and Sobel administrator. An edge methodology is ordinarily straightforward and quick. Sobel administrator for edge discovery gives constructive outcomes on the picture. The confinement of tags through

morphologically based methodologies is not defenceless to clamour but rather is moderate in execution.

After the limitation of the tag comes the character division process. Normal character division procedures depend on histogram investigation and thresholding. Other late methodologies proposed are the utilization of counterfeit neural systems.

The last phase of an ALPR framework is the character acknowledgment process. To manage various varieties found in characters crosswise over various tags will require the portioned character to experience some pre-handling steps, for example, standardization and skew amendment. These extra strides end up being useful as it significantly diminishes the required calculation time.

1.3 Types of license plate recognition system

1. Online ALPR framework: In an online ALPR framework, the limitation and elucidation of tags occur promptly from the approaching video outlines, enabling Real-time tracking through the surveillance camera.
2. Offline ALPR framework: A logged off ALPR framework, interestingly, catches the shovel, dumper number plate pictures and stores them in a concentrated information server for further preparing, i.e. for translation of tags.

1.4 Challenges in ALPR

In the created nations the qualities of the tags are entirely kept up. For instance, the measure of the plate, shade of the plate, text style face/size/shade of every character, dispersing between ensuing characters, the quantity of lines in the tag, script and so on are kept up particularly. A portion of the pictures of the standard tags utilized as a part of created nations.

Automatic license plate recognition has two major technological requirements

1. The quality of the license plate recognition algorithms.
2. The quality of the image acquisition (camera and the illumination conditions).

The better algorithms are:

1. Higher is the recognition accuracy.
2. Faster is the processing speed.

3. Wider is the range of picture quality it can be used on.

By and large, one LPR programming can read plates from one specific nation just. This is on the grounds that the geometrical structure of the plate and introduction, text styles, and grammar were imperative parts of the system LPR system. Without the earlier information of the plate geometry (character distribution, character spacing, plate colour, dimension ratios etc.), the algorithm may out not even find the plate in the captured image.

Furthermore, there are wide variety plate types:

1. Black characters on the white/light background.
2. White characters on the black/dark background.
3. Single row plates.
4. Multi row plates.

If the LPR system cannot utilize such useful information like the plate structure, it loses a helpful aid derived from its input data. This could result in the reduction of the license plate recognition system accuracy. Without using prior information of the plate, the remaining part of the recognition system should be significantly more robust and this leads to more challenges.

The image acquisition technique determines the captures image quality of the license plate with which the detection algorithms have to work. Better the quality of the acquired images, higher is the accuracy one can achieve.

A well captured image has the following properties:

1. Good spatial resolution,
2. Good sharpness,
3. High contrast,
4. Adequate lighting conditions,
5. The Decent angle of view

Chapter - 2

LITERATURE REVIEW

2. AN OVERVIEW OF THE SYSTEM

2.1 Plate Detection

In this step, we have to detect all the plates in the current camera frame. Two broad categories in which they can be defined are:

- Segmentation
- Classification

2.1.1 Segmentation

Segmentation is the process of dividing an image into multiple segments. This process is to simplify the image for analysis and make feature extraction easier. One important feature that can be exploited from Number plates is the high number of vertical edges.

2.1.2 Classification

This technique is utilized to distinguish the potential license plate region from the given picture. The principle target of such sort of systems is to confine the license plate region from images of the Shovel Dumper that are captured from the camera mounted on the Raspberry Pi 4. The quality of the image forms an important part of this technique so pre-processing the image helps in improving the quality.

Number plates usually appear to have high contrast areas in the image (black-and-yellow or black-and-white). The numbers and lettes are placed in the same row (i.e. at an identical vertical level), which results in frequent changes in intensity horizontally. This provides the basis for detecting the changes in the horizontal intensity horizontally. This provides the basis for detecting the changes in the horizontal intensity, as the rows that will contain the number plate are expected to show sharp variations. The reason for this sharp variation is the contrast between the letters and its background.

2.2 Optical Character

The project I about optical character recognition. It is process of classifying optical patterns with respect to alphanumeric or other characters. Optical character recognition process includes segmentation, feature extraction and classification.

Text capture converts Analog text-based resources to digital text resources. And then these converted resources can be used in several ways like searchable text in indexes so as to identify documents or images.

As the first stage of text capture a scanned image of a page is taken. And this scanned copy will form basis for all other stages. The very next stage involves implementation of technology Optical Character Recognition for converting text content into machine understandable or readable format.

OCR analysis takes the input as digital image which is printed or hand written and converts it to machine readable digital text format. Then OCR processes the digital image into small components for analysis of finding text or word or character blocks. And again, the character blocks are further broken into components and are compared with dictionary of characters.

Python is a programming language it will provide an environment where this problem can be solved. It has a huge library we can import the library for performing OCR task. A use of python includes analysis, algorithm development, computation and much more It helps us to solve our problem in no time and provides an easy solution.

The OCR text is written into a pure text file that is then imported again to a search engine. The text is used as index searching of the information. Accuracy rates are measured in several ways and the ways they are measured impact the accuracy rate.

Chapter - 3

METHODOLOGY

3.1 COMPONENTS REQUIRED

3.1.1 Raspberry Pi 4

Raspberry pi is a credit card sized single board computer which was firstly developed in UK by raspberry pi foundation. Raspberry pi has total 40 pins in which 27 pins is of GPIO (General purpose input and output) and remaining 13 pins are used for VCC and GND. It is the minicomputer which it has inbuilt operating system, but it requires inbuilt SD card for booting and long-term storage. Due to this drawback one can use desktop computer.

Specification of BCM2837 (ARM Cortex)

- SoC – Broadcom BCM2837 64bit ARMv8 quad core Cortex A53 processor @ 1.2GHz with dual core Video Core IV GPU @ 400 MHz supporting OpenGL ES 2.0, hardware-accelerated OpenCV, and 1080p30 H.264 high-profile decode. Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure.
- System Memory – 1GB LPDDR2.
- Storage – micro SD slot.
- Video & Audio Output – HDMI 1.4 and 4-pole stereo audio and composite video port.
- Connectivity – 10/100M Ethernet, Wi-Fi 802.11 b/g/n up to 150Mbps and Bluetooth 4.1 LE (BCM43438 module).
- USB – 4x USB 2.0 host ports (with better power management, allowing higher power peripherals), 1x micro USB port for power.

FIG 2 Raspberry pi 4 module

3.1.2 Buzzer

Buzzer is an electrical device that makes a buzzing noise and is used for signalling. It produces noisy sound irrespective of the voltage variation applied to it.

Specification

- Provides 3 to 27 volt.
- Provides sound in the range 2 to 4KHZ

FIG 3. Buzzer

3.1.3 Ultrasonic sensor

Ultrasonic sensor is an device that can measures the distance to an object by using sound waves. It measures distance by sending out a sound wave at specific frequency and listening for that sound to bounce back. In this project ultrasonic sensor is used to calculate distance from 2cm to 400cm.Ultrasonic sensor is used to calculate to distance which works on 5v. It has 4 pins that is eco, trigger, ground and VCC. Here eco pin transmit ultrasonic sound waves.

Specification

- Model- HC-SR04.
- Working voltage 5 volt.
- Static current less than 2mA.
- Sensor angle not more than 15 degrees.
- Detection Distance 2cm to 400cm.

FIG 4 Ultrasonic sensor

3.1.4 Raspberry pi Camera

In this project, raspberry camera is used to capture the image. This camera is able to capture an image of 8Mp.

Specification

- Resolution of camera 8 Megapixel.
- It is capable of 2592*1944-pixel static images and also supports 1080P30, 720P60 and 640*480P 60/90 video.

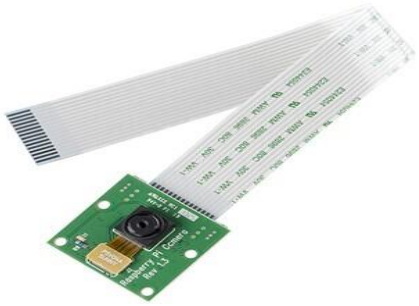

FIG 5. Raspberry pi camera

3.1.5 Stepper motor

A stepper Motor is a rotary actuator or linear actuator that allows precise control of angular or linear position, velocity and acceleration and trigger receives ultrasonic signal by using formula, here 34300(cm/sec) =Sound speed. Stepper motors are controlled by sending a PWM (pulse width modulation) using controls wire. It rotates 0 to 180 degree. It mostly used in Robotic Arm, solar tracking system and Automatic door and gate opener.

Specification

- Operating voltage 2.8 to 6.0 volt.
- Operating speed 0.12sec/60degrees.

FIG 6 Stepper motor

3.1.6 Motor driver (L293D)

The L293D is a popular 16-Pin Motor Driver IC. As the name suggests it is mainly used to drive motors. A single L293D IC is capable of running two DC motors at the same time; also, the direction of these two motors can be controlled independently.

- Contains four half H-Bridges that can operate as two full H-Bridges.
- Operate 2 motors with direction and speed control or 4 motors with speed control only.
- Can supply 600mA current per channel continuous and 1.2A peak.
- 4.5 to 36V motor voltage.
- 5V compatible on logic pins.

FIG 7 Motor driver

3.2 DESIGN & IMPLEMENTATION

The objective of this project is Usage of image authentication technology, Capturing of Vehicle number plate details using camera, unauthorized authentication and alerting through buzzer alarm, Number plate recognition indication even through LED indicator.

3.3 BLOCK DIAGRAM

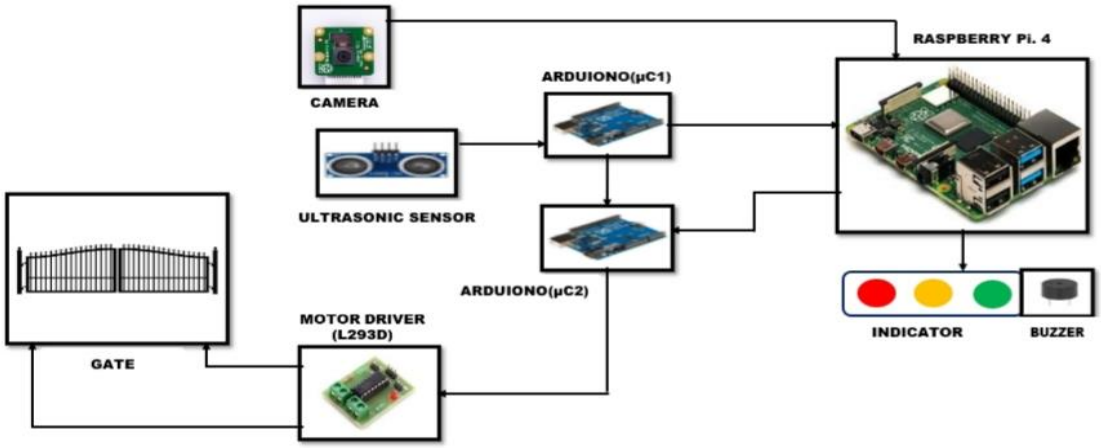

FIG 8 Block diagram

Above figure shows that the block diagram of recognition of vehicle number plate using Raspberry pi 4. In this system Raspberry pi 4 is the heart of project and we have installed Linux operating system some important library and packages have installed to convert image to text like OpenCV OCR. Raspberry pi is SoC device. Here we interface camera to Raspberry pi on a port where we interface camera. The camera is performing main role in this system. When vehicle comes in range with ultrasonic sensor automatically the image of number plate get capture and converts into text using OCR and open CV. Then compare the text into exiting number plate. If number plate gets match servo motor opens the gate else buzzer blows to in built operator that vehicle is unknown. Figure 8 shows the execution of recognition of vehicle number plate using Raspberry pi 4.

3.4 ELECTRONIC CIRCUIT

This project makes use of an on-board computer, which is commonly termed as Raspberry Pi processor. The on-board computer can efficiently communicate with the output and input modules which are being used. The Raspberry Pi is a credit card sized single-board computer among which we are using 3 pins are used for led, buzzer and motor. For running motor, the driver used is L293D.It has 16 pins.3rd and 6th pin are used to rotate motor in clockwise and anticlockwise.4th and 5th pins are grounded. Process flow chart for gate operation is shown in Fig.9

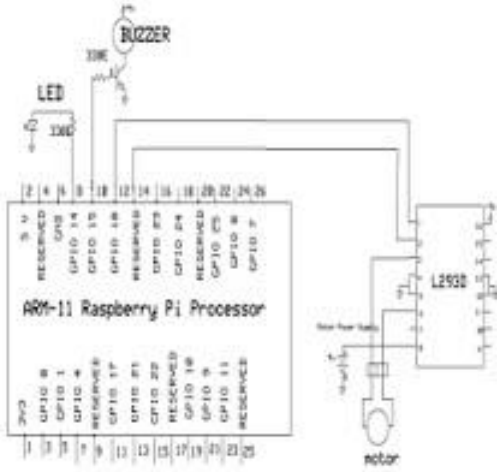

FIG 9 Raspberry pi circuit diagram

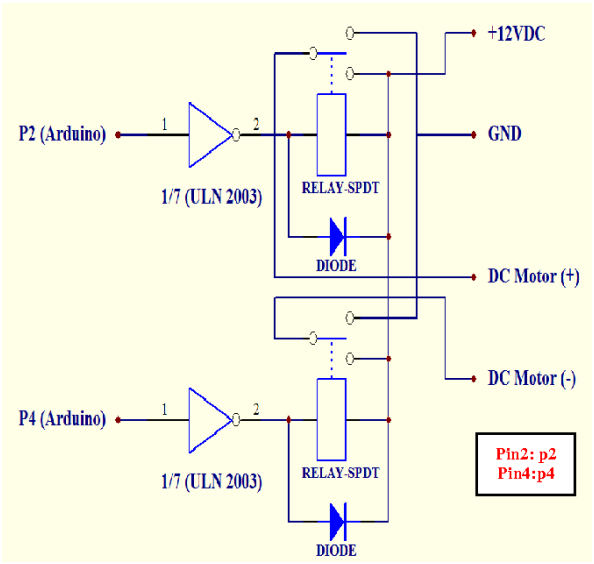

FIG 10 Arduino circuit diagram

The electronic circuit used in the model of proposed system to open, close, and stop the building gate is illustrated in Fig.10

The system procedure begins with a picture captured by a digital camera for the vehicle that stopped before the building gate intending to enter the building. The camera sends a captured picture to the computer. The computer recognition program runs to recognize the vehicle registration number plate it with sorted database. In case of the matching occurred, the system will permit the vehicle to enter, by sending a signal (means open) to the electromechanical part (which is fixed to the gate to control opening and closing the gate), then the gate will open.

Opening the gate: In case of that the patch of the vehicle matches one in database, the computer will sends a signal(5V) to pin2 in the Arduino board which in its role sends a signal(5V) to the ULN2003 in the driver board, the driver sends(5V) to the fist relay, the first relay sends(12V) to the motor which will work to open the gate after getting the current. See Fig.10

Closing the gate: When the time of entering the vehicle to the building through the gate is out, the segment of code that related to closing the gate will be executed in the computer and sends a (5V) signal to the Arduino board which in its role sends a signal to the ULN2003 in the driver board. The ULN2003 sends a signal to the second relay. The second relay sends (12V) to the motor which begins to move to close the gate by switching the polarity of the (12V). See Fig.10

Stopping the motor: When opening time is out, the segment of code related to stop the motor will be execute in the computer and sends a (0V) signal to in the Arduino board which in its role sends these signals to the ULN2003 which sends signals to the first and second relays which in their role stop the motor current and the motor will stop, see Fig.10

3.5 FLOW CHART

When the sensor detects the motion, the raspberry pi camera takes a photo. After that, pi sends a request to open ALPR with the car photo to be identified. Then the open ALPR API returns the car details like: plate number, model, colour, and the confidence of the result.

After identifying a car, we'll do some verifications, and if we found an authorized car, we'll trigger an event (that can be open the garage, for example). The following image contains a flowchart showing the process.

Here's what happens: after the car has been identified by Open ALPR, we'll check if the license plate and the car model match. If they match, we'll check if the car is in the list of authorized vehicles. If it is, we'll trigger an event. For example: open the garage. After that, we wait a determined period of time until the car enters the garage. Then, you need to add several verifications to check if the car has already entered the garage. If yes, you can close the garage.

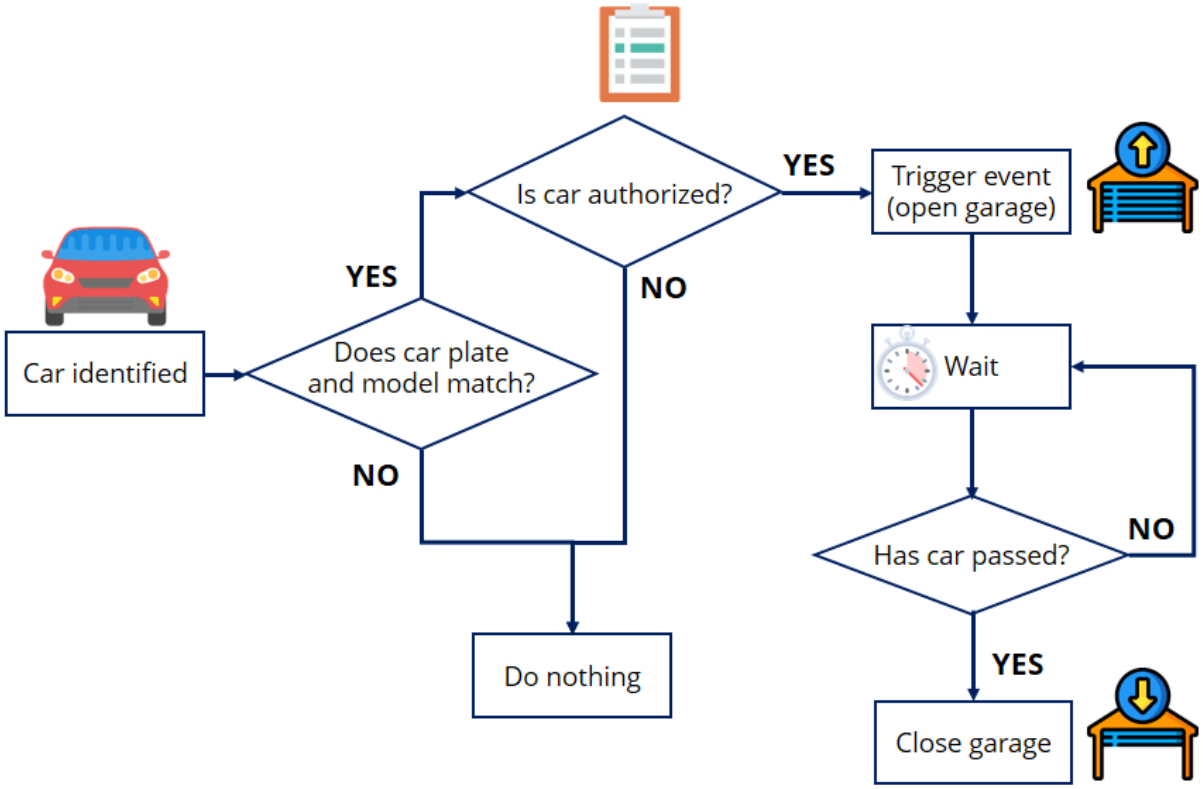

FIG.11 flow chart

Chapter - 4

CODING

4.1 Coding of Raspberry pi

```
#ALPR imports
import requests
import base64
import json
#GPIO imports
import RPi.GPIO as GPIO # Import Raspberry Pi GPIO library
GPIO.setwarnings(False) # Ignore warning for now
GPIO.setmode(GPIO.BOARD) # Use physical pin numbering

#Camera imports
from picamera import PiCamera
from time import sleep

def operate():
   print("This Vehicle is valid!")
   GPIO.output(11, GPIO.LOW)#off red light
   GPIO.output(12, GPIO.LOW)#off orange light
   GPIO.output(13, GPIO.HIGH)#on green light
   GPIO.output(16, GPIO.HIGH)#trigger gate

GPIO.setup(40, GPIO.IN, pull_up_down=GPIO.PUD_DOWN) # Set pin 40 to be an input pin
and set initial value to be pulled low (off)
GPIO.setup(11, GPIO.OUT)#red light
GPIO.setup(12, GPIO.OUT)#orange light
GPIO.setup(13, GPIO.OUT)#green
GPIO.setup(16, GPIO.OUT)#gate
GPIO.setup(15, GPIO.OUT)#buzzer
```

```
GPIO.output(15, GPIO.LOW)#buzzer off
GPIO.output(12, GPIO.LOW)#off orange light
GPIO.output(13, GPIO.LOW)#off green light
GPIO.output(16, GPIO.LOW)#gate trigger off
GPIO.output(11, GPIO.HIGH)#on red light

SECRET_KEY = 'sk_fdc6**********8ac2'#ALPR secret code
camera = PiCamera()



while True:# Run forever
   if (GPIO.input(40) == GPIO.HIGH):
      GPIO.output(11, GPIO.LOW)#off red light
      GPIO.output(12, GPIO.HIGH)#on orange light
      sleep(1)
      camera.start_preview()
      sleep(5)

camera.capture('/home/pi/Number_Plate_Recognition_usng_OpenALPR_API/realtime_ima
ge.jpg')
      camera.stop_preview()


      with open('realtime_image.jpg', 'rb') as image_file:
         img_base64 = base64.b64encode(image_file.read())
      url =
'https://api.openalpr.com/v2/recognize_bytes?recognize_vehicle=1&country=ind&secret_k
ey=%s' % (SECRET_KEY)
      a = requests.post(url, data = img_base64)
      try:
```

```
    s = a.json()['results'][0]['plate']
    print(s)
    if(s=="KA18Z6350"):
        operate()
    elif(s=="KA18S250"):
        operate()
    elif(s=="HR26DK8337"):
        operate()
    elif(s=="num_4"):
        operate()
    else:
        print("Invalid vehicle")
        GPIO.output(11, GPIO.HIGH)#on red light
        GPIO.output(12, GPIO.LOW)#off orange light
        GPIO.output(15, GPIO.HIGH)#buzzer on
        sleep(1)
        GPIO.output(15, GPIO.LOW)#buzzer off
        GPIO.output(13, GPIO.LOW)#off green light
        GPIO.output(16, GPIO.LOW)#gate trigger off

except:
    print("No number plate found")
    GPIO.output(11, GPIO.HIGH)#on red light
    GPIO.output(15, GPIO.HIGH)#buzzer on
    sleep(1)
    GPIO.output(15, GPIO.LOW)#buzzer off
    sleep(1)
    GPIO.output(15, GPIO.HIGH)#buzzer on
    sleep(1)
```

```
        GPIO.output(15, GPIO.LOW)#buzzer off
        GPIO.output(12, GPIO.LOW)#off orange light
        GPIO.output(13, GPIO.LOW)#off green light
        GPIO.output(16, GPIO.LOW)#gate trigger off

else:
    GPIO.output(15, GPIO.LOW)#buzzer off
    GPIO.output(12, GPIO.LOW)#off orange light
    GPIO.output(13, GPIO.LOW)#off green light
    GPIO.output(16, GPIO.LOW)#gate trigger off
    print("no vehicle detected")
    GPIO.output(11, GPIO.HIGH)#on red light
```

Chapter - 5

RESULT

RESULT

Images produced after the Python code has been inserted to the Raspberry Pi 4. as shown in Fig.12

Fig.12 capturing the image

- The code dumped success
- KA18Z6350, KA18S250, HR26DK8337 these registration numbers are verified & as the coloured registration number as shown in fig.12
- It traced all the number plates and showed good result

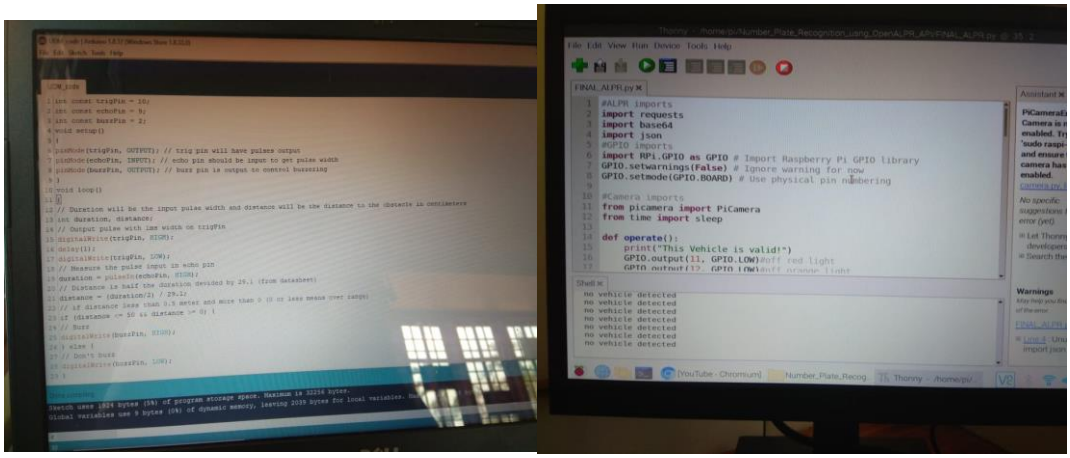

Fig.13complition of coding setup

Chapter – 6

CONCLUSION

CONCLUSION

There are frequent situations in which a system able to recognise registration numbers can be useful. This paper presents few such situations, a system designed to satisfy the requirements, and some experimental results obtained with this system.

The main features of system presented are:

- Controlled stability-plasticity behaviour (optional external supervisory input)
- Controlled reliability threshold (optional external validation input)
- Both off-line and on-line learning
- Self-assessment of the output reliability
- High reliability based on multiple feedback.

The system has been designed using a modular approach which allows easy upgrading and/or substituting of various sub-modules thus making it potentially suitable for a large range of vision applications. The performances of the system make it a valid choice among its competitors especially in those situations when the cost of the application has to be maintained at reasonable levels. Furthermore, the modular architecture makes Visicar extremely flexible and versatile

Chapter-7

REFERENCES

REFERENCES

1. M. Kadam, "Detection of Hidden Object In Speech Based on Fast Fourier Transform Algorithm", Eng. & Tech. Journal, vol.29, No.2, 2011, pp 375- 385.
2. Chirag N. Paunwala, Suprava Patnaik, “A Novel Multiple License Plate Extraction Technique for Complex Background in Indian Traffic Conditions”, In Proceedings of International Journal of Image Processing, vol.4, issue2, 2007.
3. Aishwarya Agarwal, “Automatic License Plate Recognition using Raspberry Pi,” IEEE International Interdisciplinary Conference on Science Technology Engineering Management Singapore, 22nd, 23rd April 2017.
4. Pixel connectivity-wikipedia” http://en.wikipedia.org/wiki/Pixel_connectivity.
5. Satadal Saha1, Subhadip Basu, Mita Nasipuri, Dipak Kumar Basu,” License Plate Localization from Vehicle Images:An Edge Based Multi-stage Approach”, International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.
6. Loumos, V.; Kayafas, E.,”License plate recognition from still images and video sequences: A survey” IEEE Transactions on Intelligent Transportation Systems, volume 9, issue 3, pages 377-391, September 2008.
7. Ganapathy and W.L.D. Lui, "A Malaysian Vehicle License Plate Localization and Recognition System", Journal of Systemic, Cybernetics and Informatics, Vol. 6, No. 1, 2008
8. Wisam Al Faqheri and Syamsiah Mashohor, "A Real-Time Malaysian Automatic License Plate Recognition (M-ALPR) using Hybrid Fuzzy" ,IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.2, February 2009.
9. Jameson, H. S. Abdullah, S. Norul, A. N. Ghazali, N. Nur, and N. A.Zamani, "Multiple Frames Combination Versus Single Frame Super Resolution Methods for CCTV Forensic Interpretation," Journal of Information Assurance & Security, vol. 8, 2013.