AUTOMATED TRAFFIC CONTROL SYSTEM WITH EMERGENCY VEHICLE OVERRIDE USING ARDUINO UNO

Dissertation submitted in fulfillment of the Degree of

Bachelor of Technology

UNDER THE GUIDANCE OF

Mr. Pardeep Garg

Submitted By

Girish Verma: 121054

Rashi Taneja: 123006

Akshay Modi: 123008

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY,

CERTIFICATE

This is to certify that project report entitled **"AUTOMATED TRAFFIC CONTROL SYSTEM WITH EMERGENCY VEHICLE OVERRIDE USING ARDUINO UNO"**, submitted by **Girish Verma (121054), Rashi Taneja (123006) and Akshay Modi (123008)** in fulfilment for the award of degree of Bachelor of Technology in Electronics and Communication Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date:

Supervisor's Name: Mr. Pardeep Garg

Designation: Assistant Professor
Department of ECE

Signature:

ACKNOWLEDGEMENT

We owe a great many thanks to a great many people who have helped and supported us during this project. Our deepest thanks to **Mr. Pardeep Garg (Assistant Professor)**, our project guide for his exemplary guidance, monitoring and constant encouragement throughout the course of this project work. He has taken great pains to go through the project and make necessary corrections as and whenever needed. We are also grateful to **Mr. Mohan (ECE Project lab**) for his practical help and guidance. We would also like to thank all the faculty members of ECE department without whom the progress of this project would have been a distant reality. We also extend our heartfelt thanks to our family and well-wishers.

Date:

Name of the students:
Girish Verma (121054)
Rashi Taneja (123006)
Akshay Modi (123008)

List of Figures

Abstract

Traffic congestion is a severe problem in many major cities across the world and it has become a nightmare for the commuters in these cities. The project is designed to develop a density based dynamic traffic signal system. The signal timing changes automatically on sensing the traffic density at the junction.

Conventional traffic light system is based on fixed time concept allotted to each side of the junction which cannot be varied as per varying traffic density. Junction timings allotted are fixed. Sometimes higher traffic density at one side of the junction demands longer green time as compared to standard allotted time. The image captured in the traffic signal is processed and converted into grayscale image then its threshold is calculated based on which the contour has been drawn in order to calculate the number of vehicles present in the image.

After calculating the number of vehicles we will came to know in which side the density is high based on which signals will be allotted for a particular side. Besides,

This system is implemented to make human work easier besides can reduce the uses of human power and because of its potential application. The development of automatic car license plate recognition system will have resulted greater efficiency for vehicle monitoring system. Car plate recognition systems are used commercially, both in overseas and locally. This system is implementing by using MATLAB7.1 Image Processing Toolbox, which uses optical character recognition on images to read the license plates on vehicles. The system is an online system where the image will automatically have extracted once after the image is captured by webcam using image processing technique. First, the image is converted into a binary image and then the chosen area will be cropped so that only the plate number is left. Next, the image is compliment so that the black plate background becomes white while the white plate number becomes black because the system can only detect binary image where the background should be white while the plate number should be black. One of the important step is the integration between image processing and Graphical User Interface (GUI) where, the output of this project will displayed using GUI.

TABLE OF CONTENTS

CHAPTER -1

INTRODUCTION

In modern life we have to face with many problems one of which is traffic congestion becoming more serious day after day. It is said that the high volume of vehicles, the inadequate infrastructure and the irrational distribution of the development are main reasons for increasing traffic jam. The major cause leading to traffic congestion is the high number of vehicle which was caused by the population and the development of economy. Traffic congestion is a condition on road networks that occurs as use increases, and is characterized by slower speeds, longer trip times, and increased vehicular queuing. The most common example is the physical use of roads by vehicles. When traffic demand is great enough that the interaction between vehicles slows the speed of the traffic stream, these results in some congestion.

As demand approaches the capacity of a road (or of the intersections along the road), extreme traffic congestion sets in. When vehicles are fully stopped for periods of time, this is colloquially known as a traffic jam or traffic snarl-up. Traffic congestion can lead to drivers becoming frustrated and engaging in road rage.

In traffic environments, Traffic Sign Recognition (TSR) is used to regulate traffic signs, warn the driver, and command or prohibit certain actions. A fast real-time and robust automatic traffic sign detection and recognition can support and disburden the driver, and thus, significantly increase driving safety and comfort. Generally, traffic signs provide the driver various information for safe and efficient navigation.

Automatic recognition of traffic signs is, therefore, important for automated intelligent driving vehicle or driver assistance systems. However, identification of traffic signs with respect to various natural background viewing conditions still remains challenging tasks.

Real time automatic vision based traffic light control has been recently the interest of many researchers, due to the frequent traffic jams at major junctions and its resulting wastage of time. Instead of depending on information generated by costly sensors, economic situation calls for using available video cameras in an efficient way for effective traffic congestion estimation.

Thus, given a camera sequence, the task of vision based traffic light control list:

1) Analyze image sequences

2) Estimate traffic congestion and

3) Predict the next traffic light interval. We may focus on one or more of these tasks, and they may also choose different measures for traffic structure or add measures.

For more comprehensive review on vision based traffic light control Due to the massive growth in urbanization and traffic congestion, intelligent vision based traffic light controller is needed to reduce the traffic delay and travel time especially in developing countries as the current automatic time based control is not realistic while sensor based traffic light controller is not reliable in developing countries. Traffic congestion is now considered to be one of the biggest problems in the urban environments. Traffic problems will be also much more widely increasing as an expected result of the growing number of transportation means and current low-quality infrastructure of the roads.

In addition, many studies and statistics were generated in developing countries that proved that most of the road accidents are because of the very narrow roads and because of the destructive increase in the transportation means. This idea of controlling the traffic light efficiently in real time has attracted many researchers to work in this field with the goal of creating automatic tool that can estimate the traffic congestion and based on this variable, the traffic sign time interval is forecasted.

The heart of any system is the processing unit which controls the working of various hardware components. In our project we are using Arduino UNO as the processing unit. The Arduino UNO is a Microcontroller board based on the ATmega328. Arduino is an open source electronics prototyping platform based on flexible, east-to-use hardware and software. The microcontroller board is programmed using the Arduino programming language under an open-source license; we are free to adapt them according to our needs. Arduino UNO microcontroller board based on the ATmega328 are then programmed as per our need to perform the desired operations.

The Arduino UNO Microcontroller Development Board was selected as it satisfies all the requirements. As was fundamental to the course, the assembly language was utilized for the project. The ARDUINO programmer is based on the processing IDE and uses a variation of the C and C++ programming languages. It was more than adequate to satisfy design objectives while enhancing level of understanding of the programming language.

Another aspect of this project is to present a practical design of an automatic system for opening a gate without mounting any signal transmitter on the car. License plate recognition (LPR) is an image processing technology used to identify vehicles by their license plates. This project's goal is to build a practical prototype system, which is capable of recognizing a license plate number from a standard license plate, in case of any emergency or VIP needs. The system will be based on a personal computer and software packages available such as MATLAB and a digital camera that helps in capturing images of vehicles. The software recognizes the plate number, compares the plate number with a built-in database, and decides whether a vehicle is allowed for emergency override or not. The general algorithm involves the following steps: Image capturing which can be achieved by a digital camera. Plate localization and extraction to obtain the vehicle plate sub image. Character segmentation to determine exactly where characters exist inside the plate. Recognition which identifies the numbers contained in the plate. Evaluating the performance of the algorithm. Designing a database to store the numbers of authorized vehicles that are allowed to enter the parking. Designing a graphical user interface (GUI) to simplify the interaction with the software.

1.1 MOTIVATION

a) Smart City Mission launched by PM of India under Ministry of Urban development.
b) To fulfill the demands of Smart city model
c) In modern life we have to face many problems one of which is traffic congestion, thus solution to this problem will solve many interlinked issues like fuel consumption, time management, ease of transportation and environmental pollution.
d) Core infrastructure elements in a Smart City include "efficient urban mobility and transportation system".

1.2 SCOPE OF THE PROJECT

This project is a basic step towards smart city plan of India. In today's world where basic electronics is finding its place in many days to day fields like home automation, car automation, automatic water storage system and so on, this will also take us one step further in Smart City plan. Recently government of India is spending round about 2 crores per city on 98 cities in India which are competing to become India's first Smart city thus this concept of traffic light control and transferring the traffic congestion scenarios of the city on real time basis on devices like smart phones can prove to be a good use of technology to solve day to day problems. The scope of this project is not just limited till controlling of traffic light timings based on density of vehicles but can go further by using GSM technology to transfer traffic density information to the mobile phones of travelers or controlling traffic based on priority.

CHAPTER- 2

Background Information

In this section of the report, background information concerning the project will be provided.

2.1 PROBLEM DEFINITION AND LIMITATION

Researchers now are so much interested in automatic real-time traffic congestion estimation tool as it is the most significant factor on which intelligent transportation systems are based. Some of the researchers have focused in their work on traffic flow estimation. It is measured as the rate at which vehicles pass a fixed point (e.g. vehicles per minute). They used spot sensors such as loop detectors and pneumatic sensors to quantify the traffic flow however; the sensors are very expensive and need a lot of maintenance especially in developing countries because of the road ground de-formations.

In addition, metal barriers near the road might prevent effective detection using radar sensors. It is also found that traffic congestion also occurred while using the electronic sensors for controlling the traffic. In contrast, video based systems are much better compared to all other techniques as they provide more traffic information and they are much more scalable with the progress in image progressing techniques. This is the main reason for the motivation to develop vision based tool for traffic light control in this work.

In recent years, vision based traffic light control, which is based on video processing for traffic flow or traffic density estimation, has attracted the attention of many researchers. The value of traffic density measures only the ratio between the density of the vehicles and the total density of the road.

So based on this measure, the traffic control system will compare between different roads in the intersection to take the decision for the traffic light and the time interval given.

However, most of the previous vision based monitoring systems suffered from lack of robustness on dealing with continuously changing environment such as lighting conditions, weather conditions and unattended vehicles. All these mentioned factors considerably affect the traffic

density estimation. Changes in lightening conditions and weather conditions have been tackled in many of the previous approaches and they are going to be considered also in our proposed approach, but the problem that has never been addressed before and has a significant effect on the traffic pace is the stationary vehicles, especially the unattended ones. The problem with the traffic density measurement is that the traffic density of a road with stationary or unattended vehicles is the same as the traffic density of a road with no stationary vehicles.

Traffic flow counts the number of vehicles that passes through the frame during a certain time interval. However, it may give an empty road a higher priority than a congested road, because fewer vehicles are passing though the given point in that empty road. Therefore, we will concentrate on the detection of the delayed and unattended vehicles in the proposed approach for computing more informative metric about the traffic congestion in order to have more effective way of traffic. This metric is very similar to the traffic density, but with taking the traffic flow into consideration. So it can be considered as a combination of both traffic density and traffic flow.

CHAPTER- 3

HARDWARE AND SOFTWARE DESCRIPTION

This section will outline the requirements of the project. As well, the components will be explained.

3.1 Requirements

3.1.1 Hardware

a) Arduino UNO

Arduino is an open source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, hobbyists, and anyone interested in creating interactive objects or environments. It's an open-source physical computing platform based on a microcontroller board, and a development environment for writing software for the board and we can see the image of Arduino Uno in figure 2.1[12].

Figure 1: ARDUINO UNO

In simple words, Arduino is a small microcontroller board with a USB plug to connect to your computer and a number of connection sockets that can be wired up to external electronics, such as motors, relays, light sensors, laser diodes, loudspeaker, microphones, etc., They can either be powered through the USB connection from the computer or from a 9V battery. They can be controlled from the computer or programmed by the computer and then disconnected and allowed to work independently. Anyone can buy this device through online auction site or search engine. Since the Arduino is an open-source hardware designs and create their own clones of the Arduino and sell them, so the market for the boards is competitive. The name "Arduino" is reserved by the original makers. However, clone Arduino designs often have the letters "duino" on the end of their name, for example, Freeduino or DFR duino. The software for programming your Arduino is easy to use and also freely available for Windows, Mac, and LINUX computers at no cost.

Microcontroller can be described as a computer embedded on a rather small circuit board. To describe the function of a microcontroller more precisely, it is a single chip that can perform various calculations and tasks, and send/receive signals from other devices via the available pins. Precisely what tasks and communication with the world it does, is what is governed by what instructions we give to the Microcontroller. It is this job of telling the chip what to do, is what we refer to as programming on it. However, the microcontroller by itself cannot accomplish much; it needs several external inputs: power, for one; a steady clock signal, for another. So typically, a microcontroller is used along with a circuit which provides these things to it; this combination is called a microcontroller board. The Arduino Uno that we are using is one such microcontroller board. The actual microcontroller at its heart is the chip called Atmega328. The advantages that Arduino offers over other microcontroller boards are largely in terms of reliability of the circuit hardware as well as the ease of programming and using it.

Open-source hardware shares much of the principles and approach of free and open-source software. The founders of Arduino wanted people to study their hardware, to understand how it works, make changes to it, and share those changes with the world. To facilitate this, they release all of the original design files (Eagle CAD) for the Arduino hardware. These files are licensed under a Creative Common Attribution Share-Alike license, which allows for both personal and commercial derivative works, as long as they (people) credit Arduino and release their designs

under the same license. The Arduino software is also open source. The source code for the Java environment is released under the GPL and the C/C++ microcontroller libraries are under the LGPL.

- **Arduino Pin Diagram**

Pin diagram of Arduino Uno is shown in figure.

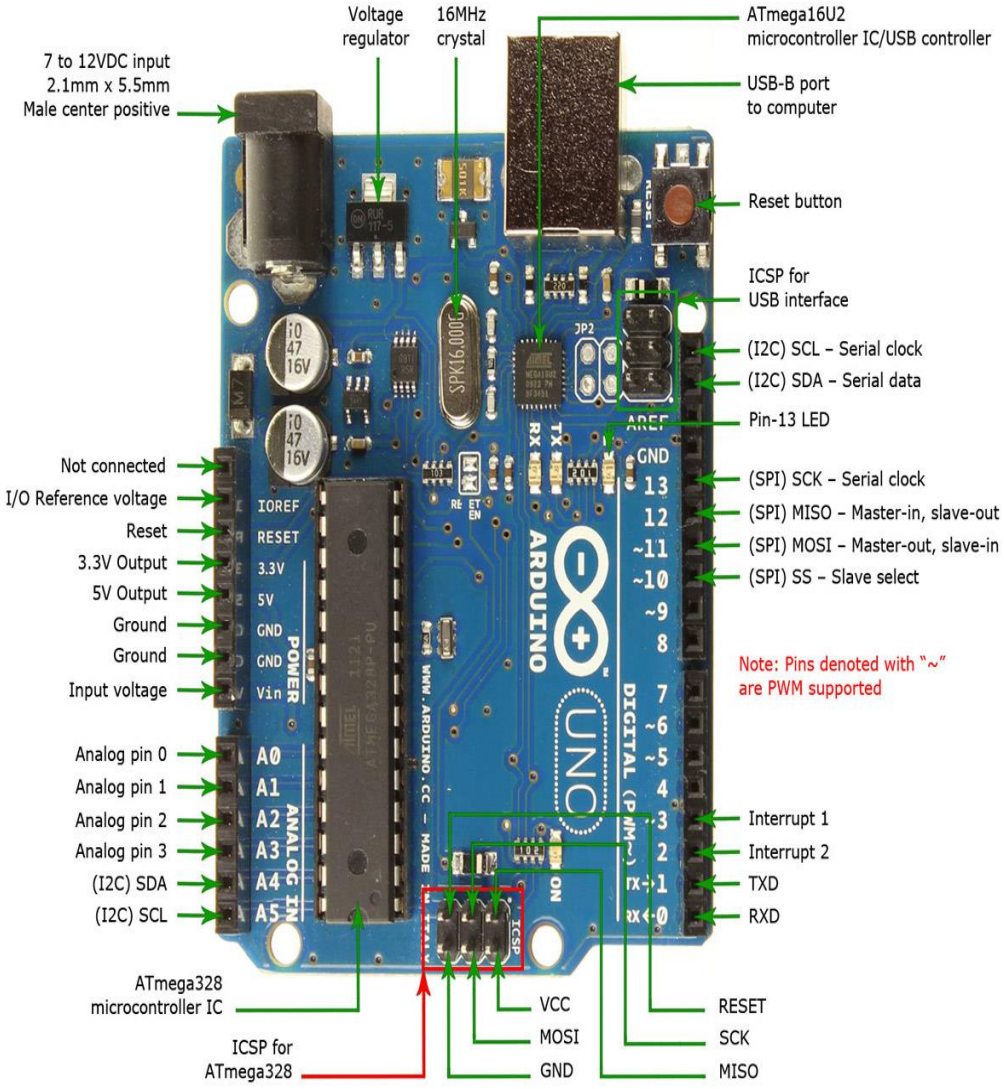

Figure 2: Arduino pin diagram

b) Microcontroller

The ATmega328 is a single chip microcontroller created by Atmel as shown in figure. As of 2013 the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models. A common alternative to the ATmega328 is the "Pico Power" ATmega328P. A comprehensive list of all other member of the megaAVR series can be found on the Atmel website.

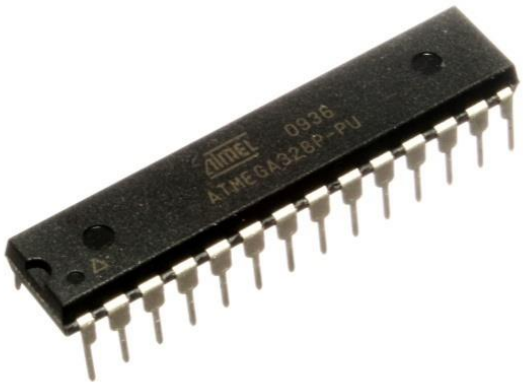

Figure 3: ATmega328 (Microcontroller)

The Atmel 8-bit AVR RISC-based microcontroller combines 32 kB flash memory with read-while-write capabilities, 1 kB EEPROM, 2 kB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter , programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. The device achieves throughput approaching 1 MIPS per MHz

c) Power Supply

The Arduino's on-board regulator can actually handle up to 20V or more, so you can actually use an adapter that puts out 20V DC. The reasons you don't want to do that are twofold: you'll lose most of that voltage in heat, which is terribly inefficient. Secondly, the nice 9V pin on the Arduino board will actually be putting out 20V or so, which could lead to potential disaster when you connect something expensive to what you thought was the 9V pin. Our advice is to stick with the 9V or 12V DC adapter.

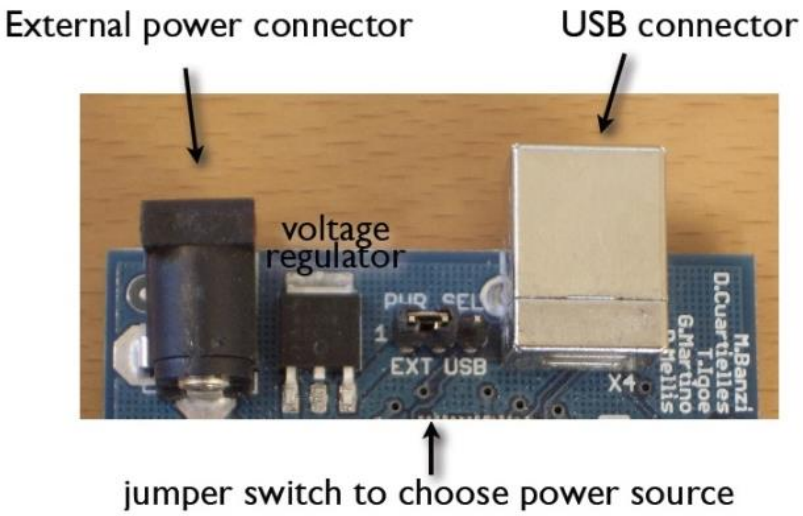

Figure 4: External Power

The power requirement for ARDUINO is 9 to 12V DC, 250mA or more, 2.1mm plug, center pin positive.

The OFF-the shelf adapter

- must be a DC adapter (i.e. it has to put out DC, not AC)
- should be between 9V and 12V DC
- must be rated for a minimum of 250mA current output, although you will likely want something more like 500mA or 1A output, as it gives you the current necessary to power a servo or twenty LEDs if you want to.
- must have a 2.1mm power plug on the Arduino end, and
- The plug must be "centre positive", that is, the middle pin of the plug has to be the + connection.

Current rating: Since you'll probably be connecting other things to the Arduino (LEDs, LCDs, servos) you should get an adapter that can supply at least 500mA, or even 1000 mA (1 ampere). That way you can be sure you have enough juice to make each component of the circuit function reliably.

- LED and wires
- Breadboard
- Webcam
- PC

3.1.2 Software

The software used by the Arduino is Arduino IDE .he Arduino IDE is a cross-platform application written in Java, and is derived from the IDE for the Processing programming language and the Wiring project. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and is also capable of compiling and uploading programs to the board with a single click. There is typically no need to edit make files or run programs on a command-line interface. Although building on command-line is possible if required with some third-party tools such as ino.

The Arduino IDE comes with a C/C++ library called "Wiring" (from the project of the same name), which makes many common input/output operations much easier. Arduino programs are written in C/C++, although users only need define two functions to make a runnable program.

a) MATLAB

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language. A proprietary programming language developed by Math Works, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces including C and interfacing with programs written in other languages like C++, Java, Fortran and Python.

CHAPTER- 4

Methodology

Digital image processing encompasses processes whose inputs and outputs are images and, in addition, encompasses processes that extract attributes from images, up to and including recognition of the individual objects. [3] An image may be defined as two-dimensional function f (x, y), where x and y are spatial (plane) coordinates, and the amplitude off at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image. The field of digital image processing refers to the processing of the digital images by means of digital computer. A digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements or pixel.

4.1 Image Acquisition

Images are generated by the combination of an "illumination" source and the reflection or absorption of energy from that source by the elements of the "scene" being imaged. An image sensor is a device that converts an optical image into an electronic signal. It is used mostly in digital cameras, camera modules and other imaging devices. Early analog sensors were video camera tubes, most currently used are digital charge-coupled device (CCD) or complementary metal–oxide–semiconductor (CMOS) active pixel sensors. Most digital still cameras use either a CCD image sensor or a CMOS sensor. Both types of sensor accomplish the same task of capturing light and converting it into electrical signals. A CCD image sensor is an analog device. When light strikes the chip it is held as a small electrical charge in each photo sensor. The charges are converted to voltage one pixel at a time as they are read from the chip. Additional circuitry in the camera converts the voltage into digital information. A CMOS imaging chip is a type of active pixel sensor made using the CMOS semiconductor process. Extra circuitry next to each photo sensor converts the light energy to a voltage. Additional circuitry on the chip may be included to convert the voltage to digital data. A digital camera (or digicam) is a camera that takes video or still photographs by recording images on an electronic image sensor. Most cameras sold today are digital and digital cameras are incorporated into many devices ranging from PDAs and mobile

phones (called camera phones) to vehicles. Digital and film cameras share an optical system, typically using a lens with a variable diaphragm to focus light onto an image pickup device. The diaphragm and shutter admit the correct amount of light to the imager, just as with film but the image pickup device is electronic rather than chemical. However, unlike film cameras, digital cameras can display images on a screen immediately after being recorded, and store and delete images from memory. Many digital cameras can also record moving video with sound. Some digital cameras can crop and stitch pictures as well as perform other elementary image editing.

4.2 Image Enhancement

Image enhancement comprises the algorithms that make necessary changes to the original images so that they can be made more useful for further processing. It eliminates the noise introduced in the images either during acquisition or during digitization. Apart from this, it is used to make the lighting adjustments in the images that are taken under unfavorable lighting conditions such as during gloomy day or in a poorly illuminated room. With regards to this project, image enhancement has been implemented to eliminate the noise using the median and low-pass Gaussian filter and adjust the shortcomings of lighting conditions using the histogram equalization technique.

I) Median Filter

In median filtering, the neighboring pixels are ranked according to brightness (intensity) and the median value becomes the new value for the central pixel. Median filters can do an excellent job of rejecting certain types of noise, in particular, "shot" or impulse noise also called "salt and pepper" in which some individual pixels have extreme values. In the median filtering operation, the pixel values in the neighborhood window are ranked according to intensity, and the middle value (the median) becomes the output value for the pixel under evaluation. There is no reduction in contrast across steps, since output values available consist only of those present in the neighborhood (no averages). Median filtering does not shift boundaries, as can happen with conventional smoothing filters (a contrast dependent problem). Since the median is less sensitive than the mean to extreme values (outliers), those extreme values are more effectively removed. The median is, in a sense, a more robust "average" than the mean, as it is not affected by outliers (extreme values). Since the output pixel value is one of the neighboring values, new "unrealistic" values are

not created near edges. Since edges are minimally degraded, median filters can be applied repeatedly, if necessary.

II) Gaussian Filter

Edges and other sharp transitions (such as noise) in the gray levels of an image contribute significantly to the high-frequency component of its Fourier transform. Hence, smoothing is achieved in the frequency domain by attenuating a specified range of high-frequency components in the transform of a given range. 16 Basic model of the filtering in the frequency domain is given by the following equation:

$$G(u, v) = H(u, v)\, F(u, v)$$

F (u, v) is the Fourier transform of the image to be smoothed. H (u, v) is the filter transfer function. G (u, v) is output of the process. Gaussian low-pass filter is one of the filters used to eliminate the edges and other sharp transitions (such as noise) in the gray levels from an image. The form of the Gaussian low-pass filtering in the 2-D is given. D (u, v) is the distance from the origin of the Fourier transform, which is assumed to be shifted to the center of the frequency rectangle.

III) Histogram Equalization

Histogram equalization is a method in image processing of contrast adjustment using the image's histogram. This method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values. This allows for areas of lower local contrast to gain a higher contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. The method is useful in images with backgrounds and foregrounds that adjustment, the intensities can be better distributed on the histogram.

4.3 Morphological Processing

The basic morphological transformations are called dilation and erosion, and they arise in a wide variety of contexts such as removing noise, isolating individual elements, and joining disparate

elements in an image. Morphology can also be used to find intensity bumps or holes in an image and to find image gradients.

I) Dilation

Dilation is a morphological transformation of the image that causes bright regions within an image to grow. Dilation is a convolution of some image (or region of an image), say A, with some kernel, say B. The kernel, which can be any shape or size, has a single defined anchor point. Most often, the kernel is a small solid square or disk with the anchor point at the center. The kernel can be thought of as a template or mask, and its effect for dilation is that of a local maximum operator. As the kernel B is scanned over the image, the maximal pixel value overlapped by B is computed and the image pixel under the anchor point is replaced with that maximal value. This causes bright regions within an image to grow. This growth is the origin of the term "dilation operator".

II) Erosion

Erosion is the converse of dilation. The action of the erosion operator is equivalent to computing a local minimum over the area of the kernel. Erosion generates a new image from the original using the following algorithm: as the kernel B, similar to the one mentioned above, is scanned over the image, the minimal pixel value overlapped by B is computed and the image pixel under the anchor point is replaced with that minimal value. This causes the narrow connected regions in the image to diminish. In general, whereas dilation expands region A, erosion reduces region A. Moreover, dilation will tend to smooth concavities and erosion will tend to smooth away protrusions. 18 Of course, the exact result will depend on the kernel, but these statements are generally true for the filled convex kernels typically used.

4.4 Segmentation

Segmentation subdivides an image into its constituent regions or objects. The level to which subdivision is carried depends on the problem being solved. That is, segmentation should stop when the object of interest in an operation has been isolated. In this project, the thresholding algorithms have been extensively used for the purpose of segmentation.

- Thresholding

Consider an image, f (x, y), composed of light objects on a dark background, in such a way that object and background pixels have gray levels grouped into two dominant modes. One obvious way to extract the objects from the background is to select a threshold T that separates these modes. Then any point (x, y) for which f (x, y)>T is called an object point; otherwise, the point is called a background point. When T depends only on f (x, y) (that is, only gray-level values) the threshold is called global. If T depends on both f (x, y) and p (x, y), which is some local property of the point (x, y), the threshold is called local. If, in addition, T depends on the spatial coordinates x and y, the threshold is called dynamic or adaptive.

Basic Global Thresholding

The simplest of all thresholding technique is to partition the image histogram by using a single global threshold, T. Segmentation is then accomplished by scanning the image pixel by pixel and labelling each pixel as object or background, depending upon whether the gray level of that pixel is greater or less than the value of T. The success of this method depends entirely on how well the histogram can be partitioned. The following algorithm is used to obtain T:

1. Select an initial estimate of T

2. Segment the image using T. This will produce two groups of pixels: G1 consisting of all pixels with gray level values >T and G2 consisting of pixels with values ≤ T. Compute the average gray level values μ1 and μ2 for the pixels in the regions G1 and G2.

3. Compute the new threshold value:

4. Repeat steps 2 through 4 until the difference in T in successive iterations is smaller than a predefined parameter T0.

Adaptive Thresholding

Adaptive thresholding is used in case when a histogram cannot be partitioned effectively by a single global threshold. An approach for handling such a situation is to divide the original image into sub-images and then utilize a different threshold to segment each sub image. The image is subdivided into smaller sub-images, such that the illumination of each sub-image is approximately uniform. Since the threshold used for each pixel depends on the location of the pixel in terms of the sub-images, this type of thresholding is called adaptive thresholding.

CHAPTER- 5
Density Measurement

5.1 Processing Operations on Captured Image

Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subfield of digital signal processing, digital image processing has many advantages over analog image processing; it allows a much wider range of algorithms to be applied to the input data, and can avoid problems such as the build-up of noise and signal distortion during processing. The different levels at which one may want to interact with the system.

1) User interface: Running the GUI, Creation of batch jobs for applications using mat lab batch, Batch management, batch execution, including use of MATLAB scripts to run batch jobs on multiple datasets with very little user interaction

2) Application development: Requirements on code structure of application

3) Introduction: Internal representation of batch configurations, Introduction to writing batch configuration scripts, Integration of an application into configuration management and GUI

5.2 Source Image

In this system the source image is the RGB image which can be given by the users for getting the contour image and the vehicle count in output screen.

Figure 5: **Source image**

5.3 Grayscale Image

Grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest. [1]

The grayscale image can be used to display the objects in the format of black and white. In this system the output will display the grayscale image after getting the source image only, because source image only gets converted into the grayscale image.

***Figure 6*: Gray Scale Image**

5.4 Threshold Image

Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images

The threshold image brightness or contrast of the grayscale image. In this system we can convert the grayscale image to threshold image.

A binary image is a digital image that has only two possible values for each pixel. Typically, the two colors used for a binary image are black and white, though any two colors can be used. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color.[1] In the document-scanning industry, this is often referred to as "bi-tonal".

***Figure 7*: Threshold Image**

5.5 Canny Image

Canny image is the image one of the edge detector that can be used to outline the edges of the objects. It can be help full for find out the objects. Here we have convert the threshold image to canny image.

Figure 8: Canny image

5.6 Erode Image

The Erode image also like the canny image it can be used find the edges with the darken lines. In our system the edges of the vehicles are detect with the darken lines Before converting the canny image to Erode image, the canny image will be destroyed.

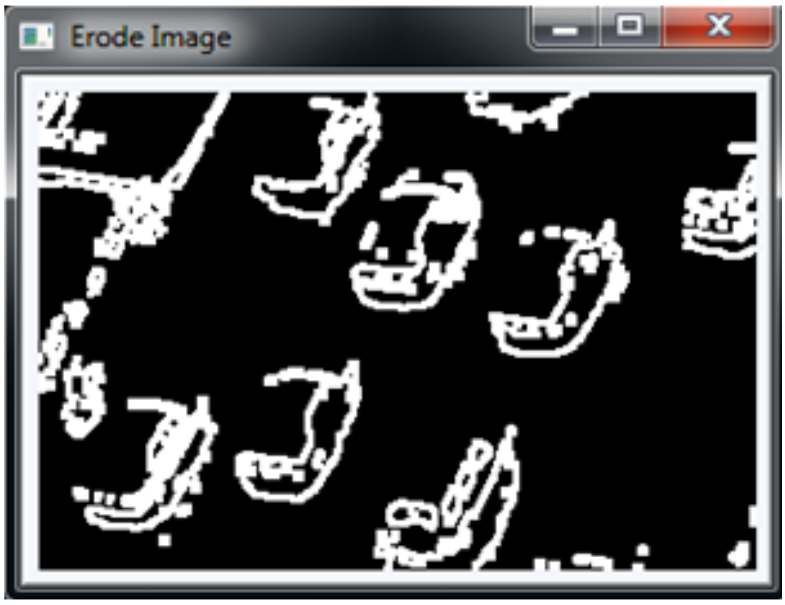

Figure 9: Erode image

5.7 Contour Image

Before showing the vehicles count and output screen the Erode Image converted into the contour image. This image is the final step to find the vehicle counts and output screen.

Figure 10: Contour image

5.8 Blob Detection

A blob is a region of an image in which some properties are constant or approximately constant; all the points in a blob can be considered in some sense to be similar to each other. One main reason to study blob detection is to provide complementary information about regions, which is not obtained from edge detectors or corner detectors.

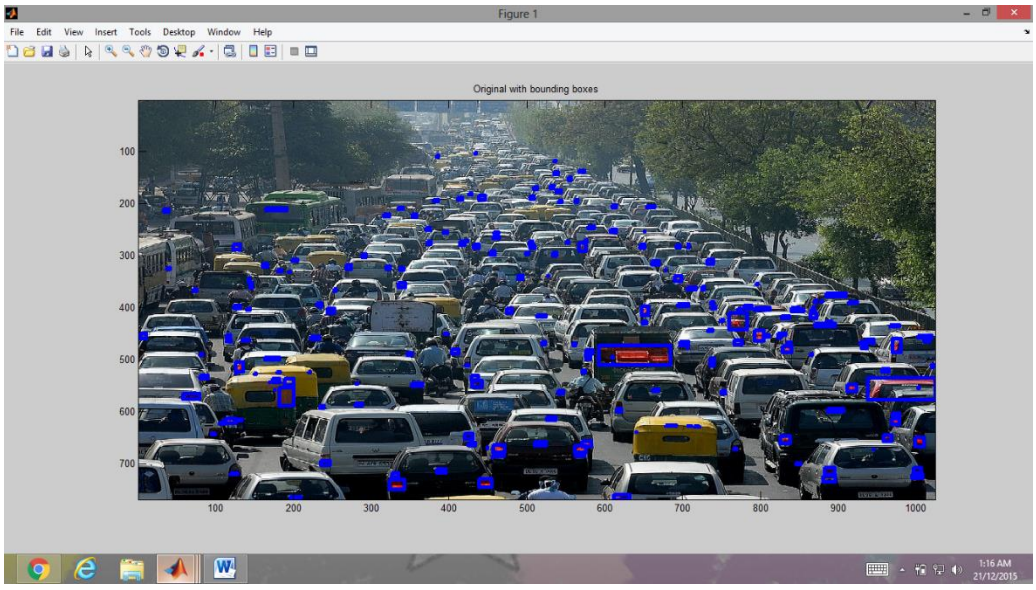

***Figure 11*: Original image with bonding boxes**

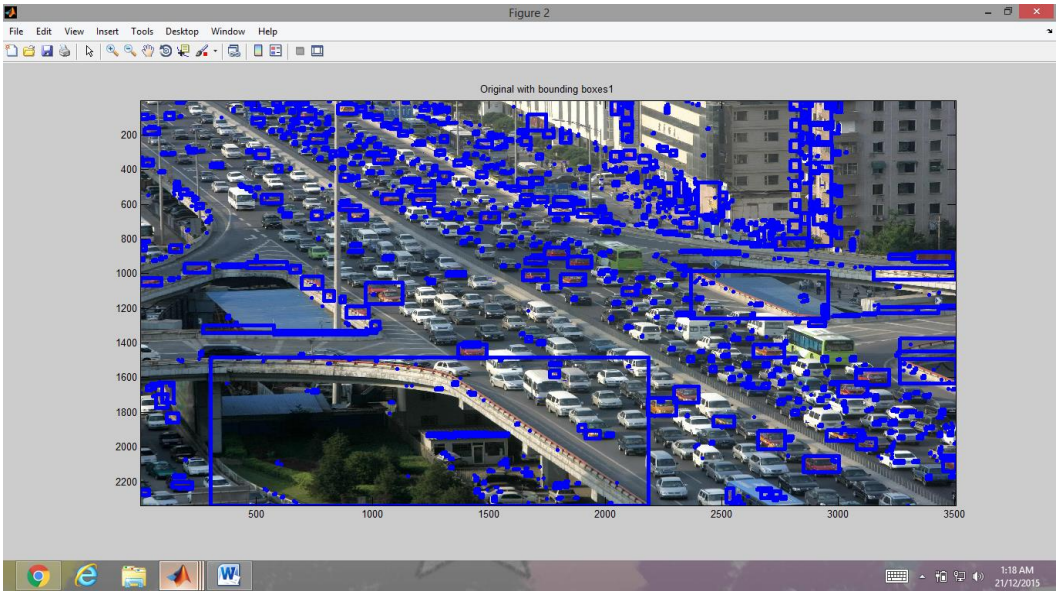

Figure 12: Blob image

Output screen

The two types of output screens are displayed in this system. They are:

1) First output screen displays the output image. In this image will display the original RGB Image and in this screen the vehicles are boxed for the find the count.

2) Another output screen is the command prompt. In this command prompt will be open when the user run this system, in final stage after getting the output image the command prompt will display the vehicle counts.

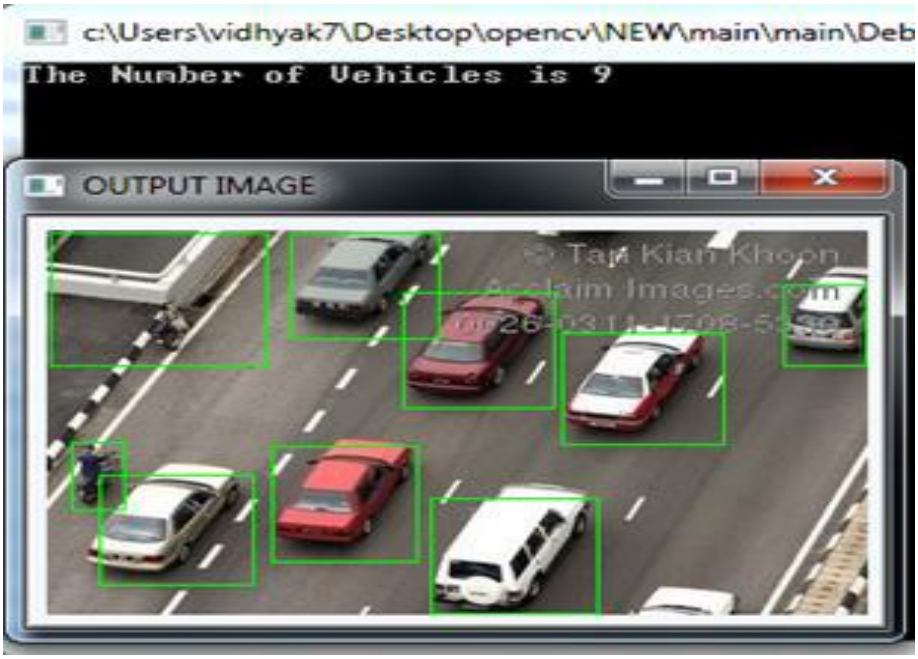

***Figure 13*: Output Image**

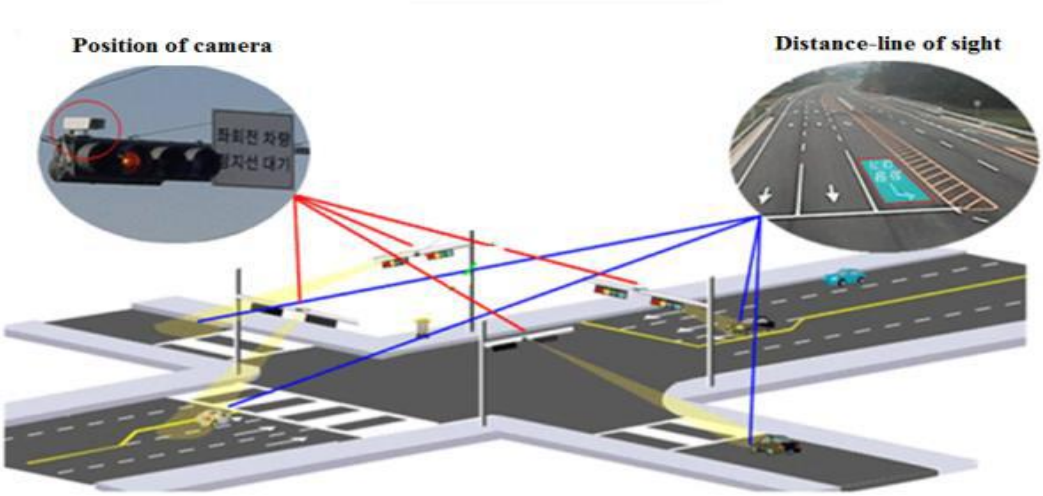

Figure 14: System Architecture

In this architecture camera is placed on the top of the signal to get the clear view of traffic on the particular side of the signal so that it will capture the image and analyze the traffic in that particular side and get the count of the number of vehicle. With this count the density of that particular side will be determined and corresponding signal will be provided. 4) Implementation: Classes used, Methods, Details about job management

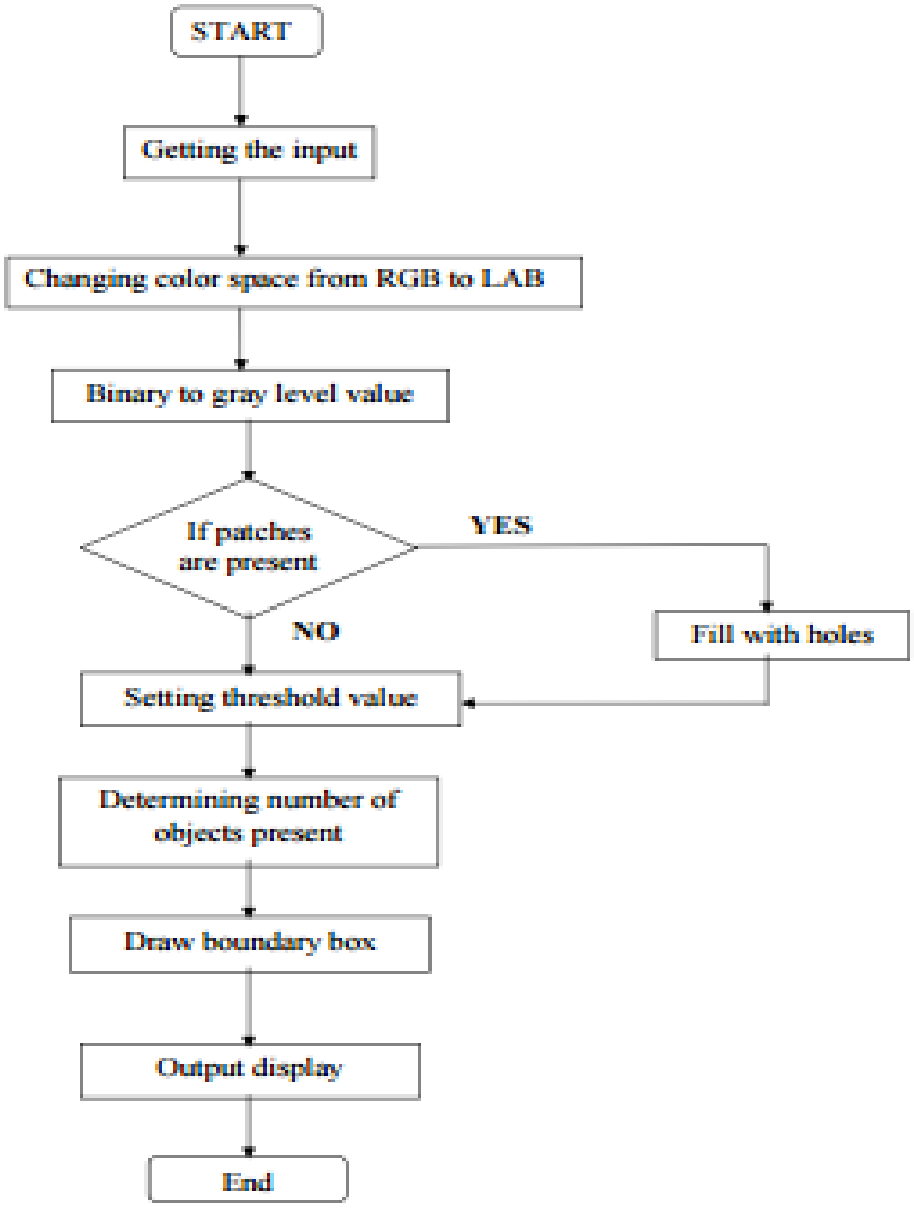

Figure 15: Flow chart for code implemented

5.9 Vehicle counting

Transportation research involves counting number of vehicles on road as well as finding the density of traffic in a particular area. There are many methods of detecting vehicles on road such as motion detection, installing lasers on both sides of the road, etc., which is tedious and involves large number of hardware. This method uses image processing techniques to count the number of vehicles on road and estimate the density. The number of vehicles found can be used for surveying or controlling the traffic signal.

5.2.3 Block Diagram for Image Processing

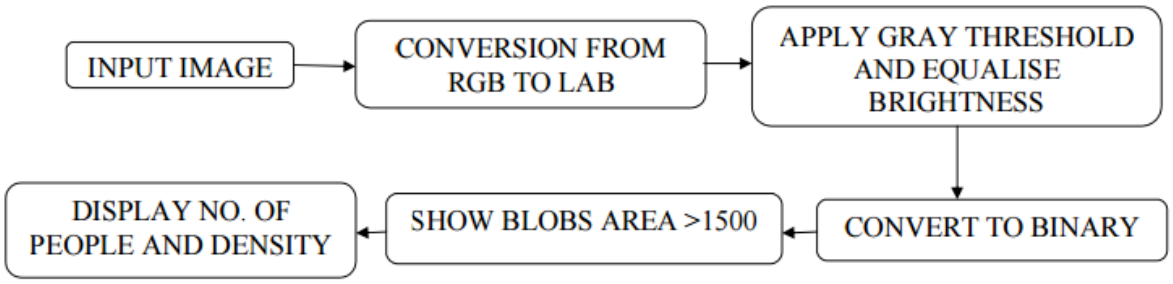

Figure 16: **Block diagram for vehicle count**

First the input image is fed into MATLAB. The image is converted from RGB to LAB. Then the gray threshold is applied to the converted image and brightness equalization is done. Then the equalized image is converted to binary. The blobs in the image are opened when the blobs area is greater than 1. The number of people and density are displayed.

I) Implementation Steps

This section gives the stepwise explanation of the tasks performed for implementing the code.

1: Start the program.

2: Get the input image.

3: Change the color space from RGB to Grayscale.

4: Convert to binary based on gray level value.

5: If patches are present fill with holes.

6: Set a threshold value for opening area above some value.

7: Use connected components method to determine the number of objects present.

8: Draw bounding box.

9: Display output count and density.

10: End the program.

The flow diagram describing the scenario is shown in the following page.

Chapter-5

Emergency override

A. Background

Vehicle license plate recognition is an image processing system whereby it is used to recognize the vehicles by identifying the license plate. It can be used for traffic control, security purposes, and parking access control. The license plate system works as follows: Firstly, the vehicle will stop in the traffic. The cycle will start when the vehicle steps over the detector. It will activate a signal to the vehicle license plate system for the presence of the vehicles.

Secondly, illumination will be activated and images of the front picture of the vehicle will be taken. Then the system will read the information pixels of the vehicle and run the recognition process.

Thirdly, the system will apply certain algorithm to analyze the vehicle image. If the registration plate number is inside the predefine list, the green light is offered to the same lane and the others are made red or amber for a few seconds till the emergency vehicles has passed through. Otherwise, the regular routine of the lights works and the cycle continues for every loop.

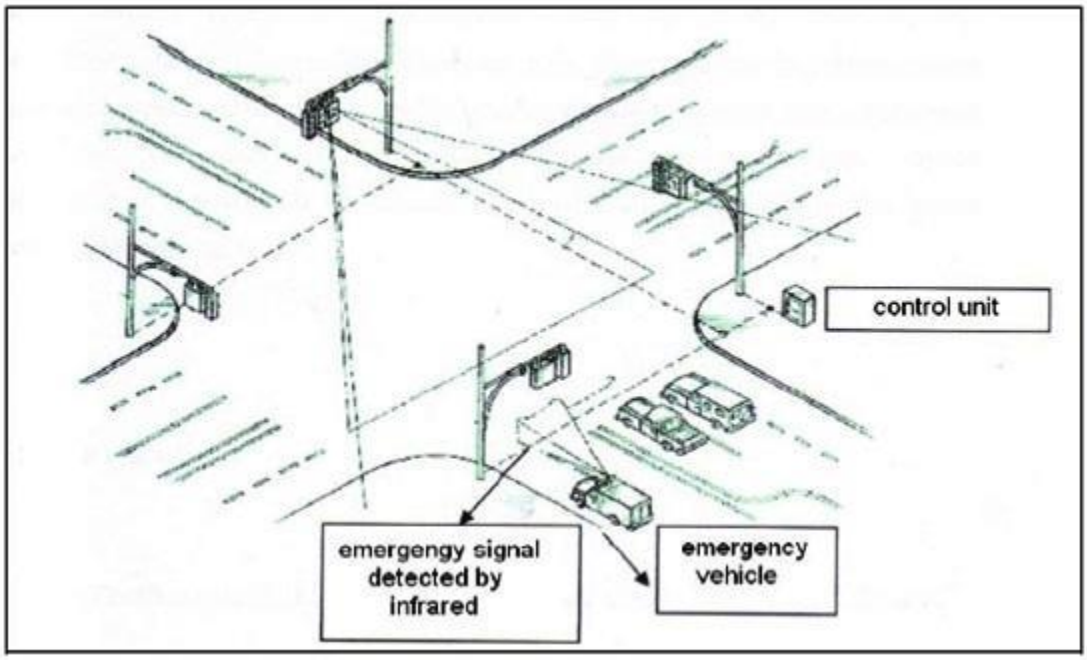

Figure 17: Overview

Over the recent years, there were quite number of methods for character/numeric and vehicle registration recognition developed. However, identification of vehicle registration plate number with respect to various natural viewing conditions still remains a challenging task. This is because:

1) Vehicle registration plate number acquired is of different front sizes.

2) Color of the vehicle registration plate number affected by varying illumination.

3) Vehicle registration plate character might be chipped off.

4) Algorithms must be able to be used in real time implementation.

There are difficulties for vehicle license plate recognition in which it will affect the efficiency and accuracy of the system. It is essential and important to determine the facts which will able to influence the operations and recognition proficiency. We also need to look into other facts of variables that are not constant. The following are the non-constant variables which affect the accuracy of recognition:

a) Weather condition.

b) Type of Vehicle.

c) Distance between vehicle license plate and the camera.

d) Type of plate (rectangular, bent type)

e) Vehicle license plate orientation.

B. Objectives

The overall objective of this project is to develop a system that recognizes vehicle license plate from a car. The software could lead to a cheaper and faster way of enhancing and determined the performance of the recognition system. The system will be based on a personal computer which will generate a report on the vehicle license plate that has been captured. Once the vehicle license plate is captured, the characters will be recognized and displayed on the graphical user interface. This algorithm development should be able to meet the requirements and goals as stated below: Able to detect the vehicle registration plate at a faster speed. Able to accurately recognize the vehicle registration plate. There is definitely a lot more room for further improvements.

6.1 THEORETICAL PRINCIPLES

The following flow chart simplifies the operation of the system:

A. Image Acquisition

The initial phase of the image processing for vehicle license plate recognition is to obtain images of vehicles. Electronic devices such as optical (digital/video) cameras, webcams,

etc. can be used to capture the acquired images. The images will be stored as colour JPEG format on the camera. Next, we might proceed in using the Matlab function to convert the vehicle JPEG image into gray scale format.

B. Plate Localization

1. Converting the Image to a Gray Scale

We now convert the image to a gray scale, which is important as we will take the edge that needs this form of image. Then we enhance the image, this step produces good results in some cases and approximately provides the same results in other cases.

2. Taking the Complement of the Image

In this step, we take the complement of the image to enhance it, and will use the canny method to find the edge. The Canny method finds edges by looking for local maxima of the image's gradient. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method takes sigma as the standard deviation of the Gaussian filter, which controls the amount of details of the image; if sigma increases the details will decrease. In our practical part, we choose sigma of 0.95. However, if the process cannot get the plate then we change it until we reach the correct result.

3. Filtering the Image from Small Objects

We now convert the image into a double class because the functions that we use need this class. Then, we filter the image from any object of less than 280 pixels and close the image.

4. Labelling the Image to Objects

We now have an image that consists of objects. In order to decide the correct plate from these objects, we label them and find the characters of each object

5. Separating Every Object

We then separate, fill, and rotate every object to give it specific characteristics that help in knowing the plate like the picture below:

6. Recognizing the Plate

By a trial and error, we could specify the suitable description solidity, the ratio between the height and width, convex area and bounding box. The last step is to display the plate by multiplying the plate object with the original image.

C. Character Segmentation

Character Segmentation is an important step in the vehicle registration plate recognition system. However, with the influence factors such as image noise, spacing, brightness, different sizes and so on, there are quite a lot of difficulties faced. Character segmentation is basically the process of separating out the individual digits from a stretch of numbers images capture.

1. Threshold of Numeral Images

Before any character segmentation to be carried out, we have to convert the images captured into grayscale. This is basically changing all the pixels to either just black or white pixels depending on whether the pixels are above or below a defined threshold. This threshold selection has to be a function of the intensity range of the pixels in the image. The average of the minimum and maximum threshold values is normally enough to optimize the conversion.

2. Extracting Connected Components

To begin the segmentation, we have to extract out each of the connected components by finding black pixels and then using a search algorithm to find all the pixels which are connected with them. This process is carried out until every black pixel in the image is being classified as part of a connected component. If any of the captured digits is broken/ defaced, then any of the connected components would represent a single character.

3. Taking Horizontal and Vertical Projections for Each Number

We then take the horizontal and vertical projections (number of pixels versus its position) for each number to identify it. We take every number after filling it to increase the properties to describe any number.

4. Segmenting the Numbers by Taking the Boundary Box

Now we have an image that consists of only the numbers that we want. So, we segment the number by taking the boundary box for everyone.

D. Numbers Recognition

1. Normalization

In this phase, the extracted characters are0 resized to fit the characters into a window. In our practical work, each character is normalized to the size of 50x30 binary image then reshaped to standard dimensions before sending the data set to a neural network for training. It is very important to expand the training database size for neural network because of the fact that increasing the database size results in an improved efficiency and accuracy of the network. Segmented characters have very much variation in size. In this phase, all the characters are normalized to predefined height (vertical length) in pixels. As the characters always have variable width (horizontal length), each character image is normalized to a size of 32x32 using an image mapping technique.

2.Features Extraction

The goal of the feature vector is to define distinguishing features of the characters. Selecting the most relevant feature of each character cannot only facilitate data visualization and data understanding, but also reduce the measurement, storage requirements, training, and utilization time, particularly when the features are redundant. Initially, the centroid of the character image is determined. With respect to centroid, the number of transitions along the axes, 0 to 1 and 1 to 0, up to the boundary of character are counted. Transitions are specified for axes with predetermined angles.

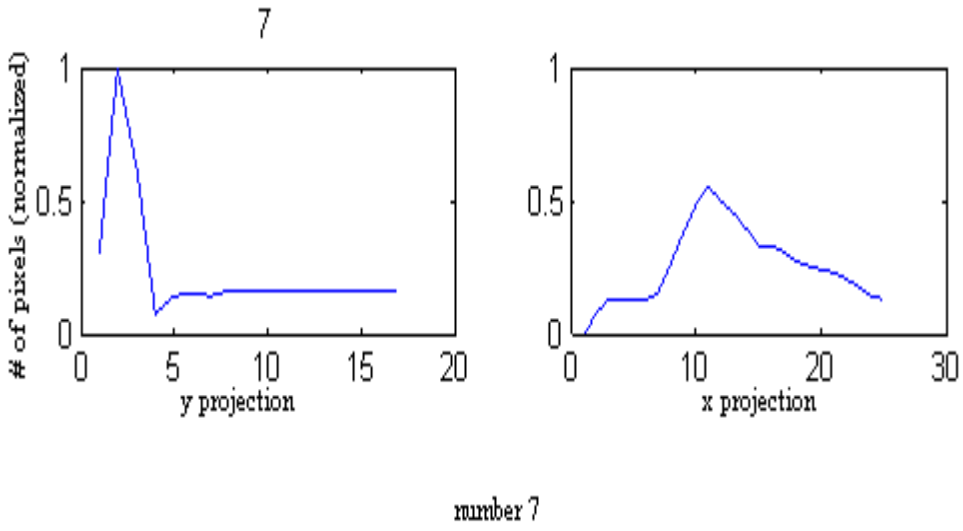

Figure 18: Detection of Number 7

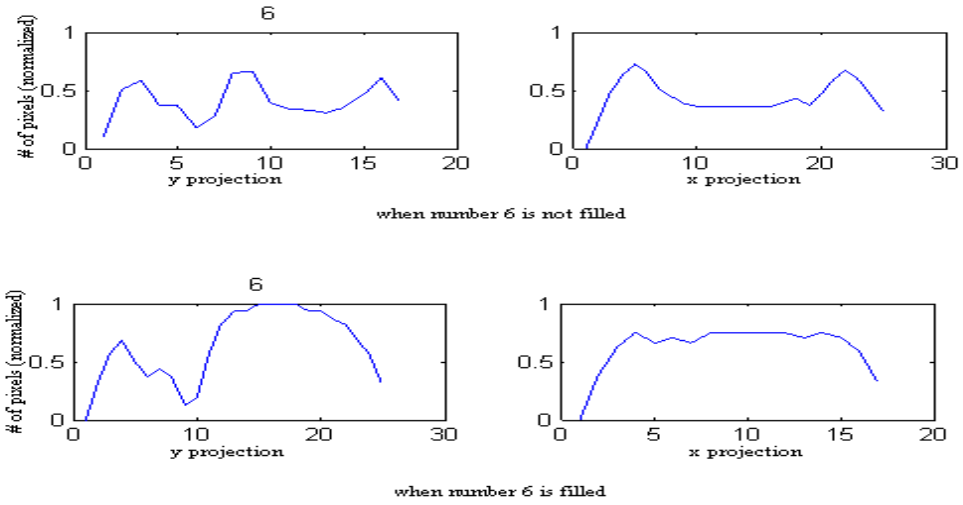

Figure 19: Detection of Number 6

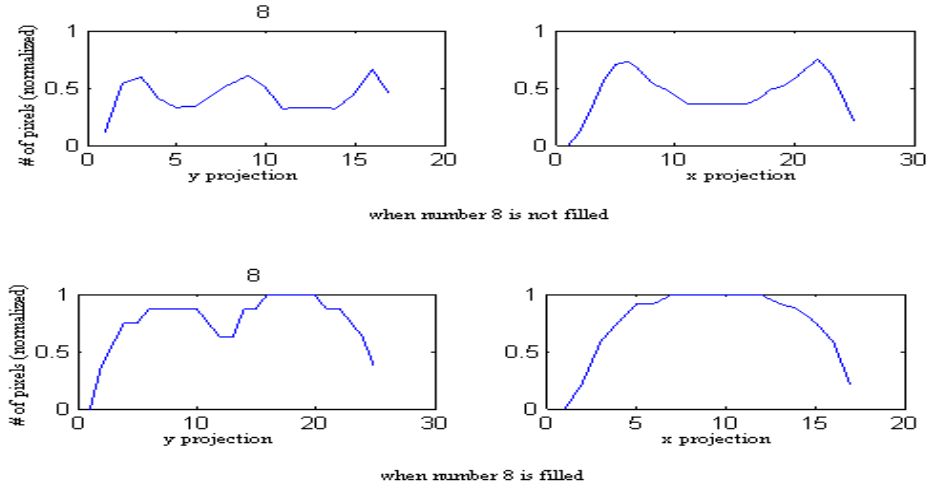

Figure 20: Detection of Number 8

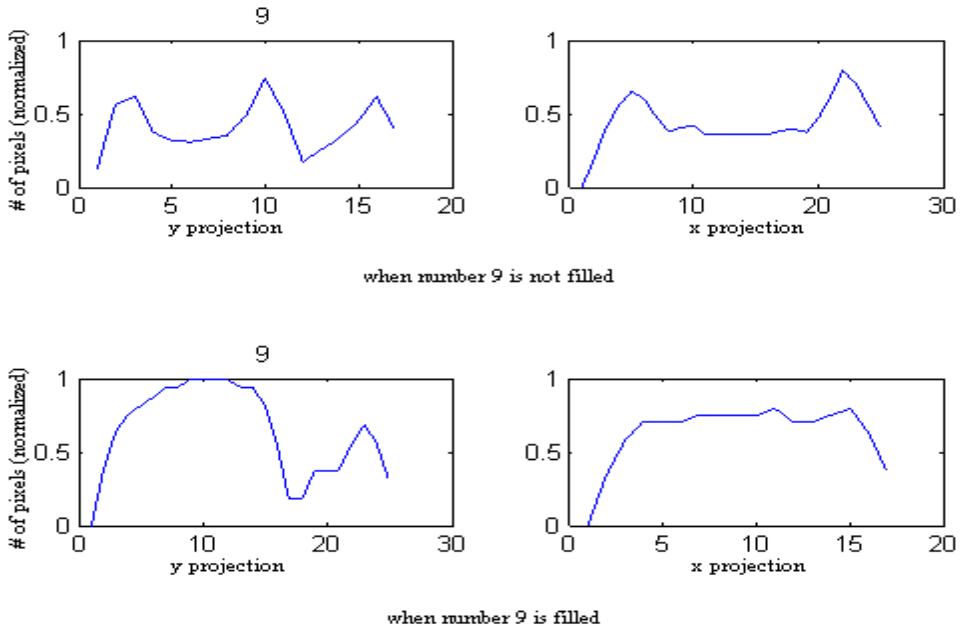

Figure 21: Detection of Number 9

E. Database System

The database system is a collection of information or data which is being orderly organized, thus it can easily be accessed and updated. The database can be in the form of text, contents, and images. A simple database system can be created, which contains the information of vehicles that are allowed to access a designated area for this access control application. The basic functions like "Add", and "Delete" should be also available to make it simple for the user to organize the database easily.

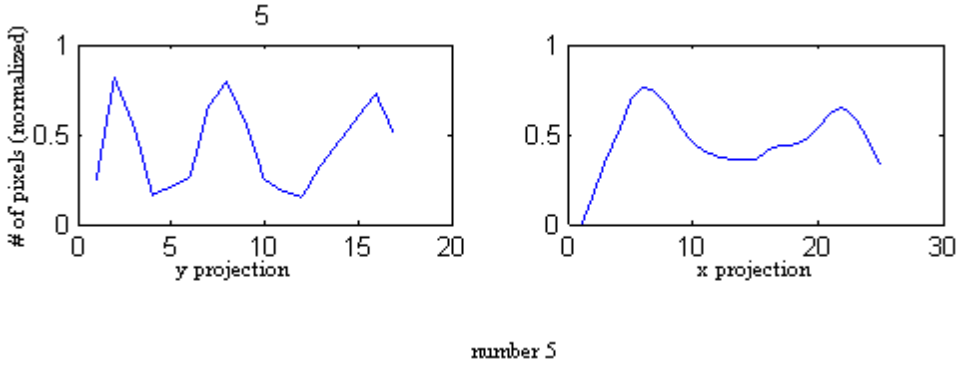

Figure 22: Detection of Number 5

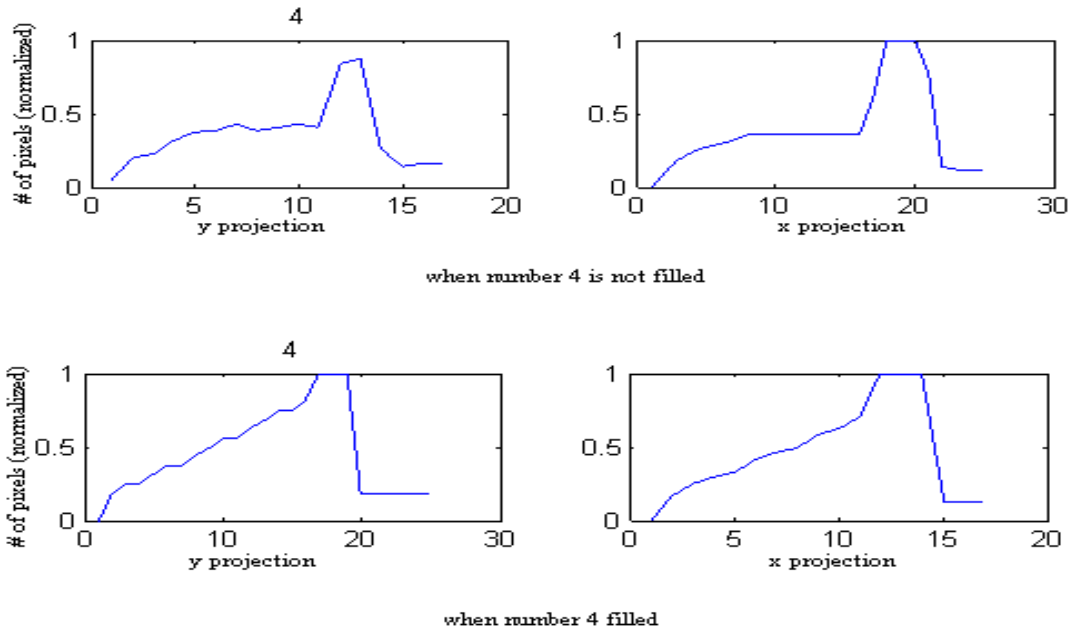

Figure 23: Detection of Number 4

6.2 HARDWARE DESIGN AND IMPLEMENTATION

IP Camera

IP stands for Internet Protocol, which is a protocol for transmitting data across a network. An IP Camera is a camera that plugs directly into a network router, and is not reliant on a PC to work. Data from the IP camera is transmitted through the network, and can be securely viewed at a remote location. An IP camera requires a high speed connection (such as DSL), a router, and an Ethernet cable. A computer is needed to view the data; however, the camera works independently of the computer.

Motion Detectors

A motion detector is a piece of equipment that can be used alone or as part of a complete business or home security system. Infrared detectors, passive infrared detectors, and outdoor motion detectors are intended to sense any movement that occurs and notify the owner of the location when a motion is detected. Some motion sensors are small parts of a more complicated security system, while others are simple and basic, giving the owner of the location an audible sound during motion detection.

6.3 SOFTWARE DESIGN AND SIMULATIONS

Matlab programming is used as it is a user friendly and the image processing can be easily performed via it. We also use the graphical user interface (GUI) to build up a graphical display to run the system. The GUI is a graphical display that contains devices or components which enable the user to perform interactive tasks.

A. Setting up the Graphical User Interface

1) A quick start of the GUI at the MATLAB: We are able to create either a blank GUI or open an existing GUI by typing “guide” in the Matlab command.

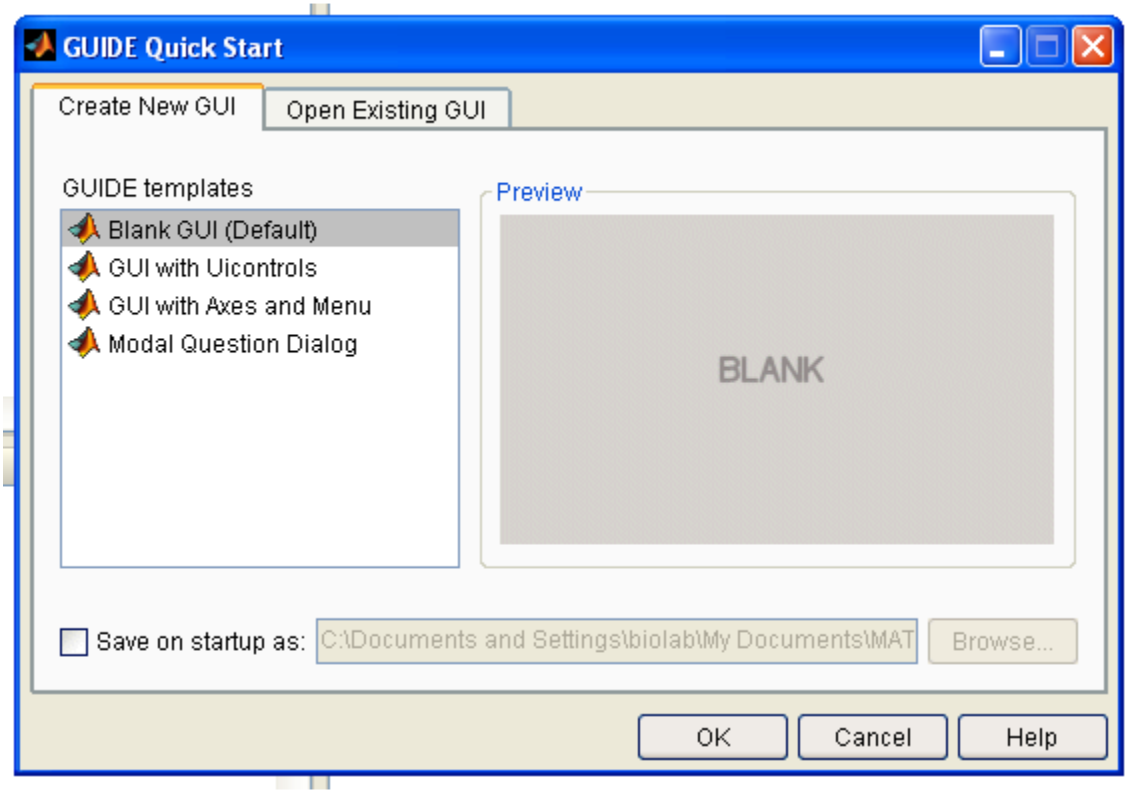

Figure 24: Guide to GUI

2) After selecting the blank GUI, it will bring us into the images below. At the stages, users are able to insert whatever components that are wanted into the program.

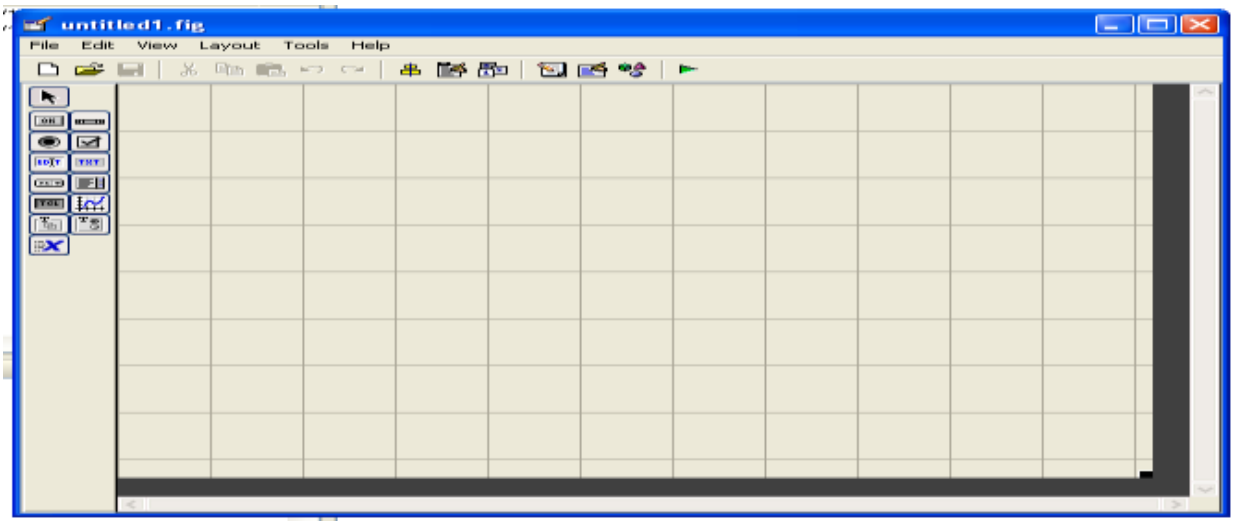

Figure 25: A blank GUI

3) This is the design output of the GUI

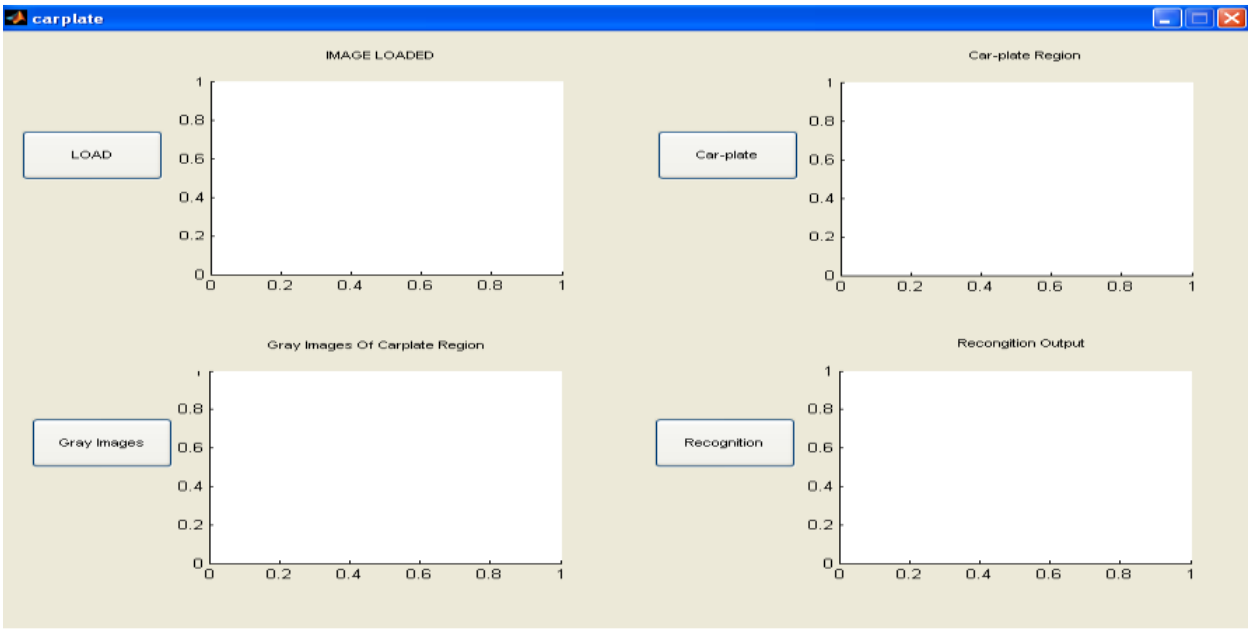

Figure 26: Design output

4) After which, we program the individual pushbutton and will be able to display the individual function at the axis assigned to it. The below diagram shows the output of the GUI after programming.

Figure 27: Final GUI

	Input	Description	System's Status	Result
	Plate 1	The plate is found in the database	Available	Access is granted
	Plate 2	The plate is not found in the database	Available	Access is denied
	Plate 3	The plate is in the database, but it is not recognized	Available	No change

We now consider the three cases that may occur while examining our practical work:

We have three cases: The first case happens when the plate is recognized and also found in the data base (access is granted). The second case occurs when the plate is not found in the data base (access is denied). The third case takes place when the program cannot recognize the plate due to many factors such as whether, light intensity, and so on (no change). In our practical part, we have performed this test for about 100 vehicles and resulted in an average percentage of success of about 91%.

Chapter-7

Result and conclusion

7.1 Result

With the help of MATLAB, the code gave us the count of the cars in the image. Whenever there will be vehicles in the image the code will detect them and after processing the image, it will display the number of vehicles.

7.2 Conclusion

System is estimated to be accurate 80% and even more depending on the accuracy of ROI used to estimate occupancy. Major advantage is the variation in signal time which control appropriate traffic density using artificial vision. The accuracy in calculation of time due to single moving camera depends on the registration position while facing road every time. The handling of emergency with the help of assigning priority has an advantage since safety human is maintained. Limitation of GSM network may sometime create problem of delay in delivery of message, but is not frequent phenomenon and could be taken care with the help of service provider. Automatic caution mode is new feature of this system that will ensure the withdrawal of signal control at night after predefined time when the traffic is lowest and brings it normal when needed. Thus the system will be close to traditional system with improved efficiency and safety.
The project presents the method of traffic light control through image processing. The earlier techniques had a drawback of time being wasted on green light on the empty road. Our implemented system avoids this problem. We have successfully implemented real time image processing based traffic light controller This project illustrates that image processing is the best way to control traffic when it comes to real time feedback. The key feature of this paper is that it removes the need of hardware sensors such as infrared sensors and RFID tags.

LPR technology is a significant tool in the arsenal of law enforcement and public safety agencies. It automates a tedious, distracting, and manual process that officers regularly complete in their daily operations, and vastly improves their efficiency and effectiveness in identifying vehicles of interest among the hundreds or thousands they observe in routine patrol. Moreover, it generates a rich and enduring record of vehicle sightings, complete with time, date, and geographic location information for each observation. This data can substantially enhance the investigative capacity of law enforcement, and greatly contribute to intelligence collection and analysis functions. Realizing the core business values that LPR promises, however, can only be achieved through proper planning, implementation, training, deployment, use, and management of the technology and the information it provides. Like all tools and technologies available to law enforcement, LPR must also be carefully managed. Agencies must clearly articulate their strategic goals and tactical objectives for the technology, and this strategy should be tightly aligned with the broader strategic plans of the agency. Thorough and ongoing training is required to ensure that the technology performs effectively, and that users are well versed in the operational policies and procedures defined and enforced by the agency. Policies must be developed and strictly enforced to ensure the

quality of the data, the security of the system, compliance with applicable laws and regulations, and the privacy of information gathered. Building robust auditing requirements into agency policies will help enforce proper use of the system, and reassure the public that their privacy interests are recognized and respected.

7.3 FUTURE SCOPE

UTILISATION OF TECHNOLOGIES

It is based on evaluation of an integrated traffic control and route diversion strategy. Traffic density management using GPS is an attempt for the creation of a smooth driving environment through a variety of recent technologies such as GPS and WI-FI. With these technologies we can make people not to wait for a long time in traffic signals when there are fewer vehicles on the road. This technology makes the signal system to work on the vehicle density basis whereas the existing system is fixed.
This system can be extended to personnel vehicles also. This technology can serve its purpose by use GPS as one can use it to choose shortest path between two place and can know traffic density of any road in the city so that he/she can decide the route to the destination accordingly. This technology can play a vital role in fuel savings and pollution control. Moreover, no longer delays for any emergency vehicle such ads an ambulance or a fire brigade or a police gypsy, making it more efficient and worth a while for the citizens.

PRIORITY BASED TRAFFIC SYSTEM

Consider a scenario of highly congested area where many vehicles such as personal transport, public transport and emergency vehicles (Ambulance, Fire brigade, VIP cars and other rescue vehicles) have to wait for long for the change of traffic signals at intersection points. Existing traffic light systems have timers that are set at regular intervals. This leads to the wastage of precious time especially in case of rescue vehicles for emergency conditions.
In order to control this situation, we can propose a system consisting of two parts: Smart Traffic Light Control System (STLC) and Smart Congestion Avoidance System (SCA) during emergencies. STLC System controls the change of traffic lights at intersection points giving high priority to emergency vehicles. SCA System is a smart traffic routing system that chooses the shortest routes having the least congestions.

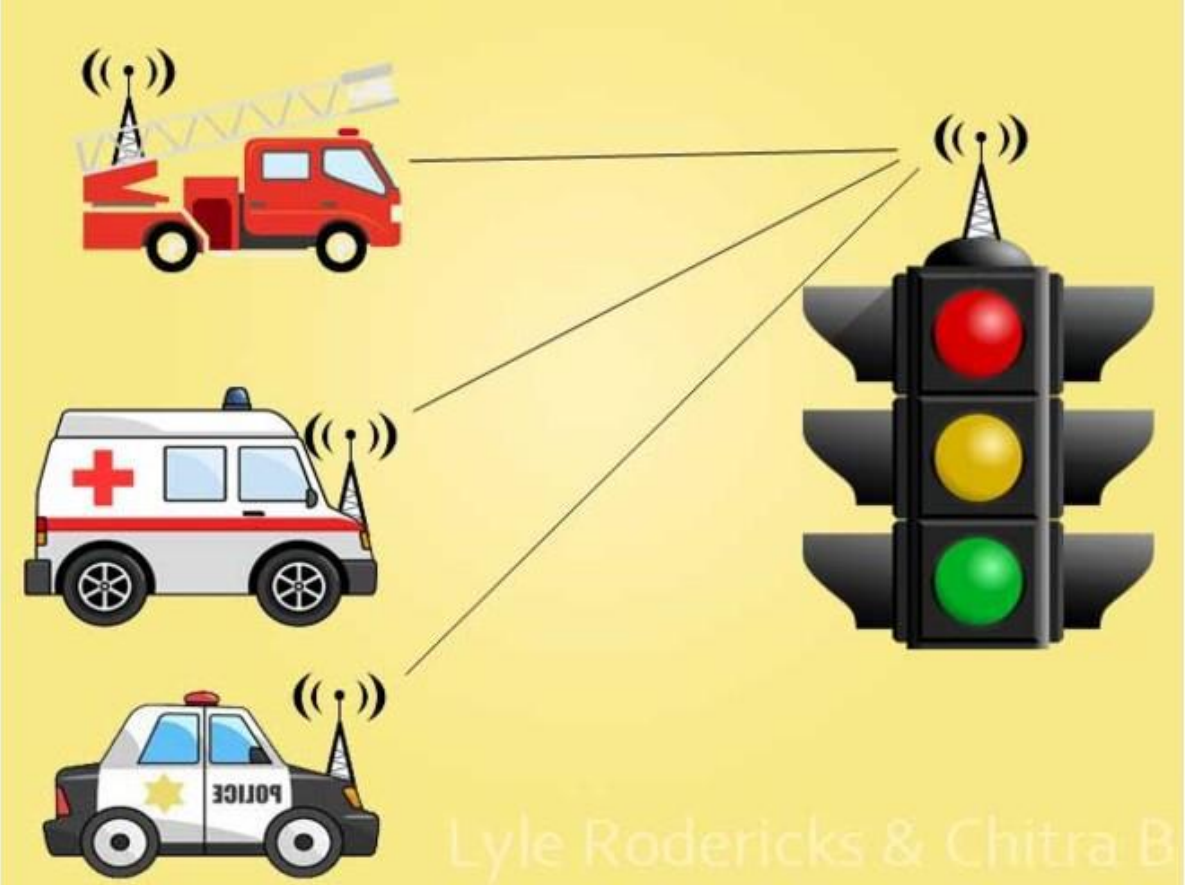

Figure 28: Overlook for Emergency vehicle

The vehicles are fitted with GPS as shown in figure that will transmit the vehicle geo-location which will then be received by the road side unit at signal and it will start the green light and this light will be in ON state till the vehicle is within the GPS range. This will be useful for the emergency vehicles such as Fire Brigade, Ambulance, Police cars

SMART ROUTING SYSTEM

In the last decade, intelligent transportation systems have progressed at a rapid rate, which aim to improve transportation activities in terms of safety and efficiency. Car-to-car and car-to-infrastructure communications are important components of the ITS architecture. GSM modules for communication and location tracking, speed and acceleration sensors for speed monitoring and display module for real time traffic pattern display Communication between cars and traffic lights is one of the important applications which help to have dynamic and automatic traffic lights that can create several benefits such as minimizing the traffic jam, reducing fuel consumption and emissions, etc. Here it deals with decreasing the response time of the emergency cars by changing the traffic lights status with employing the communication technologies.
Additionally, several statistics about traffic simulation is created for each car such as traveling time, waiting time, emissions, fuel consumption; or complete amount of car emissions in the street

during the simulation, fuel consumption, and number of vehicles and so on, for each street. Here it presents evaluation of an integrated traffic street. Here it presents evaluation of an integrated traffic control and route diversion strategy. By using all this data, we can provide information about the traffic congestion and can provide the best suitable route for their destination which can save a lot of time and fuel too.

CHAPTER-8

References

1. Ahmed S. Salama, Bahaa K. Saleh, Mohamad M. Eassa, “Intelligent Cross Road Traffic Management System (ICRTMS),” 2nd Int. Conf. on Computer Technology And Development, Cairo, Nov 2010.
2. MD. Hazrat ALI, Syuhei KUROKAWA, et al, “Autonomous Road Surveillance System Proposed Model for Vehicle Detection and Traffic Signal Control “in Procedia Computer Science 19(2013)
3. P. Pongpaibool, P. Tangamchit and K. Noodwong, "Evaluation of Road Traffic Congestion Using Fuzzy Techniques, “Proceeding of IEEE TENCON 2007, Taipei, Taiwan, October 2007.
4. H.R. Kashani and G.N. Saridis “Intelligent Control for Urban Traffic Systems” International Federation for Automatic Control Automatic, vol.19
5 https://www.ncjrs.gov/pdffiles1/nij/grants/239604.pdf
6. www.google.com
7. www.mathsworks.com
8. en.wikipedia.org
9. http://www.ijcse.com/docs/IJCSE11-02-01-031.pdf
10. www.researchgate.net
11. K. Vidhya, A. BazilaBanu, "Density Based Traffic Signal System", International Journal of Innovative Research in Science, Engineering and Technology, 2014 International Conference on Innovations in Engineering and Technology (ICIET’14), Volume 3, Special Issue 3, March 2014

12. Mario Collotta, Giovanni Pau, Gianfranco Scata, Tiziana Campisi, “a dynamic traffic light management system based on wireless sensor networks for the reduction of the red-light running phenomenon”, Transport and Telecommunication Institute, Lomonosova1, Riga, LV -1019, Latvia Transport and Telecommunication, 2014, volume 15, no 1, 1–11