**SMART TRAFFIC OPTIMIZATION USING IMAGE PROCESSING**

**ABSTRACT**

One way to improve traffic flow and Safety of the current transportation system is to apply automation and Intelligent control methods. Image processing is better technique to control traffic density in real time system. It shows that it can decrease traffic congestion and avoids time being wasted by green light on an empty roads. It is also more reliable in estimating vehicles presence because it uses actual traffic images etc. The lane images of all the 4 sides of the road are captured using camera simultaneously. We need to eliminate the background objects by using thresholding and background subtraction method and morphological operations. Vehicles counting process results in density estimation of the traffic at regular interval of time. This is achieved through object counting method. The number of objects will be counted for each lane and compared. Among the 4 lanes, the lane with more number of vehicles will be allowed to move forward after receiving the Green signal and the other 3 lanes with least number of vehicles will be stopped at their respective places. The red light is passed at this condition. The information such as LANE-1 ON, LANE-2, 3, 4-OFF will be passed to the hardware. The proposed approach is suitable for real-time applications.

**CHAPTER 1**

**INTRODUCTION**

**1.1GENERAL**

Digital image processing is the use of computer algorithms to perform image processing on digital images. The 2D continuous image is divided into N rows and M columns. The intersection of a row and a column is called a pixel. The image can also be a function other variables including depth, color, and time. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

* + 1. **THE IMAGE PROCESSING SYSTEM**

Mass Storage

Digitizer

Operator Console

Digital Computer

Image Processor

Display

Hard Copy Device

FIG 1.1 BLOCK DIAGRAM OF IMAGE PROCESSING SYSTEM

* **Digitizer**

Digitizing or digitizationis the representation of an object, image, sound, document or a signal (usually an analog signal) by a discrete set of its points or samples. Digital information exists as one of two digits, either 0 or 1. These are known as bits.

An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations.

* Microdensitometer
* Flying spot scanner
* Image dissector
* Videocon camera
* Photosensitive solid- state arrays.
* **Digital computer**

A computer is an electronic device that accepts raw data, processes it according to a set of instructions and required to produce the desired result. Mathematical processing of the digitized image such as convolution, averaging, addition, subtraction, etc. are done by the computer.

* **MASS STORAGE**

Mass storage devices used in desktop and most server computers typically have their data organized in a file system.The secondary storage devices normally used are floppy disks, CD ROMs etc.

* **OPERATOR CONSOLE**

The operator console consists of equipment and arrangements for verification of intermediate results and for alterations in the software as and when require. The operator is also capable of checking for any resulting errors and for the entry of requisite data.

* **Display**

Popular display devices produce spots (display elements) for each pixel:

* Cathode ray tubes (CRTs).
* Liquid crystal displays (LCDs).
* Printers.

Spots may be binary (e.g., monochrome LCD), achromatic (e.g., so-called black-and-white, actually grayscale for intensity), pseudo color or false colors (e.g., for intensity or hyper spectral data), or true color (color data displayed as such).

* + 1. **IMAGE PROCESSING FUNDAMENTAL**

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form. They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

**1.1.2.1 Fundamental steps in image processing**

* 1. Image acquisition
  2. Image preprocessing
  3. Image segmentation
  4. Image representation
  5. Image description
  6. Image recognition
  7. Image interpretation
* **Image acquisition**

First we need to produce a digital image from a paper envelope. This can be done using either a CCD camera, or a scanner

* **Image preprocessing**

This is the step taken before the major image processing task. The problem here is to perform some basic tasks in order to render the resulting image more suitable for the job to follow. In this case it may involve enhancing the contrast, removing noise, or identifying regions likely to contain the postcode.

* **Image segmentation**

Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

* **Image representation**

Image process is the process of convert the input data to a form suitable for computer processing

* **Image description**

Image description is the process of extract features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another.

* **Image recognition**

Image recognition is the process of assign a label to an object based on the information provided by its descriptors.

* **Image interpretation**

Image interpretation is the process of assign meaning to an ensemble of recognized objects.

**1.1.2.2 Image types**

There are several ways of encoding the information in an image.

1. Binary image
2. Grayscale image
3. Indexed image
4. True color or RGB image

* **Binary image**

Each pixel is just blackor white. Since there are only two possible values for each pixel (0, 1), we only need one bitper pixel.

* **Grayscale image**

Each pixel is a shade of gray, normally from 0 (black) to 255(white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power of 2.

* **Indexed image**

An indexed image consists of an array and a color map matrix. The pixel values in the array are direct indices into a color map. By convention, this documentation uses the variable name X to refer to the array and map to refer to the color map.

* **True Color or RGB image**

Each pixel has a particular color; that color is described by the amount of red, greenand bluein it. If each of these components has a range 0–255, this gives a total of 2563different possible colors. Such an image is a “stack” of three matrices; representing the red, greenand bluevalues for each pixel. This means that for every pixel there correspond 3 values.

**1.1.2.3 image processing goals**

In virtually all image processing applications, however, the goal is to extract information from the image data. Obtaining the information desired may require filtering, transforming, coloring, interactive analysis, or any number of other methods.

To be somewhat more specific, one can generalize most image processing tasks to be characterized by one of the following categories:

Problem Domain

Knowledge

Base

Segmentation

Preprocessing

Image Acquisition

Recognition & interpretation

Representation & Description

Result

FIG 1.2 BLOCK DIAGRAM OF FUNDAMENTAL SEQUENCE INVOLVED IN AN IMAGE PROCESSING SYSTEM

1. Image enhancement
2. Image restoration
3. Image analysis
4. Feature extraction
5. Image registration
6. Image compression
7. Image synthesis

* **image enhancement**

This simply means improvement of the image being viewed to the (machine or human) interpreter's visual system. Image enhancement types of operations include contrast adjustment, noise suppression filtering, application of pseudo color, edge enhancement, and many others.

* **image restoration**

The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. Degradation comes in many forms such as motion blur, noise, and camera misfocus. In cases like motion blur, it is possible to come up with a very good estimate of the actual blurring function and "undo" the blur to restore the original image. In cases where the image is corrupted by noise, the best we may hope to do is to compensate for the degradation it caused.

* **image analysis**

Image analysis is the extraction of meaningful information from images. Image analysis operations produce numerical or graphical information based on characteristics of the original image. They break into objects and then classify them. They depend on the image statistics. Common operations are extraction and description of scene and image features, automated measurements, and object classification. Image analyze are mainly used in machine vision applications.

* **feature extraction**

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a [classification](http://en.wikipedia.org/wiki/Statistical_classification) algorithm which [over fits](http://en.wikipedia.org/wiki/Overfitting) the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

* **image registration**

Image registration is the process of overlaying two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors. It geometrically aligns two images the reference and sensed images. The present differences between images are introduced due to different imaging conditions. Image registration is a crucial step in all image analysis tasks in which the final information is gained from the combination of various data sources like in image fusion, change detection, and multichannel image restoration.

Typically, registration is required in remote sensing (multispectral classification, environmental monitoring, change detection, image mosaicing, weather forecasting, creating super-resolution images, integrating information into geographic information systems (GIS)), in medicine (combining computer tomography (CT) and NMR data to obtain more complete information about the patient, monitoring tumor growth, treatment verification, comparison of the patient’s data with anatomical atlases), in cartography (map updating), and in computer vision (target localization, automatic quality control), to name a few.

* **image compression**

The objective of image compression is to reduce irrelevance and redundancy of the image data in order to be able to store or transmit data in an efficient form.

Image compression may be lossy or lossless. Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. This is because lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossy methods are especially suitable for natural images such as photographs in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate. The lossy compression that produces imperceptible differences may be called visually lossless.

* **image synthesis**

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

**1.1.2**.**4 Applications of image processing**

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

* **Medicine**

Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyotypes. In medical applications, one is concerned with processing of chest X-rays, cineangiograms, projection images of transaxial tomography and other medical images that occur in radiology, nuclear magnetic resonance (NMR) and ultrasonic scanning. These images may be used for patient screening and monitoring or for detection of tumors’ or other disease in patients.

* **Agriculture**

Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables distinguishing good and fresh produce from old.

* **DOCUMENT PROCESSING**

It is used in scanning, and transmission for converting paper documents to a digital image form, compressing the image, and storing it on magnetic tape. It is also used in document reading for automatically detecting and recognizing printed characteristics.

* **RADAR IMAGING SYSTEM**

Radar and sonar images are used for detection and recognition of various types of targets or in guidance and maneuvering of aircraft or missile systems.

* **DEFENSE/INTELLIGENCE**

It is used in reconnaissance photo-interpretation for automatic interpretation of earth satellite imagery to look for sensitive targets or military threats and target acquisition and guidance for recognizing and tracking targets in real-time smart-bomb and missile-guidance systems.

* 1. **OBJECTIVE**
* The objective of the proposed system is to design an approach to control the traffic and optimize which are the most traffic area and least traffic area.
* To improve the efficiency of existing traffic signaling system.
  1. **EXISTING SYSTEM**

Most of the existing systems attempt to design the smart traffic optimization system, but failed to design due to the complexity of the work. Due to poor lighting effect and low contrast, we can’t able to predict any information

* + 1. **DISADVANTAGES OF EXISTING SYSTEM**
* Difficulty in taking the snaps in all 4 lanes
* Lighting illumination
* The noise content is high.
* Segmentation results are not proper.
* Miscounting of objects

**PROBLEM IDENTIFICATION**

Most of the existing systems attempt to design the smart traffic optimization system, but failed to design due to the complexity of the work. Due to poor lighting effect and low contrast, we can’t able to predict any information.

Mismanagement and traffic congestion results in long waiting times, loss of fuel and money. It is therefore almost necessary to have a fast, economical and efficient traffic control system for national development. The monitoring and control of city traffic is becoming a major problem in many countries. With the ever increasing number of vehicles on the road, the Traffic Monitoring Authority has to find new methods of overcoming such a problem. One way to improve traffic flow and Safety of the current transportation system is to apply automation and Intelligent control methods. Image processing is better technique to control traffic density in real time system. It shows that it can decrease traffic congestion and avoids time being wasted by green light on an empty roads.

* + 1. **LITERATURE SURVEY**

# [1] Pallavi choudekar, Sayanti banerjee “Implementation of image processing in real time traffic light control,” 3rd International conference on Electronics Computer Technology (ICECT), 2011.

Traffic parameter estimation has been an active research area for the development of intelligent Transportation systems (ITS). For ITS applications traffic-information needs to be collected and distributed. Various sensors have been employed to estimate traffic parameters for updating traffic information. Magnetic loop detectors have been the most used technologies, but their installation and maintenance are inconvenient and might become incompatible with future ITS infrastructure. It is well recognized that vision-based camera system are more versatile for traffic parameter estimation. In addition to qualitative description of road congestion, image measurement can provide quantitative description of traffic status including speeds, vehicle counts, etc. Moreover, quantitative traffic parameters can give us complete traffic flow information, which fulfills the requirement of traffic management theory. Image tracking of moving vehicles can give us quantitative description of traffic flow

**Drawbacks**

* Cost is high.
* Computational complexity is high.

**[2] V.parthasarathi and M.surya, “Smart control of traffic signal using image processing,” Indian journal of science and technology (IJST), Vol.8, No.16, July 2015.**

This paper focuses on the necessity of intelligent traffic system and the peculiar way of implementation with embedded system tools. Here it is implemented using an object counting methods and detection of emergency vehicles simultaneously thereby control the traffic signals based on the priority outcome. Methods/Statistical analysis: Presently various system do provide a cost effective solution, but the rate of successful operation is bad. Inductive loop detectors installed under surface, this fails in case of poor road condition and so. In the other system, the density is found by installing IR detectors along the side of the lane. These systems operate less efficiently where they are not applicable in the real time process. This method uses image acquisition method using a real time live stream and the algorithm is processed using MATLAB.

**Drawbacks**

* Cost is high.
* Fails for bad road conditions.
* Less efficient.

**[3] Chandrasekar and saikrishna, “Traffic control using digital image processing,” International journal of Advanced Electrical and Electronics Engineering (IJAEEE), Vol.2, No.5, 2013.**

Automatic traffic control and surveillance are important for road usage and management. Timers for each stage are the simplest way to control the traffic. Another way is to use electronic sensors in order to find out vehicles, and produce signal. Here we suggest a system that implement image processing algorithm in real time traffic light control which will control the traffic light efficiently. A web camera is placed in each stage of traffic light that will capture the still images of the road where we want to control the traffic. Then those captured images are successively matched using image matching with a reference image which is an empty road image. The traffic is governed according to percentage of matching.

**Drawbacks**

* Use of timer had a drawback that the time is being wasted by green light on the empty
* Segmentation results are not proper.
* Low image quality

**[4] Omkar Ramdas Gaikwad and Anil vishwas Rao, “Image processing based Traffic light control,” International journal of Science, Engineering and Technology Research (IJSETR), Vol.3, No.4, April 2014.**

Timer for every lane is the simplest way to control traffic. And if those timers are predicting exact time then automatically the system will be more efficient. This paper represents the project that has been implemented by using the Matlab software and it aims to prevent heavy traffic congestion. This project does not actually measure the number of vehicles present on the road, but measures the area covered by vehicles on the road. Moreover, for implementing this project following steps must be considered: 1) image acquisition 2) RGB to grayscale transformation 3) image enhancement. At first, film of highway is captured by a camera has been installed in highway

**Drawbacks**

* Use of timer had a drawback that the time is being wasted by green light on the empty
* Segmentation results are not proper.
* Not proper object count.

# [5] Pallavi choudekar, Sayanti banerjee “Real time traffic light control using image processing,” 3rd International journal of computer science and engineering (IJCSE), 2011.

Traffic parameter estimation has been an active research area for the development of intelligent Transportation systems (ITS). For ITS applications traffic-information needs to be collected and distributed. Various sensors have been employed to estimate traffic parameters for updating traffic information. Magnetic loop detectors have been the most used technologies, but their installation and maintenance are inconvenient and might become incompatible with future ITS infrastructure. It is well recognized that vision-based camera system are more versatile for traffic parameter estimation. In addition to qualitative description of road congestion, image measurement can provide quantitative description of traffic status including speeds, vehicle counts, etc. Moreover, quantitative traffic parameters can give us complete traffic flow information, which fulfills the requirement of traffic management theory. Image tracking of moving vehicles can give us quantitative description of traffic flow

**Drawbacks**

* Cost is high.
* Computational complexity is high.
* Computational time is high.
  1. **PROPOSED METHOD**

In this paper, we present an automatic traffic control system and smart traffic optimization using image processing. The objective of the proposed system is to design an approach which automatically detects and counts the vehicles. The lane images of all the 4 sides of the road are captured using camera simultaneously. We need to eliminate the background objects by using thresholding and background subtraction method and morphological operations. Vehicles counting process results in density estimation of the traffic at regular interval of time. This is achieved through object counting method. The number of objects will be counted for each lane and compared. Among the 4 lanes, the lane with more number of vehicles will be allowed to move forward after receiving the Green signal and the other 3 lanes with least number of vehicles will be stopped at their respective places. The red light is passed at this condition. The information such as LANE-1 ON, LANE-2, 3, 4-OFF will be passed to the microcontroller. The proposed approach is suitable for real-time applications**.** The control was moved to hardware via USB to RS-232 converter.

**PROPOSED BLOCK DIAGRAM**

**THRESHOLDING**

**PREPROCESSING (IMAGE ENHANCEMENT)**

**INPUT IMAGE ACQUSITION (ALL 4 SIDES)**

**RGB TO GRAY CONVERSION**

**OBJECT EXTRACTION**

**MORPHOLOGICAL OPERATIONS**

**BINARY IMAGE**

**OBJECT COUNTING**

**HARDWARE**

**GREEN LIGHT**

**MORE VEHICLES**

**COMPARISON**

**RED LIGHT**

**LEAST VEHICLES**

**FIG. DETAILED BLOCK DIAGRAM**

* + 1. **PROPOSED SYSTEM ADVANTAGES**
* It automatically counts the number of vehicles.
* It results in high accuracy output
* The segmentation results are better.
* The noise content is removed and enhances the contrast of the image.

**CHAPTER 2**

**PROJECT DESCRIPTION**

**2.1 GENERAL**

**INTRODUCTION**

Fast transportation systems and rapid transit systems are nerves of economic developments for any nation. Mismanagement and traffic congestion results in long waiting times, loss of fuel and money. It is therefore almost necessary to have a fast, economical and efficient traffic control system for national development. The monitoring and control of city traffic is becoming a major problem in many countries. With the ever increasing number of vehicles on the road, the Traffic Monitoring Authority has to find new methods of overcoming such a problem. One way to improve traffic flow and Safety of the current transportation system is to apply automation and Intelligent control methods.

**2.2 METHODOLOGIES**

**2.2.1 MODULE NAMES**

* IMAGE ACQUSITION/RESCALING
* RGB TO GRAY CONVERSION
* IMAGE ENHANCEMENT USING CLAHE
* IMAGE SEGMENTATION
* MORPHOLOGICAL OPERATIONS
* OBJECT COUNTING
* DECISION MAKING

**2.2.2 MODULE DESCRIPTIONS**

**MODULE 1**

**IMAGE ACQUSITION/RESIZING/RESCALING**

Initially, we design a road like setup. The images are acquired by the webcam. The stepper motor is attached with the webcam and it will be rotated at 90,180,270 and 360. It will capture all the 4 lane of the road setup respectively. Then all the four images are given as the input to PC. The size of the input image was rescaled to a square matrix such as [256 256] or [512 512]. Since the input image given for the feature extraction should be in square matrix form.

**MODULE 2**

**COLOR CHANNEL CONVERSION**

The input is the RGB (24bits/pixel) image. The output is the red channel, green channel and blue channel image (8 bits/pixel). We can also convert the RGB image into grayscale image. This process is known as RGB to gray conversion.

**MODULE 3**

**IMAGE ENHANCEMENT**

Contrast Limited Adaptive Histogram Equalization (CLAHE) is used to enhance the image. CLAHE algorithm is applied for all the channels in the image. The result is the enhanced image.

**Contrast Limited Adaptive Histogram Equalization**

Contrast limited adaptive histogram equalization (CLAHE) is a popular technique in biomedical image processing, since it is very effective in making the usually interesting salient parts more visible. The image is split into disjoint regions, and in each region local histogram equalization is applied. Then, the boundaries between the regions are eliminated with a bilinear interpolation.

The main objective of this method is to define a point transformation within a local fairly large window with the assumption that the intensity value within it is a stoical representation of local distribution of intensity value of the whole image. The local window is assumed to be unaffected by the gradual variation of intensity between the image centres and edges. The point transformation distribution is localised around the mean intensity of the window and it covers the entire intensity range of the image.

Consider a running sub image W of N X N pixels centred on a pixel P (i,j) , the image is filtered to produced another sub image P of (N X N) pixels according to the equation below

Where

and *Max* and *Min* are the maximum and minimum intensity values in the whole image, while and indicate the local window mean and standard deviation which are defined as:

As a result of this adaptive histogram equalization, the dark area in the input image that was badly illuminated has become brighter in the output image while the side that was highly illuminated remains or reduces so that the whole illumination of the image is same.

**MODULE 4**

**IMAGE SEGMENTATION**

The grayscale image was converted into binary image using thresholding level. This process is known as binarization. For computing the thresholding level, we implement otsu thresholding algorithm..

**THRESHOLDING**

Binarization is a process where each pixel in an image is converted into one bit and you assign the value as '1' or '0' depending upon the mean value of all the pixel. If greater then mean value then its '1' otherwise its '0'.

Image binarization converts an image of up to 256 gray levels to a black and white image. The simplest way to use image binarization is to choose a threshold value, and classify all pixels with values above this threshold as white, and all other pixels as black. The problem then is how to select the correct threshold. In many cases, finding one threshold compatible to the entire image is very difficult, and in many cases even impossible. Therefore, adaptive image binarization is needed where an optimal threshold is chosen for each image area.

**MODULE 5**

**MORPHOLOGICAL OPERATIONS**

The morphological operations are used to remove unwanted objects in the image. This is done to eliminate the background portion in the image and to extract only the vehicle objects. This process is also known as ROI extraction

**MODULE 6**

**OBJECT COUNTING**

Each vehicle is considered as one object. After the image segmentation and morphological operation process, the objects at each lane will be present. We need to count the number of objects using object counting method.

**MODULE 7**

**DECISION MAKING**

Based on the number of object counts of each lane, we are going to make the decision such that allowing the vehicles to move forward in particular lane, where the number of objects is more. The GREEN light is visible for that lane, and for the rest of the lane, RED light will be visible.

**CHAPTER 3**

**SOFTWARE SPECIFICATION**

**3.1 general**

**MATLAB** (**mat**rix **lab**oratory) is a numerical computing environment and [fourth-generation programming language](http://en.wikipedia.org/wiki/Fourth-generation_programming_language). Developed by Math Works, MATLAB allows [matrix](http://en.wikipedia.org/wiki/Matrix_(mathematics)) manipulations, plotting of [functions](http://en.wikipedia.org/wiki/Function_(mathematics)) and data, implementation of [algorithms](http://en.wikipedia.org/wiki/Algorithm), creation of [user interfaces](http://en.wikipedia.org/wiki/User_interface), and interfacing with programs written in other languages, including [C](http://en.wikipedia.org/wiki/C_(programming_language)), [C++](http://en.wikipedia.org/wiki/C%2B%2B), [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), and [Fortran](http://en.wikipedia.org/wiki/Fortran).

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) [symbolic engine](http://en.wikipedia.org/wiki/Computer_algebra_system), allowing access to [symbolic computing](http://en.wikipedia.org/wiki/Symbolic_computing) capabilities. An additional package, [Simulink](http://en.wikipedia.org/wiki/Simulink), adds graphical multi-domain simulation and [Model-Based Design](http://en.wikipedia.org/wiki/Model_based_design) for [dynamic](http://en.wikipedia.org/wiki/Dynamical_system) and [embedded systems](http://en.wikipedia.org/wiki/Embedded_systems).

In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of [engineering](http://en.wikipedia.org/wiki/Engineering), [science](http://en.wikipedia.org/wiki/Science), and [economics](http://en.wikipedia.org/wiki/Economics). MATLAB is widely used in academic and research institutions as well as industrial enterprises.

MATLAB was first adopted by researchers and practitioners in [control engineering](http://en.wikipedia.org/wiki/Control_engineering), Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of [linear algebra](http://en.wikipedia.org/wiki/Linear_algebra) and [numerical analysis](http://en.wikipedia.org/wiki/Numerical_analysis), and is popular amongst scientists involved in [image processing](http://en.wikipedia.org/wiki/Image_processing). The MATLAB application is built around the MATLAB language. The simplest way to execute MATLAB code is to type it in the Command Window, which is one of the elements of the MATLAB Desktop. When code is entered in the Command Window, MATLAB can be used as an interactive mathematical [shell](http://en.wikipedia.org/wiki/Shell_(computing)). Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a [script](http://en.wikipedia.org/wiki/Shell_script) or encapsulated into a [function](http://en.wikipedia.org/wiki/Functional_programming), extending the commands available.

MATLAB provides a number of features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

**3.2 features of matlab**

* High-level language for technical computing.
* Development environment for managing code, files, and data.
* Interactive tools for iterative exploration, design, and problem solving.
* Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration.
* 2-D and 3-D graphics functions for visualizing data.
* Tools for building custom graphical user interfaces.
* Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++, Fortran, Java™, COM, and Microsoft Excel.

MATLAB is used in vast area, including signal and image processing, communications, control design, [test and measurement](http://www.mathworks.in/applications/t_m), financial modeling and analysis, and computational. Add-on toolboxes (collections of special-purpose MATLAB functions) extend the MATLAB environment to solve particular classes of problems in these application areas.

MATLAB can be used on personal computers and powerful server systems, including the [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha) compute cluster. With the addition of the Parallel Computing Toolbox, the language can be extended with parallel implementations for common computational functions, including for-loop unrolling. Additionally this toolbox supports offloading computationally intensive workloads to [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha) the campus compute cluster. MATLAB is one of a few languages in which each variable is a matrix (broadly construed) and "knows" how big it is. Moreover, the fundamental operators (e.g. addition, multiplication) are programmed to deal with matrices when required. And the MATLAB environment handles much of the bothersome housekeeping that makes all this possible. Since so many of the procedures required for Macro-Investment Analysis involves matrices, MATLAB proves to be an extremely efficient language for both communication and implementation.

**3.2.1 INTERFACING WITH OTHER LANGUAGES**

MATLAB can call functions and subroutines written in the [C programming language](http://en.wikipedia.org/wiki/C_(programming_language)) or [FORTRAN](http://en.wikipedia.org/wiki/Fortran). A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "[MEX-files](http://en.wikipedia.org/wiki/MEX_file)" (for **M**ATLAB **ex**ecutable).

Libraries written in [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), [ActiveX](http://en.wikipedia.org/wiki/ActiveX) or [.NET](http://en.wikipedia.org/wiki/.NET_Framework) can be directly called from MATLAB and many MATLAB libraries (for example [XML](http://en.wikipedia.org/wiki/XML) or [SQL](http://en.wikipedia.org/wiki/SQL) support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with MATLAB extension, which is sold separately by Math Works, or using an undocumented mechanism called JMI (Java-to-Mat lab Interface), which should not be confused with the unrelated Java that is also called JMI.

As alternatives to the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) based Symbolic Math Toolbox available from Math Works, MATLAB can be connected to [Maple](http://en.wikipedia.org/wiki/Maple_(software)) or [Mathematica](http://en.wikipedia.org/wiki/Mathematica" \o "Mathematica).

Libraries also exist to import and export [MathML](http://en.wikipedia.org/wiki/MathML" \o "MathML).

**Development Environment**

* Startup Accelerator for faster MATLAB startup on Windows, especially on Windows XP, and for network installations.
* [Spreadsheet Import Tool](http://www.mathworks.in/videos/matlab/new-spreadsheet-import-tool-in-r2011b.html?type=shadow) that provides more options for selecting and loading mixed textual and numeric data.
* Readability and navigation improvements to warning and error messages in the MATLAB command window.
* [Automatic variable and function renaming](http://www.mathworks.in/videos/matlab/new-automatic-variable-and-function-renaming-in-r2011b.html?type=shadow) in the MATLAB Editor.

**Developing Algorithms and Applications**

MATLAB provides a high-level language and development tools that let you quickly develop and analyze your algorithms and applications.

**The MATLAB Language**

The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory. In many cases, MATLAB eliminates the need for ‘for’ loops. As a result, one line of MATLAB code can often replace several lines of C or C++ code.

At the same time, MATLAB provides all the features of a traditional programming language, including arithmetic operators, flow control, data structures, data types, object-oriented programming (OOP), and debugging features.

MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution. For fast execution of heavy matrix and vector computations, MATLAB uses processor-optimized libraries. For general-purpose scalar computations, MATLAB generates machine-code instructions using its JIT (Just-In-Time) compilation technology.

This technology, which is available on most platforms, provides execution speeds that rival those of traditional programming languages.

Development Tools

MATLAB includes development tools that help you implement your algorithm efficiently. These include the following:

**MATLAB Editor**

Provides standard editing and debugging features, such as setting breakpoints and single stepping

**Code Analyzer**

Checks your code for problems and recommends modifications to maximize performance and maintainability

**MATLAB Profiler**

Records the time spent executing each line of code

**Directory Reports**

Scan all the files in a directory and report on code efficiency, file differences, file dependencies, and code coverage

**Designing Graphical User Interfaces**

By using the interactive tool GUIDE (Graphical User Interface Development Environment) to layout, design, and edit user interfaces. GUIDE lets you include list boxes, pull-down menus, push buttons, radio buttons, and sliders, as well as MATLAB plots and Microsoft ActiveX® controls. Alternatively, you can create GUIs programmatically using MATLAB functions.

**3.2.2 ANALYZING AND ACCESSING DATA**

MATLAB supports the entire data analysis process, from acquiring data from external devices and databases, through preprocessing, visualization, and numerical analysis, to producing presentation-quality output.

**Data Analysis**

MATLAB provides interactive tools and command-line functions for data analysis operations, including:

* Interpolating and decimating
* Extracting sections of data, scaling, and averaging
* Thresholding and smoothing
* Correlation, Fourier analysis, and filtering
* 1-D peak, valley, and zero finding
* Basic statistics and curve fitting
* Matrix analysis

**Data Access**

MATLAB is an efficient platform for accessing data from files, other applications, databases, and external devices. You can read data from popular file formats, such as Microsoft Excel; ASCII text or binary files; image, sound, and video files; and scientific files, such as HDF and HDF5. Low-level binary file I/O functions let you work with data files in any format. Additional functions let you read data from Web pages and XML.

**Visualizing Data**

All the graphics features that are required to visualize engineering and scientific data are available in MATLAB. These include 2-D and 3-D plotting functions, 3-D volume visualization functions, tools for interactively creating plots, and the ability to export results to all popular graphics formats. You can customize plots by adding multiple axes; changing line colors and markers; adding annotation, Latex equations, and legends; and drawing shapes.

**2-D Plotting**

Visualizing vectors of data with 2-D plotting functions that create:

* Line, area, bar, and pie charts.
* Direction and velocity plots.
* Histograms.
* Polygons and surfaces.
* Scatter/bubble plots.
* Animations.

**3-D Plotting and Volume Visualization**

MATLAB provides functions for visualizing 2-D matrices, 3-D scalar, and 3-D vector data. You can use these functions to visualize and understand large, often complex, multidimensional data. Specifying plot characteristics, such as camera viewing angle, perspective, lighting effect, light source locations, and transparency.

3-D plotting functions include:

* Surface, contour, and mesh.
* Image plots.
* Cone, slice, stream, and isosurface.

**3.2.3 PERFORMING NUMERIC COMPUTATION**

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed by experts in mathematics, are the foundation of the MATLAB language. The core math functions use the LAPACK and BLAS linear algebra subroutine libraries and the FFTW Discrete Fourier Transform library. Because these processor-dependent libraries are optimized to the different platforms that MATLAB supports, they execute faster than the equivalent C or C++ code.

MATLAB provides the following types of functions for performing mathematical operations and analyzing data:

* Matrix manipulation and linear algebra.
* Polynomials and interpolation.
* Fourier analysis and filtering.
* Data analysis and statistics.
* Optimization and numerical integration.
* Ordinary differential equations (ODEs).
* Partial differential equations (PDEs).
* Sparse matrix operations.

MATLAB can perform arithmetic on a wide range of data types, including doubles, singles, and integers.

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 GENERAL**

Matlab is a program that was originally designed to simplify the implementation of numerical linear algebra routines. It has since grown into something much bigger, and it is used to implement numerical algorithms for a wide range of applications. The basic language used is very similar to standard linear algebra notation, but there are a few extensions that will likely cause you some problems at first.

**4.2 CODE IMPLEMENTATION**

%%%%% load the image %%%%%

[filename pathname]=uigetfile( {'\*.png'; '\*.bmp';'\*.tif';'\*.jpg'});

x=imread([pathname filename]); %%%% read the image %%%%%

figure,imshow(x,[]); %%%% show the image %%%%%

title('ORIGINAL IMAGE');

[row col dim]=size(x);

[filename pathname]=uigetfile( {'\*.png'; '\*.bmp';'\*.tif';'\*.jpg'});

y=imread([pathname filename]); %%%% read the image %%%%%

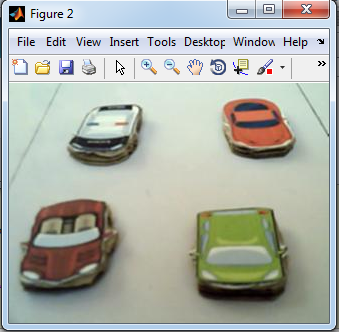
figure,imshow(y,[]); %%%% show the image %%%%%

title('ORIGINAL IMAGE');

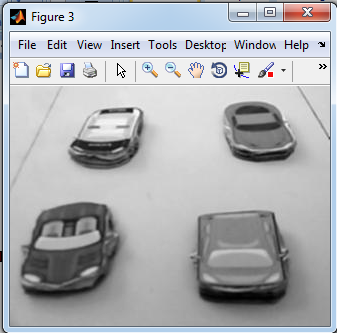
[row1 col1 dim1]=size(y);

**4.3 SNAPSHOTS**

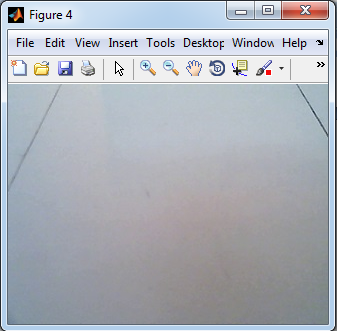
**INPUT IMAGE-LANE 1**

****

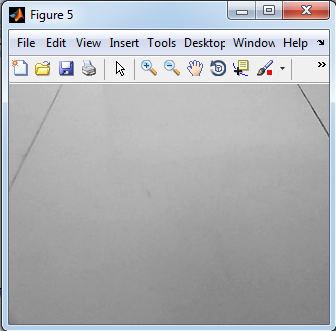
**GRAYSCALE IMAGE**

****

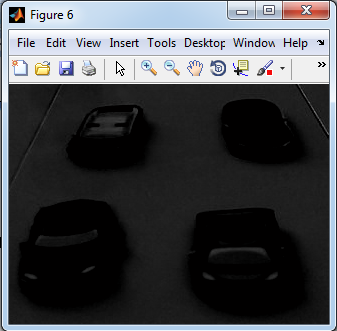
**BACKGROUND IMAGE**

****

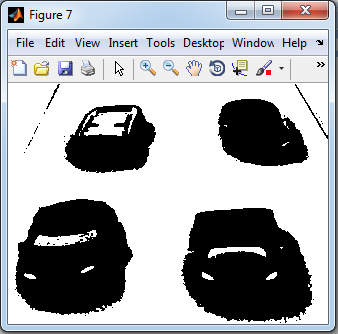
**BACKGROUND IMAGE-GRAYSCALE**

****

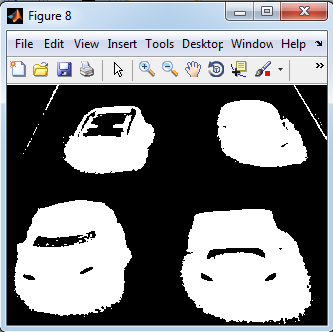
**BACKGROUND SUBTRACTED IMAGE**

****

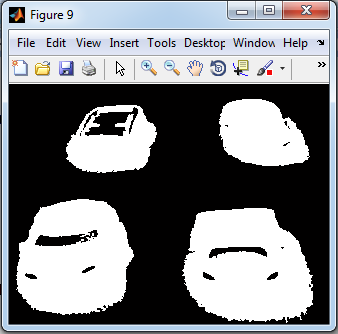
**THRESHOLDED IMAGE**

****

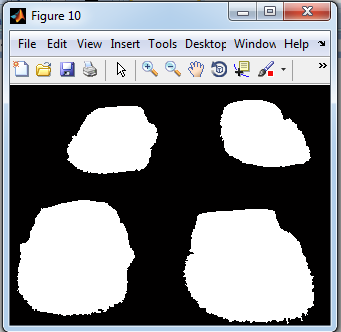
**INVERSE THRESHOLDED IMAGE**

****

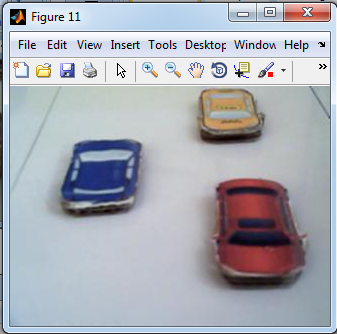
**UNWANTED OBJECT REMOVED IMAGE**

****

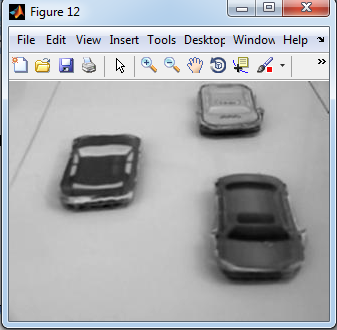
**SEGMENTED VEHICLES IN IMAGE**

****

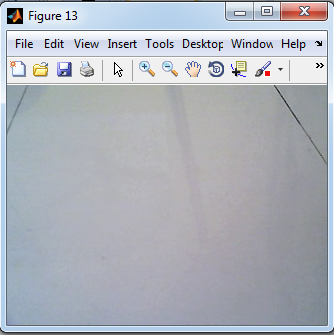
**INPUT IMAGE-LANE 2**

****

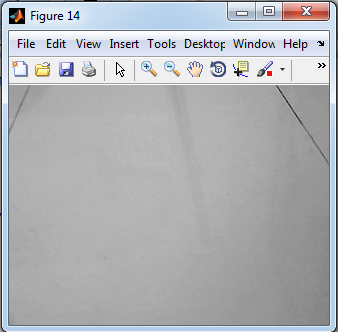
**GRAYSCALE IMAGE**

****

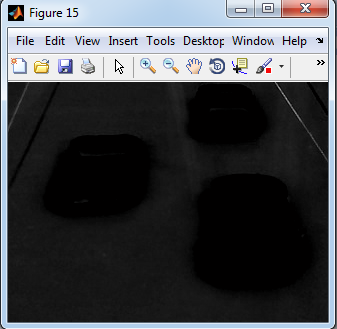
**BACKGROUND IMAGE**

****

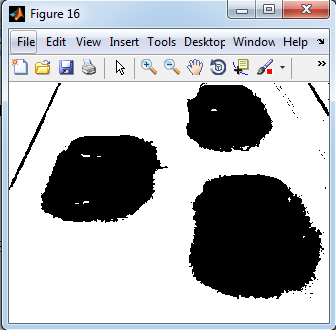
**BACKGROUND IMAGE-GRAYSCALE**

****

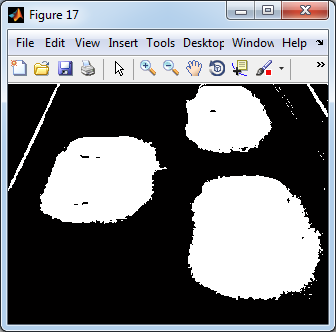
**BACKGROUND SUBTRACTED IMAGE**

****

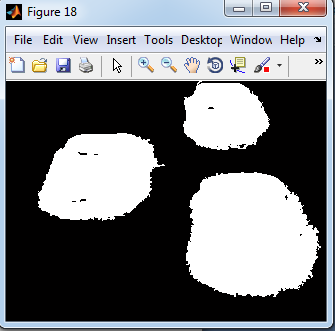
**THRESHOLDED IMAGE**

****

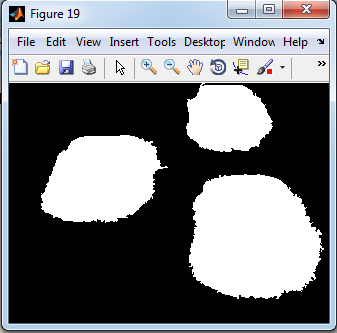
**INVERSE THRESHOLDED IMAGE**

****

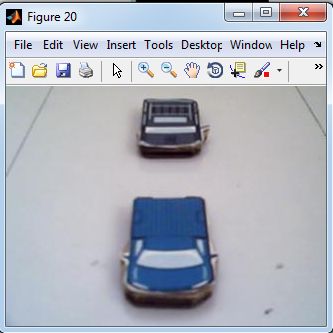
**UNWANTED OBJECT REMOVED IMAGE**

****

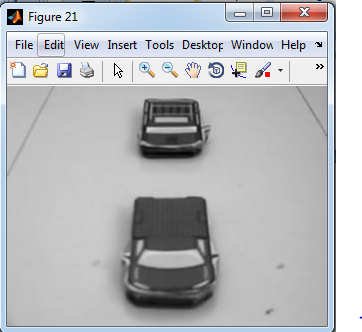
**SEGMENTED VEHICLES IN IMAGE**

****

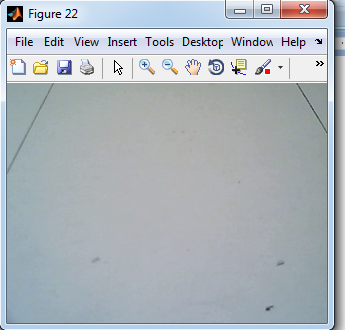
**INPUT IMAGE-LANE 3**

****

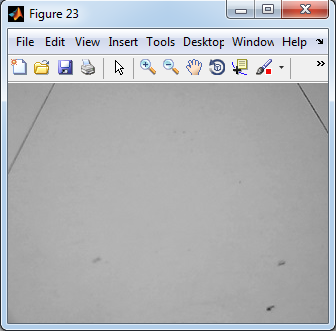
**GRAYSCALE IMAGE**

****

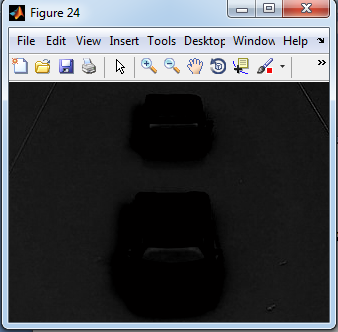
**BACKGROUND IMAGE**

****

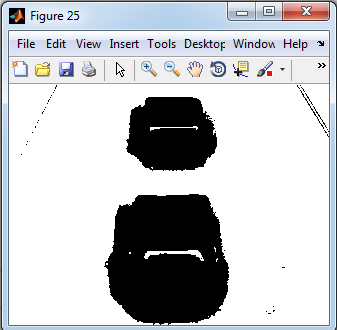
**BACKGROUND IMAGE-GRAYSCALE**

****

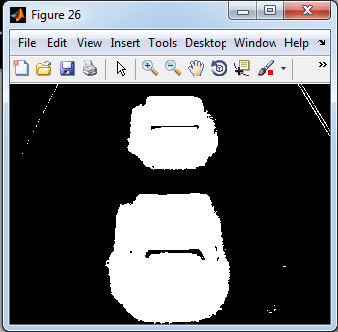
**MEAN SUBTRACTED IMAGE**

****

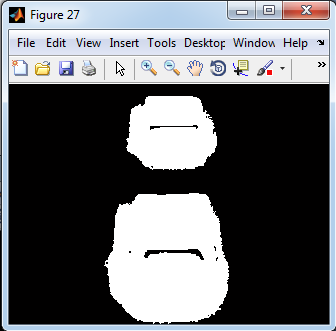
**THRESHOLDED IMAGE**

****

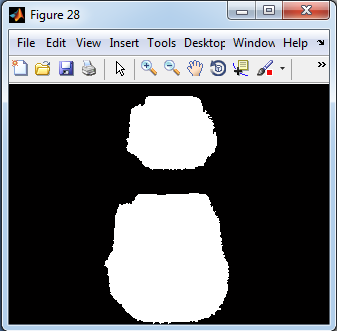
**INVERSE THRESHOLDED IMAGE**

****

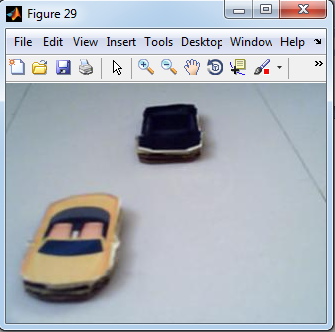
**OBJECT REMOVED IMAGE**

****

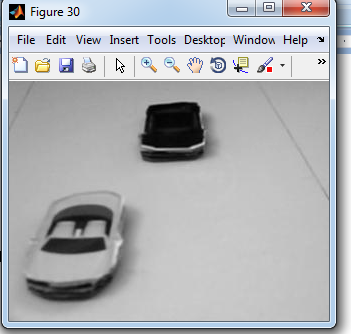
**SEGMENTED VEHICLES IN IMAGE**

****

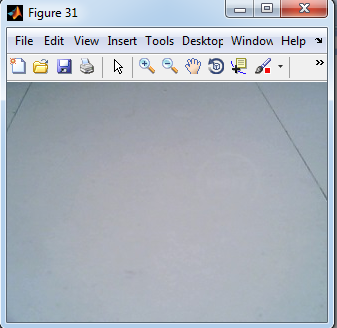
**INPUT IMAGE-LANE 4**

****

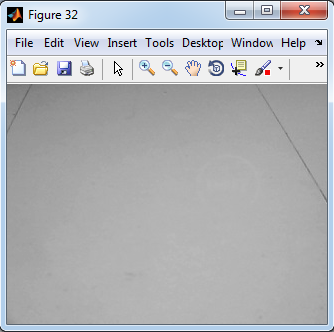
**GRAYSCALE IMAGE**

****

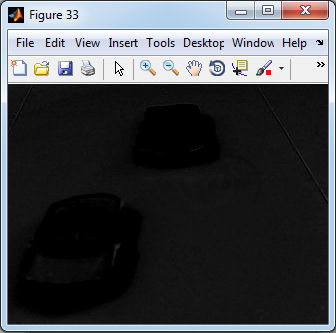
**BACKGROUND IMAGE**

****

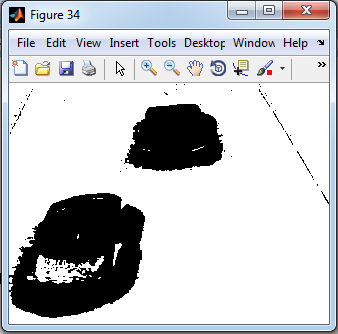
**BACKGROUND IMAGE-GRAYSCALE**

****

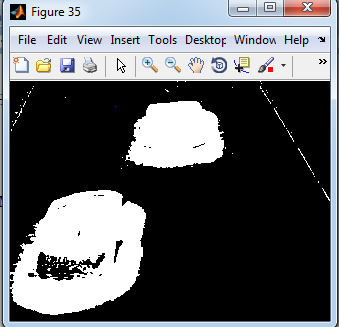
**MEAN SUBTRACTED IMAGE**

****

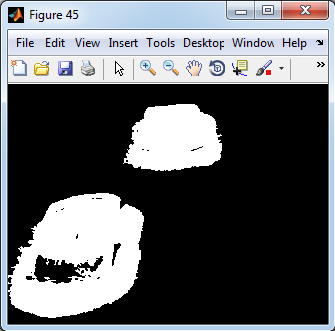
**THRESHOLDED IMAGE**

****

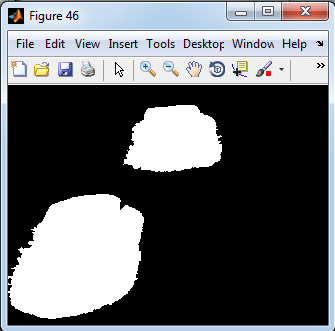
**INVERSE THRESHOLDED IMAGE**

****

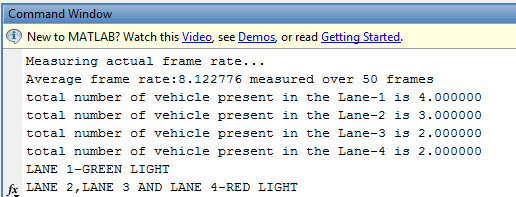
**OBJECT REMOVED IMAGE**

****

**SEGMENTED VEHICLES IN IMAGE**

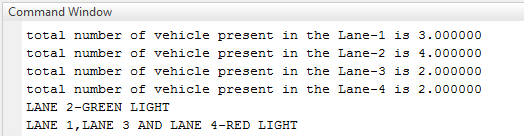
****

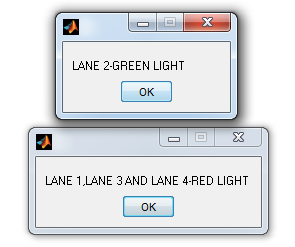
**OUTPUT-LANE1 MAX VEHICLE**

****

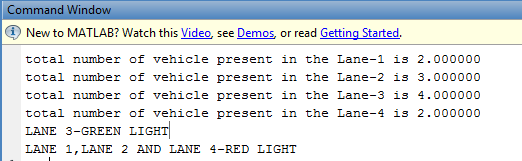
****

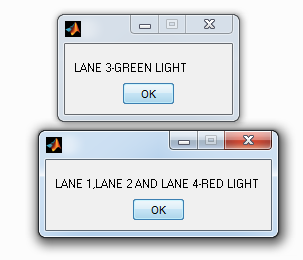
**OUTPUT-LANE2 MAX VEHICLE**

****

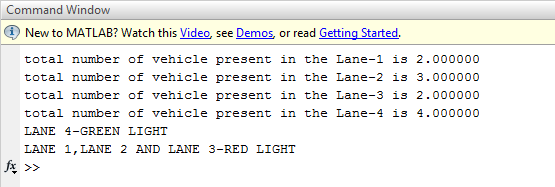
****

**OUTPUT-LANE3 MAX VEHICLE**

****

****

**OUTPUT-LANE4 MAX VEHICLE**

****

****

**CHAPTER 6**

**CONCLUSION AND REFERENCES**

**6.1 CONCLUSION**

This paper has proposed an automatic traffic control system and smart traffic optimization using image processing. The objective of the proposed system is to design an approach which automatically detects and counts the vehicles. The lane images of all the 4 sides of the road are captured using camera simultaneously. We need to eliminate the background objects by using thresholding and background subtraction method and morphological operations. Vehicles counting process results in density estimation of the traffic at regular interval of time. This is achieved through object counting method. The number of objects will be counted for each lane and compared. Among the 4 lanes, the lane with more number of vehicles will be allowed to move forward after receiving the Green signal and the other 3 lanes with least number of vehicles will be stopped at their respective places. The red light is passed at this condition. The information such as LANE-1 ON, LANE-2, 3, 4-OFF will be passed to the microcontroller. The proposed approach is suitable for real-time applications**.** The control was moved to hardware via USB to RS-232 converter.

**6.2 REFERENCES**

# [1] Pallavi choudekar, Sayanti banerjee “Implementation of image processing in real time traffic light control,” 3rd International conference on Electronics Computer Technology (ICECT), 2011.

[2] V.parthasarathi and M.surya, “Smart control of traffic signal using image processing,” Indian journal of science and technology (IJST), Vol.8, No.16, July 2015.

[3] Chandrasekar and saikrishna, “Traffic control using digital image processing,” International journal of Advanced Electrical and Electronics Engineering (IJAEEE), Vol.2, No.5, 2013.

[4] Omkar Ramdas Gaikwad and Anil vishwas Rao, “Image processing based Traffic light control,” International journal of Science, Engineering and Technology Research (IJSETR), Vol.3, No.4, April 2014.

[5] Pallavi choudekar, Sayanti banerjee “Real time traffic light control using image processing,” 3rd International journal of computer science and engineering (IJCSE), 2011.