

AUTOMATIC TRAFFIC VOLUME CENSUS USING VIDEO

By

Muhammad Syazwan Bin Baharuddin

Final Year Project

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfilment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

©Copyright 2010
by
Muhammad Syazwan Bin Baharuddin, 2010

CERTIFICATION OF APPROVAL

AUTOMATIC TRAFFIC VOLUME CENSUS USING VIDEO

by

Muhammad Syazwan Bin Baharuddin

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL & ELECTRONICS ENGINEERING)

Approved by,

(Dr. Fawnizu Azamadi B. Hussin)
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

December 2010

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

MUHAMMAD SYAZWAN BAHARUDDIN

ABSTRACT

Image processing has been applied to traffic analysis in recent years, with different goals. In this report, an automatic system using video is proposed for extracting numbers of vehicles and vehicle classification. The aim of this project is to improve the traditional method of traffic volume census and to promote the application of image processing with obvious result of cost and quality. Through this project, the MATLAB software will be used to develop algorithm for video processing and a standalone executable file will be created based on the MATLAB algorithm. The result is expected that proper system can be developed to show the function of vehicle classification and counting. So, the result will provide accurate data which is important in traffic management system and road planning.

ACKNOWLEDGEMENTS

First and foremost, there were many people, individuals and parties that involved while working for my Final Year Project 2 which has contributed immeasurable and invaluable amount of help, guidance, assistance, advice, supports, and also supervision. A very high gratitude and appreciation to all of them for the commitments and help me to achieve the objectives and then made my Final Year Project 2 completed successfully.

Special thanks to Dr. Fawnizu Azmadi Bin Hussin, my final year project supervisor for his willingness to become my supervisor. I also would like to say thank you again to him for given me lots of supportive, constructive advices, and sharing his knowledge. Without his guidance, the project cannot be completed.

Special thanks also go to both my parents for their endless support and encouragement. Their advices always motivate me to work very hard and never give up until my project finish.

I also gratefully acknowledge all my friends who have given invaluable review and feedbacks on my project especially those who helped me working with MATLAB software. All your guidance, support and cooperation are very important in completing the Final Year Project 2.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL.....	ii
CERTIFICATION OF ORIGINALITY.....	iii
ABSTRACT.....	iv
ACKNOWLEDGEMENT.....	v
LIST OF FIGURES.....	viii
CHAPTER 1: INTRODUCTION.....	1
1.1 Background of Study.....	1
1.2 Problem Statement.....	2
1.3 Objectives	2
CHAPTER 2: LITERATURE REVIEW AND THEORY.....	3
2.1 Introduction	3
2.2 Vehicle Detection.....	3
2.2.1 Automatic Background Estimation	4
2.2.2 Vehicles detection based on background extraction.....	5
2.2.3 Vehicles detection based on background estimation.....	5
CHAPTER 3: METHODOLOGY.....	7
3.1 Flowchart.....	7
3.2 Detail of the Flowchart.....	8
3.3 MATLAB algorithm development.....	9
CHAPTER 4: RESULTS AND DISCUSSION.....	10
4.1 Data gathering	10
4.2 Vehicle counting	13
4.3 Standalone executable file.....	17

CHAPTER 5: CONCLUSION AND RECOMMENDATION.....	19
5.1 Conclusion.....	19
5.2 Recommendation.....	19
 REFERENCES.....	 20

LIST OF FIGURES

Figure 1: Automatic background extraction.....	4
Figure 2: Workflow of the project	7
Figure 3: Workflow of algorithm development.....	9
Figure 4: Digital compact camera	10
Figure 5a: Side View	12
Figure 5b: Top View	12
Figure 6: Original image	13
Figure 7: Greyscale image	13
Figure 8:MATLAB Algorithm.....	14
Figure 9: Foreground Image.....	14
Figure 10:MATLAB Algorithm.....	15
Figure 11: Final Image.....	15
Figure 12: Result in MATLAB.....	16
Figure 13: Compiler Configuration.....	17
Figure 14: Result in Standalone Executable File.....	18

LIST OF ABBREVIATIONS

SVM	Support Vector Machine
HOS	Higher Order Statistics
GHT	Generalized Hough Transform
PCA	Principal Component Analysis
DP	Difference Product
MOR	Moving Object Region
EF	Extraction Flag
ETA	Extraction Target Area

CHAPTER 1

INTRODUCTION

1.1 Background

Nowadays, a lot of traffic analysis has been done using image processing with different goals such as queue detection, incident detection, vehicle classification, and vehicle counting [1].

This report explicitly recognizes that traffic volume is an important data for traffic management system and road planning. Relatively few efforts have attempted to detect moving vehicles by using video images. For example, Gupte et al [2] presents algorithms for vision-based detection and classification of vehicles in monocular image sequences of traffic scenes recorded by a stationary camera.

Tsai et al [3] describes a novel vehicle detection approach from static images using color and edges. Unlike traditional methods, which use motion features to detect vehicles, this method introduces a new color transform model to find important ‘vehicle color’ for quickly locating possible vehicle candidates. Anan et al [4] develops the video vehicle detection algorithm based on virtual-line group. The algorithms uses virtual-line group to optimize the states of detection lines in both spatial domain and temporal domain.

In [5], a moving object recognition method is described that uses an adaptive background subtraction technique to separate vehicles from the background. The background is modelled as low time-varying image sequence, which allows it to adapt to changes in lighting and weather conditions.

1.2 Problem Statement

Vehicle detection and classification of vehicles play an important role in decision making for the purpose of traffic control and management. Traffic management system and road planning rely on the accuracy of data for traffic volume census. However, the current situation is described by outdated data since the census is not conducted in daily or monthly basis. In addition, the traditional method requires human operators to count vehicles manually at a specific street. Thus, it cannot provide continuous data due to certain limitation in term of cost and human resources. For certain task like highway management, continuous data is very important in order for the service provider to manage their operation. Even in large metropolitan areas, there is need for data about vehicle classes that use a particular street. So, a classification system like the one proposed here can provide important data for a particular design scenario.

1.3 Objective and Scope of Work

Objective of this project is to develop an Automatic Traffic Volume Census using Video. Such an automatic system is essential to improve the traditional method of traffic census. In addition, the use of an automated system can lead to accurate design of pavements with obvious results in cost and quality. The scope of work will cover several tasks such as develop algorithms using MATLAB and develop standalone executable program.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Traffic volume census is the best method to provide information such as numbers of vehicles and type of vehicles, which are very important for traffic management and road planning. The census can be classified into two ways on how it is conducted. The first way is done manually by a human operator and the other way is by using automatic equipment. For this section, the methods which are using video or image processing will be discussed in detail.

2.2 Vehicle detection

Z. Kim et al. [6] presented one approach for vehicle detection using edge-based techniques. For this approach, the vehicle is represented by 3D model. This 3D model depends on the edge detection of the vehicle. However, it only can be applied for passenger vehicles under certain conditions. The edge detection also is not perfect due to several factors such as vehicle shadows, dark colors and ambient lights.

Saad et al. [7] proposed background subtraction and edge detection technique for vehicle detection. This method consists of two procedures: First, automatic background extraction procedure, in which the background is extracted automatically from the successive frames; Second vehicles detection procedure, which depend on edge detection and background subtraction.

2.2.1 Automatic Background Extraction

For this procedure [7], enough successive frames must be available to extract the background automatically. There are twelve steps involved and Fig. 1 below summarizes the steps of this process.

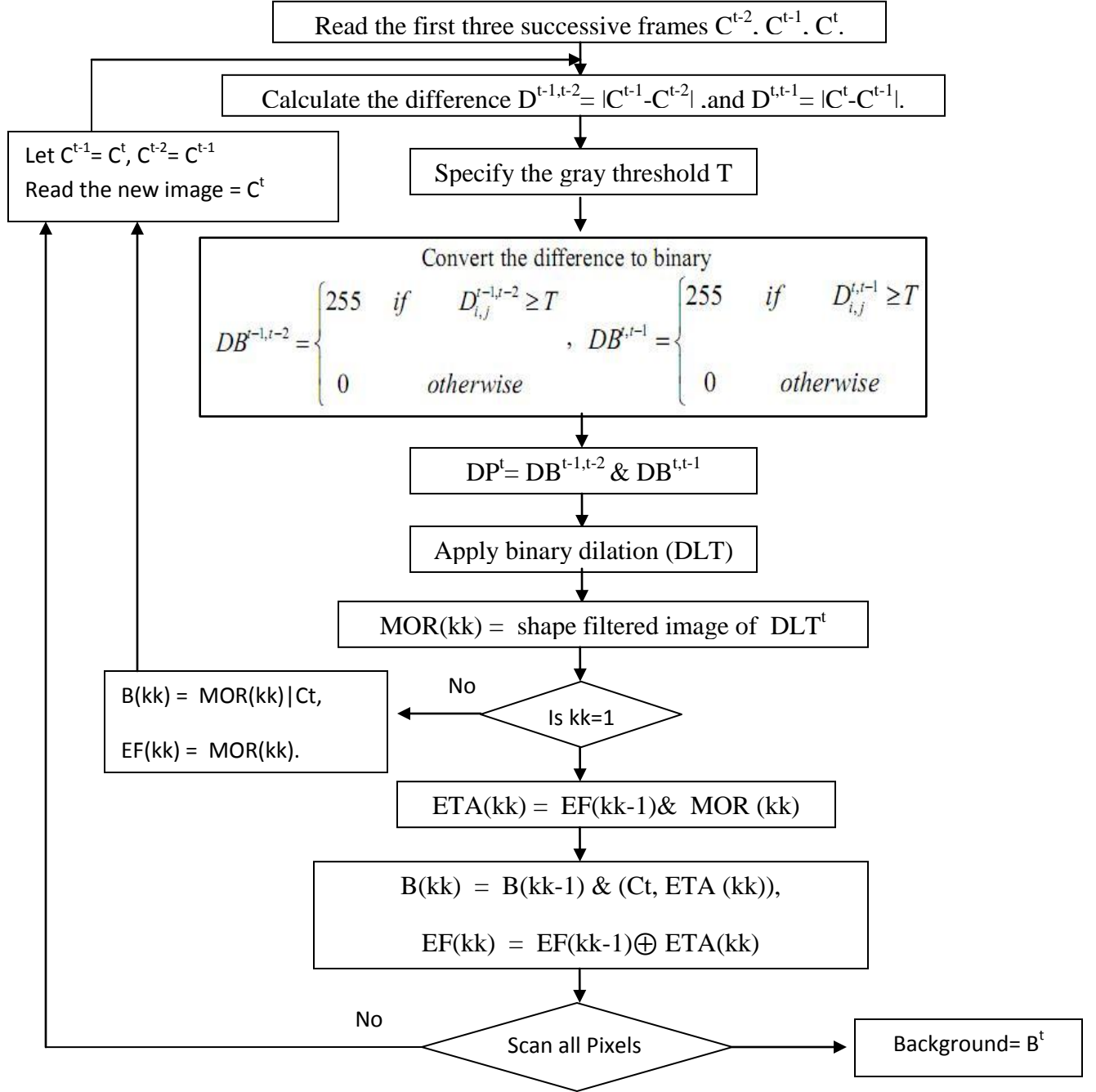


Figure 1 Flow chart for Automatic Background extraction

2.2.2 *Vehicles detection based on background extraction*

Saad et al. [7] also explained that the extracted background must be subtracted from the current image in order to detect vehicles. This method is summarized as in the following steps:

- Step 1: Subtract the extracted background from the current image.
- Step 2: Find the edge of the current image and the background image.
- Step 3: Subtract the edge of the background image from the edge of the current image.
- Step 4: Fill the resulted images in steps 2 and 3. Implement logical AND operation for the results in steps 2 and 3.
- Step 5: Filter the resulted image.
- Step 6: Count the resulted moving vehicles.

2.2.3 *Vehicles detection based on background estimation method*

Vibha L et al. [8] describes a solution to detect vehicles using background estimation method. After a video clip is read, it will be converted to frames. Then, there are several processes involved as following:

2.2.3.1 *Frame difference*

Frame differences are computed by finding the difference between consecutive frames. For this method [8], the background is assumed as stationary. Hence the difference between frames at regular intervals (k) is considered. If there are n frames, the frame differences (FD) will be (n/k) .

2.2.3.2 *Background elimination*

After the frame differences are computed, the value for pixels that belong to the background region is zero since the background is assumed stationary. However, it may not be zero due to several factors like noise. So, the values are set to zero by comparing any two frame differences. Thus, the background region is eliminated and only the moving object region will contain non-zero pixel values.

2.2.3.3 *Foreground detection*

Foreground dynamic objects can be obtained by subtracting the background image from the video frames. However, some noise regions exist in the image due to camera noise and irregular object motion. So, median filter is used to replace the value of a pixel by the median of the gray levels in the neighbourhood of that pixel. After applying the median filter, the resulting image is converted into a binary image.

2.2.3.5 *Vehicle counting*

The tracked binary image forms the input image for counting. When a new object is encountered, it is first checked to see whether it is already registered in the buffer. The counting will be incremented if the object is not registered. This process will keep continue for the entire image

CHAPTER 3

METHODOLOGY

3.1 Project identification

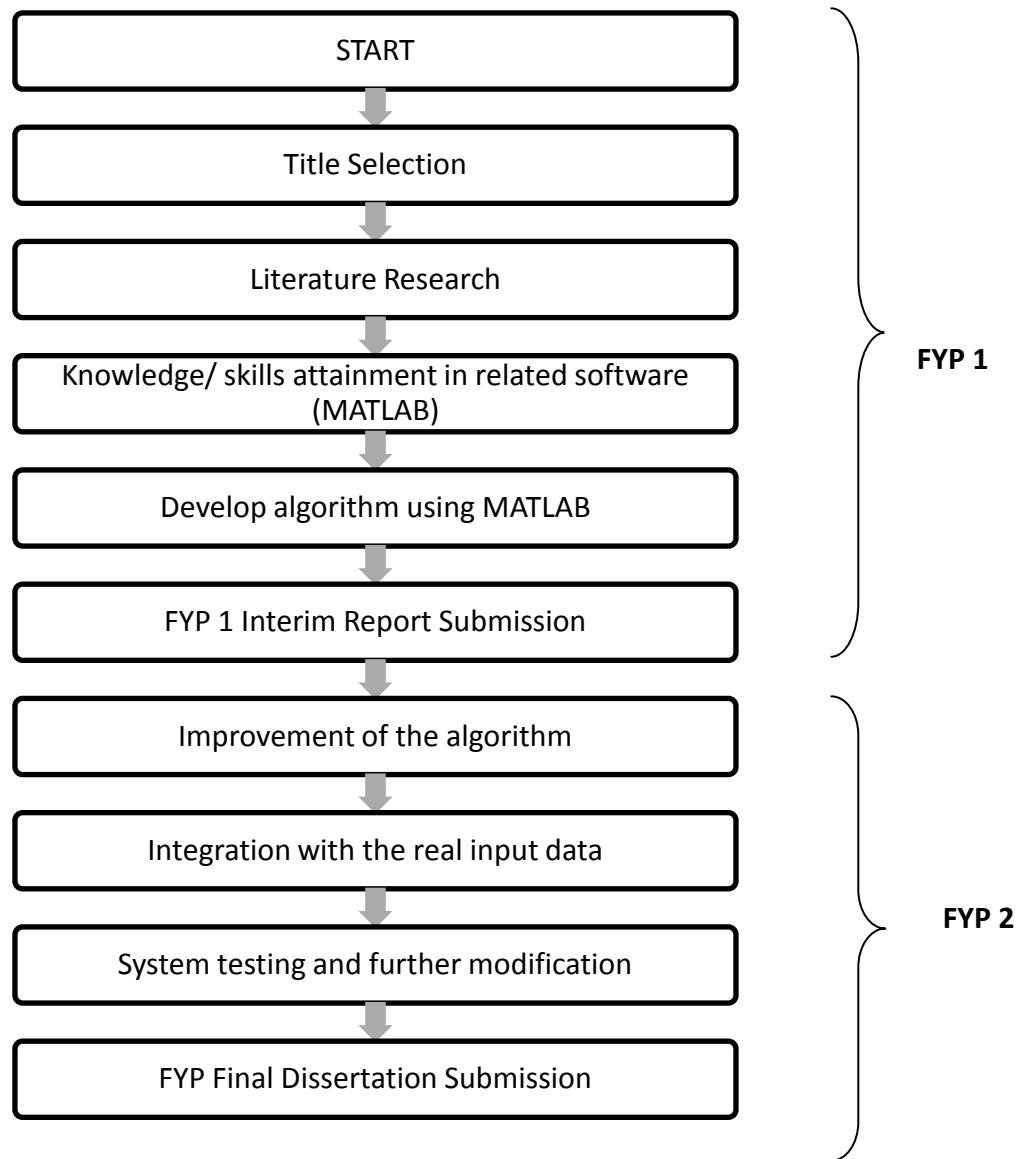


Figure 2: Workflow of the project

3.2 Detailed of the Flowchart

In order to realize the objective of this project and to have proper project co-ordination, the following strategies have been adopted:

Literature review has been done to find out the available method related to this project. At this stage, various sources such as research paper, textbooks and experimental theories have been referred in order to have basic understanding of video/ image processing.

Mathematical Software, MATLAB would be use for algorithm development and simulation of this project. MATLAB was chosen due to its greater flexibility. It is an interactive computer program containing various commands and functions that aid mathematical analysis of problems in engineering and science. MATLAB has excellent graphics capability, enabling the user to visualize complicated functions in a variety of ways to aid analysis.

As a high level language, MATLAB facilitates design exploration and algorithm development. Due to the fact that MATLAB is an interpreted language, M-files execute slower than compiled programs written in other languages, such as C, C++, and Fortran. Furthermore, M-files require the presence of MATLAB to run. Thus, standalone executable file is needed to show the work on computer that has no MATLAB environment in it.

3.3 MATLAB Algorithm development

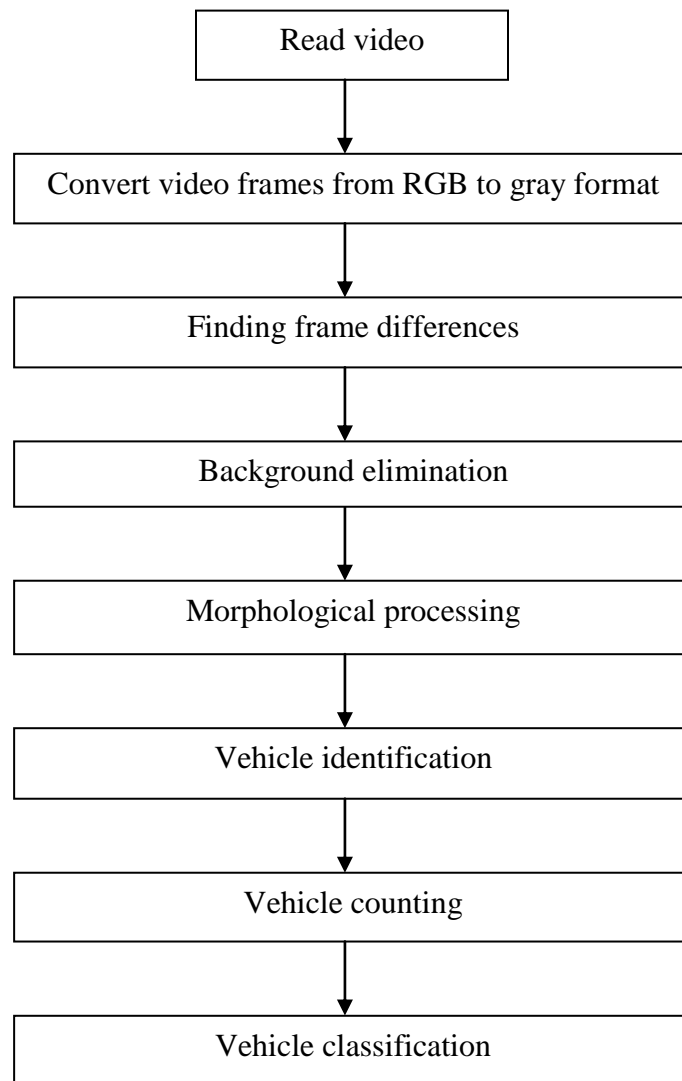


Figure 3: Workflow of algorithm development

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Data gathering

Several video samples had been taken as input for this project. These videos were recorded with different angles of view using digital compact camera. For the purpose of data gathering, the specification of camera should be standardized. Hence, BenQ DSC E800 as shown in Figure 4 had been used.



Figure 4

The detail specification of BenQ DSC E800 can be referred to Appendix A, but the main features are as following Table 1;

Image	8.1M Pixels
Zoom	Optical: 3X / Digital: 5X
Movie mode	640 x 480(15 frames per second)
Movie format	M_JPEG

Table 1

Figure 5 below shows on how data gathering had been done. As stated earlier, data gathering was done by recording the video from two different angles.



Figure 5a: Side view



Figure 5b: Top view

4.2 Vehicle counting

Several steps are involved in order to count number of vehicle. The counting process can be done with various methods as stated in literature review. However, the methods are not suitable for this project. For example, some methods only can detect and count one type of vehicle which is car while this project requires any type of vehicle can be detected and counted.

The first step is to convert the original image to grayscale image which can be done by eliminating the hue and saturation information while retaining the luminance. Figure 6 below shows the original image and Figure 7 shows grayscale image.



Figure 6 : Original Image



Figure 7: Grayscale Image

The next step is to remove background of image using following MATLAB algorithms:

```
% Take small portion
small_portion1=border(Top:Bottom,Left:Right);

% Remove dark Points.
darkValue=150;
noDark =
imextendedmax(im_portion1,darkValue);
```

Figure 8 : MATLAB Algorithm

The following figure shows the result after the code is applied.



Figure 9: Foreground Image

Next, all the small structures and non-disk shaped structures are removed from the image using following code:

```
% Remove lane markings and other non-disk shaped
structures.
sedisk=strel('disk',35);
noSmallStructures = imopen(noDark, sedisk);

% Remove small structures.
noSmallStructures1 = bwareaopen(noSmallStructures,
30);
noSmallStructures2= bwlabel(noSmallStructures1);
p=mean(noSmallStructures2);
q=mean(p);
```

Figure 10: MATLAB Algorithm

The result of previous code is shown in figure below.



Figure 11: Final Image

Finally, vehicle can be count by comparing the mean value of previous image with certain define value. If the value is not same, it means one vehicle had pass by the camera. The following figure shows the final result of vehicle counting.

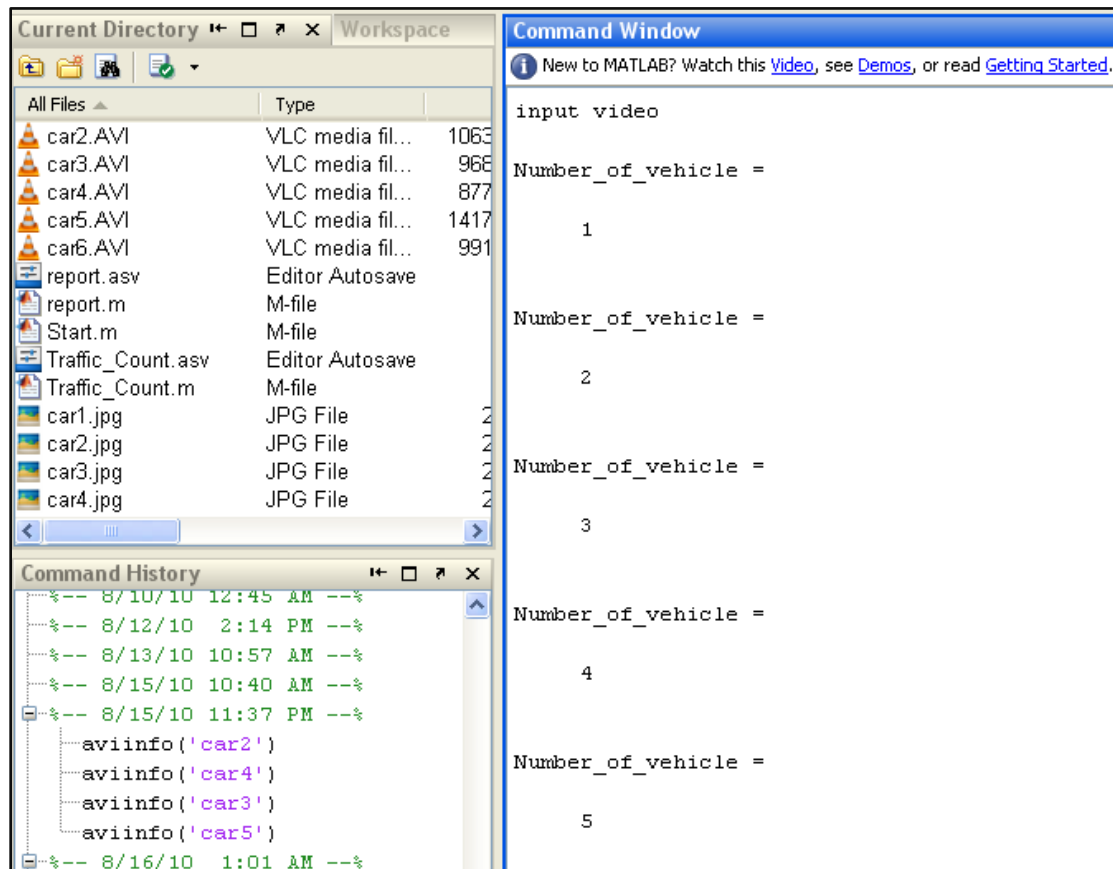


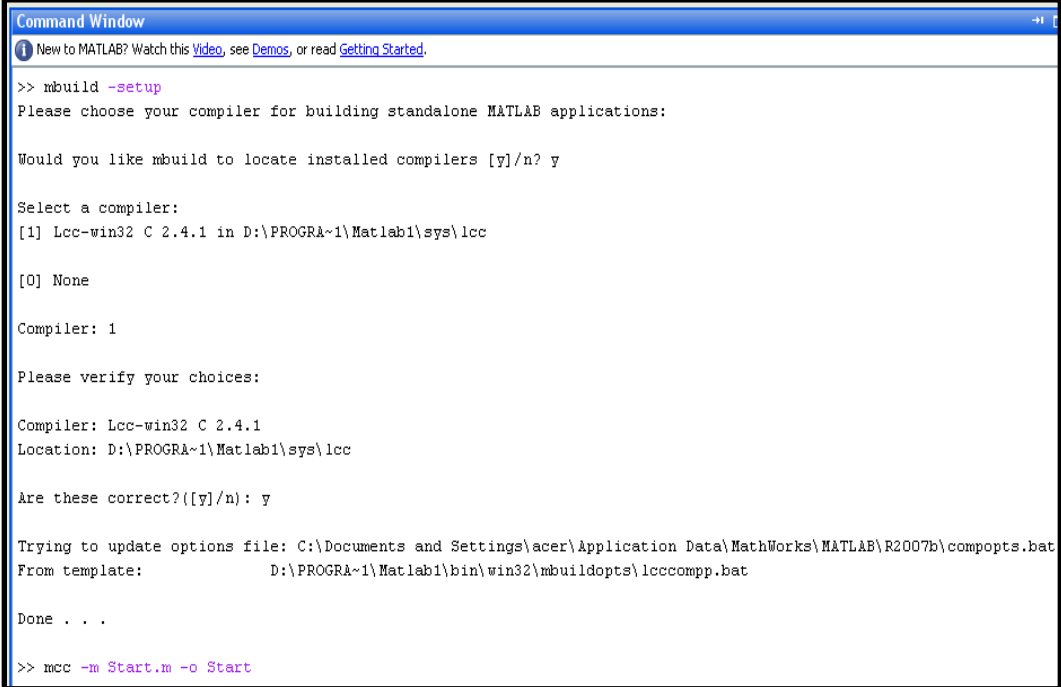
Figure 12: Result in MATLAB

4.3 Standalone executable file

MATLAB is high performance interactive system that allows technical computation, analysis and program development. Due to the fact that MATLAB is an interpreted language, M-files execute slower than compiled programs written in other languages, such as C, C++, and Fortran. Furthermore, M-files require the presence of MATLAB to run. Thus, standalone executable file is needed to show the work on computer that has no MATLAB environment in it.

The first thing need to do is configure the compiler. The Compiler supports the following ANSI C and C++ PC Compilers:

- Lcc C version 2.4 (included with MATLAB)
- Watcom C/C++ versions 10.6 and 11.0
- Borland C++ versions 5.0, 5.2, 5.3, 5.4, and 5.5
- Microsoft Visual C++ (MSVC) versions 5.0 and 6.0



```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>> mbuild -setup
Please choose your compiler for building standalone MATLAB applications:

Would you like mbuild to locate installed compilers [y]/n? y

Select a compiler:
[1] Lcc-win32 C 2.4.1 in D:\PROGRA~1\Matlab1\sys\lcc
[0] None

Compiler: 1

Please verify your choices:

Compiler: Lcc-win32 C 2.4.1
Location: D:\PROGRA~1\Matlab1\sys\lcc

Are these correct?([y]/n): y

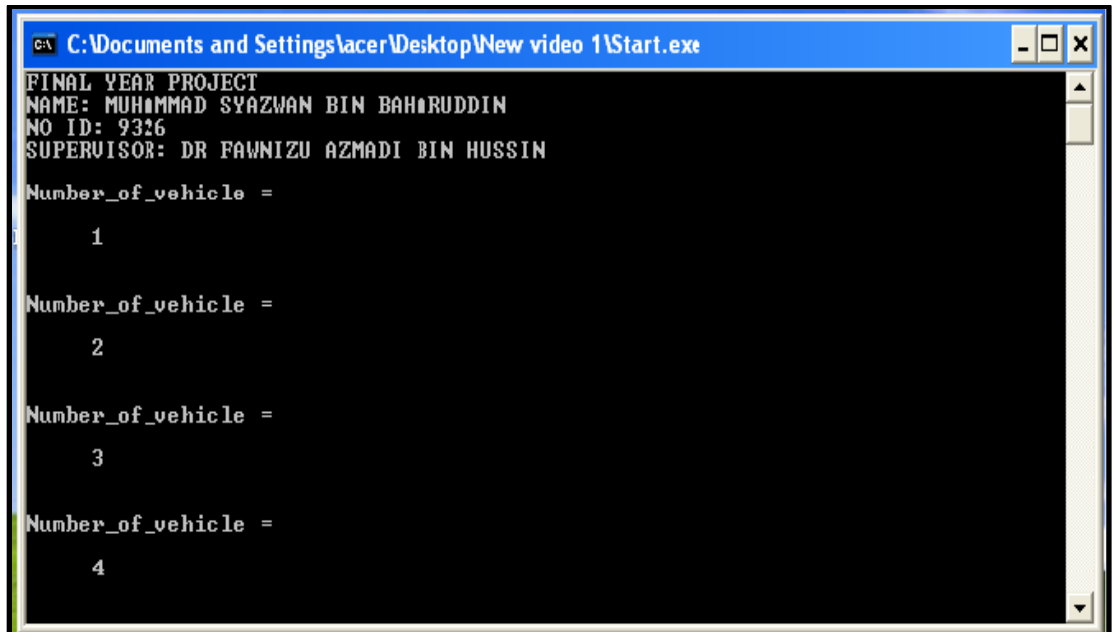
Trying to update options file: C:\Documents and Settings\acer\Application Data\MathWorks\MATLAB\R2007b\compopts.bat
From template:          D:\PROGRA~1\Matlab1\bin\win32\mbuildopts\lcccomp.bat

Done . . .

>> mcc -m Start.m -o Start
```

Figure 13: Compiler Configuration

Once the compiler configuration is done, then MATLAB code will be compiled. Interpreter will create the necessary linker files for the MAT files. After the process is completed, standalone executable is created for the compiler. Figure 14 below shows the result after the standalone file is executed.



```
C:\Documents and Settings\acer\Desktop\New video 1\Start.exe
FINAL YEAR PROJECT
NAME: MUHAMMAD SYAZWAN BIN BAHARUDDIN
NO ID: 9326
SUPERVISOR: DR FAWNIZU AZMADI BIN HUSSIN
Number_of_vehicle =
    1
Number_of_vehicle =
    2
Number_of_vehicle =
    3
Number_of_vehicle =
    4
```

Figure 14: Result in Standalone Executable File

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The author successfully completes a simulation-based vehicle tracking and counting system using video and image processing. The system is expected to be capable of detecting, counting and classifying vehicles while requiring minimal human operators. In addition to the vehicle category, the scope of the project can be extended in future to provide location and velocity information for each vehicle.

5.2 Recommendation

Although the models mentioned above have the similar function, but the current model does not fully meet the requirements of this project. For example, the operation of this model only can detect and count cars while the purpose of this project is to count and classify type of vehicles. So, the succeeding works will focus on how to integrate the current model with the objective of this project.

REFERENCES

- [1] Kilger, M. Video-Based Traffic Monitoring. International Conference on Image Processing and its Applications, 7-9 April 1992, Maastricht, Netherlands, pp. 89-92.
- [2] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and Classification of Vehicles," in IEEE Transactions on Intelligent Transportation Systems, 3(1):37-47, 2002.
- [3] L. Tsai, J. Hseih and K. Fan, IEEE Transactions on Image Processing, 3 March 2007, pp. 850-864
- [4] Liu Anan, Yang Zhaoxuan, and Li Jintao, "Video Vehicle Detection Algorithm based on Virtual-Line Group" in Transactions on Intelligent Transportation Systems, 7(2):175-187, 2006.
- [5] K.P.Karmann and A.von Brandt, "Moving object recognition using an adaptive background memory," in Proc. Time-Varying Image Processing and Moving Object Recognition, vol.2, V.Capellini, Ed., 1990.
- [6] Z. Kim and M.Y. Siyal, "A new segmentation technique for classification of moving vehicles," in Proceedings of Vehicular Technology Conference, 1:323-326, 2000.
- [7] Saad M. Al-Garni and Adel A. Abdenmour, "Moving vehicles detection using automatic background extraction," in World Academy of Science, Engineering and Technology, 24:323-326, 2006.
- [8] Vibha L, Chetana Hegde, P Deepa Shenoy, Venugopal K R, L M Patnaik, "Dynamic Object Detection, Tracking and Counting in Video Streams for Multimedia Mining", in IAENG International Journal of Computer Science, 35:3, 2008

APPENDICES

APPENDIX A

GANTT CHART

Final Year Project Planning

No	Task	2010											
		Jan	Feb	March	April	May	June	July	Aug	Sept	Oct	Nov	Dec
Develop Algorithms using MATLAB													
1	Familiarization with image processing using MATLAB <ul style="list-style-type: none">Learn how to access image dataOrganize image data for visualization												
2	image&Video processing with Simulink <ul style="list-style-type: none">Constructing a model using block diagram interfaceAcquiring live video into the model												
3	Familiarization with Wavelet Toolbox <ul style="list-style-type: none">Compression and denoising of signals and imagesSignal classification and multisignal analysis												
4	Use Simulink for video processing <ul style="list-style-type: none">Learn how to prototype ideasExplore “what-if” scenarios												
5	Image Acquisition&Processing using MATLAB <ul style="list-style-type: none">Connect to & configure image acquisition hardware from MATLAB												
Develop prototype or physical product													
1	Comparison study between image acquisition hardware and Graphical user interface <ul style="list-style-type: none">Study on the algorithm and software that need to be used in both approaches												
2	Develop embedded system												
3	Add extra feature into the system												

APPENDIX B

CAMERA SPECIFICATION

BASIC SPECIFICATIONS	
Manufacturer	BenQ
Model	DC E800
Camera Type	Compact
Dimensions	3.65 x 2.27 x 0.82 in (93 x 58 x 21 mm)
Weight	4.9 oz (140 grams)
Waterproof	No
Announce Time	April 25, 2008
SENSOR	
Total Pixels	8.1 Megapixels
Optical Sensor Type	CCD
Light Sensitivity	50 - 1600 ISO, Auto ISO
Max Resolution	3264 x 2448
FEATURES	
Digital Zoom	5x
Video Recording	Yes
Manual setting of shutter speed and aperture	No
Face Detection	Yes
Features	Tripod mount
White Balance	Auto, Manual, Presets
Max Shutter Speed	8 s
Min Shutter Speed	1/2000 s
Digital Video Codec	MJPEG
Built-in Microphone	Yes
Focal Length Equivalent to 35mm Camera	34 - 102 mm
Macro Shooting	Yes
Minimal Focus Range	0.1 m
LENS SYSTEM	
Optical Zoom	3x
Lens Aperture Range	F2.9 - F5.2
Image Aspect Ratio	4:3, 3:2, 16:9
Video Max Resolution	640x480
CAMERA FLASH	
Built-in Flash	Yes
Red Eye Reduction	Yes
DISPLAY	
Display Form Factor	Built-in
Display Pixels	230000
Display Size	2.7 inch
STORAGE	
Supported Flash Memory	SD, SDHC
Built-in Memory	10 Mb

BATTERY	
Battery Capacity	700 mAh
ADDITIONALS FEATURES	
Self Timer	Yes
Self Timer Delay	2, 10 s
Cables Included	USB cable, A/V cable
Included Accessories	Hand strap, Battery charger

APPENDIX C

MATLAB ALGORITHM

```
function d = Traffic_Count(video)
```

```
    Top=250;
```

```
    Bottom=350;
```

```
    Left=1;
```

```
    Right=600;
```

```
Number_of_vehicle=0;
```

```
    c=0;
```

```
    b=0;
```

```
for i=1:5:(length(video)-5)
```

```
    imwrite(video{i},'car1.jpg')
```

```
    imwrite(video{i+1},'car2.jpg')
```

```
    imwrite(video{i+2},'car3.jpg')
```

```
    imwrite(video{i+4},'car4.jpg')
```

```
    I1=imread('car1.jpg');
```

```
    I2=imread('car2.jpg');
```

```
    I3=imread('car3.jpg');
```

```
    I4=imread('car4.jpg');
```

```
    Ia=(I1+I2+I3)./3;
```

```
    I5 = imsubtract(I4,Ia);
```

```
    I6=rgb2gray(I5);
```

```
% Take small portion
```

```
im_portion1=I4(Top:Bottom,Left:Right);
```

```
% Remove dark Points.
```

```

darkValue=200;
noDark = imextendedmax(im_portion1, darkValue);

% Remove lane markings and other non-disk shaped structures.
sedisk=strel('disk',25);
noSmallStructures = imopen(noDark, sedisk);

% Remove small structures.
noSmallStructures1 = bwareaopen(noSmallStructures, 30);
noSmallStructures2= bwlabel(noSmallStructures1);
p=mean(noSmallStructures2);
q=mean(p);

%Display section
figure(1);
imshow(I4)
if q~=1
    %disp('vehicle')
    b=i;
else
    %disp('No vehicle')
    c=i;
end
d=c-b;
if d==5
    Number_of_vehicle=Number_of_vehicle+1
end
end

```