MOVING VEHICLE NUMBER PLATE DETECTION FOR TRAFFIC CONTROL

A Project report submitted in partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

M.Manikanta (315126512107) N.Sandeep(315126512117)

L.Rajani (315126512083) M.Sajida (315126512101)

P.Revanth(315126512134)

Under The Esteemed Guidance Of

V.V.K.RAJU(B.Tech,M.Tech.,(Ph.D))

ASSISTANT PROFESSOR

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, bheemili mandal, visakhapatnam dist.(A.P)

2018-2019

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)
(*Affiliated to Andhra University, Approved by AICTE and Accredited by NBA)*
Sangivalasa, Bheemilimandal, Visakhapatnam Dist.(A.P)

PRAGNANAM BRAHMA

ANITS

CERTIFICATE

*This is to certify that the project report entitled***"Moving Vehicle Number Plate Detection For Traffic Control"**submitted by **M.Manikanta (315126512107), L.Rajani (315126512083), N.Sandeep (315126512117), M.Sajida (315126512101), P.Revanth (315126512134)**in partial fulfillment of the requirements forthe award of the degree of **"Bachelor of Technology"**in **Electronics & Communication Engineering** ofAndhra University, Visakhapatnam is a record of bonafide work carried out under my guidanceand supervision.

Project Guide	**Head of the Department**
V.V.K.RAJU	**Dr. V.Rajyalakshmi**
B.Tech,M.Tech,(Ph.D)	M.E,Ph.D,MIEEE,MIE,MIETE
Department of E.C.E	Department of E.C.E
ANITS	ANITS

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **V.V.K.Raju, Assistant professor**, Department of Electronics and Communication Engineering, ANITS, for his guidance with unsurpassed knowledge and immense encouragement. We are grateful to **Dr. V. Rajyalakshmi**, Head of the Department, Electronics and Communication Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa,** for their encouragement and cooperation to carry out this work.

We express our thanks to all **teaching faculty** of Department of ECE, whose suggestions during reviews helped usin accomplishment of our project. We would like to thank **all non-teaching staff** of the Department of ECE, ANITSfor providing great assistance in accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

M.Manikanta(315126512107),
L.Rajani(315126512083),
N.Sandeep(315126512117),
M.Sajida(315126512101),
P.Revanth(315126512134)

ABSTRACT

Basically video surveillance system is used for security purpose as well as monitoring systems. ButDetection of moving object is a challenging part of video surveillance. Video surveillance system is used for Home security, Military applications, Banking /ATM security, Traffic monitoring etc. Now a day's due to decreasing costs of high quality video surveillance systems, human activity detection and tracking has become increasingly in practical. Accordingly, automated systems have been designed for numerous detection tasks, but the task of detecting illegally parked vehicles has been left largely to the human operators of surveillance systems. The detection of Indian vehicles by their number plates is the most interesting and challenging research topic from past few years. It is observed that the number plates of vehicles are in different shape and size and also have different colour in various countries.

This project proposes a method for the detection and identification of vehicle number plate that will help in the detection of number plates of authorized and unauthorized vehicles. This project presents an approach based on simple but efficient morphological operation and Sobel edge detection method. This approach is simplified to segmented all the letters and numbers used in the number plate by using bounding box method. After segmentation of numbers and characters present on number plate, template matching approach is used to recognition of numbers and characters. The concentrate is given to locate the number plate region properly to segment all the number and letters to identify each number separately.

Keywords: Number Plate Recognition (NPR), Support Vector Machine (SVM), Automated Teller Machine

LIST OF ABBREVVATIONS

1.ALPR-Automatic License Plate Recognition.

2.MATLAB-Matxix Laboratory.

3.FFT-Fast Fourier Transform.

4.JPEG-Joint Picture Expert Group.

5.ASCII-American Standard Code for Information Interchange.

6.HDR-High Dynamic Range.

7.RGB-Red Green Blue.

8.HSV-Hue Saturation Value.

9.OCR-Optical character Recognition.

10.ANN-Artificial Neural Network.

11.BPNN-Back Propagation Neural Network.

12.BPA-Back Propagation Algorithm.

13.CCD-Charge Couple Device.

14.CMOS-Complementary Metal Oxide Semiconductor.

15.IR-Infra Red.

16.ROI-Region Of Interest.

17.MLP-Multi Layer Perceptron.

List of figures

CONTENTS

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 Introduction:

License Plate Recognition (LPR) is a combination of image processing, character segmentation and recognition technologies used to identify vehicles by their license plates. Since only the license plate information is used for identification, this technology requires no additional hardware to be installed on vehicles. LPR technology is constantly gaining popularity, especially in security and traffic control systems. License Plate Recognition Systems are utilized frequently for access control in buildings and parking areas, law enforcement, stolen car detection, traffic control, automatic toll collection and marketing research.

LPR applications apply image processing and segmentation algorithms for license plate extraction, and each operation involves lots of computation. Government regulations standards employed in the license plates can reduce the computational requirements substantially and improve the accuracy. Constraints contain range of values instead of exact measures, since the license plate text size, style and orientation can vary substantially in different images.

The license plate recognition systems have two main points: the quality of license plate recognition software with recognition algorithms used and the quality of imaging technology, including camera and lighting.

Elements to be considered: maximum recognition accuracy, achieve faster processing speed, handling as many types of plates, manage the broadest range of image qualities and achieve maximum distortion tolerance of input data.

Ideally, for extreme conditions and with serious problems of normal visibility, would have special cameras ready for such an activity, such as infrared cameras that are much better to address these goals and achieve better results. This is because the infrared illumination causes reflection of light on the licence plate made of special material which causes a different light in that area of the image relative to the rest of it, causing it to be easier to detect.

1.2 Elements of Typical LPR System:

LPR systems normally consist of the following units:

Camera:-It takes image of a vehicle from cither front or rear end

Computer: Normally a PC, It runs the LPR application that controls the images, analyses and identifies the plate, and interfaces with other applications and systems.

Software: -The Application and the recognition package(MATLAB)

1.3 Problem Definition:

Automatic recognition of license plates requires several image processing and algorithms to be utilized within a single application. Text localization, extraction and enhancement, character segmentation and recognition operations are used to determine the license plate number in a given image or video frame. Only a few of the previous studies involved all the steps of a typical LPR system, from image acquisition to verification. In this research, a complete license plate recognition system, which is based on constraints and operates in real time, has been designed and implemented.

License plate localization and extraction are the most time consuming stage of a typical system. Assumptions as well as optimizations are required in order for LPR systems to be able to locate license plates in real time. However, the computational requirements increase in parallel. To minimize this side-effect, constraints and prior knowledge are utilized. After extracting the license plate area, the resulting region is further processed for character segmentation and recognition.

1.4 Overall description about project:

1.4.1 Structure of the Proposed System

The system presented is designed to recognize license plates from vehicle. Input to the system is an image acquired by a camera that consists of a license plate and its output is the recognition of characters on the license plate in a separate notepad window.

The system consists of the standard six main modules in an LPR system, viz. Estimation of vehicle speed, Image acquisition, License plate extraction, License plate segmentation and Licence plate recognition. The first task acquires the image. The second task extracts the region that contains the license plate. The third task isolates the characters, letters and numerals (total of 10 digits), as in the case of Indian License Plates. The last task identifies or recognizes the segmented characters.

1.4.2 Background Subtraction Method

Background subtraction is a useful and effective method for detecting moving objects in video images. Since this method assumes that image variations are caused only by moving objects (i.e., the background scene is assumed to be stationary), however, its applicability is limited. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called "background image", or" background model". Background subtraction is mostly done if the image in question is a part of a video stream. There are many challenges in developing a good background subtraction algorithm. First, it must be robust against changes in illumination. Second, it should avoid detecting non-stationary background objects such as moving leaves, rain, snow, and shadow cast by moving objects. Finally, its internal background model should react quickly to changes in background such as starting and stopping of vehicles.

1.4.3 Optical Flow Method:

Optical flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene. Optical flow is used for the perception of movement by the observer in the world; perception of the shape, distance and movement of objects in the world and the control of locomotion. Sequences of ordered images allow the estimation of motion as either instantaneous image velocities or discrete image displacements i.e, it corresponds to the movement of pixels in an image. The optical flow methods try to calculate the motion between two image frames which are taken at times t and t+at at every vowel position. However, the methods used in optical flow are very slow and have unsharp boundaries with many errors.

1.4.4 Adaptive Contrast Change Detection:

Adaptive change detection improves change detection by combining both the temporal difference method and the background subtraction method into one algorithm with the advantages of one of the algorithms compensating for the disadvantages of the other. Since, temporal difference method detects only the continuously moving objects and fails when an object stops, and due to the in efficiency of background detection method because of changing backgrounds and changing light conditions, the above method is useful to some extent. However, this method can be used only for static background cases.

1.4.5 Image Acquisition:

This is the first phase in an LPR system. This phase deals with acquiring an image by an acquisition method. In our proposed system, we used a digital camera to acquire the input image.

1.4.6 License Plate Extraction:

License Plate Extraction is a key step in an LPR system, which influences the accuracy of the system significantly. This phase extracts the region of interest, i.e, the license plate, from the acquired image. The proposed approach involves histogram based analysis for detecting the location of the license plate in the image acquired

1.4.7 License Plate Segmentation:

License Plate Segmentation, which is sometimes referred to as Character Isolation takes the region of interest and attempts to divide it into individual characters.

1.4.8 License Plate Recognition

Alter splinting the extracted License plate into individual character images, the character in each image can be identified. There are many methods used to recognize isolated characters. In this system we use Template matching was performed by using two sets of templates. One set was made up of perfect characters. Other set includes the segmented license plate characters. A correlation function was used to compare the match between the character segment and the templates in the database. The character that retuned the highest match was output as the recognized character.

BLOCK DIAGRAM :

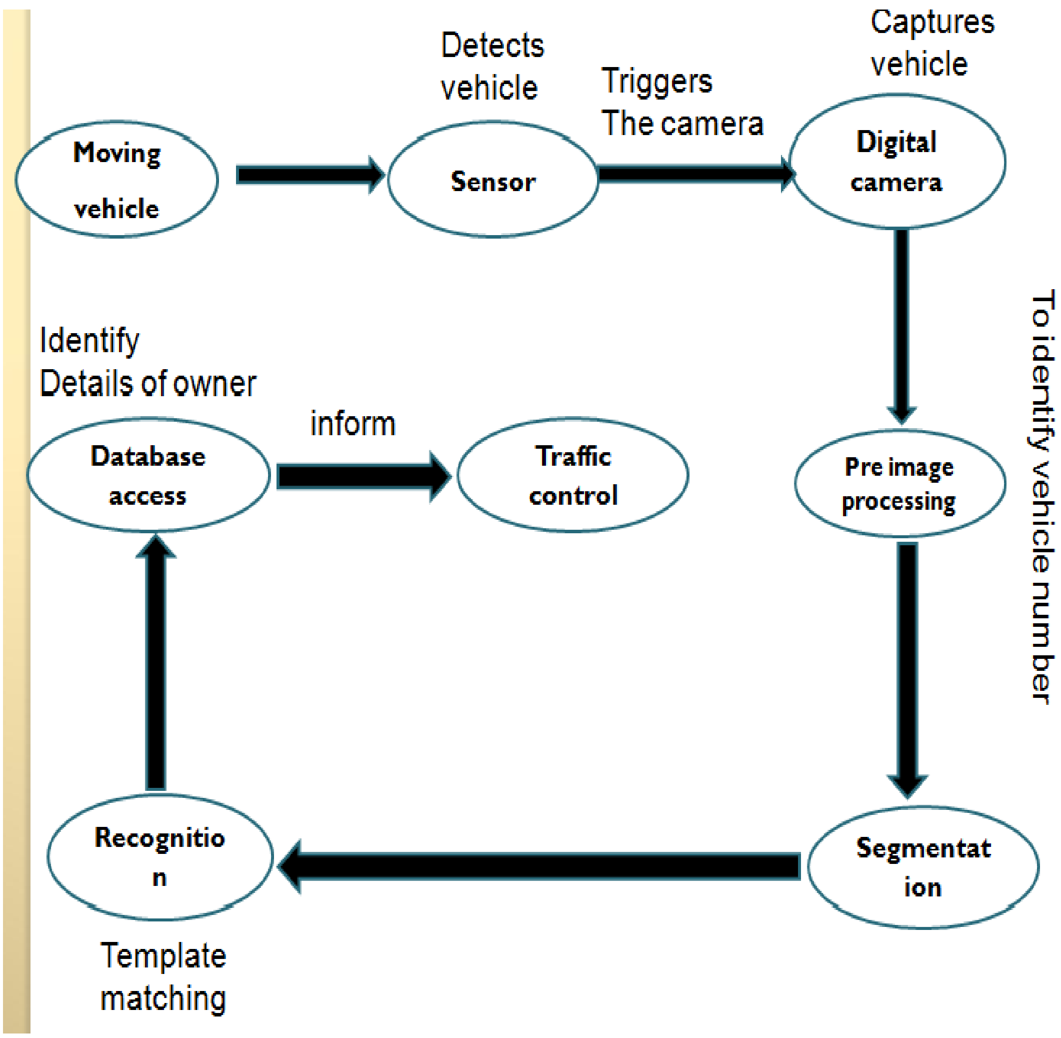


Figure: 1.4 Licence plate recognition

1.5 Application of LPR Systems:

There are numerous applications of license plate recognition systems for any given country. They include highway electronic toll collection, automatic parking attendant e.g. in banks, hotels, airports and fleet vehicle compounds, customer identification enabling personalized services e.g. in golf clubs, leisure centres, petrol station surveillance, speed limit enforcement and security among others. The following section illustrates the application of such systems:

1.5.1 Law Enforcement:

In this case the license plate is used to produce a violation fine on the speeding vehicles, illegal use of bus lanes, and detection of stolen or wanted vehicles. A speeding car's image can be caught by a traffic camera and the image sent to the system for processing to try and identify the particular car. The violators are presented with image of the vehicle as well as the speeding information as proof.

1.5.2 Parking:

The LPR system is used to automatically enter pre-paid members and calculate parking for non-members by comparing entry and exit times. So, a vehicle will be recognized as it enters the parking lot and it's information as well as time is stored. During exit, the car is recognized again and its fee is calculated.

CHAPTER 2

DIGITAL IMAGE PROCESSING

2. DIGITAL IMAGE PROCESSING

2.1 Introduction to Digital Image Processing

Digital image processing is the use of computer algorithms to create process, display digital images, Digital image processing algorithms can be used to:

- Convert signals from an image sensor into digital images
- Improve clarity, and remove noise and other artefacts
- Extract the size, scale, or number of objects in a scene
- Prepare images for display or printing
- Compress images for communication across a network

Image processing toolbox in MATLAB provides a comprehensive set of reference algorithms and graphical tools for image processing, analysis, visualization and algorithms development. It can restore the noisy and degraded images, enhance image, improve intelligibility, extract features, analyse shapes and textures. It has an ability to inspect algorithms, modify the source code and to create our own custom functions.

It supports engineers, scientists in the areas such as biometrics, remote sensing, microscopy semiconductor testing, image sensor design, colour science and material science.

2.2 Key Features:

- Image analysis, including segmentation, morphology, statistics, and measurement
- Image enhancement, filtering, and deblurring
- Geometric transformations and intensity-based image registration methods
- Image transforms, including FFT, DCT, Radon, and fan-beam projection
- Large image workflows, including block processing, tiling, and multi resolution display
- Visualization apps, including Image Viewer and Video Viewer
- Multi core and GPU-enabled functions, and C-code generation

2.1 Exploration and Discovery:

Image Processing Toolbox supports images and video generated by a wide range of devices, including webcams, digital cameras, satellites and airborne sensors, medical imaging devices, microscopes, telescopes, and other scientific instruments. Also these are functions to visualize, analyse, and process these images in many data types.

2.2 Standard and Specialized Vile Formats:

MATLAB supports standard data and image formats including: AVI, JPEG, M4V, MOV, MP4, PNO,TIFF, ASCII, Binary files

It also supports the multiband image formats BIP and BIL, as used by LANDSAT. Low-level I/O und memory mapping functions enable you to develop custom routines for working with any data format.

Image Processing Toolbox supports a number of specialized image file formats. For medical images, it supports DICOM files, including associated metadata.

2.2.3 Image Enhancement

Image enhancement techniques in Image Processing Toolbox enable to increase the signal-to-noise ratio and accentuate image features by modifying the colours or intensities of an image.

The toolbox includes specialized filtering routines and a generalized multidimensional filtering function that handles integer image types, offers multiple boundary padding options, and performs convolution and correlation.

Using predefined filters and functions there is a possibility to

- Filter with morphological operators
- De-blur and sharpen
- Remove noise with linear, median, or adaptive filtering
- Perform histogram equalization
- Remap the dynamic range
- Adjust the gamma value
- Adjust contrast

Specialized filtering routines and multi-dimensional filtering techniques are also present. Predefined filters and also the functions for designing and implementing the filters are present.

2.2.4 Image De-blurring:

Image de-blurring algorithms also include blind, Lucy-Richardson, Wiener, and regularized filter de-convolution, as well as conversions between point spread and optical transfer functions
These functions help correct blurring caused by out-of-focus optics, movement by the camera or the subject during image capture, atmospheric conditions, short exposure time, and other factors. All de-blurring functions work with multidimensional images.

2.2.5 Image Analysis:

Image analysis is the process of extracting meaningful information from images such as finding shapes, counting objects, identifying colours, or measuring object properties.

Image Processing Toolbox provides a comprehensive suite of reference -standard algorithms and visualization functions for image analysis tasks such as statistical analysis, feature extraction and property measurement.

2.2.7 Device-Independent Color Management:

Device-independent Color management enables to accurately represent color independently from input and output devices.

This is useful when analysing the characteristics of a device, quantitatively measuring color accuracy, or developing algorithms for several different devices. With specialized functions in the toolbox, you can convert images between device-independent color spaces, such as RGB, XYZ,L*a*b*, uvL and L*ch.

2.3 Image Conversions:

Image related applications often require the conversion between the data classes and image types. There are different types of images supported by MATLAB:

2.3.1 Indexed Images:

An indexed image consists of a data matrix, X, and a color map matrix, map. Map is an m-by-3array of class double containing floating-point values in the range [0, 1]. Each row of map specifies the red, green, and blue components of a single color. An indexed image uses "direct mapping" of pixel values to color map values.

The relationship between the values in the image matrix and the color map depends on the class of the image matrix. If the image matrix is of class double, the value 1 points to the first row in the color map, the value 2 points to the second row, and so on. If the image matrix is of classuint8 or uint16, there is an offset- the value 0 points to the first row in the color map, the value 1points to the second row, and so on. The offset is also used in graphics file formats to maximize the number of colors that can be supported. In the preceding image, the image matrix is of class double.

2.3.2 Intensity Images:

An intensity image is a data matrix, whose values represent intensities within some range. An intensity image is represented as a single matrix, with each element of the matrix corresponding to one image pixel. The matrix can be of class double, uint8,or uintl6. While intensity images are rarely saved with a color map, a color map is still used to display them. In essence, handles intensity images are treated as indexed images.

2.3.3 RGB (True color) Images

An RGB image, sometimes referred to as a true color image, is stored as an m-by-n-by-3 data array that defines red, green, and blue color components for each individual pixel. RGB images do not use a palette. The color of each pixel is determined by the combination of the rod, green, and blue intensities stored in each color plane at the pixel's location. Graphics file formats store RGB images as 24-bit images, where the red, green, and blue components are 8 bits each. This yields a potential of 16 million colors.

2.4 Statistical Functions:

Statistical functions let you analyse the general characteristics of an image by:

- Computing the mean or standard deviation
- Displaying an image histogram.

2.5 Image Segmentation:

Image segmentation is the process of dividing an image into multiple parts. This is typically used to identify objects or other relevant information in digital images. There are many different ways to perform image segmentation, including:

- Thresholding methods such as Otsu's method
- Color-based segmentation such as K-means clustering

2.6 Template matching:

All motorised road vehicles in India are tagged with a registration or license number. The Vehicle registration plate (commonly known as **number plate**) number is issued by the district-level Regional Transport Office (RTO) of respective states — the main authority on road matters. The number plates are placed in the front and back of the vehicle. By law, all plates are required to be in modern Hindu-Arabic numerals with Latin letters. Other guidelines include having the plate lit up at night and the restriction of the fonts that could be used. In some states such as Sikkim, cars bearing outside plates are barred from entering restricted areas. The international vehicle registration code for India is **IND**.

Example:

Figure: 2.1 standard number plate for private vehicles

Figure : 2.2 standard number plate for commercial vehicles

2.1 Standard Templates:

Figure :2.3 Standard Templates For Numeric Characters

Figure :2.4 Standard Templates For Alphabets

CHAPTER 3

ESTIMATION OF VELOCITY

3.1FRAMES TO VIDEO CONVERSION

The final RGB images obtained after the complete color image processing are to be converted into video. The final video contains, only the detected object.

3.2 VELOCTTY ESTIMATION:

Velocity of the moving objects in a video can be determined using several methods edge-preserving forms of model regularization, canny edge detection in which difference between images is considered to be the estimated velocity

In the early stages of prospect evaluation, an inexpensive interval velocity estimate is often desired The Dix equation (Dix, 1952) analytically inverts root-mean-square (RMS) velocity for interval velocity as a function of time. In addition to many physical shortcomings (assumption of a stratified (vz) earth), Dix inversion suffers from numerical problems that lead to poor velocity estimates Dix inversion is unstable when RMS velocities vary rapidly, and may produce interval velocities with unreasonably large and rapid variations For this reason, the problem is often cast as a least-squares problem, which is regularized in time with a differential operator to penalize rapid velocity variations and to produce a smooth result.

(Clapp et al., 1998). While temporal velocity smoothness may often be justified from a geological point of view, in some cases however, it can change abruptly (e.g., carbonate layers, salt bodies, strong faulting). In these situations the desire for a regularization technique that yields smooth velocities while preserving geologic interval velocity sharp contrasts. In addition, no pre-defined boundaries should be supplied.

In this project, finding the velocity of the object by considering the maximum of mean of the difference of resultant images per second obtained after performing edge detection on them. Among many edge detectors, Canny edge detection method is widely considered to be an ideal detector Canny edge detector finds edges by looking for local maxima of the gradient of optical image I. The gradient is calculated using the derivative of a Gaussian filtert7] 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 is therefore less likely than the others to be fooled by noise, and more likely to detect true weak edges.

Canny edge detector is widely considered as the ideal edge detector [3]. This is justified by the motivation underlying the developing of the algorithm. The motivation for Cranny's edge operator is to derive an 'optimal' operator in the sense that it

- Minimizes the probability of repeatedly detecting an edge;
- Minimizes the probability of failing to detect an edge;
- Minimizes the distance of the reported edge from the true edge.

The first two of these criteria address the issue of detection, the third criterion addresses the issue of localization, that is, how accurately the position of an edge is reported. The Canny edge detector is based on computing the squared gradient magnitude. The mathematics behind the whole optimization process is rather complex [9]. However, the optimal edge detector turns cut to have a simple approximate implementation: edges are detected by smoothing the image with a Gaussian low-pass filter and local maxima of the gradient magnitude that are above some threshold are then identified as edges. The low-pass filtering prior to calculating the gradients significantly contributes to reduction in noise sensitivity of the edge detector. The threshold local peak detection method is called non-maximum suppression, or NMS. The algorithm of canny edge detection can be summarized as the following steps of processing for a digital image I (x, y):

1. Smooth the image using a 2D Gaussian, Is=G*I

2. Compute the gradient and squared magnitude of the smoothed image

$$\nabla Is = (\partial Is/\partial x, \partial Is/\partial y)$$

$$M(x,y) = (\partial Is/\partial x)^2 + (\partial Is/\partial y)^2$$

3. Use the unit vector $\nabla Ix/|\nabla Ix|$ $(\partial x, \partial y)$ at each point to estimate the gradient magnitude in the gradient direction and opposite of the gradient direction. This is done by a weighted average of the neighbouring pixels in the direction$(\partial x,\partial y)$ or more simply by selecting the neighbouring pixel closest to the direction $(x+ \partial x, y+\partial y)$.

4. Let $p=m(x, y)$,$p+ = m(x+ \partial x, y+\partial y)$, and $p\_= m(x-\partial x, y-\partial y)$. Define a peak as $((p>p+)(p\geq p\_))V((P>P\_)(p\geq p+))$

5. Apply threshold to pick out strong peaks in order to get rid of little peaks due to noise, etc.

Using $|\nabla Is|$ as the measure of edge strength, a hysteresis mechanism with two thresholds on edge strength, lo and hi, is then carried out to threshold out weak points but retain weak points with connected edges. In practice, the convolution with a two-dimensional Gaussian in Step l is separated into two convolutions with one-dimensional Gaussians due to the rotationally symmetry property of Gaussian, this greatly reduces the implementation complexity. Therefore, the first two steps of the algorithm can be implemented as:

(I) Create ID Gaussian mask G

(2) Convolve the image I (x, y) with G along the rows to give the x component image I_x, and down the columns to give the y component I_y.

(3) Create a ID mask for the first derivative of the Gaussian in the x and y directions, resulting in G_x and G_y

4) Convolve I_x with G_x to give I_x^1, and convolve I_y with G_y to give $I_y^{1.}$

After performing edge detection on the images taken, the difference in the resultant images is obtained which is considered per second and the maximum of mean of the resultant is said to be the velocity of the required object. The velocity of moving object in the frames is defined in pixels per second. Thus, velocity of the object is determined using this method.

CHAPTER 4

FEASIBILITY OF PROJECT

4. FEASIBILITY STUDY

All the projects will be feasible, if given unlimited resources and infinite time. But development of software is plagued by the scarcity of resources and difficult delivery rates. It is both and prudent to evaluate the feasibility of the project at the earliest possible time.

4.1.Key Considerations

Three key considerations are involved in feasibility analysis.

4.1.1.Economic Feasibility

It determines whether the project goal can be within the resource limits allocated to it. It also determines whether it is worthwhile to process with the all project or the benefit obtained from the new system. All it requires is a computer which can run MATLAB software. Now a day's computers are not considered as economic issue. Hence this technique is well within the budget of almost all the people

4.1.2.Technical Feasibility

Technical feasibility centres on the existing computer system hardware, software etc. And what extent it can support the proposed edition. Technical constraint is not serious constraint for feasibility of this technique.

This system is feasible on the following grounds:

- All necessary technology is available to those who want to develop an application.
- The existing resources are capable of holding all the necessary in an efficient way.
- The system is too flexible and can be expanded further whenever new modules are added.

4.1.3. Operational Feasibility

This determines if the proposed system has satisfied user objectives and can be fitted into current operation. The proposed will certainly satisfy the user requirements and will also enhance their capability. It can be best fitted into current operations. Also the maintenance of the system is very easy and requires minimum effort. Therefore, the system is operationally feasible.

4.2. Feasibility Study of This Project

The requirements of engineering process starts with feasibility study. The input of the feasibility study is an outline description of the system. The result of feasibility study is a report which recommends whether or not it is worth carrying on with requirements engineering and the system development process. Carrying out the feasibility study involves information assessment, information collection and report writing.

This project requires a computer which can run the MATLAB software 2015 version and has the ability to upload the images from the digital cameras.

CHAPTER 5

SOFTWARE AND ITS SPECIFICATIONS

5. SOFTWAREREQUIREMENT SPECIFICATION

5.1. INTRODUCTION

The main purpose of this project is to detect characters from license plate

Image provided by a camera. An efficient algorithm is developed to detect a license plate in various luminance conditions. This algorithm extracts the license plate data from an image and provides it as an input to the stage of Car License Plate Recognition. The image of a vehicle is given as an input.

5.1.2. Scope:

The scope of this project is to detect the license plate from the given image and observe the output on MATLAB. This project can work as a base for future improvements in the field of image processing, especially in license plate extraction and plate number recognition.

5.1.3. Developers Responsibilities:

The main responsibility lies in obtaining the result for input image which consists of different characters in license plate.

5.2. General Description

5.2.1. Product Perspectives:

For the given input image our project focuses mainly on applying morphological operations to increase the possibility of extracting the license plate region and to ease the segmentation process.

5.2.2. Product Functions:

Steps involved in developing our project:

- The project starts with the literature study on morphological operations in image processing
- Then a detailed study on MATLAB is done
- The image is read using MATLAB
- Morphological processing is done.
- Segmentation of the required region is done.
- Segmented images are used for template matching
- Extracted characters of the license plate are displayed

5.2.3. User Characteristics:

Before implementing the project the user must have a basic idea in MATLAB software. User need not have the entire knowledge of the software but he must know some of the image processing techniques in the software. Without having the basic knowledge of the software it is difficult to implement the image processing. In order to analyse the project, the user must know the basic terms like grey scale images, pixel values regarding the image processing.

In this project the user intervention is very less. The user just needs to provide the input video and run the code to see the output.

5.3. Specific Requirements:

5.3.1. Non-Functional Requirements:

a) Usability:

The user can give a normal clarity image as input and can interpret output with minimum knowledge of basics.

b) Performance:

The proposed system after giving the required input takes less time to process and produces output accurately.

c) Supportability:

The system can be expanded further to add new factors that may effect. The proposed system can work under any environment without any problems.

5.3.2. Design Requirements:

a) HARDWARE REQUIREMENTS:

Processor : Any Intel or AMD,1 GHz or faster processor

RAM : 4GB

Hard disk : 1 TB

b) Software requirements:

Operating system : Windows XP, WINDOWS 10

Language : MATLAB

5.4 Introduction to MATLAB:

MATLAB is a high-performance language for technical computing. It integrates computation visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation
- Algorithm development
- Modelling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in & fraction of the time.

The name MATLAB stands for matrix laboratory, MATLAB features a family of applications solutions called toolboxes. Toolboxes are comprehensive collections of MATLAB(M-files) that extend the MATLAB environment to solve particular classes of problem.

It is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB's support for object-oriented programming includes classes, inheritance, virtual dispatch, packages, pas-by-value semantics, and pass-by-reference semantics. A wrappers function is created allowing MATLAB data types to be passed and returned.

5.5 The MATLAB system:

The system mainly concentrates on

- Mathematics

Linear algebra, basic statistics, differentiation and integrals, Fourier transforms, and other mathematics

- Programming Scripts and Functions

Program files, control flow, editing, debugging

- Data and File Management

Data import and export, workspace, files and folders

The MATLAB system consists of these main parts:

5.5.1 The MATLAB language:

Versions such as R2009, R2010, R2012, R2013, R2015,R2017 are available in both 'a' and 'b' versions. This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

5.6 M-files:

5.6.1 M-file scripts:

To solve complicated problems, typing all commands in an ASCI file named with .m extension can be done. It is called a script file or an M-file. In MATLAB, cd path can be used to change the directory and the commands in this file will be executed. Pwd in MATLAB is used to see the present working directory. The variable names and the names of MATLAB built-in functions are to be avoided while choosing a filename for an M-file. MATLAB help can be used to know the details of the in built functions.

5.7 MATLAB program control:

MATLAB is also a programming language. Like other computer programming languages, MATLAB has some decision making structures for control of command execution. These decision making or control flow structures include for loops, while loops, and if-else-end constructions. Control flow structures are often used in script M-files and function M files.

By creating a file with the extension .m, we can easily write and run programs. We do not need to compile the program since MATLAB is an interpretative (not compiled) language. MATLAB has thousands of functions, and you can add your own using Mfiles. MATLAB provides several tools that can be used to control the flow of a program.

5.8 Types of Functions:

MATLAB offers several different types of functions to use in your programming.

5.8.1 Anonymous Functions:

An anonymous function is a simple form of the MATLAB function that is defined within a single MATLAB statement. It consists of a single MATLAB expression and any number of input and output arguments. You can define an anonymous function right at the MATLAB command line, or within a function or script. This gives you a quick means of creating simple functions without having to create a file for them each time.

5.8.2 Primary and Sub functions:

Any function that is not anonymous must be defined within a file. Each such function file contains a required primary function that appears first, and any number of sub functions that may follow the primary. Primary functions have a wider scope than sub functions. That is, primary functions can be called from outside of the file that defines them (e.g., from the MATLAB command line or from functions in other files) while sub functions cannot. Sub functions are visible only to the primary function and other sub functions within their own file.

5.9 Plotting in MATLAB:

MATLAB provides a variety of functions for displaying vector data as line functions for annotating and that produce basic line plots. These functions differ in the way they scale the accepts input in the form of vectors or matrices and automatically scales the axes to accommodate the data. The mesh and surf commands create 3-D surface plots of matrix data. Surface object properties provide additional control over the visual appearance of the surface and also the edge line styles, vertex markers, face colouring, lighting, characteristics can also be specified.

5.10 Data Types in MATLAB:

By default, MATLAB stores most data in arrays of class double. The data in these arrays is stored as double precision (64-bit) floating-point numbers. All MATLAB functions and capabilities work with these arrays.

For images stored in one of the graphics file formats supported by MATLAB, however, this data representation is not always ideal. The number of pixels in such an image may be very large; for example, a 1000-by-1000 image has a million pixels. Since each pixel is represented by at least one array element, this image would require about 8 megabytes of memory if it were stored as class double. To reduce memory requirements, MATLAB supports storing image data in arrays of class uint8 and uintl6. The data in these arrays is stored as 8-bit or 16-bit unsigned integers. These arrays require one-eighth or one-fourth as much memory as data in double arrays.

5.11 Implementation:

Implementation of License Plate Detection algorithm using MATLAB. MATLAB is a very powerful software tool used to implement the tasks that require extensive computation. It provides easy and quicker implementation of algorithms compared to C and C++. The key feature in MATLAB is that it contains a rich library functions for image processing and data analysis. This makes MATLAB an ideal tool for faster implementation and verification of any algorithm before actually implementing it on a real hardware. Sometimes, debugging of errors on actual hardware turns out to be a very painful task. MATLAB provides an easy approach for de-bugging and correction of errors in any algorithm. Other than this, MATLAB contains many features including workspace, plot, imread, imhist, imshow, etc. for data analysis and image processing, which makes it a better choice over other software languages like C and C++.Considering the above advantages, we are p implementing an algorithm for License Plate Detection using MATLAB. The algorithm initially uses various inbuilt functions and uses few user defined routines related to image processing. Once the algorithm was developed, it was verified with multiple input images containing car number plates. The input images contained number plates that were aligned horizontally as well as at some angle from horizontal axis. Once the algorithm was completely verified, the in-built functions of MATLAB were replaced by user defined functions.

CHAPTER 6

PROJECT IMPLEMENTATION

6. PROJECT IMPLEMENTATION

6.1 Image Acquisition:

The first stage of any vision system is the image acquisition stage. After the image has obtained, various methods of processing can be applied to the image to perform the many different vision tasks required today. However, if the image has not been acquired satisfactorily then the intended tasks may not be achievable, even with the aid of some form of image enhancement, Image Acquisition enables you to acquire images and video from camera frame grabbers directly into MATLAB and Simulink . You can detect hardware automatically and configure hardware properties. Advanced workflows let you trigger acquisition while processing in-the-loop, perform background acquisition, and synchronize sampling across several multimodal devices. With support for multiple hardware vendors and industry standards, you can use imaging devices ranging from inexpensive. Web cameras to high-end scientific and industrial devices that meet low-light, high-speed, and other challenging requirements. Image acquired by camera will be indicated in MATLAB as a matrix of 0's and 1's i.e, binary.

For this purpose we can use following types of cameras

6.1.1 CMOS Cameras:

CMOS (Complementary Metal-Oxide Semiconductor) is a technology used in fabricating integrated circuit chips. But in the context of photography, it is one of the two alternative technologies used in digital camera image sensors; the other being CCDs.

Early digital cameras tended to use CCD sensors, as this was the more mature technology. The evolution of CMOS sensors had a steeper learning curve, and originally their use was confined to lower-cost products where image quality was not an overriding concern. Canon and Sony refined CMOS sensors through several generations, today (2011) when comparing sensors of equal size, CMOS sensors generally exceed CCDs in image quality measures. Typically a CMOS design uses active circuitry placed on directly on the chip itself which can speed image readout and help with noise reduction. The added circuit complexity does increase the up-front costs to the chip developer, thus we see sensor manufacturer. Sony trying to recoup some of those costs by selling sensor chips to its nominal competitors in the camera market: Pentax, Nikon, Leica, etc.

Figure:6.1CMOS Camera

6.1.2 CCD Cameras:

A Charge-Coupled Device (CCD) is a device for the movement of electrical charge, usually from within the device to an area where the charge can be manipulated, for example conversion into a digital value. This is achieved by "shifting" the signals between stages within the device one at a time. CCDs move charge between capacitive bins in the device, with the shift allowing for the transfer of charge between bins.

The CCD is a major piece of technology in digital imaging. In a CCD image sensor, pixels are represented by p-doped MOS capacitors. These capacitors are biased above the threshold for inversion when image acquisition begins, allowing the conversion of incoming photons into electron charges at the semiconductor-oxide interface; the CCD is then used to read out these charges. Although CCDs are not the only technology to allow for light detection, CCD Image sensors are widely used in professional, medical, and scientific applications where high-quality image data is required. In applications with less exacting quality demands, such as consumer and professional digital cameras, active pixel sensors (CMOS) are generally used; the large quality advantage CCDs enjoyed early on has narrowed over time.

Figure:6.2 CCD Cameras

6.1.3 Infrared Cameras:

Infrared (IR) light is electromagnetic radiation with longer wavelengths than those of visible light, extending from the nominal red edge of the visible spectrum at 500 nanometres (nm) to1 mm. This range of wavelengths corresponds to a frequency range of approximately530 THz down to 300 GHz. Most of the thermal radiation emitted by objects near room temperature is infrared.

Infrared light is emitted or absorbed by molecules when they change their rotational-vibrational movements. Infrared energy elicits vibrational modes in a molecule through a change in the dipole moment, making it a useful frequency range for study of these energy states for molecules of the proper symmetry. Infrared spectroscopy examines absorption and transmission of photons in the infrared energy range.

Infrared light is used in industrial, scientific, and medical applications. Night-vision devices using active near-infrared illumination allow people or animals to be observed without the observer being detected. Infrared astronomy uses sensor-equipped telescopes to penetrate dusty regions of space, such as molecular clouds; detect objects such as planets, and to view highly red-shifted objects from the early days of the universe. Infrared thermal-imaging cameras are used to detect heat loss in insulated systems, to observe changing blood flow in the skin, and to detect overheating of electrical apparatus.

Based on our application any of the above camera can be used. Here in this project we use images taken by a CMOS camera.

Below figures shows the one of the images we use for the project and how the MATLAB reads the image.

Figure:6.3 IR Cameras

6.2 License Plate Extraction:

Localization of potential license plate regions from vehicle images is a challenging task due to huge variations in size, shape, color, texture and spatial orientations of license plate regions in such images. In general, objective of any Automatic License Plate Recognition (ALPR) system is to localize potential license plate region(s) from the vehicle images captured through a road-side camera and interpret them using template matching method to get the license number of the vehicle. In an online ALPR system, the localization and interpretation of license plates take place instantaneously from the incoming still frames, enabling real-time tracking of moving vehicles through the surveillance camera. An offline ALPR system, in contrast, captures the vehicle images and stores them in a centralized data server for further processing, i.e. for interpretation of vehicle license plates.

6.2.1Pre processing techniques:

6.2.1.1 Reduction of colors:

A problem arises while processing image is that color can appear differently in different lightning conditions. To overcome this we reduce the number of colors around 50. Following can be reasons for color reduction:

- Images with more than 256 colors will need to be dithered or mapped and, therefore, might not display well.
- Image has more than 256 colors, MATLAB cannot store the image data in a uint8 array but generally uses an array of class double instead, and making the storage size of the image much larger (each pixel uses 64 bits).

6.3 Gray scale conversion:

From the 24-bit color value of each pixel (i, j) the R, G and B components are separated and the8-bit Gray value is calculated using the formula:

Gray(i, j) = 0.59 * R(i, j) + 0. 30 G(i, j) +0.11 * B(i, j) (1)

Figure:6.4Gray Scale Image

6.4 Dilation:

Dilation allows objects to expand thereby further highlighting shapes/geometry of objects in an image. Structuring element is simply a matrix of 0s and 1s that could be of any arbitrary shape and size.

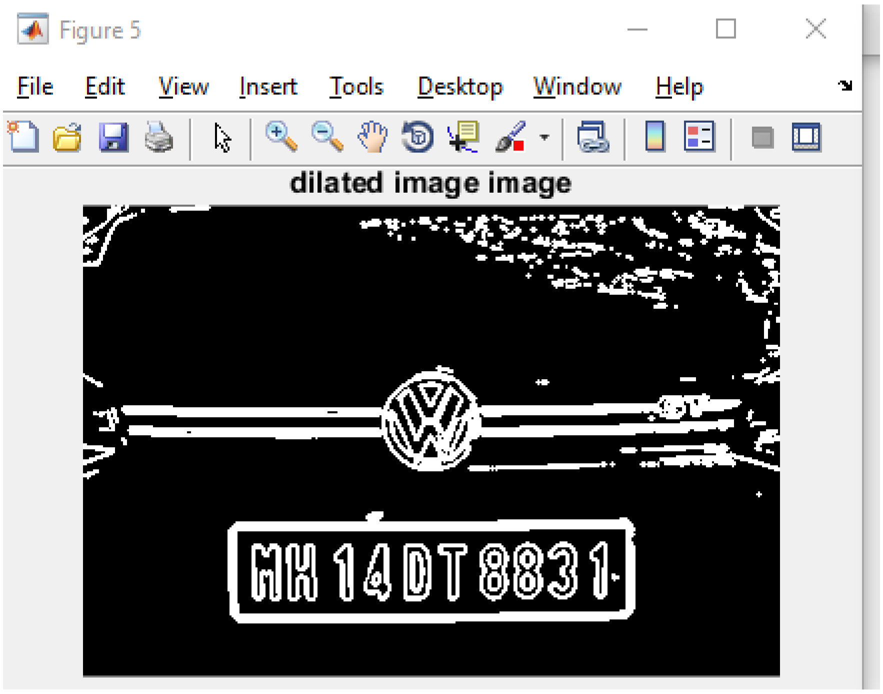
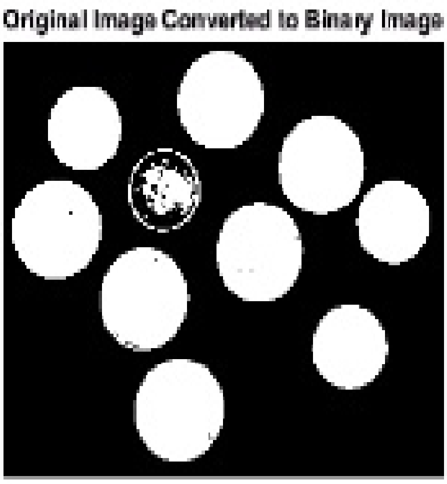


Figure:6.5 Dilated image

6.5 Filling:

Dilated image is filled with 'holes' to emphasize region of interest.

Following examples shows images before dilate and erode operation



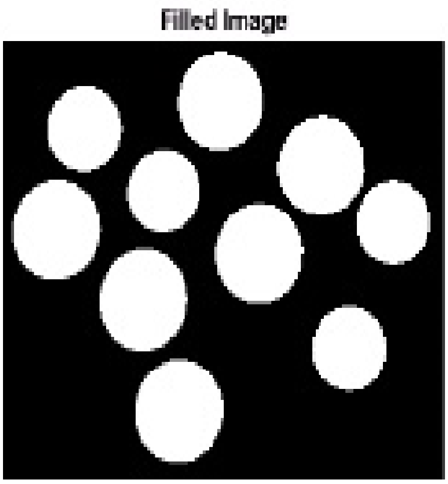


Figure:6.6 image filling

6.6 Edge Detection:

In this work, we have extracted the edges created by the characters within the license plate. It is seen that when the characters of the license number are written horizontally the vertical edges of each character appear at regular interval and they have a specific height. The pattern and concentration of the vertical edges also remains in conformity with the pattern of the license number. This appearance of vertical edge pattern is statistically seen to occur within the license plate, nowhere else within the natural scene of the image. The area of the region should not be less than specified threshold values. Length and Breadth or aspect ratio should lie within 10:1.

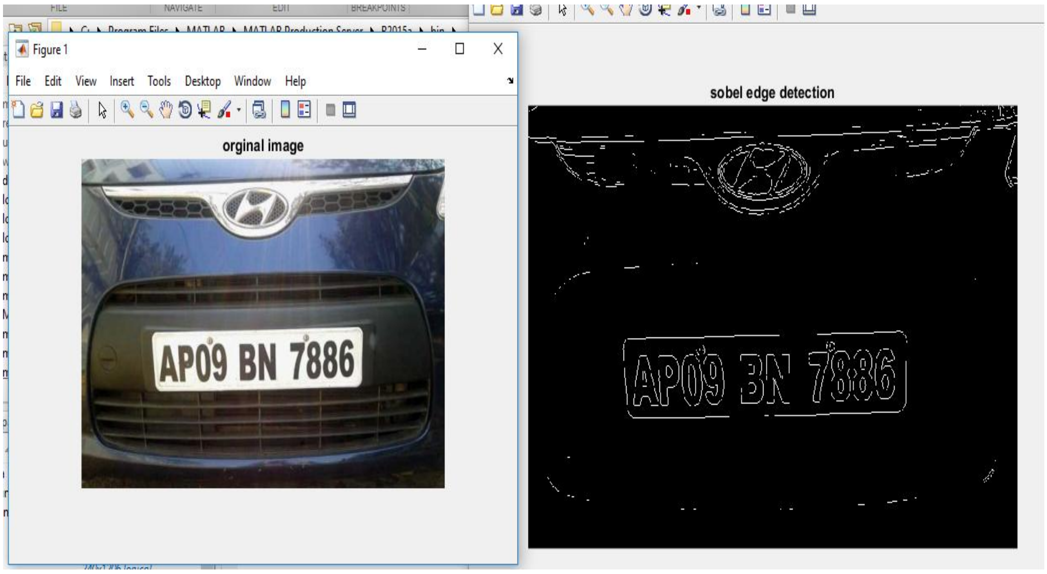


Figure:6.7 Edge Detection Example

Normalization and contrast enhancement:

6.7.1 Median filtering

Median filter is a non-linear filter, which replaces the gray value of a pixel by the median of the gray values of its neighbours. We have used 3x3 masks to get eight neighbours of a pixel and their corresponding gray values. This operation removes salt-and-pepper noise from the image.

6.7.2 Contrast enhancement:

Contrast of each image is enhanced through histogram equalization technique, Total 256 numbers of gray levels (from 0 to 255) are used for stretching the contrast. Let total number of pixels in the image be N and the number of pixels having gray level k be nk. Then the probability of occurrence of gray level k is, Pk= nk/N. The stretched gray level Sk is calculated using the cumulative frequency of occurrence of the gray level k.

6.8 Segmentation:

Image segmentation is the process of partitioning an image into parts or regions. This division into parts is often based on the characteristics of the pixels in the image. For example, one way to find regions in an image is to look for abrupt discontinuities in pixel values, which typically indicate edges. These edges can define regions. Another method is to divide the image into regions based on color values. Segmentation is done using blob analysis which involves following commands. 'Bwlabel' command labels and gives you the number of pixels connected together in a sequence to form a group of connected objects. "regionprops" measures a set of properties for each connected component. Regionprops computes only the 'Area', 'Centroid', and 'Bounding Box' measurements. Usage of both of these commands to label objects is used in rectangle as Bounding Box around the objects. Boundaries of each labelled objects and using these boundary coordinates cropped the characters from the license plate. At this stage, cropped character may also contain garbage objects as well as useful LP characters and numbers; few of the extracted objects.

Figure:6.8 Segmented Number Plate

6.9 Commands used in segmentation:

Imshow:

Syntax:

1. imshow(1): displays the image I in a figure, where I is a gray scale, RGB (true color), or binary image. For binary images, imshow displays pixels with the value 0 (zero) as black and 1 as white.

2. imshow(filename): displays the image stored in the graphics file filename. The file must be in the current directory imread or dicomread. imshow calls imread or dicomread to read the image from the file, but does not store the image data in the MATLAB workspace. If the file contains multiple images, imshow displays only the first one

3.Imdilate():

Syntax:

1. **IM2 =imdilate (IM, SE**) dilates the gray scale, binary, or packed binary image IM, returning the dilated image, IM2. The argument SE is a structuring element object, or array of structuring element objects, returned by the strel function. If IM is logical and the structuring element is flat, **imdilate** performs binary dilation; otherwise, it performs gray scale dilation. If SE is an array of structuring element objects, imdilate performs multiple dilations of the input image, using each structuring element in SE in succession.

2. **IM2= imdilate(IM, NHOOD)** dilates the image IM, where NHOOD is a matrix of 0's and1's that specifies the structuring element neighbourhood. This is equivalent to the syntax **imdilate (IM, strel (NHOOD)).** The imdilate function determines the centre element of the neighbourhood by **floor ((size (NHOOD)+1)/2)).**

3. Imfill():

Syntax:

1.**BW2 = imfill(BW)** displays the binary image BW on the screen and lets you define the region to fill by selecting points interactively by using the mouse. To use this interactive syntax, BW must be a 2-D image. Press Backspace or Delete to remove the previously selected point. A shift-click, right-click, or double-click selects a final point and starts the fill operation. Pressing Return finishes the selection without adding a point.

2. BW2= imfill(BW, 'holes') fills holes in the binary image BW. A hole background pixels that cannot be reached by filling in the background from the edge of the image.

3.Imerode():

Syntax:

1. IM2= imerode(IM,SE) erodes the gray scale, binary, or packed binary image IM, returning the eroded image IM2. The argument SE is a structuring element object or array of structuring element objects returned by the strel function. If IM is logical and the structuring element is flat, imerode performs binary erosion; otherwise it performs gray scale erosion. If SE is an array of structuring element objects, imerode performs multiple erosions of the input image, using each structuring element in SE in succession.

4.regionprops():

Syntax:

1. STATS= regionprops(L, properties) measures a set of properties for each labelled region in the label matrix L. Positive integer elements of L correspond to different regions. For example, the set of elements of L equal to 1 corresponds to region 1; the set of elements of L equal to 2 corresponds to region 2; and so on.

2. STATS = regionprops(BW, properties) measures a set of properties for each connected component (object) in the binary image, BW. The image BW is a logical array; it can have any dimension.

RESULTS

Results:

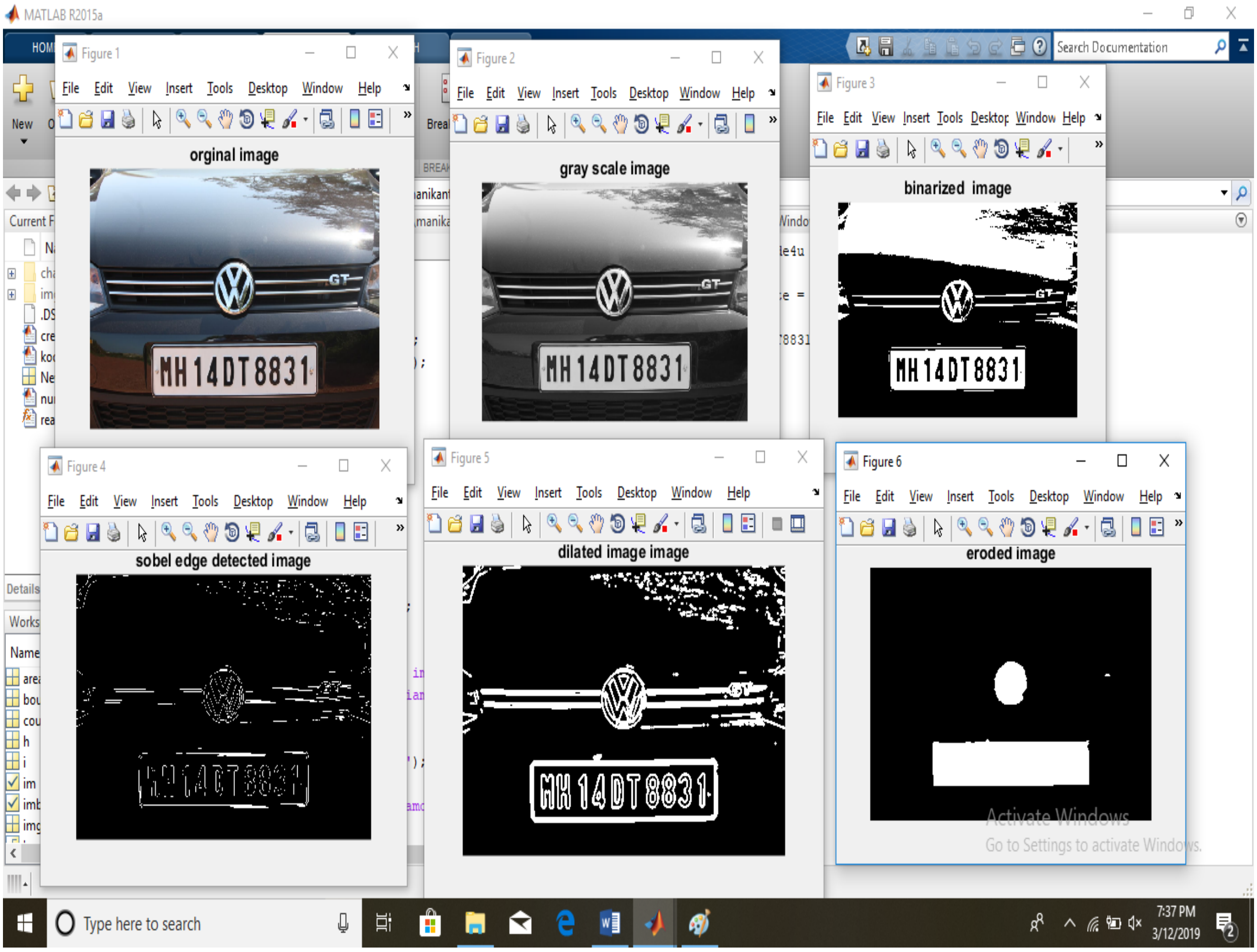


Figure:7.1 Different stages in number plate extraction

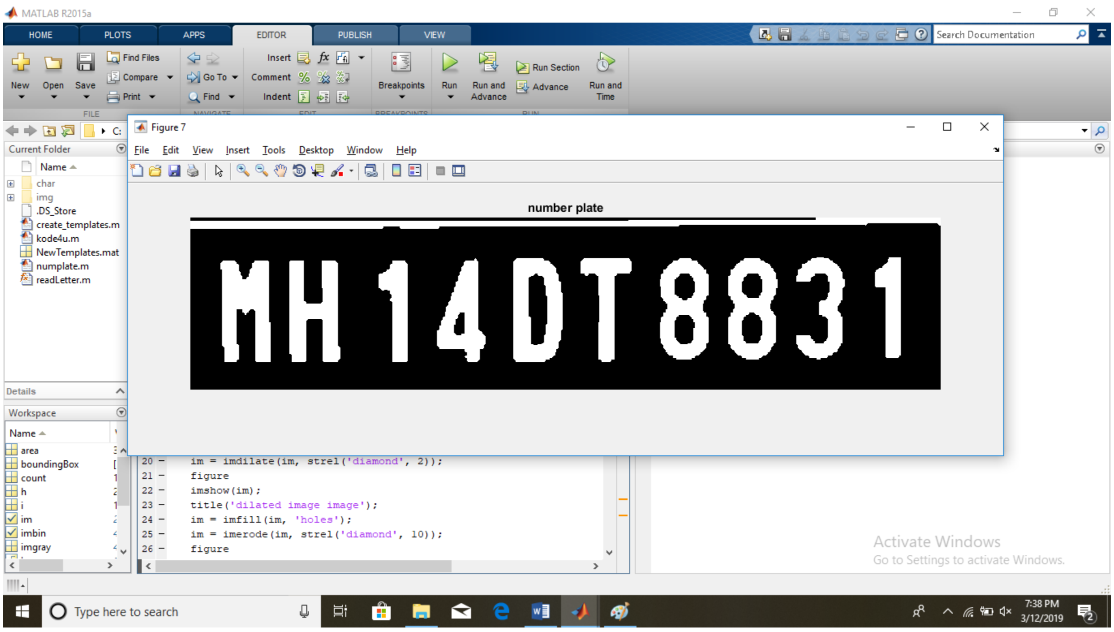


Figure: 7.2 Segmented Number Plate

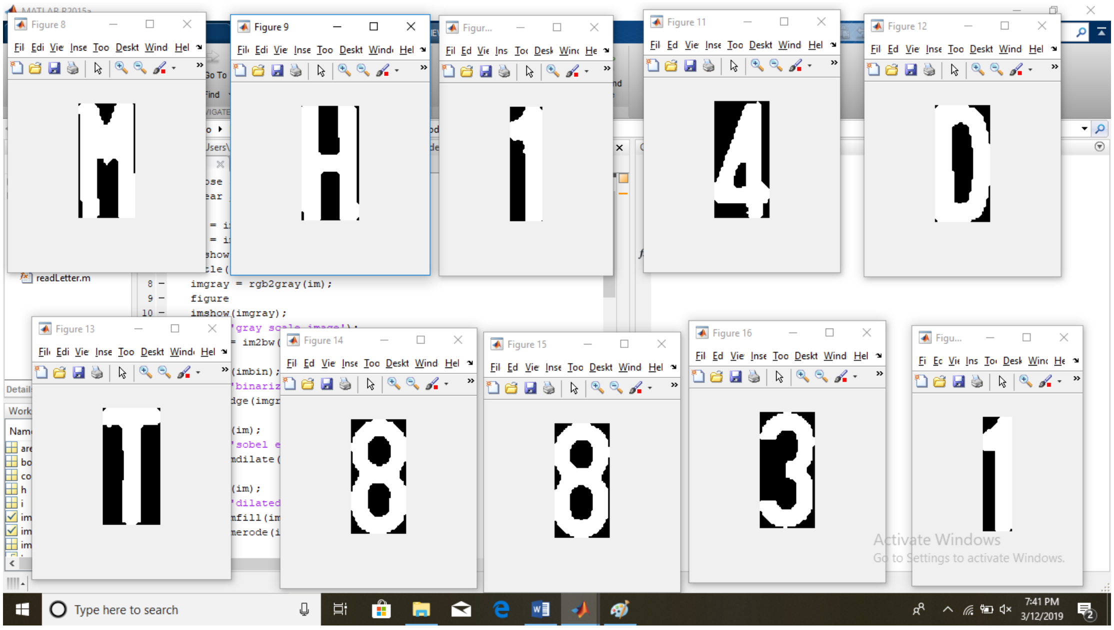


Figure: 7.3 Segmented Characters

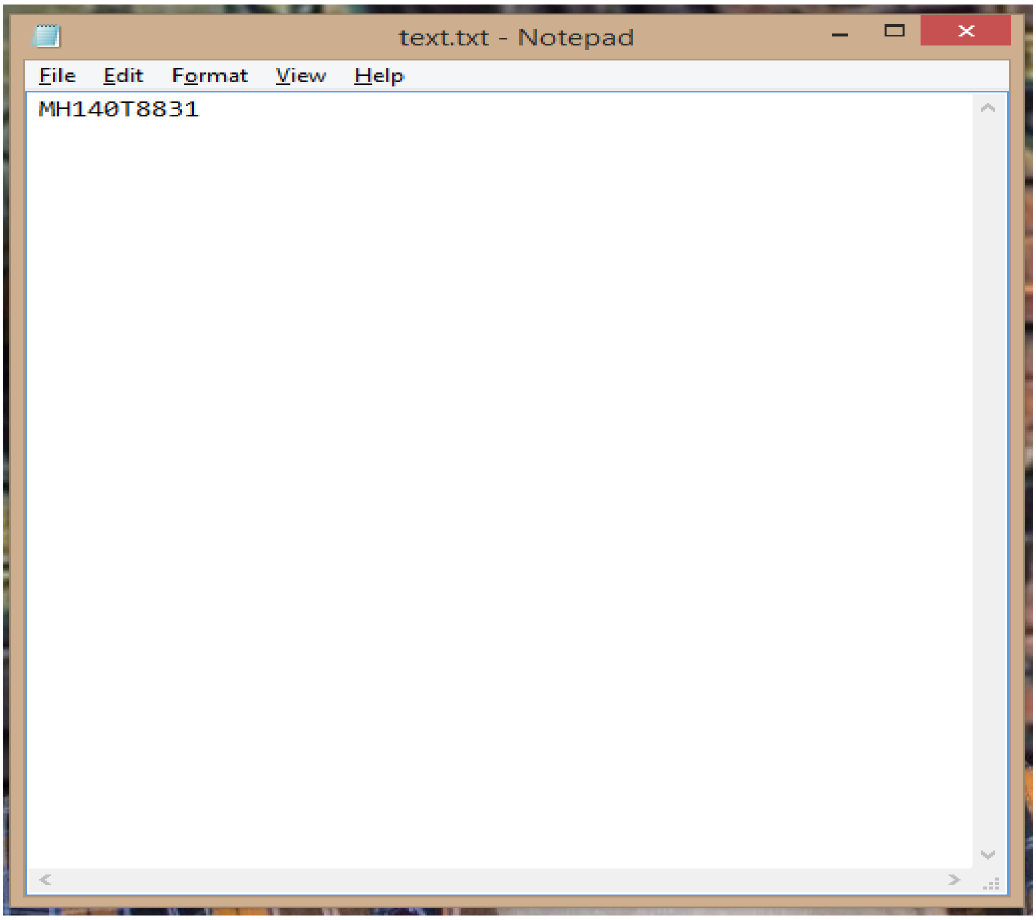


Figure:7.4 Text Converted Number Plate

- First we have executed our program in MATLAB then command window appears.
- Then user enters the file name in the format of "file name jpg".
- Then need to specify our requirement whether to display each stage output or only final output by either entering (y or n).
- As soon as we choose our requirement the program imports the image and program performs processing operations according to program.
- Masking and localization of plate are performed according to specified threshold values.
- The characters in the localized plate are cropped and segmented and segmented outputs are displayed in figure window.
- The number extracted from the number plate is saved in a text file for the future use.

7.1 CONCLUSION:

- Our work mainly proposes a plate localization and extraction technique from vehicle number plates.
- Firstly extraction of plate location, then separation of the plate characters individually by performing different pre-image processing techniques and segmentation, finally the segmented numbers is correlated with the standard templates in the library.
- In order to extract the plate location a bounding box method is used. And also each character is also segmented using the same bounding box method.
- The Segmented characters are identified by using Template Matching Method.
- The suggested method is tested with various types of vehicles like four wheelers and with yellow and white background.
- The number plates with additional unnecessary data are also segmented with great accuracy.

7.2 FUTURE SCOPE:

- The future work will involve in recognizing the individual characters from the plate with other colours back grounds with the standard templates issued by the government.
- This mainly uses in high traffic areas to control the traffic without help of Human.
- It can be used for the monitoring Parking Areas.
- In order to punish the one who violates the traffic rules such as crossing the speed limits, signal jumping and unauthorised vehicles.

8. REFERENCES:

[1] Digital image processing by Rafacl C.Gonzalez and Richard E. Woods published by pearson education.

[2] R.Radha1 and C.P.Sumathi2, “A Novel approach to extract text from license plate of vehicle”, Signal & Image Processing: An International Journal (SIPIJ) Vol.3, No.4, August 2012

[3] Kumar Parasuraman and P.Vasantha Kumar, “An Efficient Method for Indian Vehicle License Plate Extraction and Character Segmentation”, IEEE International Conference on Computational Intelligence and Computing Research,2010.

[4] Lekhana G.C, R.Srikantaswamy ,“Real time license plate recognition system”, International Journal of Advanced Technology & Engineering Research (IJATER), National Conference on Emerging Trends in Technology (NCETTech) ISSN, Volume2,Issue4,ISSN No: 2250-3536, July 2012.