A Project report on

VEHICLE NUMBER PLATE DETECTION SYSTEM

A Dissertation submitted to JNTU Hyderabad in partial fulfillment of the academic requirements for the award of the degree.

Bachelor of Technology

In

Computer Science & Engineering

<u>Submitted by</u>

S.SAI KARUN
(14H51A05N7)

Under the esteemed guidance of
Ms.PREETY SINGH
Assistant Professor (guide)

Department of Computer Science & Engineering

CMR COLLEGE OF ENGINEERING ANDTECHNOLOGY
(An Autonomous Institution under UGC &JNTUH , Approved by AICTE, Permanently Affiliated to JNTUH, Accredited by NAAC with 'A' Grade.)
KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.
2014- 2018
CMR COLLEGE OF ENGINEERING & TECHNOLOGY
KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report entitled **" VEHICLE NUMBER PLATE DETECTION SYSTEM"** being submitted by **S.Sai Karun** *(14H51A05N7)* in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** is a record of bonafide work carried out his/her under my guidance and supervision.

The results embodies in this project report have not been submitted to any other University or Institute for the award of any Degree.

Ms. Preety Singh   **Dr. K.Vijaya Kumar**
Assistant Professor   **Professor and HOD**
Dept. of CSE   **Dept. of CSE**

Submitted for viva voice Examination held on   _____

External Examiner

<u>Acknowledgment</u>

With great pleasure I want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project work a grand success.

I am grateful to **Ms. Preety Singh,** Assistant Professor, Dept of Computer Science and Engineering for her valuable suggestions and guidance during the execution of this project work.

I would like to thank **Dr. K. Vijaya Kumar**, Head of the Department of Computer Science and Engineering, for his moral support throughout the period of my study in CMRCET.

I am highly indebted to **Dr. V.A. Narayana**, Principal CMRCET for giving permission to carry out this project in a successful and fruitful way.

I would like to thank the Teaching & Non- teaching staff of Department of Computer Science and Engineering for their co-operation

Finally I express my sincere thanks to **Mr. Ch. Gopal Reddy**, Secretary, CMR Group of Institutions, for his continuous care. I sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

S.Sai Karun

14H51A05N7

Contents

List of Figures

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 Pi 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 Pi 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 Pi 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 are fed as input to the Raspberry Pi processor. The Processor takes responsibility to check the authentication details of every shover and dumper. 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 processor is programmed using embedded 'Raspbian'.

CHAPTER1

INTRODUCTION

1. INTRODUCTION

1.1 Problem Statement

To design an Automated Licence Plate Recognition System Using Raspberry Pi2 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)
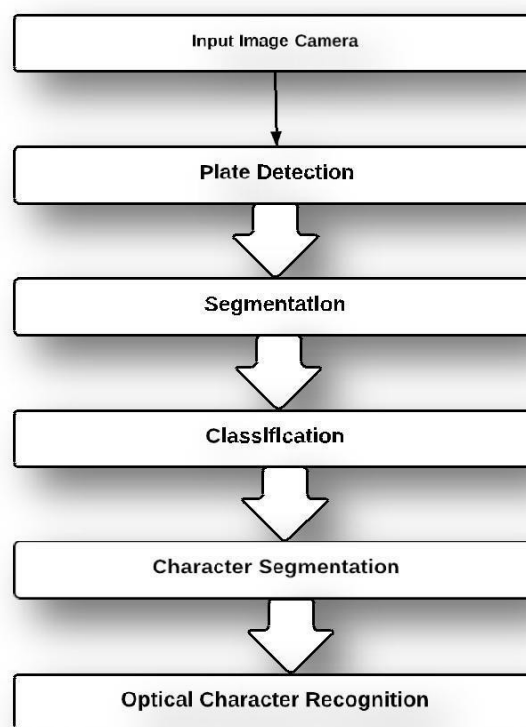
Figure 1.2. ALPR process

- 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 defenseless to clamor but rather is moderate in execution. [3]

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. [2]

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.[8]

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 color, dimension ratios etc.), the algorithm may out not even find the plate in the captured image.

Furthermore, there are wide variety of 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.

Example of some images which may create problem are shown in Fig 2.

Fig: 1.4. (a) Low spatial resolution (too small characters on the plate)

Fig: 1.4. (b) Blurred image

Fig: 1.4. (c) Low contrast image

Fig: 1.4. (d) Overexposure image

Fig: 1.4. (e) image showing Bad lighting conditions (shadow and strong light)

Fig: 1.4. (f) Highly distorted image

Fig1.2: Examples of image which may causer problem during ALPR (

http://www.platerecognition.info/1102.htm**)**

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

2.1.0   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. But before that, we need to do some handy pre-processing of the current image namely:

1. Conversion to Gray Scale: The Red, Green, and Blue components are separated from the 24-bit color value of each pixel. To convert the image captured into Gray Scale code can be written in Python IDLE:

```
img = cv2.imread('2.jpg',0)
```

An example of image converted to grayscale is shown in Fig. 3(a).

2. Blur the Image: The result of blurring an image by a Gaussian functions. Using this technique image looks sharper or more detailed if we are able to perceive all the objects and their shapes correctly in it. Using Code:

```
Blur =cv2.GaussianBlur(img,(5,5),0)
```

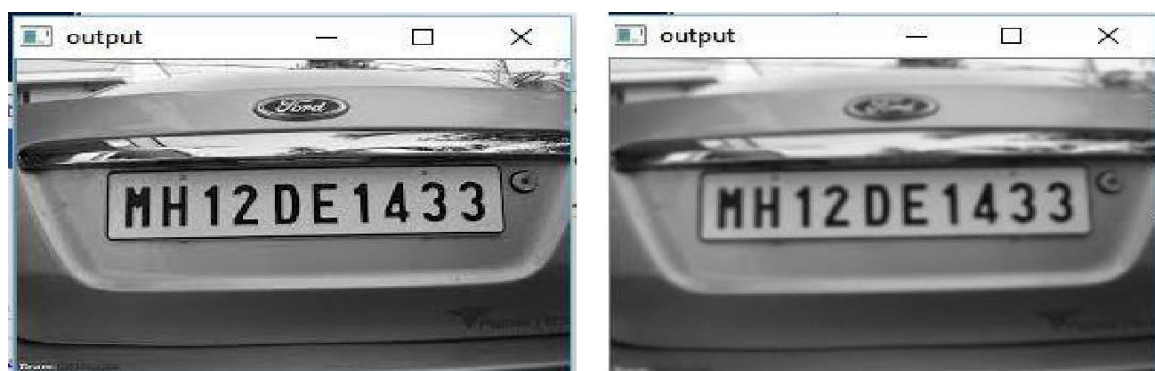An example of image blurring is shown in Fig. 3(b).

Figure 2.1.

(a) Grayscale Image   (b) Blurred Image

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 Pi2. The quality of the image forms an important part of this technique so preprocessing 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 letters 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.

2.3.1 *Edge Detection*

Edge helps to characterize the boundaries and therefore are a problem of fundamental importance while processing the image. Edges in images are the areas where strong intensity contrasts are present ,a sudden variation in the intensity from one pixel to the next Detecting the edges of an image significantly reduces the amount of data and it helps in filtering out the useless information, while preserving the important structural properties of an image. There are many ways to perform the edge detection. However, the majority of various methods can be grouped in to two different categories, gradient and laplacian. The gradient methods detect the edges by finding out the maximum and minimum in the first derivative of the image. The Laplacian method searches for the zero crossings in the second derivative of the image to find the edges. An edge has the one dimensional shape of the ramp and calculation the derivative of the image can highlight its location. Suppose we have the signal shown below in Figure 4.a. If we take the gradient of this given signal we get the following shown in Figure 4.b.[11]
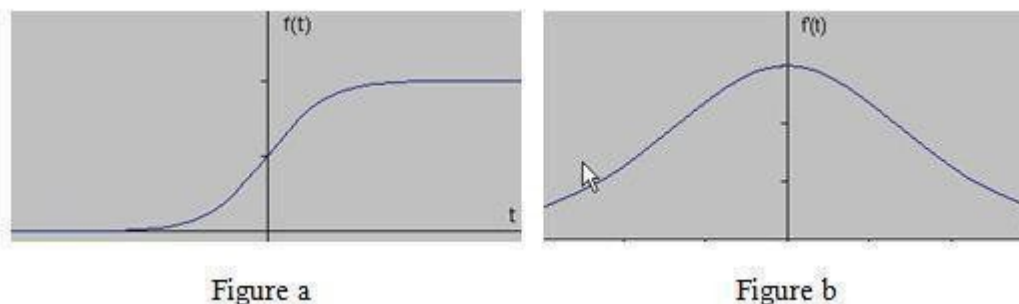

Figure a   Figure b

Fig 2.3.1: trend of signal and its derivative

As shown, the derivative shows maximum located at the centre of the edge in the original signal. Such a method of finding an edge is characteristic of the gradient filter family of the edge detection filters and it includes the Sobel method .A pixel location is declared as an edge location if the value of the gradient is exceeding some threshold. The edges will have higher pixel intensity values than their surrounding pixels. So once a threshold has been set, we can compare the gradient value of the threshold value and an edge can be detected whenever the threshold is exceeded. Also, when the first derivative is at a maximum, the second derivative is turns out to be zero.

2.3.2 Sobel Filter

The theory of the one-dimensional analysis can be carried over to two –dimensions as long as there an accurate approximation for calculating the derivative of a two-dimensional image. The Sobel operator helps to perform a 2 dimensional spatial gradient measurement on an image. Generally, it is used to find the approximate absolute gradient magnitude at each point on an input grayscale image .The Sobel edge detector make use of a pair of 3X3 convolution masks, one estimates the gradient in the x-direction(columns) and the other estimates the gradient in the y-direction(rows). [11]

Using Python code As one of the consequences, the inherent errors introduced by the edge detection mechanism are typically below 5% (1-2 pixels for a character size of 40x100 pixels) and therefore not significant for the subsequent processing.

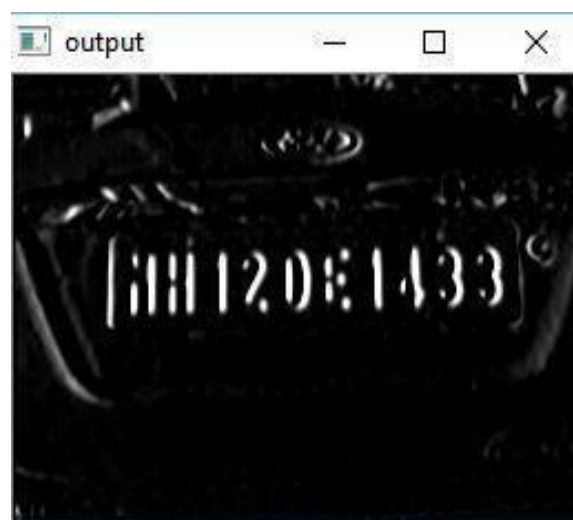sobelx=cv2.Sobel(blur, cv2.CV_8U, 1, 0, ksize=3). An image with output from sobel filter is shown in Fig 5.

Figure 2.3.2. Output of the image from Sobel Filter

After a Sobel Filter, we apply a threshold filter to obtain a binary image with a threshold value
obtained through Otsu's Method.Otsu's algorithm needs an 8 -bit input image and Otsu's meth

od automatically determines the optimal threshold value. An image with output as Threshold image from otsu's algorithm is shown in Fig 6.
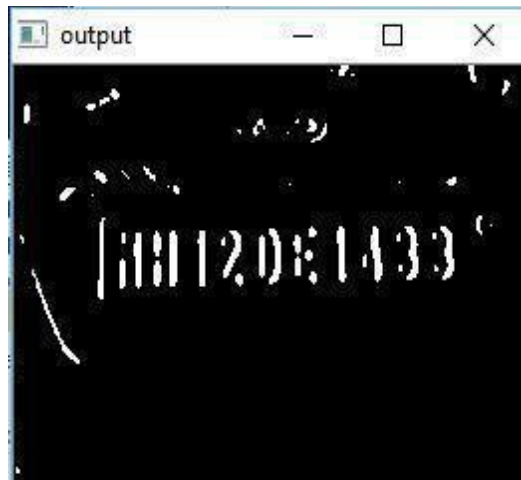

Figure 2.3.2. Threshold Image

2.3.3 *Contours*

The minimum or the smallest bounding or the enclosing box for any point set in N dimensions is the box with the smallest measure within which all points lie. When the other kinds of measure are used, the minimum box is usually called accordingly depending on the measure used.eg. "Minimum-perimeter bounding box". The minimum bounding box of any point set is same as the minimum bounding box of its convex hull, this is a fact which can be used heuristically to speed up computation. The term "box"/"hyper rectangle" has come from its usage in Cartesian coordinate system, where it can be indeed visualized as a rectangle, rectangular parallelepiped etc. In the case of two-dimensional it is called the minimum bounding rectangle. In other words it is a rectangle which has the minimum height and which that covers all the pixels present in a particular connected component or region[13]. An image with contouring of captured image is shown in Fig.7.

Figure 2.3.3. Contouring in the captured Image

2.3.4 Selecting the best Contours

In a figure there can be many bounding boxes identified but to select the best possible candidate for license plate region requires to identify few properties of the license plates as discussed below:

1. Contrast present in the Contours: As the license plate consists of dark coloured numbers to the lighter background .The centre row of the box can be scanned and total number sudden contrast change can be recorded, if the image is binarized. The box having the maximum changes in contrast within the box is the possible candidate for the license plate.[12]

2. Aspect Ratio: The aspect ratio of an image is the ratio of the width of the image to its height. The inverse of the aspect ratio should be less than 1 for any license plate. Hence, all the regions which don't satisfy this property can be rejected.[12]

3. Width of license plate: The width of the license plate region also has a threshold limit.it cannot be more than some threshold value. In this project after analysing various dataset we have used the threshold limit.

4. Total number of pixels in License plate region: Another factor that separates license plate region from the rest is the amount of pixel that it has in it. In project, the distance in between the camera is triggered when the Shovel-Dumper hits the laser. After analysis various dataset we have come to conclusion that the minimum threshold for total number of pixels in the license plate region is $1/100^{th}$ of the total number of pixels in the image and maximum threshold as $1/100^{th}$ of the total number of pixels in image plus 1500 pixels.
An image with contour satisfying properties of license plate is shown in Fig. 8.

Figure 2.3.4. Contour satisfying the properties of license plate.

2.3.5 Cropping the Contour

After identifying the best possible Contour candidate for the license plate the coordinates of the contour are noted and the box is cropped from the image and sent to character segmentation module for the further processing as shown in figure 9:

2.4 Principles of Character Segmentation

The character segmentation process acts as bridge between plate detection and optical character recognition modules. Its main function is to segment the characters in the selected candidate region (extracted license plate) such that each character can be sent to the optical character recognition module individually for recognition.

Normalized or standardized are of a fancy format the conditions of the license plates are important criteria for efficient segmentation because if numbers are of a fancy format the conditions of the license plate as described .Once the license plate is localized we proceed to obtain the individual characters .A license plate as described above has high intensity variation regions. This forms the basis for character segmentation. Sometimes it is observed that along with license numbers, various texts may be present, which have to be removed. By various observation we observed that for the license plate regions the amount of white on black is specific for the number regions and falls within a certain range[13].

Morphological technique are used to remove small white areas which escape range corrections. Finally individual characters are extracted to pass on through the optical character recognition system. Segmentation is one of the most important processes in the automatic number plate recognition, because all further steps rely on it. If the segmentation fails, a character can be improperly divided in two pieces, or two characters can be wronged merged together which would lead to the failure of following stages of recognition.

The second phase of the segmentation is an enhancement of segments. The segment of a plate contains besides the character also undesirable elements such as noise due to shadows or defects in camera equipment as well as redundant space on the sides of characters[1].

2.4.1 Preprocessing stage

Before we can proceed with the segmentation stage, we must ensure that the plate obtained is cleared off most of the unwanted characters or graphics like state name or flags etc. We proceed to do so by scanning the plate vertically and horizontally and ignoring those rows and colums which have too much white and black. This is justified as those areas containing the numbers have black areas which be in a particular range .This range by experiments was found to be between 0.2 to 0.8 times the number of pixels horizontally and vertically.

The character segmentation process takes the extracted license plate region from the preceding module as the input .The input is a coloured JPEG image. For our process, we work only binary images and thus the first part of segmentation is binarisation of the image as shown in Fig 10. Fig 11 shows character segmented image and Fig 12 shows peaks of character in image.

Figure 2.4.1. [A] Sample localized Plate   [B] Binarised Image

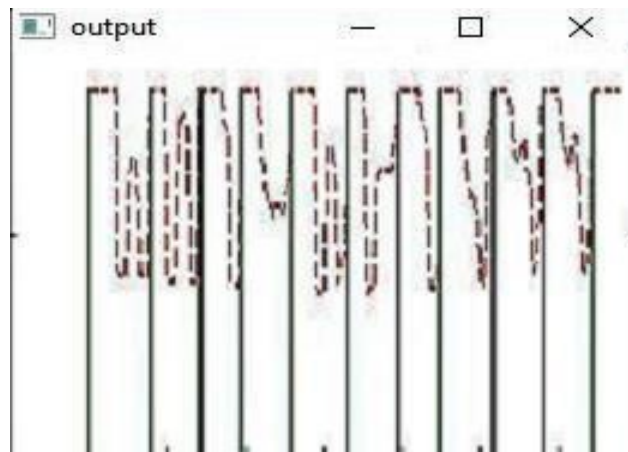**Figure 2.4.1. Character Segmented Image**

Figure 2.4. Horizontal projection showing peaks between characters

2.5 The Image Deskewing Mechanism

Quite often the localized license plate obtained from the above steps are skewed or rotated by a certain angle. This can be attributed to either:

- The camera angle during exposure.

- Orientation of the vehicle with respect to the desired camera.

Either way, Image skew could severely hamper subsequent steps of character extraction and recognition. It is thus desired to deskew the image before passing the localized plate on the next LPR step. Below is a sample image illustrating the need of deskewing techniques.

The above skewed image could result from either of the above two reasons. If this skew is left uncorrected it could lead to the failure of segmentation and extraction steps. Fig 13 shows the resultant extracted characters thus rendering segmentation process ineffective.[1]
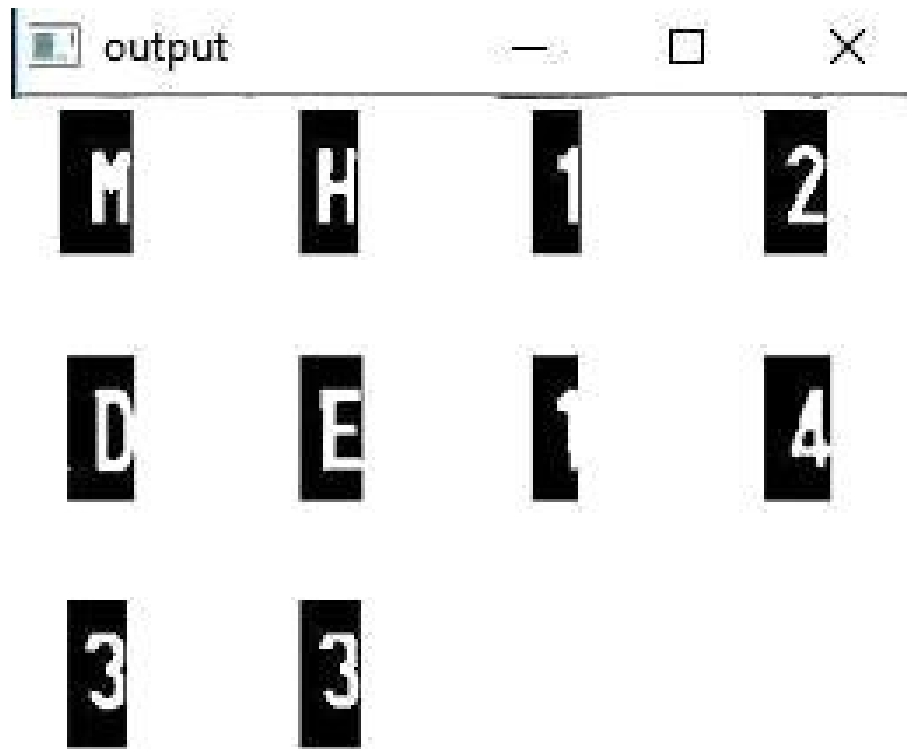
Figure 2.5. Individual Characters Obtained as output.

2.6 Optical Character Recognition

2.6.1 Introduction

The neural networks are typically made up of many artificial neurons. An artificial neurons. An artificial neuron is an analogy to biological neuron. It is simply electronically modelled to the biological neuron. The number of neurons that are used depends on the task at hand. The number of neurons used can be few as two or three or large as two or several thousand. There are many ways of connecting artificial neurons together to form a neural network. Some of the ways are discussed below.

- Feedforward network:

  In feedforward Neural Network each input into the neuron has its own weight associated with it. A weight can simply be a floating point no. and it is these that we adjust when we come to train the network. The weights in most of the neural networks can be both negative and positive, therefore, it helps in providing excitory or else inhibitory influences to each input. As each of the input enters the nucleus it is then multiplied by its weight. The nucleus sums up all these new input values and gives us the activation which is again a floating point no. and can be negative or positive. The threshold value is decided and if the activation value is greater than a threshold value the neuron outputs 1(considering these are two outcomes 1 and 0 to the input) as a signal. If the activation is less than the threshold value the neuron then outputs zero.

  A neuron can take any number of inputs from one to n, here n is the total number of inputs. The inputs, therefore, may be represented as x1,x2,x3…xn. The corresponding weights for the inputs can be represented as w1,w2,w3,w3..wn. The weighted sum of

the links and its corresponding weights is called the activation value as discussed above.

$a = x1w1+x2w2+x3w3…xnwn$[14]

where, a is the activation value.

Each of the input is sent to every neuron of the hidden layer and each hidden layers neurons output is connected to every neuron of the next layer. There is no predefined number of the neurons to be present in a particular layer it can be arbitrary and it totally depends on the problem[14]. An image with functionality of neural networking is shown in Fig 14.
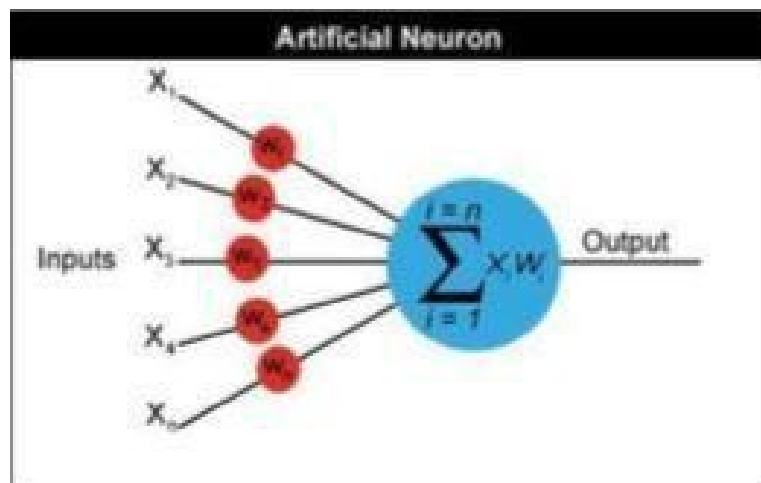

Figure 2.6.1. Functional diagram of Neural Networking

 Back Propagation Network Algorithm: A back propagation networks learns by example various sets of datasets are provided as input. The various inputs provided helps the network to calculate and recalculate the networks weight value so that when the network is trained it can give the required output. Fig 15 shows declaration of
layers for neural networking.

The network is initialized by first setting random weights which generally have very small value such as values between -1 and 1. There are two passes in the Back Propagation Algorithm. After the networks is setup with the random weights the
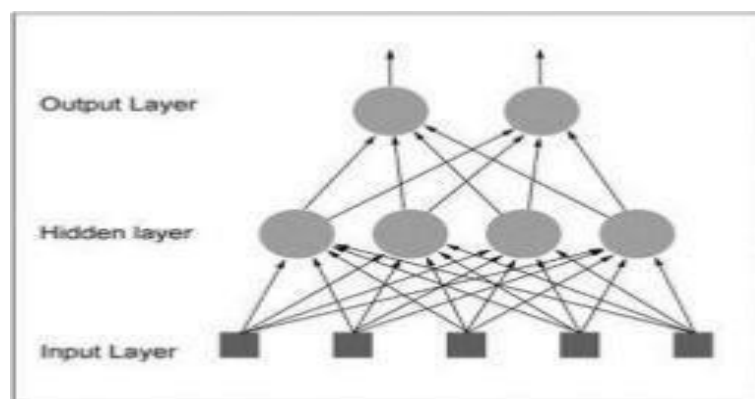

Figure 2.6.1. Layers Declaration for neural networking

output is calculated this is called the forward pass the result obtained in the forward pass may not be equal to the required result or the target and so the error is calculated for each neuron which is Target-Actual Output. The error calculated for each neuron is then mathematically used to change the weights so the next time the forward pass will have minimum the error[15].

The character is recognized after training the network with various datasets of the particular character to get maximum accuracy and minimum error.

CHAPTER 3

METHODOLOGY

3. METHODOLOGY

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. 16.
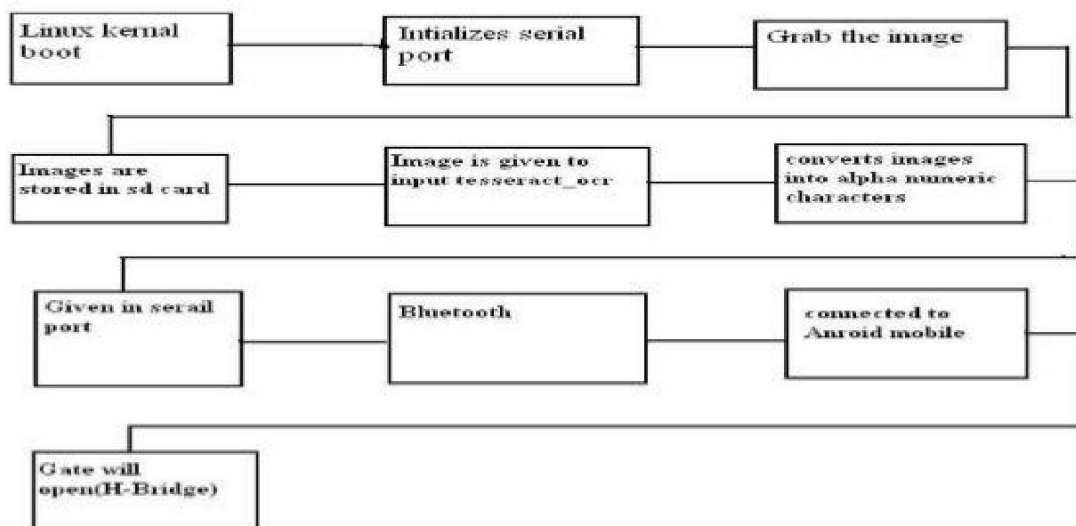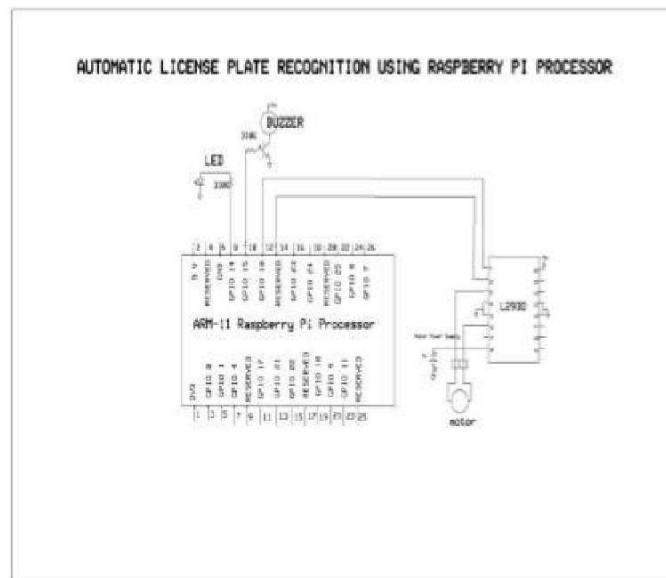
Fig.3.1. Process Flow/ Flowchart explaining gate operation

3.1 Overview of Raspberry Pi2

The ARM1176JZF-S processor incorporates an integer core that implements the ARM11 ARM architecture v6. It supports the ARM and Thumb™ instruction sets, Jazelle technology

to enable direct execution of Java bytecodes, and a range of SIMD DSP instructions that operate on 16-bit or 8-bit data values in 32-bit registers. The ARM1176™ applications processors deployed broadly in devices ranging from smart phones to digital TV's to eReaders, delivering media and browser performance, a secure computing environment, and performance up to 1GHz in low cost designs. The ARM1176JZ-S processor features ARM TrustZone technology for secure applications and ARM Jazelle technology for efficient embedded Java execution. Optional tightly coupled memories simplify ARM9™ processor migration and real-time design while AMBA 3 AXITM interfaces improve memory bus performance. DVFS support enables power optimization below the best-in-class nominal static and dynamic power of the ARM11TM processor architecture. An overview of Raspberry pi2 is shown in Fig.17.

Figure 3.1. Overview of Raspberry

Pi2 *3.2 Experimental Setup*

The RPi.GPIO library comes pre-installed with the latest version of Raspbian. In case all need to do is to install the latest version from the repositories by running the following in the terminal (holds good for Raspbian only). Fig 18 & Fig 19 shows experimental setup required for Raspberry pi2.
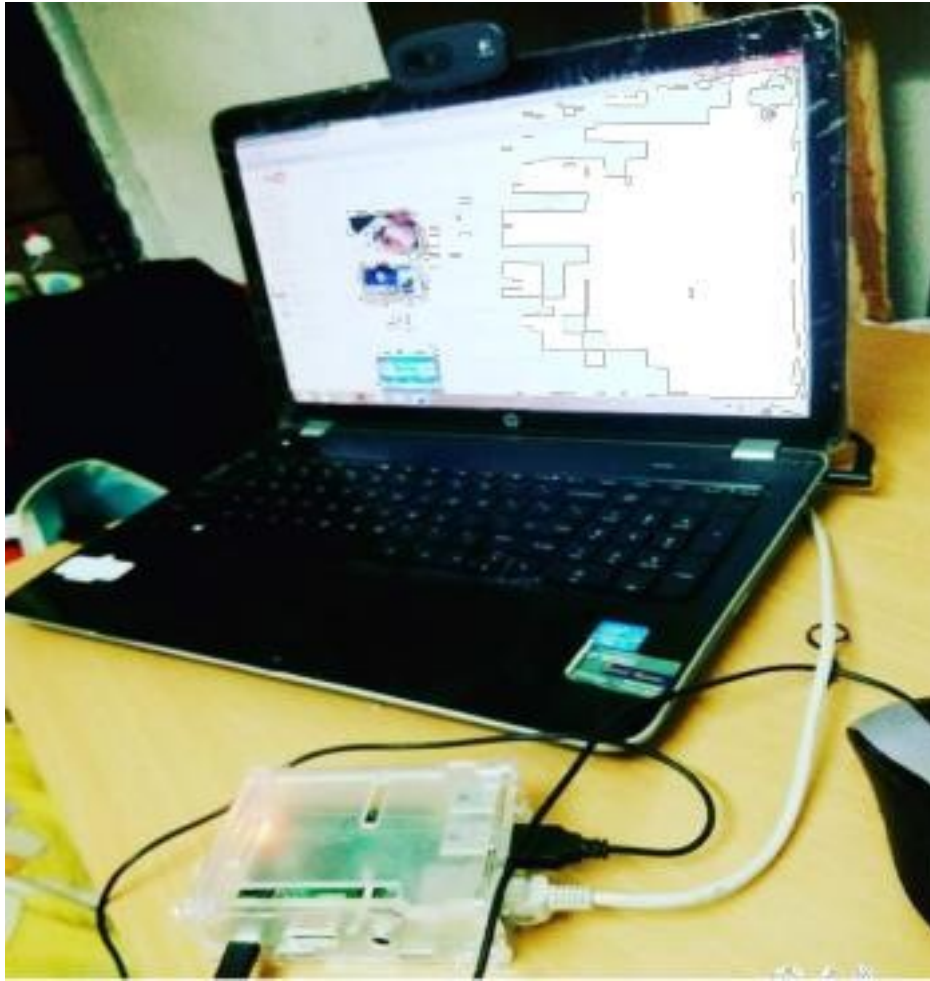

Figure 3.2.1. Experimental Setup

Figure 3.2.2. Setup with the Camera Calibration

CHAPTER 4

RESULTS AND

DISCUSSIONS

4.1 Sample Data and Results:

Images produced after the Python code has been inserted to the Raspberry Pi2 as shown in Fig 20.
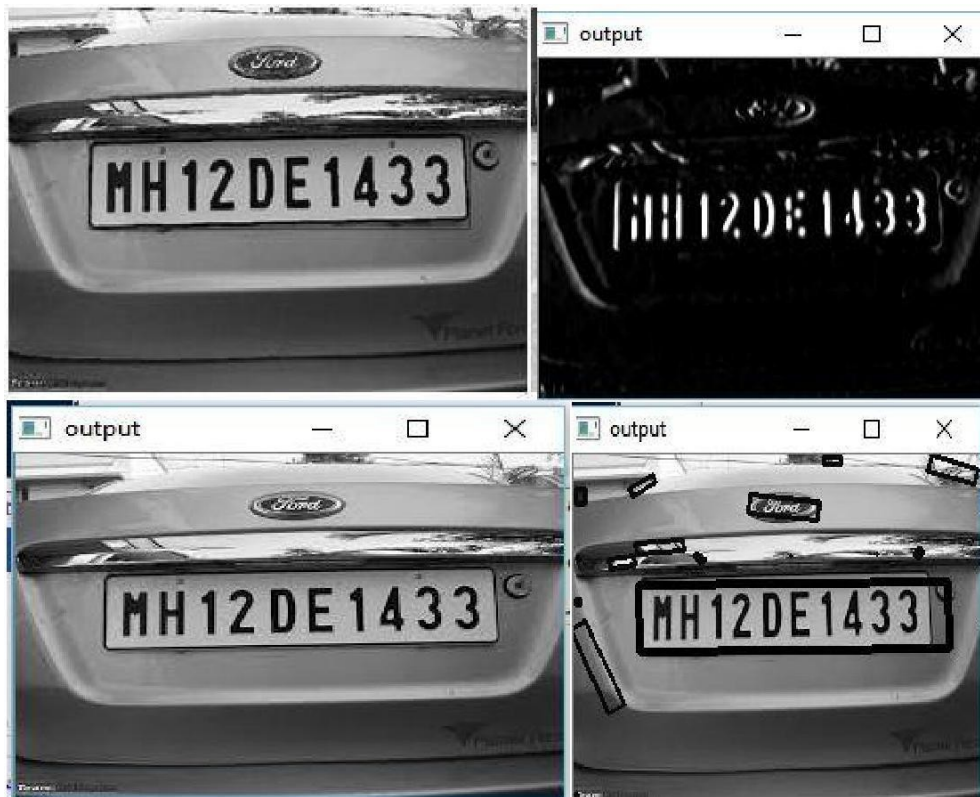

Figure 4.1. Sample data

4.2 VNC Server

Installing the VNC server on pi allows to see the raspberry pi's desktop remotely, using the mouse and keyboard as if we were sitting right in front of pi. It also means that I can put pi anywhere else in home, but still control it. Also, internet can be shared from laptop's WiFi over Ethernet. This also lets access internet on the pi and connect raspberry pi to laptop display.

Step 1: Setting up Raspberry Pi

Before moving to connect raspberry pi to laptop display, we need an SD card having the OS preinstalled. I will find lots of blogs and tutorials about preparing an SD card for the Raspberry Pi2... This will show how to install the OS for the raspberry pi.

After setting up SD Card, insert it into the raspberry pi. Next, for powering the pi connect my micro USB cable to it. Also connect my raspberry pi to the laptop via an Ethernet cable. And connect the keyboard & mouse to it. Now, connect the HDMI display (the HDMI is only required for running the pi for the first time). Now power on my Pi. And follow the connect to raspberry pi to laptop display.

Step 2 : Sharing internet over ethernet

This step explains how I can share my laptop internet with the raspberry pi via Ethernet cable.

In Windows : For sharing internet to multiple users over Ethernet, go to Network and Sharing Center. Then click on the WiFi network as shown in Fig 21:

Fig 4.2: Process of sharing internet to multiple users

Click on Properties (shown below), then go to Sharing and click on "Allow other network users to connect". Make sure that networking connection is changed to "Local Area Connection" as shown in Fig 22 & 23:
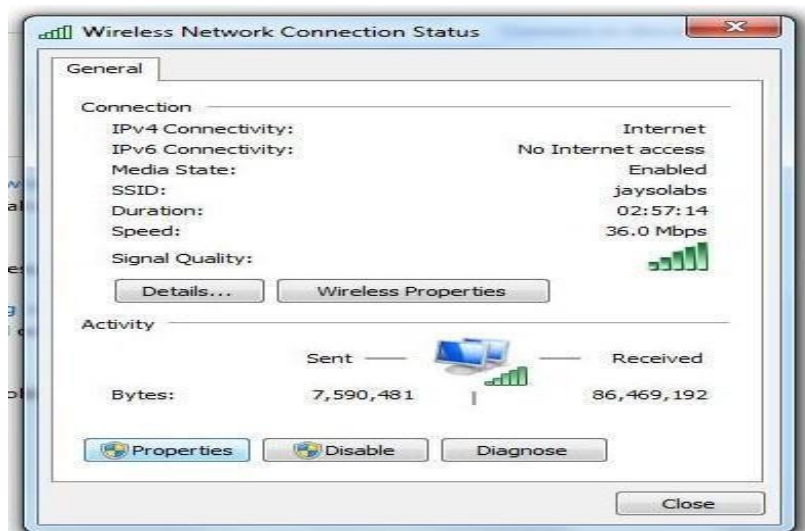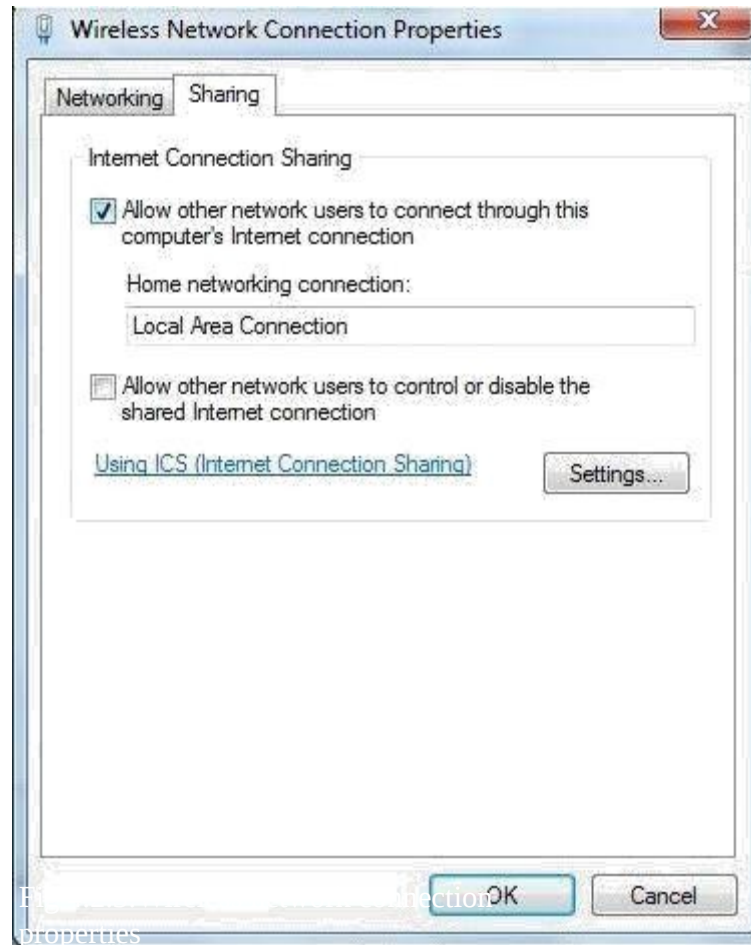

Fig.4.2.2: Wireless network connection status

Wireless Network Connection
properties

View your basic network information and set up connections

See full map

DELL-PC
(This computer)

Multiple networks

Internet

View your active networks

Connect or disconnect

jaysolabs
Home network

Access type: Internet
HomeGroup: Joined
Connections: Wireless Network Connection
(jaysolabs)

Unidentified network
Public network

Access type: No network access
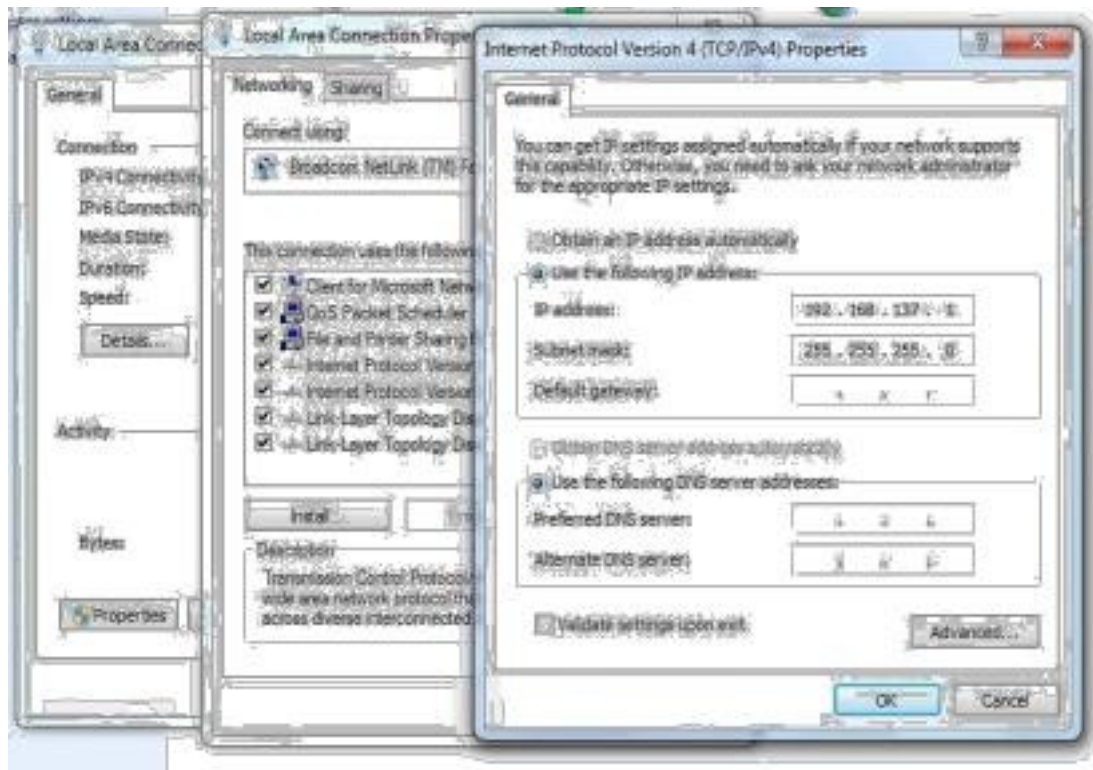Connections: Local Area Connection

Fig 4.2.4: Assigning IP to connected laptop

As shown in Fig 25, the IP assigned to my laptop is 192.168.137.1. For checking the IP assigned to the connected ethernet device, do the following. Considering that IP assigned to my Laptop is 192.168.137.1 and subnet mask is 255.255.255.0 :

- Open command prompt.

 • Ping on broadcast address of my IP. (Type) Eg: ping 192.168.137.255   •
Stop the ping after 5 seconds.
  ◻ Check the reply from device: arp –a

Step 3 : Setting up the VNC server to connect Raspberry Pi to laptop display

If I have an HDMI display: Using the connected HDMI display on my pi, I should install VNC server in raspberry pi. Open the LX-Terminal and type the following commands to install VNC:

$sudoapt-getupdate
$ sudo apt-get install tightvncserver

If I don't have an HDMI display: If I do not have a display even for one time setup, then no need to worry. Install Putty as per my windows configuration and via SSH I can connect with my raspberry pi. And, as I get access of my pi terminal, run the same commands as above to install VNC.

Starting VNC Server on Pi :

For starting VNC, enter the following command in SSH terminal: `$ vncserver :1`

Step 4 : Setting up the client side (Laptop)

Download VNC client and install it. When I first run VNC viewer was seen as shown in Fig 26.
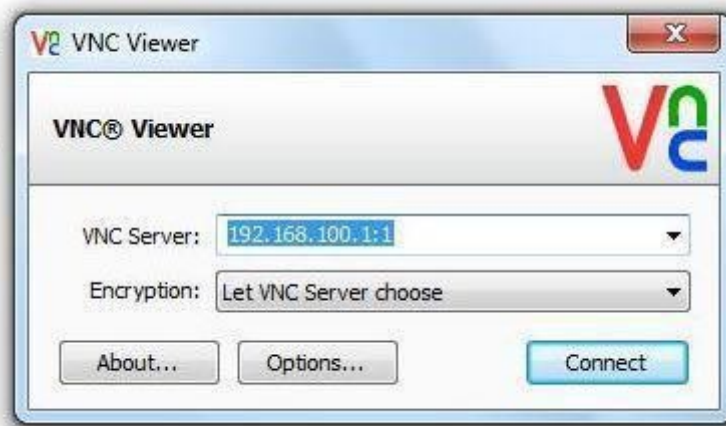

Fig 4.2.5: VNC Viewer

Enter IP address of my raspberry pi given dynamically by my laptop (I got the address from the earlier step). And append with :1 (denoting port number) and press connect. I got a warning message, press 'Continue' as shown in Fig 27:
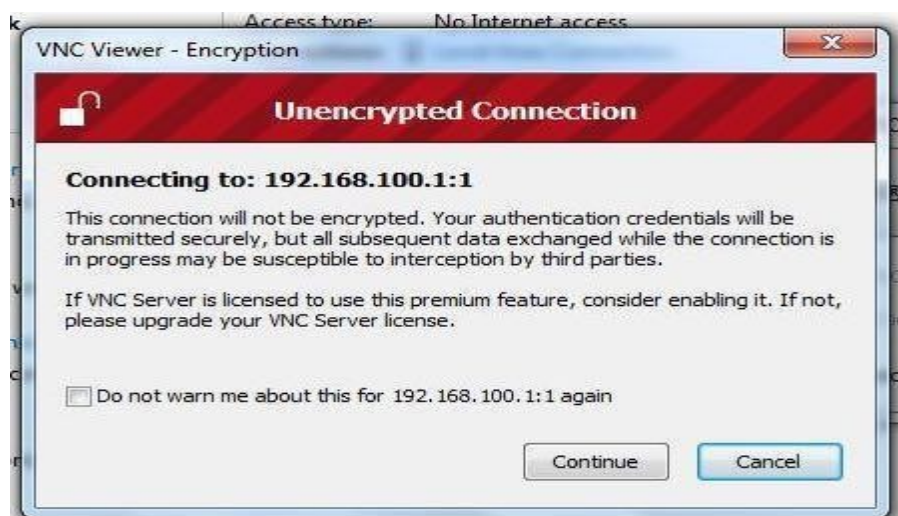

Fig 4.2.6: Warning message window

Enter the 8 digit password which was entered in VNC server installation on raspberry pi as shown in Fig 28:
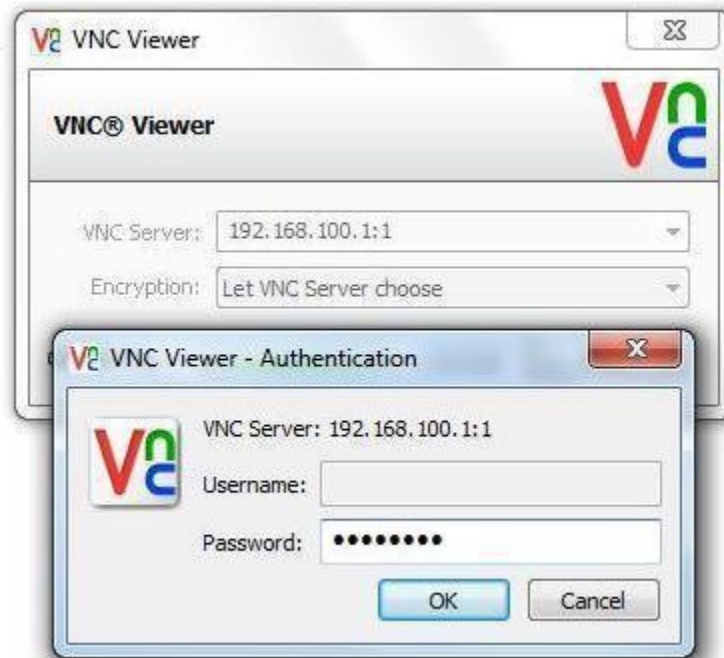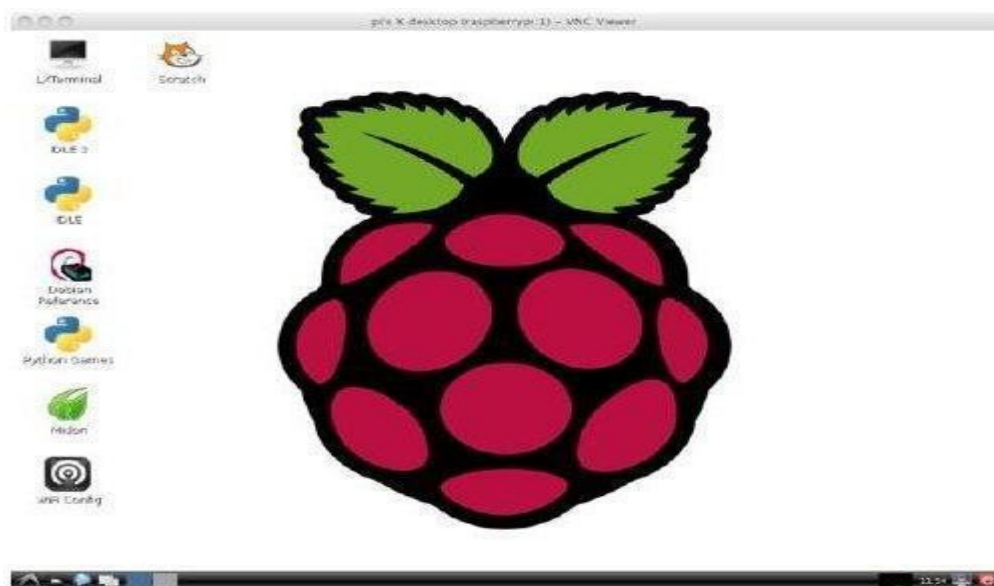

Fig 4.2.7: Server installation on raspberry pi

Finally, the raspberry pi desktop itself should appear as a VNC window as shown in Fig 29. I will be able to access the GUI and do everything, as if I were using the pi's keyboard, mouse and monitor directly. As with SSH, since this is working over my network, my pi could be situated anywhere as long as it is connected to my network.

Step 5 : Running VNC server at startup in Raspberry Pi

Connecting to my raspberry pi remotely withFi VNC is fine as long as my pi does not reboot. If it does, then I either have to connect with SSH and restart the VNC Server. Else, arrange for the VNC Server to run automatically after the raspberry pi reboots. To ensure that VNC starts automatically each time on booting up, run the following commands on terminal:

Open ".config" folder from pi's: user folder (it is a hidden folder).

Now next time reboot Pi, vncserver will start automatically. And seamlessly would connect raspberry pi to laptop display.

CHAPTER 5

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, Visicar, and some experimental results obtained with this system.[5]

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 makes 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 6

REFERENCES

REFERENCES

[1] 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.

[2] Saeed Rastegar, Reza Ghaderi, Gholamreza Ardeshipr & Nima Asadi, " An intelligent control system using an efficient License Plate Location and Recognition Approach", International Journal of Image Processing (IJIP) Volume(3), Issue(5) 252, 2009. [3] V. Ganapathy and W.L.D. Lui, "A Malaysian Vehicle License Plate Localization and Recognition System", Journal of Systemics, Cybernetics and Informatics, Vol. 6, No. 1, 2008.

[4] 3D surface tracking and approximation using Gabor filters, Jesper Juul Henriksen, South Denmark University, March 28, 2007

[5] O. Martinsky, "Algorithmic and Mathematical Principles of Automatic Number Plate Recognition System", B. Sc. Thesis, BRNO University of Technology, 2007.

[6] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Pearson Education Asia, 2002.

[7] 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

[8] Satadal Saha, Subhadip Basu, Mita Nasipuri, Dipak Kumar Basu "An Offline Technique for Localization of License Plates for Indian Commercial Vehicles"

[9] Fukunaga K.: Introduction to statistical pattern recognition, Academic Press, San Diego, USA, 1990

[10] Gonzalez R., Woods R.: Digital Image Processing, Prentice Hall, Upper Saddle River, New Jersey, 2002

[11] "Edge Detection Tutorial" http://www.pages.drexel.edu/~weg22/edge.html

[12]" Pixel connectivity-wikipedia" http://en.wikipedia.org/wiki/Pixel_connectivity

[13]"Minimum bounding box wikipedia"http://en.wikipedia.org/wiki/Minimum_bounding_box

[14] "ai-junkie" http://www.ai-junkie.com/ann/evolved/nnt2.html

[15]"The Back Propogation Algorithm" http://www4.rgu.ac.uk/files/chapter3%20-%20bp.pdf

[16] T. L. Chien, H. Guo, K.L. Su and S.V. Shiau, "Develop a Multiple Interface Based Fire Fighting Robot," IEEE International Conference on Mechatronics, May 2007.

[17] K. L. Su, "Automatic Fire Detection System Using Adaptive Fusion Algorithm for Fire Fighting Robot," IEEE International Conference on Systems, Man and Cybernetics,Vol.2006.
[18] T. L. Chien, H. Guo, K.L. Su and S.V. Shiau, "Develop a Multiple Interface Based Fire Fighting Robot," IEEE International Conference on Mechatronics, May 2007.

[19] J.H. Park, B.W. Kim, D.J. Park and M.J. Kim, "A system architecture of wireless communication for fire-fighting robots," Proceedings of the 17th World Congress The

International Federation of Automatic Control, July 2008.

[20]Benjamin C. Kuo, Step Motors and Control Systems, SRL Publishing Company, Champagne, IL, 1979