CAR PLATE DETECTION SYSTEM

SITI NUR AISHAH RODZIAH ALKHUSHAIRI

A report submitted in partial fulfillment of the requirements for the award of the
degree of Bachelor of Electrical Engineering (Electronics)

Faculty of Electrical & Electronics Engineering

Universiti Malaysia Pahang

JUNE 2012

"I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Electronics)"

Signature : _____

Name : MOHD HISYAM BIN MOHD ARIFF

Date : 20 JUNE 2012

Signature   : _____

Author   : <u>SITI NUR AISHAH RODZIAH ALKHUSHAIRI</u>

Date   : <u>20 JUNE 2012</u>

Special Dedication To

My Beloved Father

Alkhushairi bin Mohd Noor

My Beloved Mother

Loimi @ Taimi Sampuna

My brothers & sister

Mohd Al-Junaidi Rizal

Mohd Al-Azmi Fariz

Mohd Al-Nameiri Yusri

Siti Nur Fatimah

My Supervisor and Lecturers

All My Friends

For support, love and understanding

During the completion of my degree study

ACKNOWLEDGEMENT

ABSTRACT

Basically, the system using human eyes to detect the car plate and manually compare the plate number through the report file. Sometimes it will take a long time to search. Then this project is developing to make human work easily. The mainly purpose of this system is to monitor and identify vehicles by their license plate. This system can be used in widely applications such as section control, border crossings, entrance security, parking security and road traffic control. The advantage of this system is no need to add any transmitter or special sign on the vehicles because the car plate already have their own identification by referring to their registration car plate number. The image of the driver also can be captured and stored to make this system more secure. The optical character recognition (camera or video) is the main technique in identify every each of the character on the car plate and going through the process of image processing. That means this project is helping and making human easily monitoring the car and captures the image with the new development technology. The result of this system will display through the Graphical User Interface (GUI).

ABSTRAK

Pada asasnya, sebelum sistem ini dibina, pengesanan nombor plat kenderaan hanya menggunakan mata manusia dan secara manual membandingkan nombor plat melalui file simpanan. Kadang-kadang ia akan mengambil masa yang lama untuk mencari dan mengesan nombor kenderaan tersebut. Kemudian sistem ini direka khas untuk membuat kerja-kerja manusia menjadi mudah dan cepat. Tujuan utama system ini direka adalah untuk memantau dan mengenal pasti nombor plat lesen kenderaan individu. Aplikasi penggunaan sistem ini adalah secara meluas contohnya ia bole digunakan pada kawalan seksyen, lintasan sempadan, keselamatan pintu masuk, keselamatan tempat letak kereta dan kawalan lalu lintas jalanraya. Kelebihan sistem ini adalah tidak perlu untuk menambah pemancar atau tanda khas pada kenderaan kerana plat kereta sudah mempunyai pengenalan diri mereka sendiri dengan merujuk kepada plat nombor pendaftaran kereta mereka. Imej pemandu juga boleh ditangkap dan disimpan untuk membuat sistem ini lebih selamat. Pengecaman aksara optik (kamera atau video) adalah teknik utama dalam mengenal pasti setiap setiap karatek pada nombor plat kereta dan akan melalui proses pemprosesan imej. Ini bermakna projek ini membantu dan membuat manusia mudah untuk memantau kereta dan menangkap imej dengan pembangunan teknologi baru. Penggunaa Antara Muka Pengguna Grafik (GUI) adalah untuk paparan hasil pemprosesan imej pada gambar tersebut.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CPD   -   Car Plate Detection

ANN   -   Artificial Neural Network

GUI   -   Graphical User Interface

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1   Background

The Car Plate Detection Systems (CPD) is one of the most important applications in transportation. This system firstly introduced by the Police Scientific Development Branch, United Kingdom in 1976 and known as Automatic Number Plate Recognition (ANPR). The mainly purpose of this system is to monitor and identify vehicles by their license plate. This system can be used in widely applications such as section control, border crossings, entrance security, parking security and road traffic control. The advantage of this system is no need to add any transmitter or special sign on the vehicles because the car plate already has their own identification by referring to their registration car plate number. The image of the driver also can be captured and stored to make this system more secure. This may help any case in crime investigation. The optical character recognition is the main technique in identifying every each of the characters on the car plate and going through the process of image processing.

The system is to help human automatically detect plate number without using human eyes. As we know before, humans are needed to observe and list the user car plate number manually. That means this project is helping and making human easily

monitoring the car and captures the image with the new development technology. In the other hands, this system will automatically analyze and view the output result of the car status by comparing it with the database. These easily alert the user if the car is in observation or not.

1.2   Problem Statement

Nowadays, the vehicle in Malaysia has been increasing on the road today. Because of that problem, it is difficult for law enforcement and security to monitor the vehicles. For example, at the entrance gate, it is time consuming for the security to physically check the car plate every day. As a solution, this simple system is designed to identify the car plate was developed.

1.3   Objectives of project

The objectives of this project are:
i.   To develop car plate detection systems using Image Processing and MATLAB tools.
ii.   To develop a system that can determine the location of the character on the car plate.
iii.   To design a simple and easy understanding Graphical User Interface (GUI).

1.4   Scope of Project

The scopes of this project are:

i.   The car plate detection system focus on the Malaysia standard car (except for Sabah, Sarawak and Putrajaya) type of car plate and it must on single line type car plate.

ii.   The systems automatically detect and captured any car plate and recognize all the characters on the car plate.

iii.   The Graphical User Interface (GUI) is simple and easy understanding that compatible with Windows operating systems.

1.5   Thesis outline

This thesis consists of five chapters:

Chapter 1: discuss about the background of study of the project, the problem statements, the objectives and the scope of the project.

Chapter 2: discuss more on theory and literature reviews that has been done in this project. Besides, the techniques that have been used in this project also well discuss in this chapter from the image processing until the process end of the system.

Chapter 3: the methodology of the system will be discussed in this chapter including the image processing and the MATLAB software.

Chapter 4: the result and discussion will be discussed in this chapter.

Chapter 5: in this last chapter of the thesis will discuss the conclusions and the future recommendations for this system.

CHAPTER 2

LITERATURE REVIEW

Introduction

Car Plate Detection System (CPD) is a system that allows detecting the character images of number car plate that stored on hard disk. The systems used to store images of number plate of vehicles and automatically analyze the character on the image using Digital Image Processing System. It is widely used in police enforcement, traffic control, electronic toll collection, gate entrance, border crossings and etc. [1][7]

The CPD system can be used to control and recognize any vehicle that make any violation traffic rules, such as entering the private property area, UMP gate entrance, crossing red light, breaking speed limits ; etc.

Car plate detection system is a new technology in automatic monitoring traffic by using image processing. The system has two unit generic parts; Capture unit and Process unit. The capture unit responsible to capture the car plate image and process unit will process and analyze the image captured. The techniques that implemented in this system are digital image processing and artificial neural network (ANN). [7]

2.1   Capture Unit

The first step of the system is to detect the presence of the vehicle then automatically captured the image of the vehicle at the right time. There are three kinds of trigger control exist; hardware trigger, software trigger and free flow. [7]

Once the vehicle is detected, the following step is capturing the vehicle image. To make the image taken at a right time, there are several things to be considered;

    i.   Type of camera

    ii.   Lighting management (in daylight and night)

    iii.   Distance between the car plate and camera

After all above has been considered, then the image stored in hard disk to be analyzed on the next stage.

2.2   Digital Image Processing

Digital image processing is the system that use of computer algorithms to perform image processing on digital images. The advantages of using this technique are: [10]

    i.   Allows a much wider range of algorithms to be applied to the input data.

    ii.   Can avoid noises and signal distortion during processing.

    iii.   Can be modeled in the form of multidimensional systems.

    iv.   Offer more sophisticated performance of simple tasks.

The approach concerns stages of preprocessing, edge detection, filtering, detection of the plate position, slope evaluation and character segmentation and recognition. The image will be converted to single frame gray-level images for source of information.

There are four primary algorithms that the software requires for identifying a car plate number:

```
┌─────────────────────────────┐
│     Plate localization      │
└─────────────────────────────┘
              ↓
┌─────────────────────────────┐
│  Plate orientation & sizing │
└─────────────────────────────┘
              ↓
┌─────────────────────────────┐
│        Normalization        │
└─────────────────────────────┘
              ↓
┌─────────────────────────────┐
│    Character Segmentation   │
└─────────────────────────────┘
              ↓
┌─────────────────────────────┐
│    Character Recognition    │
└─────────────────────────────┘
```
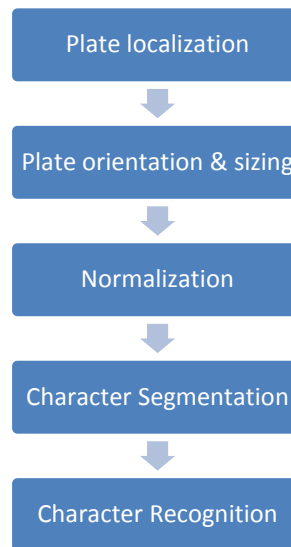
Figure 2.1: The four primary algorithms

Each primary above is the programs to determine the accuracy of the system. On the normalization phase, the systems need to use the edge detection technique to increase the picture difference between the letters and the background plates. Luckily, the standard Malaysian license plate number is in black and white color, it's easy for the system to detect and analyze it in a gray - level image. [3][4][5]

2.2.1 Plate Detection

This is the phase to identify the car plate area in an image. It will compose of image processing or plate enhancement phase. This helps to enhance the signal in the car plate area and attenuate it elsewhere and the actual plate detection phase. More powerful the enhancement of the image, the more trivial the plate detection will be. This phase faced many difficulties related to the application. Assume that the image contains little information but the car in case of automatic pay tolls, thus the car

probably the only high frequency area of the image. Other cases probably become problem with this system when at the street monitoring, it is not clear where exactly the plate will be, the background can be rather complex, then localization will be difficult.

To satisfy the performance of the system, it is useful to use the prior on the application and on the system position. Also, even more than in general purpose text detection systems. The automatic plate detection only can rely on:

i.   Car plates are meant to be read, thus letters and background are usually in a good contrast

ii.   The fonts used are usually simple

iii.   The digits are included in a rectangle or square box

iv.   Usually aligned on one or two rows

v.   Colors, fonts and size depend on the car origin country

2.2.2   Normalization

The normalization phase is used in this project is the morphological process that applied onto the image aim to remove all unrelated objects in the image and ensure that characters on the license plate are well preserved. A number of morphological processes will be applied to the image to make sure the algorithm is robust enough to cater for characters of varying sizes and scale. Also it's performed to loose upper and lower limit is specified for each process. There are three main processes applied to the image:

i.   The width

ii.   The height

iii.   The number of holes of each individual object of the binary image is checked.

Those exceeding the specified upper and lower limits will be removed from the image. It can be solved by fill, clear boundary objects and small objects removal processes. The figure below is the morphological process of the image: [2]
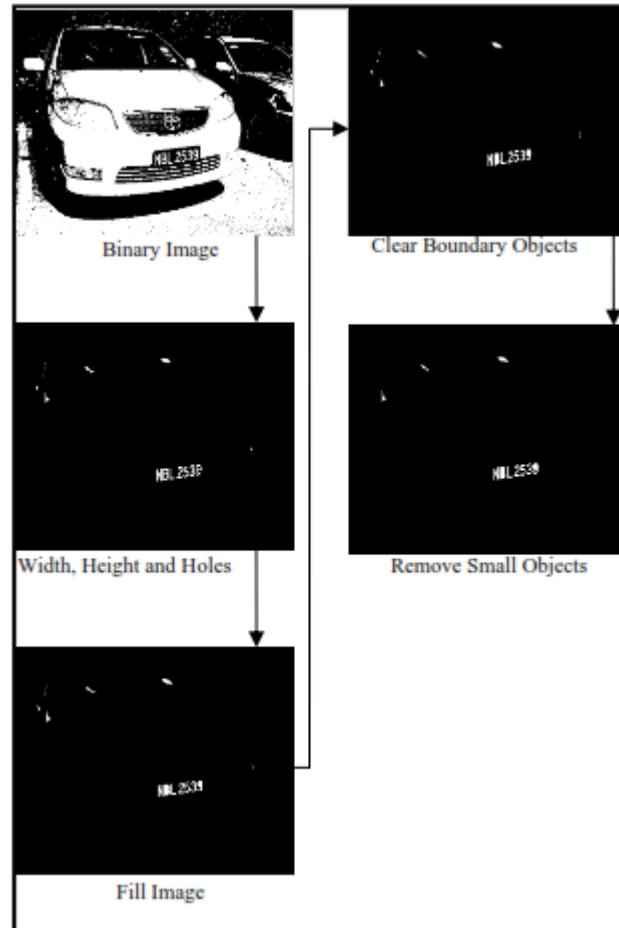
Figure 2.2: The morphological process of car plate image

2.2.3 Character segmentation

The character segmentation is a bridge between the license plate extraction and optical character recognition modules. The main function of this phase is to segmentate the characters on the chosen candidate region such that each character can be sent to the optical character recognition module individually for recognition.

The characters on the license plate can be extracted by determining the required location of the bounding box of each object in the chosen candidate region. Then, for each of the extracted bounding boxes, the object with the maximum area is retained and all other objects are removed. The process of the character segmentation illustrated below: [2]
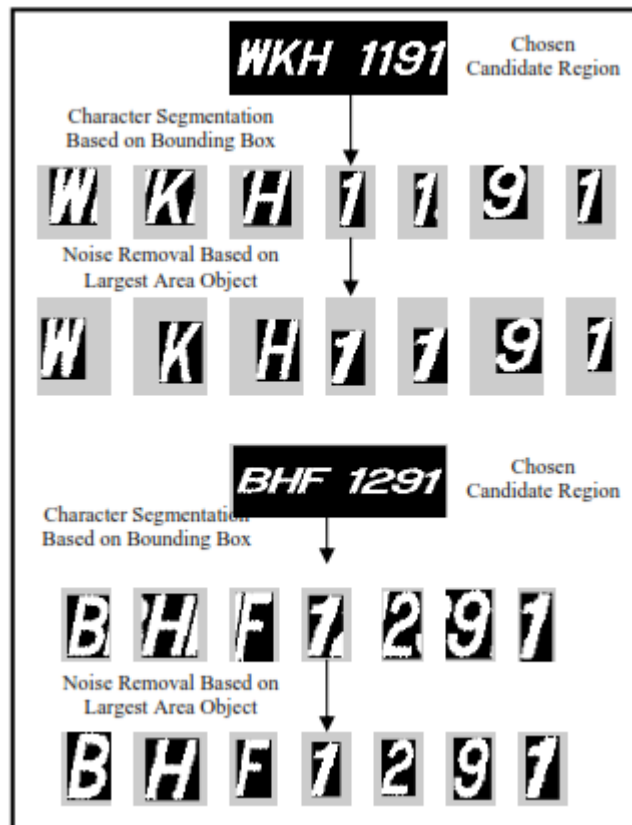
Figure 2.3: The segmentation process of car plate image

The characters are normalized into required size of pixels, this important to allow the recognition of the segmented characters via artificial neural networks in the character recognition module.

2.3   Character recognition

The problem for this phase is being posed as a classification or categorization task, where the classes are either defined by the system designer or are learned based on the similarity of the patterns. The demands of automatic pattern recognition systems are rising enormously due to the availability of large databases and stringent performance requirements. The design involves three aspects; [3][2]

i.   Data acquisition and preprocessing
ii.   Data representation
iii. Decision making

The learning from the set of examples (training sets) is an important and desired attribute of most pattern recognition systems. There are four best approaches for pattern recognition:[3]

i.   Template matching
ii.   Statistical classification
iii. Syntactic or structural matching
iv.   Neural network

2.3.1   Artificial Neural Network (ANN)

An Artificial Neural Network (ANN) is inspired by the structure and/or functional aspects of biological neural networks. In other words, the ANN is a data analysis methods and algorithms. It is an adaptive system that changes its structure based on external or internal information during the learning phase. It's suitable for making the car plate detection system. [11][3]
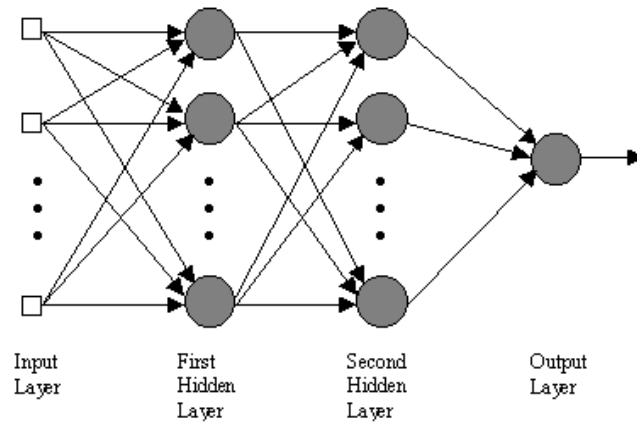
Figure 2.4: The example of three layer feed-forward Neural Network

The ANN is defined by three types of parameters. First is the interconnection pattern between different layers of neurons. Follow by The learning process for updating the weights of the interconnections. Last is the activation function that converts a neuron's weighted input to its output activation. [11][3]

2.4   MATLAB

MATLAB is a high-level language and interactive environment that enable for user to perform computationally intensive task faster that with traditional programming language, C++. The software was developed by The MathWorks and can allows matrix manipulations, plotting of functions and data, implementation of algorithm, creation of simple user interface and interfacing with program written in other languages.

The software has many applications toolbox especially the Image Processing Toolbox. It's a collection of functions that extended the capability of the MATLAB numeric computing environment. In image processing toolbox, it support various operations including the spatial image transformation, morphological operations,

neighborhood and block operation, linear filtering, filtering design, transformation, image analysis and enhancement, image registration and region interest operations.

CHAPTER 3

METHODOLOGY

This chapter will discuss about the way this system will conduct. This part is important in order to make sure the system to be function with suitable choice of equipment. The main objective of this research project is to examine and search out a suitable way to read and detect the license plate numbers of vehicles.

3.1 Introduction

The section will discuss the details about the project which has been done. The process of this system is summarized in flowchart section. Then, figure 3.1 is the process of the system. The process will begin at the image acquisition and going to the image enhancement, plate extraction, character segmentation (ANN part), character detection and GUI display. This project fully uses software implementation.

3.2   Flowchart of the System

```
                      ┌─────────────┐
                      │    START    │
                      └──────┬──────┘
                             │
                             ▼
                    ┌─────────────────┐
                    │   LOAD IMAGE    │
                    └────────┬────────┘
                             │
                             ▼
                      ◇ CROP CAR PLATE
           NO ◄───────   IMAGE      ◇────────► NO
                           │ YES                │
                           ▼                    │
                   ╱ DETECTION BUTTON: ON ╱     │
                           │                    │
                           ▼                    │
                    ┌─────────────────┐         │
                    │ IMAGE PROCESSING│         │
                    └────────┬────────┘         │
                             │                  │
                             ▼                  │
                 ◇ ANALYSIS & COMPARE CHARACTER ◇
                     SEGMENTATION DATA
                           │ YES
                           ▼
                    ┌─────────────────┐
                    │ DISPLAY RESULT ON GUI │
                    └────────┬────────┘
                             │
                             ▼
                       ┌─────────┐
                       │   END   │
                       └─────────┘
```

START

LOAD IMAGE

CROP CAR PLATE IMAGE

NO

YES

DETECTION BUTTON: ON

NO

IMAGE PROCESSING

ANALYSIS & COMPARE CHARACTER SEGMENTATION DATA

YES

DISPLAY RESULT ON GUI

END

```
┌─────────────────────────┐
│   Image Acquisition     │
└─────────────────────────┘
             ↓
┌─────────────────────────┐
│    Image Cropping       │
└─────────────────────────┘
             ↓
┌─────────────────────────┐
│   Image Enhancement     │
└─────────────────────────┘
             ↓
┌─────────────────────────┐
│     Segmentation        │
└─────────────────────────┘
             ↓
┌─────────────────────────┐
│      Detection          │
└─────────────────────────┘
             ↓
┌─────────────────────────┐
│      GUI display        │
└─────────────────────────┘
```
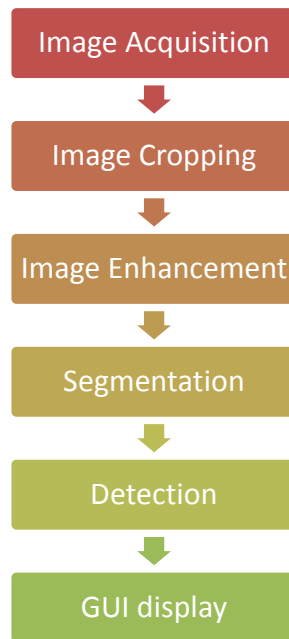
Figure 3.1: The process of the car plate detection system

3.3 Image Acquisition

In the first part of this process is the image acquisition. This part is the important phase of image processing for this system because this part needs to obtain the images of vehicles. Some electronics devices such as optical camera, webcam and handphone camera can be used to capture the acquired images.

For this project, the vehicles image is taken using a Canon PowerShot A3000IS digital camera from the consideration angle. Then the images will be stored in JPEG format in RGB color on the camera. Then, by using the image processing in MATLAB tools, the RGB image of the vehicles will be converted into a grayscale format.

Figure 3.2: The sample image of the vehicles with the consideration angle

4.3.1 Extract Digit

At first, determine the angle of the Car Plate is important. Capturing the car image need to be from the suitable angle and the car plate is clearly seen. The figures below are the wrong and the correct angle of the car image.

Figure 3.3: The wrong captured of car image angle

The angle is not suitable because it will cause distraction at the image processing part. This hard for the system to detect the important image on the car plate and make the detection failed to detect the plate number. For the best angle to capture images of the car is shown in the Figure 3.2.

3.4   Image Cropping

The second part of this process is image cropping. This part is a detection process where it will extract the smallest rectangle which will contains the edge of the license plate and license plate itself. This process will highly increase the speed of image processing. The figure below shows the cropping image of the car plate:

Figure 3.4: The car plate image cropped

3.5 Image Enhancement

This process is to increase and enclosure the visibility of the image. The image enhancement techniques consist with sharpening the edges of image process, contrast manipulation, reducing the image noise, color image processing and image segmentation.

3.5.1 Histogram Processing

This process also called as equalization process and the basis for numerous spatial domain processing techniques. The process will alter an image by changing its histogram. The histogram of an image shows the distribution gray level in the image. This process is easy to calculate in software and using an economic hardware. The formula for the histogram equalization transformation is given by:

$$S_k = T(r_k) = \sum_{j=1}^{k} p_r(r_j) = \sum_{j=1}^{k} \frac{n_j}{n}$$

Histogram equalization produced an adaptive transformation function and the histogram will not change unless there is any change in the image. Histogram

matching is very similar to Histogram equalization but it need to specify the histogram shape in which it is able to highlight the given image gray level range. Figure 3.4 is the example of the histogram process of car plate image.
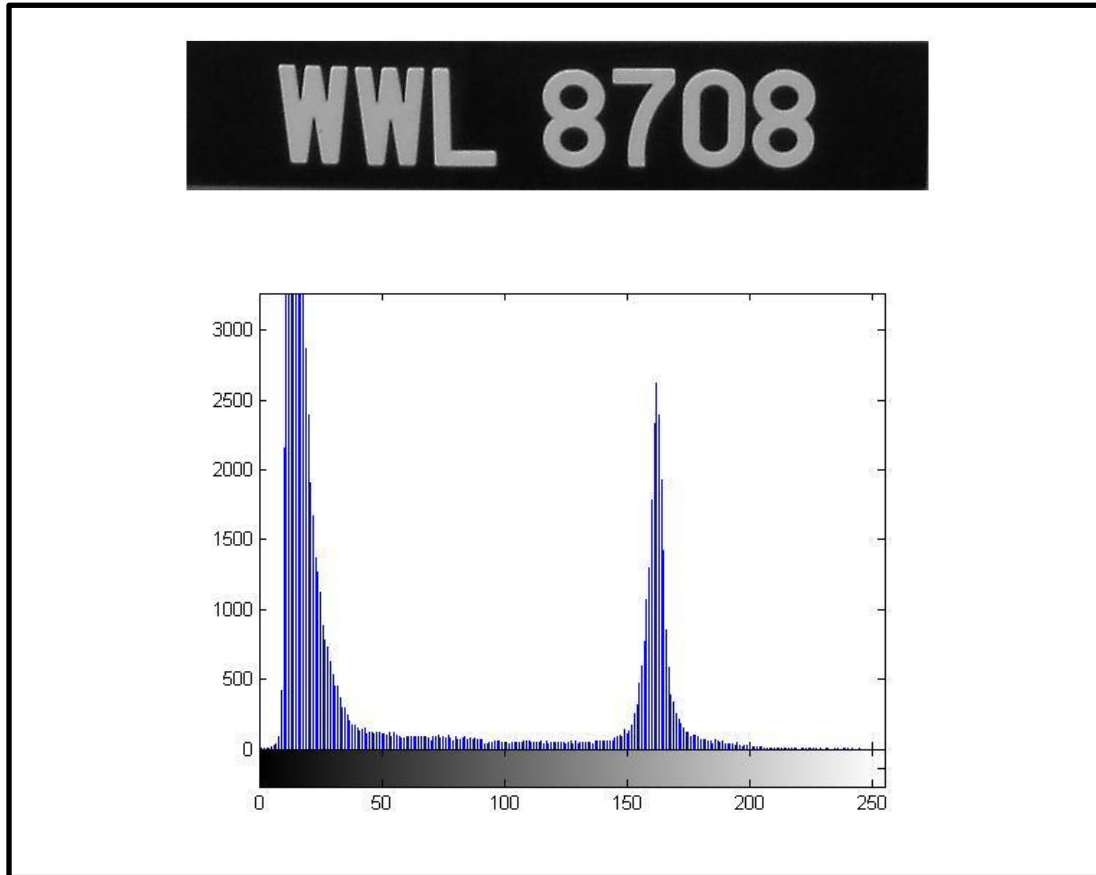

Figure 3.5: The histogram process of car plate image

3.5.2   Quantization and Equalization

To make sure the result success in recognizing the image of car license plate, the contrast image should be less. The very dark image, high contrast image and low contrast image in the binary image will not allow making the difference between the background and the digits. The figure 3.5 below show a contrast image which lead to wrong detection.
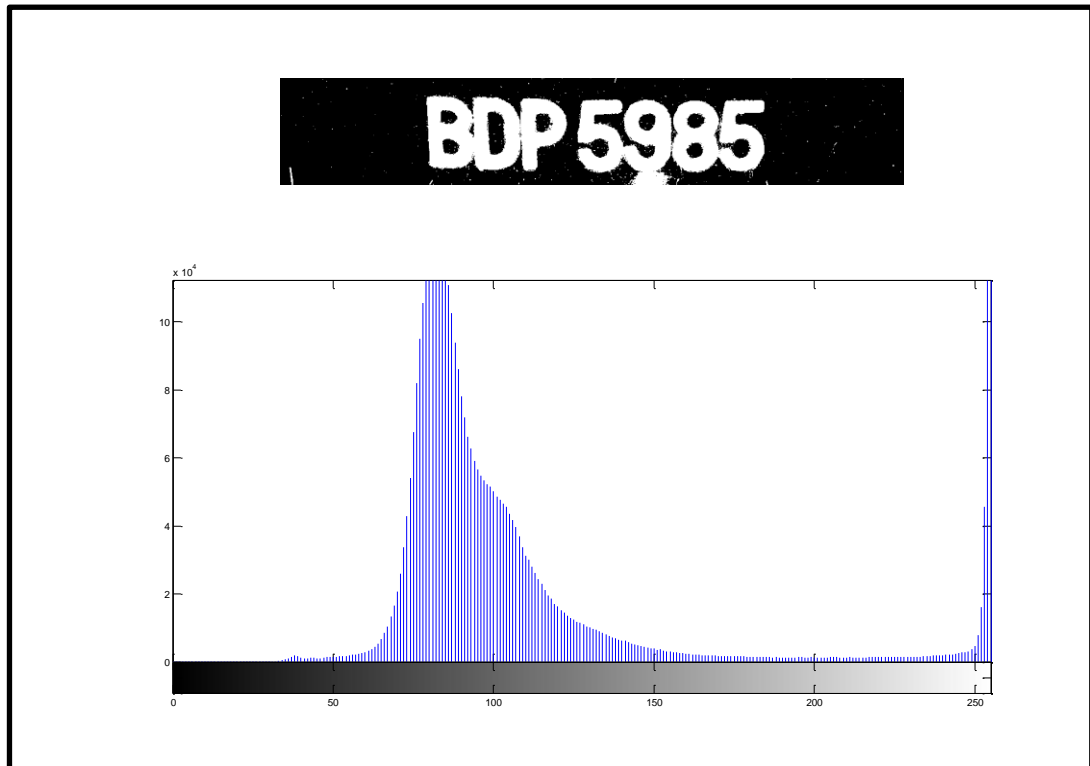
Figure 3.6: The low contrast of car plate image

3.5.3 Checking and Verification of Statistic

The objects areas in the binary image compute by using the MATLAB toolbox function BWAREA. The algorithm to compute the object area of the car plate image as written below:

```
for loop = 1:1:numOfDigits1
    total(loop) = bwarea(find(label1==loop))
end
```

The result from the algorithm is shown in figure 3.7 below:

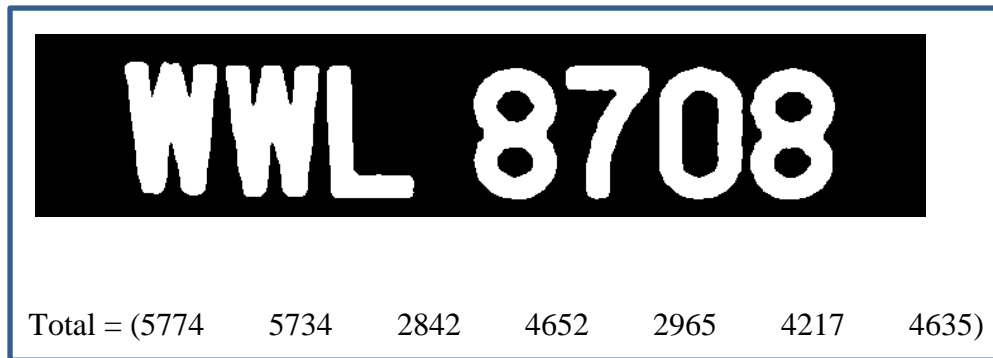Total = (5774   5734   2842   4652   2965   4217   4635)

Figure 3.7: The binary image of car plate

After that, the extracting process is use. To make sure the image is not taken any unrelated image from the car plate, the unrelated image need to remove. By using the MATLAB toolbox syntax, some adjustment on the value and applying the skill of binary area open (BWAREAOPEN) which help removing the small object on the car plate. This to make sure the clear image is obtained and the technique of erosion can be applied to remove unwanted details from the binary image. Figure 3.8 below is the example of the scenario:
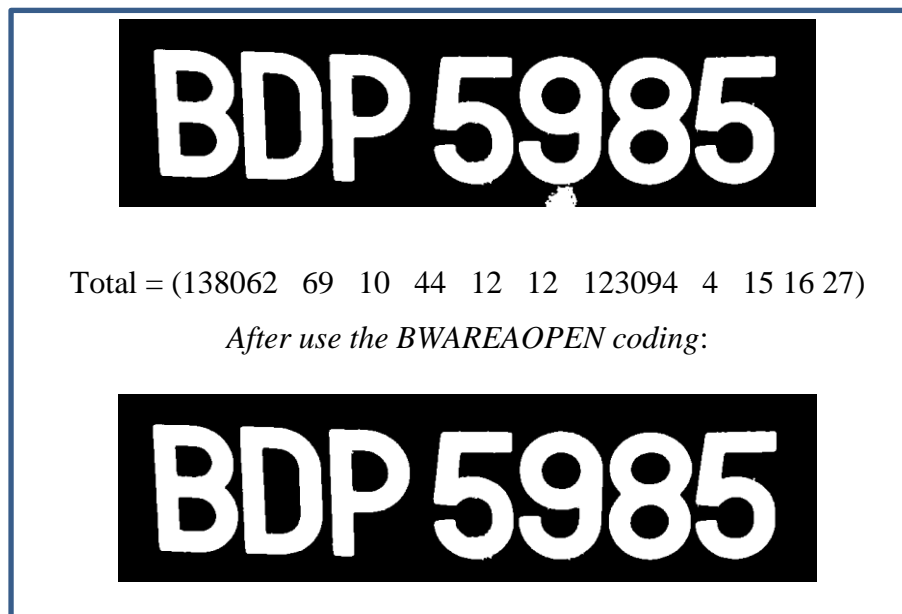

Total = (138062 69 10 44 12 12 123094 4 15 16 27)

After use the BWAREAOPEN coding:

Figure 3.8: The effect of morphologically open binary image

3.6   Image Segmentation

Image segmentation is an important and critical step which leads to the analysis of the processed image data. The process will subdividing an image into consistent regions or objects. In order to make sure the process of character detection easier, the process needs to partition each of the characters on the car plate. For this project, the horizontal and vertical projection of Malaysian standard license plate is the most suitable technique for this character partition or segmentation. The following steps are used for this process:

i.   Threshold the car plate image.

ii.   Resize the character into suitable pixels.
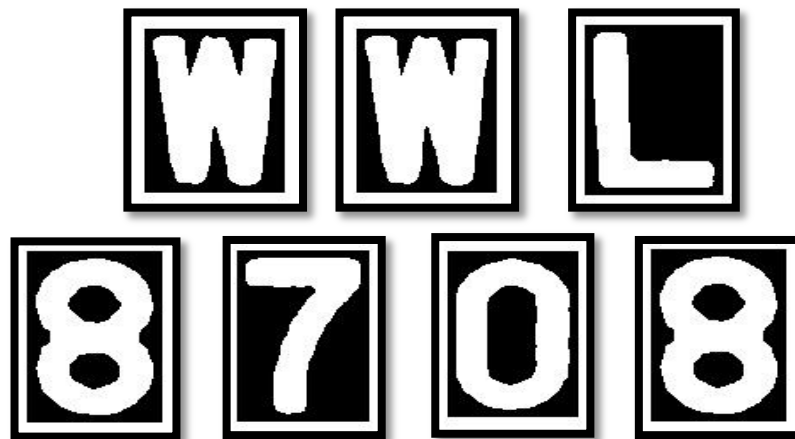

Figure 3.9: The extraction of car plate image

Figure 3.10: The character segmentation by MATLAB coding

3.7   Image Detection

The last process of this system is the image detection. Each of the segmented characters from the previous image is resized to fit the characters into a window. This detection method called template matching. The template matching is the process of comparing each character to the set of template character (reference template).

The template matching used two types of test point that is white test point and black test point. The white test point is to test the white area on the characters; meanwhile the black test point will be used to test the black area. The formula below is the test matching equation:

$$\text{Result} = \frac{\sum Match\ point}{\sum Test\ point}\ x\ 100\%$$

From the equation, the black point will fall in the black pixels, while the white points will fall in white pixels that are defined by matched point. During the template matching process, the reference template is loaded one by one to test the character image. This case called template priority where the higher priority of the template was found matched then the lower priority of the template is being ignored. In this project the samples of reference templates being used are in size 42x24 pixels.

3.8   MATLAB Programming

In this project, the system will be developed using MATLAB R2010a version. This software is designed to make the user easily control the process of locating the image, processed the car plate image and also developed a Graphical User Interface (GUI) application window as the main paned of this Car Plate Detection System.

3.8.1   Graphical User Interface (GUI)

The graphical user interface can develop using MATLAB and it is much easier than other C++ programming software. GUIDE stands for Graphical User Interface Development Environment which can be stored into two types of files:

i.   .fig file : the file contains a complete description of GUI layout and GUI components like control button, axes and etc.

ii.   .m file : the file contains of code to control the GUI. All the code initialization and callbacks are held in this file.

First step, type the word "guide" at the MATLAB command window. Then the "GUIDE Quick Start" window will appear.
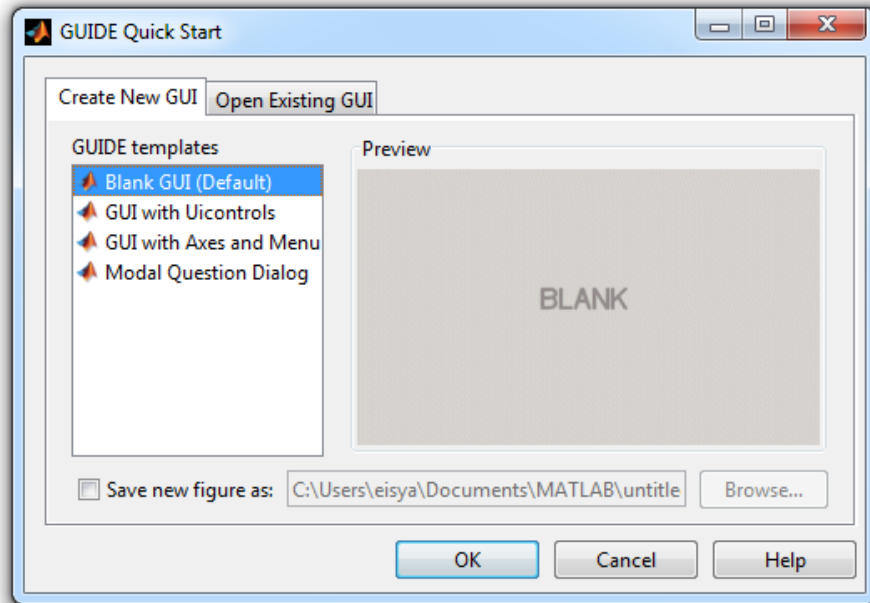
Figure 3.11: The GUIDE Quick Start window

Then, choose the "Blank GUI (Default)" to start designing the GUI. After that, the new window will appear. This window is where will the GUI will be designed and how will be it looks like. The complete GUI design of the system is shown in figure 3.16.
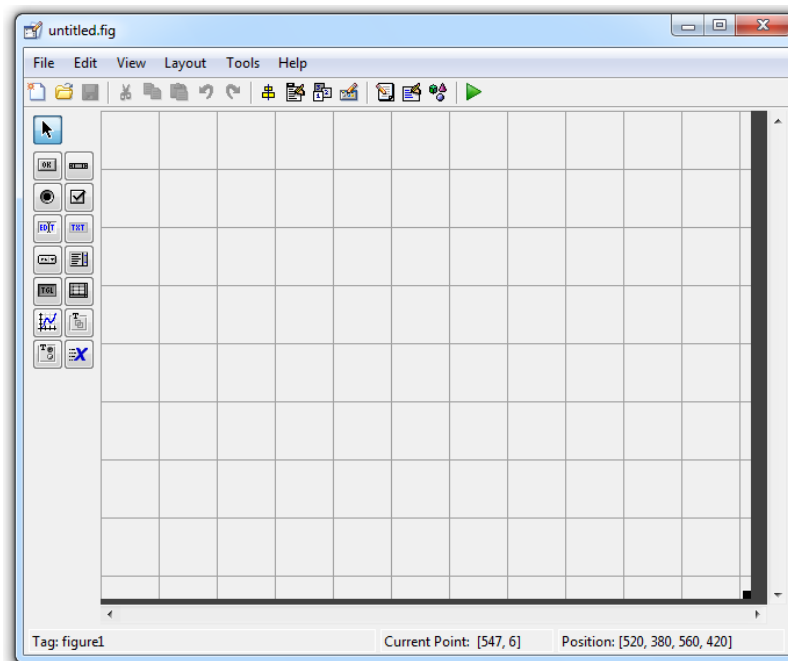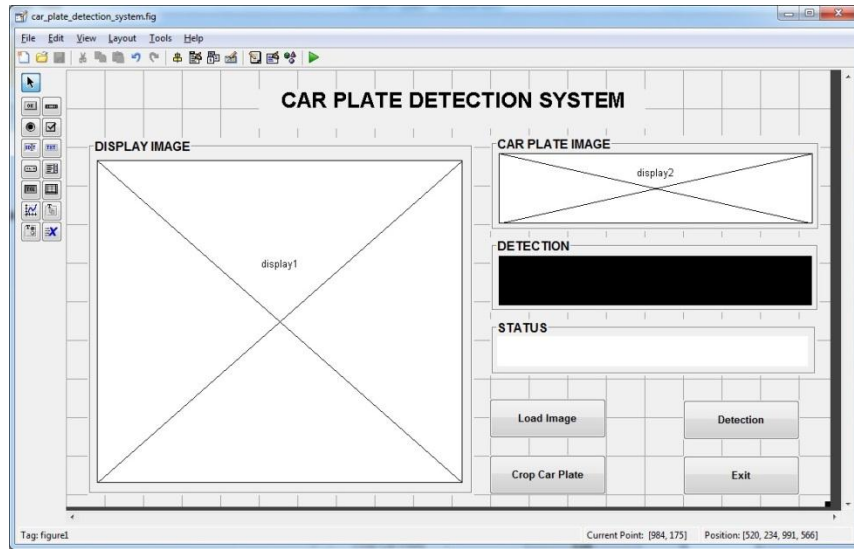

Figure 3.12: The blank GUI window design

Figure 3.13: The complete GUI design for Car Plate Detection System window

CHAPTER 4

RESULT AND DISCUSSION

This section will discuss about the result obtained during the execution of the project. Some explanation and discussion for each of the results obtained writes in this section to overview each result. Step by step using the Car Plate Detection System is also written in this section.

4.1   Result

After applying all the methodology and procedure has been done, the image taken and stored in the hard disk can be processed. The table below is the lists of the computer equipment that use for this project.

Table 4.1: The lists of computer equipped

Operating System	Windows 7
Processor	AMD Athlon™ II P340 Dual-Core Processor 2.20GHz
RAM	3.00GB
System type	32-bit Operating System
Image type	JPEG (RGB color)
Camera model	Canon PowerShot A3000IS

4.1.1 Image Display in GUI

Press the "Load Image" button to display the car image. Then, the image will display at the "Display Image" box. After that, press the "Crop Image" button and then make a small rectangular area around the car plate image. The figure below is illustrating the process of the system.
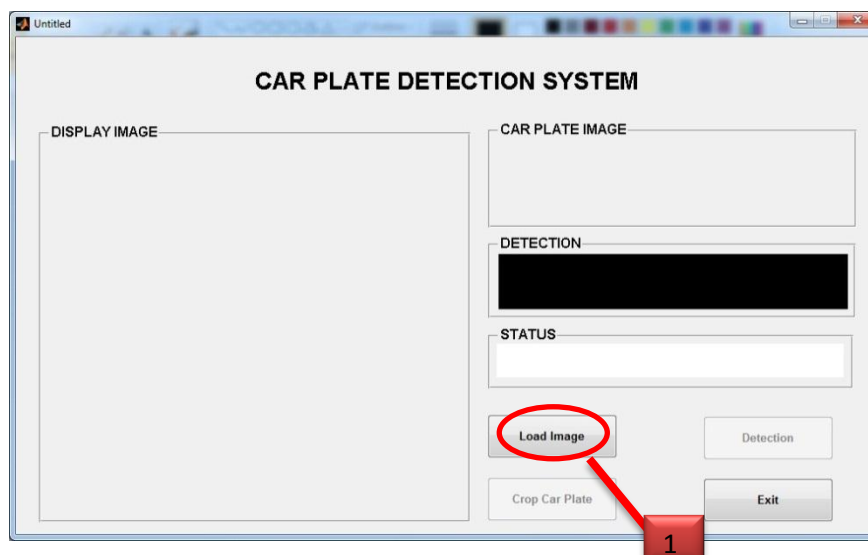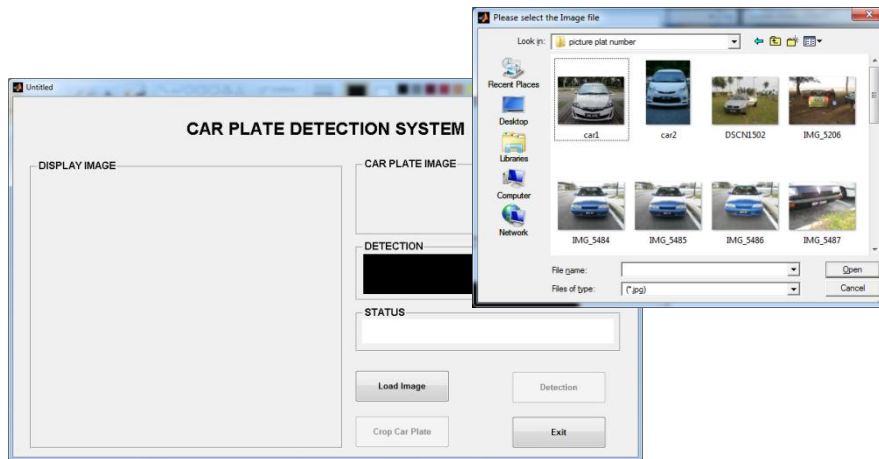

Figure 4.1: Press the "Load Image" button first
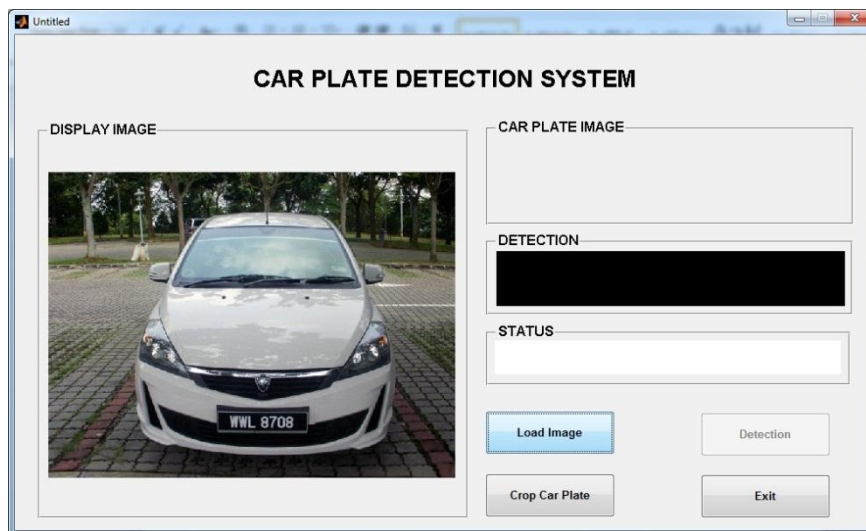
Figure 4.2: Locate the car image on harddisk

Figure 4.3: The image display at "Display Image" box

Figure 4.4: Crop image rectangle

After making the rectangle crop at the car plate area, double click the rectangle. Then, the car plate image that have been cropping will display at the "Car Plate Image" box.
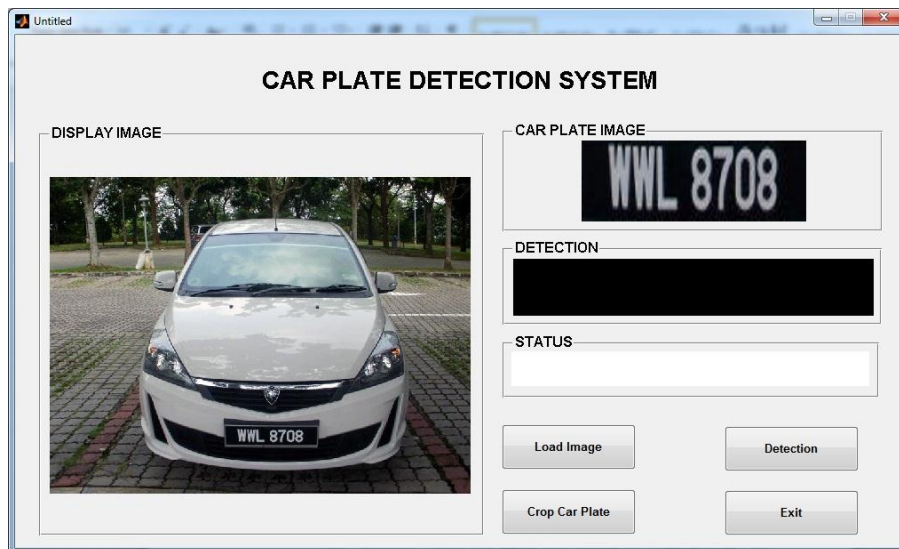

Figure 4.5: The car plate image display on the box

The "Detection" button will appear after making the cropping image. The next step is to make the character detection on the car plate image that have been cropping. Press the "Detection" button and the image processing and template matching process will take place. After finishing making the process, the character detection will display at "Detection" box and also display the status of the car

whether it is in the excel database or not. If car is in the excel database, the status will display "Access Granted", if not the status will display "Access Denied". For the full results of this system can find in Appendix C page 46.
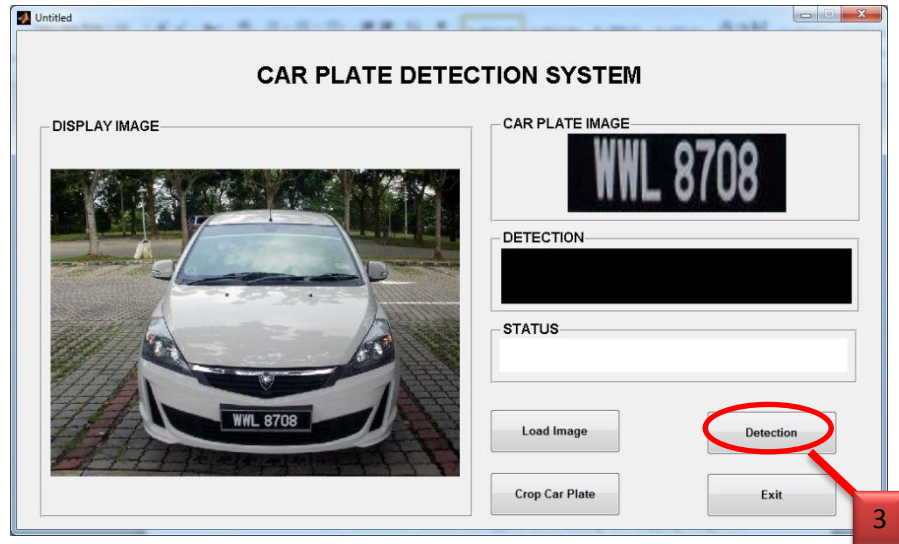

Figure 4.6: Press the "Detection" button

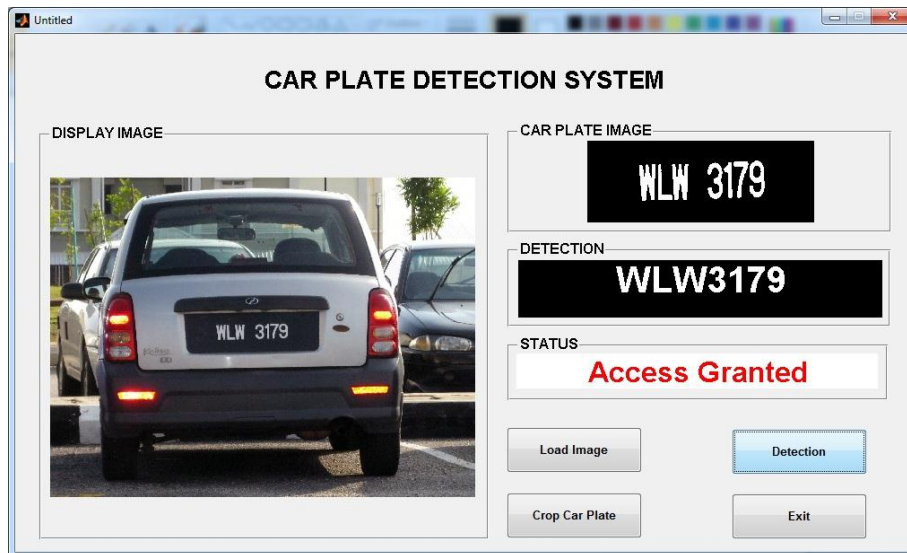Figure 4.7: The status of the car is not in database

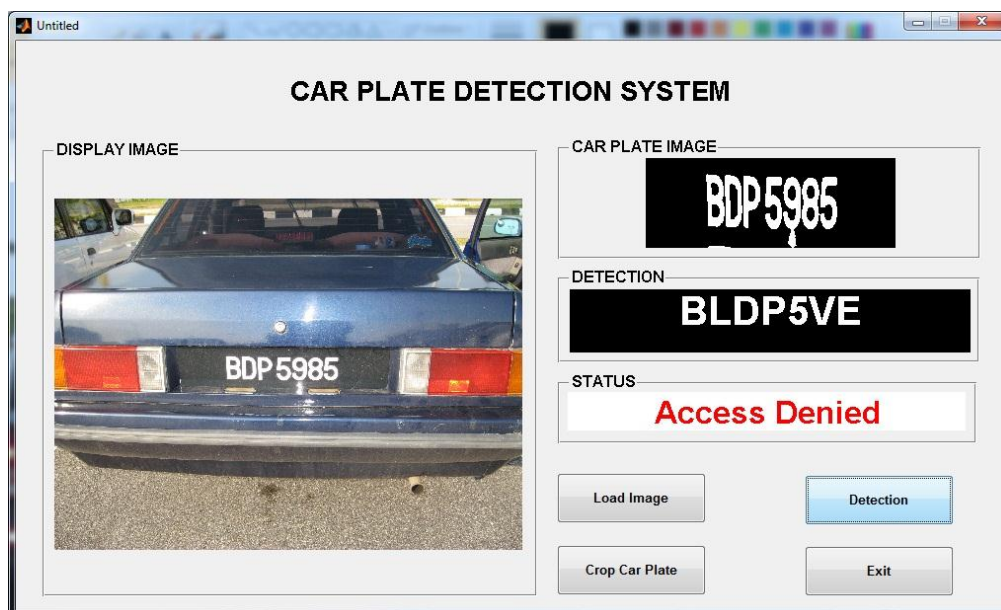Figure 4.8: The status of the car is in the excel database

Figure 4.9: The example of car image taken in bright sunlight

4.2   Discussion

There are several reasons why some of the images not able to be detected by the system. The list below is the possibility of the failed cases happen of the system:

i.   The image of the car plate is too bright because of the direct sunlight of the car plate image while taking the image. This problem makes difficulty to make a thresholding process.

ii.   The car plate character is too close and will cause problems at segmentation process. This problem makes the isolation of the car plate character became more difficult.

iii.   Image cropping is not cropping properly then the unwanted objects on the car plate image also participate in the detection process. Then, the process cannot compare the image with the template database.

CHAPTER 5

CONCLUSION

5.1   Introduction

This chapter discusses about the conclusion overall project that have been done. Thus, some future recommendations also given for further study in image processing.

5.2   Conclusion

The Car Plate Detection System in this project successfully implemented using MATLAB R2010a. The GUI helps user to easily understand and used. This system can be applied at the gate main entrance, parking security and access control system. However, there are still needs some changes and repair on the code. In the future recommendation section will discuss the future study for this project as a reference for students.

5.3   Future Recommendation

This project still needs some improvement in detecting the character on the image. For the future study and suggestion on improvements, below are the lists which are recommended:

i.   This project can only implement for the car plate in black and white image. For the future implementation, we can use RGB to HSV or RGB to CMY method to deal with another color of license car plate.

ii.   The system can be implemented in real-time and capture the image from the webcam. Then it can incorporate by using the GUI to MATLAB which in line with a webcam.

iii.   The cropping image of the car plate maybe can be crop by automatically compare to this project where the user need to cropping the car plate manually.

REFERENCES

[1]   26th September 2011, *Automatic Number Plate Recognition - Wikipedia, the free encyclopedia.* URL: http://en.wikipedia.org/wiki/Automatic_number_plate_recognition.

[2]   Salgado, L., Menendez, M. J., Rendon, E., Garcia, N., "*Automatic Car Plate Detection and Recognition Through Intelligent Vision Engineering*", Available:

[3]   Ganapathy, V., Dennis, W. L., "*A Malaysian Vehicle Licence Plate Localization and Recognition System*",

[4]   Odone F., "*Experiment on a License Plate Recognition System*",DISI, University degli Studi di Genova

[5]   Chang, L. S., Chen, S. L., Chung, C. Y., Chen, W. S., "*Automatic Licence Plate Recognition*", IEEE Transactions on Intelligent Transportation Systems, Vol. 5, No.1, March 2004, pp. 42-53. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1271288&tag=1.

[6]   Blanchet, G., Charbit, M., (2006). *Digital Signal and Image Processing Using Matlab* : ISTE Ltd.

[7]   26th September 2011, *The Automatic Number Plate Recognition Tutorial.* URL: http://www.anpr-tutorial.com/.

[8]   26th September 2011, *An Introduction to ANPR.* URL: http://www.cctv-information.co.uk/i/An_Introduction_to_ANPR .

[9]   26th September 2011, *Plate Recognition.* URL: http://www.photocop.com/recognition.htm .

[10]   27th September 2011, *Digital Image Processing – Wikipedia, the free encyclopedia*. URL: http://en.wikipedia.org/wiki/Digital_image_processing .

[11]   29th September 2011, *Artificial Neural Network – Wikipedia, the free Encyclopedia*. URL: http://en.wikipedia.org/wiki/Artificial_Neural_Network

[12]   30th September 2011, *Graphical User Interface - Wikipedia, the free Encyclopedia*. URL: http://en.wikipedia.org/wiki/Graphical_user_interface

APPENDIX

APPENDIX A

SOURCE CODE

```matlab
function varargout = car_plate_detection_system(varargin)
% CAR_PLATE_DETECTION_SYSTEM M-file for
car_plate_detection_system.fig
%      CAR_PLATE_DETECTION_SYSTEM, by itself, creates a new
CAR_PLATE_DETECTION_SYSTEM or raises the existing
%      singleton*.
%
%      H = CAR_PLATE_DETECTION_SYSTEM returns the handle to a new
CAR_PLATE_DETECTION_SYSTEM or the handle to
%      the existing singleton*.
%
%
CAR_PLATE_DETECTION_SYSTEM('CALLBACK',hObject,eventData,handles,...)
calls the local
%      function named CALLBACK in CAR_PLATE_DETECTION_SYSTEM.M with
the given input arguments.
%
%      CAR_PLATE_DETECTION_SYSTEM('Property','Value',...) creates a
new CAR_PLATE_DETECTION_SYSTEM or raises the
%      existing singleton*.  Starting from the left, property value
pairs are
%      applied to the GUI before
car_plate_detection_system_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to
car_plate_detection_system_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows
only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
car_plate_detection_system

% Last Modified by GUIDE v2.5 26-May-2012 23:42:03

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn',
@car_plate_detection_system_OpeningFcn, ...
                   'gui_OutputFcn',
@car_plate_detection_system_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```

```matlab
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before car_plate_detection_system is made
visible.
function car_plate_detection_system_OpeningFcn(hObject, eventdata,
handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to car_plate_detection_system
(see VARARGIN)

% Choose default command line output for car_plate_detection_system
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes car_plate_detection_system wait for user response
(see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = car_plate_detection_system_OutputFcn(hObject,
eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
set(handles.crop,'enable','off');
set(handles.detection, 'enable','off');


% --- Executes on button press in load.
function load_Callback(hObject, eventdata, handles)
% hObject    handle to load (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%Read image in file by locate the image in disk
[FileName,PathName] = uigetfile('*.jpg','Please select the Image
file')
car_image = imread([PathName '\' FileName]);

handles.car_image = car_image;

guidata(hObject, handles);
```

```matlab
axes(handles.display1)
car_image = imresize(car_image,[480 640]);%resize the car image into
640x480

%display the car image
imshow(car_image)

set(handles.crop,'enable','on')
set(handles.detection, 'enable','off')

cla(handles.display2)
set(handles.text5,'string','')
set(handles.status,'string','')




% --- Executes on button press in crop.
function crop_Callback(hObject, eventdata, handles)
% hObject    handle to crop (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

warning off
%clc, close all, clear all

car_image = handles.car_image;
axes(handles.display1)
set(handles.display1,'Visible','on')
[plate_RGB carplateRect] = imcrop(car_image);%crop the car plate
image manually

plate_RGB = imresize(plate_RGB,[170 473]);%resize the plate image
into 473x170


handles.plate_RGB = plate_RGB;
guidata(hObject,handles);

axes(handles.display2)

%display the image cropping
imshow(plate_RGB)

set(handles.detection,'enable','on')




% --- Executes on button press in detection.
function detection_Callback(hObject, eventdata, handles)
% hObject    handle to detection (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

plate_RGB = handles.plate_RGB;

if length(size(plate_RGB))==3
```

```matlab
        plate_BW=rgb2gray(plate_RGB);
end

%filter image noise
plate_BW = medfilt2(plate_BW);
[f c] = size(plate_BW);
plate_BW (1,1) = 255;
plate_BW (f,1) = 255;
plate_BW (1,c) = 255;
plate_BW (f,c) = 255;

%threshold the plate image
level = graythresh(plate_BW);
plate_BW = im2bw(plate_BW,level);%convert image to binary image

plate_BW_filter = bwareaopen(plate_BW,30);%remove any small objects
on the image

%display the image in binary image
imshow(plate_BW_filter)

%storage the matrix word from the plate car image
word = [ ];
re = plate_BW_filter;

%load templates file
load templates
global templates

%compute the number of letters in template file
alphanumeric=size(templates,2);


[L Ne] = bwlabel(re);

for n=1:Ne
    [r,c] = find(L==n);

    n1=re(min(r):max(r),min(c):max(c));

    img_r=imresize(n1,[42 24]); %resize the letter image into 24x42

    img_r = imcomplement(img_r); %complement the image template

    figure,imshow(img_r);%pause(0.5)

    letter=read_letter(img_r,alphanumeric);%convert image into text

    word=[word letter];

    handles.result=word;

end

%display the image result
set(handles.text5,'string',handles.result);
```

```matlab
%check database in excel database
[numCP, txtCP] = xlsread('CPDSdatabase.xls','Authorise');
txtCP = upper(txtCP);

if isempty(strmatch(handles.result,txtCP))
    set(handles.status,'String','Access Denied')
else
    set(handles.status,'String','Access Granted')
end

guidata(hObject, handles);



% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
exitbtn=questdlg('CLOSE THE SYSTEM?','Exit
Program','Yes','No','No');
switch exitbtn
    case 'Yes'
        delete(handles.display1);
        delete(handles.display2);
        delete(handles.status);
        delete(handles.text5);
        close
    case 'No'
        return
end



% --- Executes during object deletion, before destroying properties.
function display1_DeleteFcn(hObject, eventdata, handles)
% hObject    handle to display1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)



% --- Executes during object creation, after setting all properties.
function display1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to display1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: place code in OpeningFcn to populate display1
```
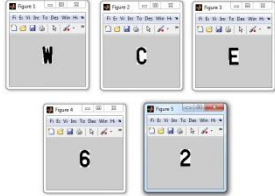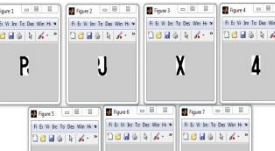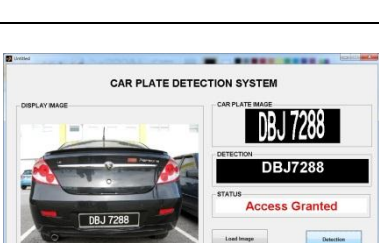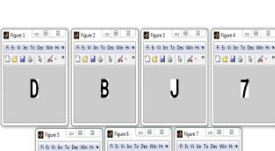
APPENDIX B

EXCEL DATABASE

Plate Number
WCE62
WLW3179
WTP3531
WKQ272
CBQ4456
CCG3680
SAA3641M
CCU7342
PJX4004
DBJ7288
AHD5046

APPENDIX C

CAR PLATE DETECTION SYSTEM RESULTS

PLATE NO	IMAGE PROCESSING	SEGMENTATION	COMMENT
WCE 62			100% detection, OK
WLW 3179			100% detection, OK
WTP 3531			100% detection, OK
PJX 4004			100% detection, OK
DBJ 7288			100% detection, OK

AHD 5046			**100% detection, OK**
WWL 8708			**100% detection, OK**
WKQ 272			**70% detection, image too bright**
BDP 5989			**80% detection, image too bright**
CBQ 4456			**Error detection, image too bright**

GANTT CHART PSM I

Task	Week														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PSM 1 Briefing	■														
Find supervisor & project title	■	■													
Analyze & understanding the topic		■	■												
Problem statement		■	■												
Literature review (article that related to this topic)		■	■	■	■	■	■	■	■	■	■	■	■		
Research methodology				■	■	■	■								
Submit draft 1						■									
Prepare proposal & presentation slide				■	■	■	■								
Submit proposal & presentation slide								■							
Data collection and analysis								■	■						
Submit draft 2									■						
Prepare for seminar PSM 1										■	■				
PSM 1 seminar											■				
Prepare report for PSM 1				■	■	■	■	■	■	■	■	■	■	■	
Submit report for PSM 1														■	

GANTT CHART PSM II

Task	Week														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PSM 2 Briefing	▨														
Tools preparation	▨	▨	▨												
Design and simulation (software)		▨	▨	▨	▨										
Assemble & testing				▨	▨	▨	▨								
Submit draft 1					▨										
Program the controller module					▨	▨	▨	▨							
Testing the programmed					▨	▨	▨	▨	▨	▨					
Submit draft 2									▨						
Finalize overall project										▨	▨	▨	▨	▨	
Submit final draft											▨				
Seminar PSM 2 preparation												▨	▨	▨	
Thesis preparation		▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	
Seminar PSM 2 & demo														▨	
Submit thesis														▨	