

# **Event-Driven Traffic Ticketing System**

JIA WANG

A thesis submitted to Auckland University of Technology  
in partial fulfillment of the requirement for the degree of  
Master of Computer and Information Science (MCIS)

2016

School of Engineering, Computer and Mathematical Sciences

# Abstract

This thesis presents a new schema to implement an event-driven traffic ticketing system. The entire system consists of four modules, namely, (1) event detection module, (2) plate number recognition module to recognize the extracted plate number, (3) database management module to execute information retrieval and (4) traffic ticket transmission module to deliver traffic tickets. Due to the importance of plate number recognition module in entire system, this thesis focuses on plate number localization and recognition. A novel plate number localization algorithm called Secondary Positioning (SP) method is proposed in this thesis. The rough position of plate number is identified at first stage by searching the red light region in HSV colour space, and then the accurate position of plate number is localized in the second stage by finding out the vertical edge of plate number. Template matching implemented for recognizing individual plate character incorporates correction coefficient calculated between templates and testing images. The test datasets include two subsets, namely, (1) 120 images for plate number localization and (2) 80 images for plate number recognition. The success rates obtained from plate number localization and recognition are approximately 75% and 70% respectively.

**Keywords:** plate number recognition, template matching, event database, traffic ticketing system

# Table of Contents

|   |      |
|---|------|
| Abstract .....  | I    |
| List of Figures .....                                   | IV   |
| List of Tables .....                                    | VI   |
| List of Algorithms .....                                | VII  |
| Attestation of Authorship .....                         | VIII |
| Acknowledgments .....                                   | IX   |
| Chapter 1 Introduction .....                            | 1    |
| 1.1 Background and Motivation .....                     | 2    |
| 1.2 Objectives .....                                    | 4    |
| 1.3 Structure of This Thesis .....                      | 6    |
| Chapter 2 Literature Review .....                       | 7    |
| 2.1 Event .....   | 8    |
| 2.2 Moving Object Detection .....                       | 11   |
| 2.3 Moving Object Tracking .....                        | 13   |
| 2.4 Plate Number Recognition .....                      | 15   |
| 2.4.1 The Challenges of Recognizing Plate Numbers ..... | 16   |
| 2.4.2 The Plate Number Format .....                     | 18   |
| 2.4.3 Plate Number Localization .....                   | 19   |
| 2.4.4 Plate Number Recognition Methods .....            | 27   |
| Chapter 3 Methodology .....                             | 33   |
| 3.1 The Related Work .....                              | 34   |
| 3.2 The Research Question .....                         | 35   |
| 3.3 Data Acquisition .....                              | 36   |
| 3.3.1 Data Collection for Experiments .....             | 36   |
| 3.3.2 Templates and Datasets for ANN .....              | 37   |
| 3.4 Evaluation Methods .....                            | 37   |
| 3.5 The Novelty of This Thesis .....                    | 38   |
| Chapter 4 Design and Implementation .....               | 40   |
| 4.1 Event Detection .....                               | 41   |
| 4.1.1 Red Light Detection .....                         | 41   |
| 4.1.2 Plate Area Detection and Tracking .....           | 43   |
| 4.2 Plate Number Recognition .....                      | 46   |
| 4.2.1 Plate Number Localization .....                   | 46   |
| 4.2.2 Character Segmentation .....                      | 54   |
| 4.2.3 Character Recognition .....                       | 56   |
| 4.3 The Database Design .....                           | 59   |
| 4.4 Information Notification .....                      | 62   |
| Chapter 5 Experimental Results and Discussion .....     | 64   |
| 5.1 Introduction .....                                  | 65   |
| 5.2 Event Detection Module .....                        | 65   |
| 5.2.1 Event Detection .....                             | 65   |

|   |    |
|---|----|
| 5.2.2 Plate Tracking.....                 | 67 |
| 5.3 Plate Number Recognition Module.....  | 70 |
| 5.3.1 Plate Number Localization.....      | 70 |
| 5.3.2 Character Recognition.....          | 71 |
| 5.4 Discussions.....                      | 72 |
| Chapter 6 Conclusion and Future Work..... | 73 |
| 6.1 Conclusion.....                       | 74 |
| 6.2 Limitations and Future Work.....      | 75 |
| References.....                           | 77 |

# List of Figures

|   |    |
|---|----|
| Figure 1.1 A Generic Framework of Event-Driven Traffic Ticketing System.....      | 5  |
| Figure 2.1 The Basic Five Components in a LPR System.....                         | 16 |
| Figure 2.2 The Size of a Common NZ Plate Number.....                              | 18 |
| Figure 2.3 Example of Horizontal and Vertical Gradient of an Image.....           | 22 |
| Figure 2.4 Results of Vertical Edge Detection.....                                | 24 |
| Figure 2.5 Spatial Variance of Two Scan Lines.....                                | 25 |
| Figure 2.6 The Detected Plate Number Area by MorphologyOperations.....            | 27 |
| Figure 2.7 The Structure of a Neuron.....   | 29 |
| Figure 2.8 The Basic Structure of BP-Neural Network.....                          | 31 |
| Figure 2.9 The Sample of Same Number of White Pixels in Character Z and N.....    | 32 |
| Figure 4.1 The Implementations of Locating ROI and Extracting Feature Points..... | 45 |
| Figure 4.2 The Tracking Results of a Number Plate.....                            | 45 |
| Figure 4.3 The Flowchart of How to Search the Candidates.....                     | 47 |
| Figure 4.4 The Original Image and HSV Colour Space of a Red Car.....              | 48 |
| Figure 4.5 The Centroid of Detected Red-pixel Region.....                         | 49 |
| Figure 4.6 The Results of Plate Number of The First Scan.....                     | 50 |
| Figure 4.7 The Flow Diagram of The Second Scan.....                               | 50 |
| Figure 4.8 Results of The Second Scan of Plate Number.....                        | 54 |
| Figure 4.9 Results of Example 1.....  | 56 |
| Figure 4.10 Results of Example 2.....   | 56 |
| Figure 4.11 Results of Character Reorganization.....                              | 57 |
| Figure 4.12 The Horizontal Projection Line of Letter ‘H’.....                     | 58 |
| Figure 4.13 Results of Information Notification.....                              | 63 |
| Figure 5.1 Results of Straight Line Detection.....                                | 66 |
| Figure 5.2 Results of Stop Line Detection and Event Detection.....                | 66 |
| Figure 5.3 An Example of Plate Tracking.....                                      | 68 |
| Figure 5.4 The Results of Plate Tracking Removing the Background.....             | 69 |

|   |    |
|---|----|
| Figure 5.5 A Plate Number Goes Through the Red Traffic Light..... | 69 |
| Figure 5.6 Samples from Website.....                              | 70 |

# List of Tables

|   |    |
|---|----|
| Table 2.1 The Sample of Event Vocabulary Set.....                     | 10 |
| Table 2.2 The Comparison of Pros and Cons of Above Three Methods..... | 13 |
| Table 2.3 Different Types of NZ Plate Number.....                     | 18 |
| Table 4.1 Categories of Plate Number Characters.....                  | 58 |
| Table 4.2 Driver Table in the Database.....                           | 60 |
| Table 4.3 Vehicle Table in the Database.....                          | 61 |
| Table 4.4 Violation Event Table in the Database.....                  | 61 |
| Table 5.1 Results of Foreground Detection.....                        | 67 |
| Table 5.2 Results of Plate Number Tracking.....                       | 67 |
| Table 5.3 Results for Edge Detection and the Proposed Method.....     | 70 |
| Table 5.4 Results from Character Recognition.....                     | 71 |

# List of Algorithms

|   |    |
|---|----|
| Algorithm 4.1 Red Light Detection.....      | 41 |
| Algorithm 4.2 Plate Number First Scan.....  | 46 |
| Algorithm 4.3 Plate Number Second Scan..... | 51 |
| Algorithm 4.4 Character Segmentation.....   | 54 |



## **Attestation of Authorship**

“I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.”

Signature Jia Wang

Date 19/05/2016

# Acknowledgments

I would like to express my deepest gratitude to my thesis supervisor Dr. Wei Qi Yan. This work will not be completed without his guidance and encouragement. He always guides me to pursue the real knowledge in the world. I also want to thank my secondary supervisor Dr. Boris Bacic for his advices. They both help me a lot in this study journey.

Finally, I would like to thank my parents and family. Thank their quiet support without any regrets in this year. I love them forever.

# Chapter 1

## Introduction

*In the first chapter of the thesis, we start introducing the background and motivation of the thesis. A novel video-based event detection application, event-driven traffic ticketing system, is proposed in this thesis. After presenting the overall relevant background information, the objectives will be discussed. Finally, the structure of this thesis is detailed.*

## 1.1 Background and Motivation

The widespread use of digital cameras and smart camera phones has verified the affection of computer vision in daily life. The photographs and videos not only record the personal memorable experiences and scenery but also can be used as the evidence in civil dispute or other controversial cases. As the old English saying “seeing is believing”, a picture or a video can provide more information and is worth a thousand words. Video cameras are ubiquitous in our society. They are installed in everywhere such as parking lots, airports, banks, traffic intersections, military control zones, motorways, etc. for surveillance, working for 24 hours a day 7 days a week and 365 days a year (Aköz & Karşligil, 2010; Bunyak, 2005; Castello, Coelho, Del Ninno, Ottaviani, & Zanini, 1999; Kumar, Vaidehi, & Chandralekha, 2013; Rashid, Musa, Rahman, Farahana, & Farhana, 2012). The aim of using surveillance is to capture events of interest in the scene. It is tedious and time consuming for human operators to extract an event from thousands of video streams. Therefore, it would be beneficial to find a way to detect the events from video streams automatically.

The field of event detection in surveillance video has drawn a great deal of attention from researchers ranging from engineering, manufacturing, computer science, machine learning and artificial intelligent and so forth. Traffic control and monitoring using event detection from video are practical issues, because the significant profit and commercial value of event detection applications. For example, it is useful to reduce congestion and help construction of plan by analyzing the video of traffic patterns on a highway. Small crimes or terrorist attacks could be prevented by pinpointing anomalies from surveillance (Xie, Sundaram, & Campbell, 2008). Traffic control and vehicle owner identification as important branch of utilizing event detection from surveillance application is popular research topic. To automatically identify the vehicle owner who violates traffic rules such as running through red light, driving too fast, making illegal U turn and so forth is one of major event driven applications. In those applications, plate

number recognition module affects the whole event detection system accuracy. After the successfully identifying the correct plate number, extra functionality modules such as violator database management, fine management or fine notification module would be implemented. The most important advantage of event driven system is that all the work would be done automatically.

There are numerous algorithms or methods used in video event detection. The algorithms used in video event detection mostly derive from statistical pattern categorization and recognition, machine learning and artificial intelligence (Vallejo, Albusac, Jimenez, Gonzalez, & Moreno, 2009; Yan & Weir, 2011). For example, the Hidden Markov Model (HMM) as one statistical pattern has been applied to visual event detection. HMM is suitable for classification of discrete state sequences (Siskind & Morris, 1996). Specifically, HMM is a double Markov Random Process, which consists of two sets of states, hidden states and observable states, and three sets of probabilities. The three sets of probabilities are  $\Pi$  vector, state transition matrix and confusion matrix. Support Vector Machine (SVM) as another statistical method or machine learning classification algorithm has been used to detect events in video (Burges, 1998). Rule-based detection methods can also be used in video event detection. Li, Chen and Wang (2009) proposed a rule-based detection method which generates the video events positions by using the combination of structural analysis and semantic tagging. The major idea of the method is that the event will follow a slow motion shot in sport video. However, this method is too restricted to specified video which must contain different shot types of video such as close-up shot, score shot and slow motion shot. If the input video changed, this method will lose efficacy. A rule-based solution is presented to detect uncertain events (Wasserkrug, Gal, Etzion, and Turchin (2012). Due to the complexity and timeliness of traffic video and the difficulty of finding correct rules and corresponding probabilities, the rule-based method may not be suitable for this research project. Trajectory-based detection method can also be utilized in video event detection (Piciarelli, Micheloni, & Foresti, 2008). A trajectory is a record of an object motion, which means it records the object position from emerging to disappearing in a

scene (Zhang, Huang, Tan, & Wang, 2007). In most scenarios of video event detection system, events can be represented by object trajectories (Jiang, 2012; Morris & Trivedi, 2008). For instance, a trajectory can be represented by using a sequence of 2-D coordinates of the object centre at every frame. The basic principle of trajectory-based method is using a statistical model of target trajectories to distinct which event is abnormal (Piciarelli et al., 2008; Zhou, Yan, & Huang, 2007; Zhu et al., 2007). A novel dynamic hierarchical clustering (DHC) method is presented to detect abnormal trajectory from surveillance video (Jiang, 2012). The thought of their method is to find out any trajectories that cannot be explained by normal trajectories models which are modelled by HMM.

## **1.2 Objectives**

Firstly, this thesis introduces computer vision techniques related to video event detection applications and plate number recognition systems. The supporting principles of computer vision techniques and the theories are linked to the developed methods that are demonstrated and evaluated in this thesis.

Secondly, this thesis presents a conceptual framework developed for traffic ticketing system, and detecting traffic rule violations related to vehicles passing through the red light at the junction. The overall objective of this thesis is to develop an event-driven traffic ticketing system which implements four key functions, including, (1) event detection from video frames, (2) plate number recognition, (3) database management, (4) information notification. Figure 1.1 shows a conceptual framework for the developed event-driven traffic ticketing system.

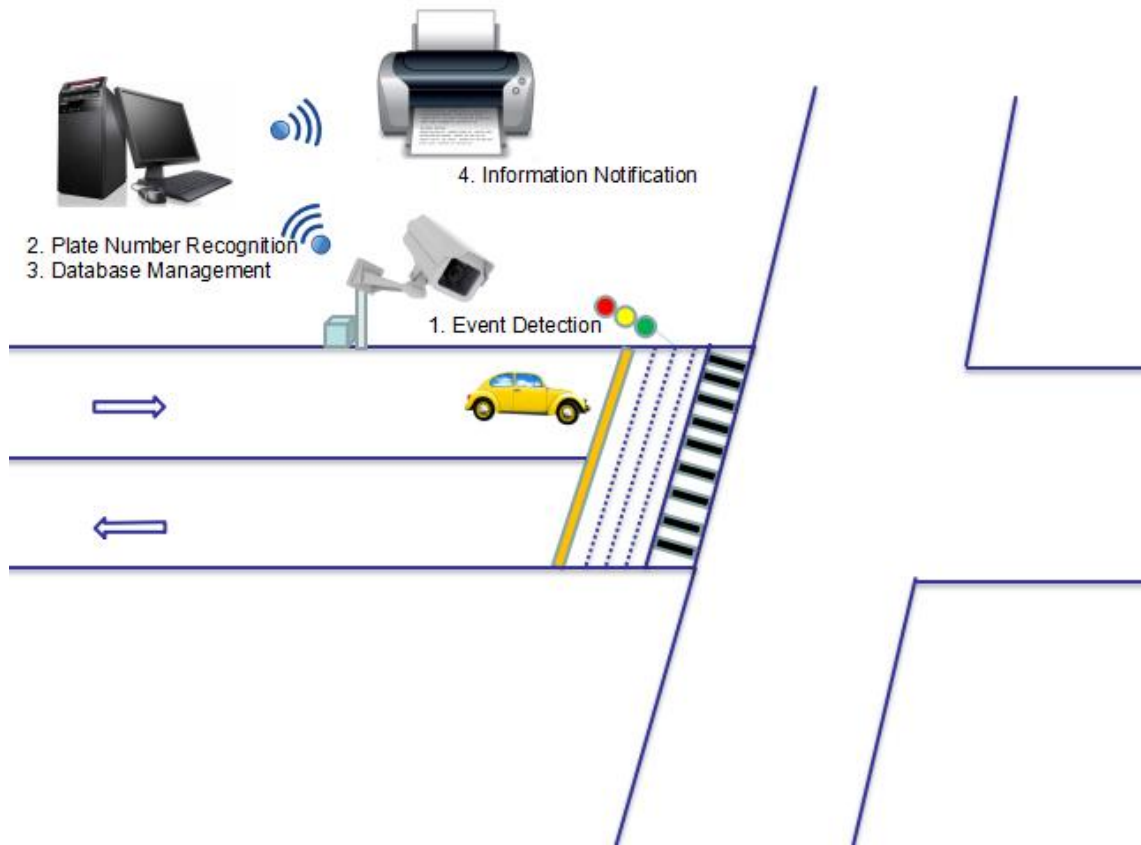


Fig. 1.1 A Generic Framework of Event-Driven Traffic Ticketing System

The steps of the system can be defined as follows. First of all, if the event that vehicles run through the red light occurs, event detection module will be triggered. As a result, an image with accident vehicle will be captured. After that, digital image processing techniques such as image enhancement, image segmentation, and colour space conversation will be utilized to handle the original captured image. Then, the plate number recognition module will identify the number automatically. Eventually, the system will automatically retrieve the database in accordance with the recognized plate number and send an email that alerts the owner of the vehicle to pay the traffic fine.

Finally, the comparison of plate number recognition algorithms will be carried out in this thesis. Due to the real intersection traffic video deficiency, this thesis focuses on the plate number recognition module. Each method involved with plate number recognition will be discussed and compared.

## 1.3 Structure of This Thesis

In Chapter 2, we discuss the literature related to the two main categories in this thesis: event detection in surveillance video and plate number recognition. Moving object detection and tracking in video are also emphasized in this chapter.

In Chapter 3, we explain the research methodology of this thesis. The research question as well as the main issues will be also proposed in this chapter. The potential solutions and answers to the problems will be presented. Finally, the experimental layout and setting, data set and implementation are described.

In Chapter 4, we present the algorithms and methods developed for the scope of this thesis. The experimental results and the outcomes are compared with related existing work.

In Chapter 5, we analyze our experimental results and outcomes. The solutions to research objectives are tested and the final experiment results are discussed. Finally, the limitations of this research and future work are also presented in Chapter 6.



# **Chapter 2**

## **Literature**

### **Review**

*The main objectives of this chapter are to explain and evaluate the research studies that focus on event detection in surveillance, moving object detection and tracking, and vehicles plate number recognition. The-state-of-the-art of moving object detection and license plate recognition methods is summarized in this Chapter. In conclusions, the considered techniques and algorithm candidates are evaluated for achieving the objectives of this thesis.*

## 2.1 Event

The advanced technologies and massive social media platforms enable people to observe and record their routine lives. As a result, the huge amount of sensory data has been captured from physical sensors and social sensors. The notion of a physical sensor refers to electrical devices such as cameras, mobile phones, RFID tags and so on. The social sensors refer mainly to social networking sites which consist of user-generated event information like text, image video, etc.(Sakaki, Okazaki, & Matsuo, 2010; Wang & Kankanhalli, 2015; Weng & Lee, 2011). Our world is being emerged by these physical and social sensors. These sensory data especially the multimedia data facilitate video event detection in real world. The task of video event detection is to identify and localize specific spatial-temporal patterns in video.

Nowadays, the word “event” frequently appears in daily lives. These events in general usually refer to something happens at a given time and place (Yan & Weir, 2011). Event is an elementary concept for both human and multimedia applications. Simply, an event can be considered as a story that involves with different objects or human. Event can be also considered as phenomena or circumstances that happen at a particular place and a certain time, which can be identified without ambiguities (Castellanos, Kalva, Marques, & Furht, 2010). An event processes in a duration, happens in a specific place and involve certain change of state (Xie, Sundaram, and Campbell, 2008). Hence, the most important parameters to describe an event should consist of time and location. Events that take place at different times or place are considered to be different events (Yan and Weir, 2011). For example, two events can be extracted from the sentence “a man gets up at the same time in Monday and Tuesday morning”. The action is the same “a man gets up at the same time in the morning”, but the action carries out in different time “Monday and Tuesday”.

Video-based event detection takes additional care on abnormal events. The

definition of event in dictionary is that “anything that happens, especially something important or unusual”. These abnormal events that have a high causality but a low frequency usually attract more attention from human (Lee & Kwon, 2012). For example, a car passing an intersection is a very common scene. The cameras in the roadside are not interested in normal driving vehicles. However, the cameras are interested in the abnormal driving cars such as running through the red light, not giving way to pedestrians, illegal turning round etc. Hence, the events that people interested and desired are special important and unusual. The tasks of finding events in surveillance are to identify abnormal events happened in specific frame in a video and a string or text message stored in database that can be used to alert operator and even can be as evidence provided in dispute court.

Event detection plays a vital role in automatic human activity understanding, which has a similar task with object detection. They both have a goal to identify and localize specified spatio-temporal patterns in video (Ke, Sukthankar, & Hebert, 2010). The event in video can be treated as long-term temporal object which usually spans over several frames. Temporal events can be divided into three classes and each class could be modelled and recognized by separate approaches. The three categories are comprised as: temporal textures, activities and motion events (Polana & Nelson, 1994; Zelnik-Manor & Irani, 2001). Specifically, motion events denote independent actions that do not repeat either in time or in space such as people smiling. Activities refer to actions that are temporally periodic but spatially restricted such as car moving. A class of temporal textures represents the events are persistent in spatial and temporal extent such as sunrise and sunset. These aforementioned three types of events are treated as temporal events (Zelnik-Manor & Irani, 2006). In this thesis, all the events such as vehicle approaching, speeding down and vehicle running through the red light are taken into account as temporal events.

There are various ways to describe an event in video. How to depict an event that belongs to a specific type of event is a semantic description problem. The task of event

semantic description is to build the event semantics that combining visual features, natural language verbs and symbols (Liu, Hu, Liu, & Aggarwal, 2013). In order to define the event, a transformed triplet which derives from linguistic theory of “subject + verb (+object)” structure is proposed (Liu, Hu, Liu & Aggarwal, 2013). By using this transfer triplet, an event can be described as a linguistic sentence. They define a vocabulary sets to store the words to describe activities. For example, there are some names such as train, door, luggage and human in “Subject Set”, and some verbs such as stand, walk, turn left, turn right, stop and so forth in “Verb Set”. When an event is detected from video scene, the corresponding word from each set would be extracted to form a “subject - verb - object” sentence. The following Table 2.1 shows an example of vocabulary sets. To form a “subject - verb - object” sentence described the event that people are running, the words “Human” and “Run” are extracted automatically from the first and second column, respectively.

Table 2.1 The Sample of Event Vocabulary Set

| <b>Subject Set</b> | <b>Verb Set</b> |
|--------------------|-----------------|
| Human              | Stand           |
| Train              | Run             |
| Door               | Shut down       |
| Luggage            | Leave           |
| Animal             | Eat             |
| ...                | ...             |

Six aspects called 5W1H, which is derived from journalism, is proposed to describe event in multimedia stream (Xie et al. (2008). 5W1H is abbreviation of six interrogatives, namely, WHO, WHEN, WHERE, WHAT, WHY and HOW. These six words could describe and support our understanding of the real-world event specifically. By using these six interrogatives, the news happened in video can be translated to events described in paper work.

Basically, event detection and object detection have similarity. According to Zelnik-Manor and Irani (2006), events known as temporal objects are usually characterized by numerous temporal dimensions. Similarly, spatial objects are characterized by numerous spatial dimensions. As shown in Figure 1.1, the task of event detection is to detect where the car appears, when the car leaves and what the car has done in the video frames. The task of object detection is to detect whether the background has new object existing. To understand the delicate difference between these two concepts, event description should be described in this thesis. Event description is not only the basis for solving an event detection problem and an event-drive application, but also a standard to structure the semantics of real-world event representation (Xie et al., 2008).

## **2.2 Moving Object Detection**

Moving object detection is a major component of event detection system. It is the first step of event detection process due to the target should be detected before the following processes such as tracking, recognition, classification and activity analysis (Bunyak, 2005; Papageorgiou, Oren, & Poggio, 1998). Moving object detection in video is a procedure to verify the presence of an object in image sequence and locate it for further recognition (Papageorgiou et al., 1998; Shaikh, Saeed, & Chaki, 2014). The primary objective of moving object detection is to analyze a video sequence to detect moving object with respect to background scene (Elhabian, El-Sayed, & Ahmed, 2008; Shaikh et al., 2014). Due to a film or movie is a collection of static images or frames, all image processing operations are able to be used on individual frame extracted from films or movies. Traditional approaches for moving object detection are broadly categorized into three classes: Background subtraction, temporal differencing and optical flow (Kulchandani & Dangarwala, 2015; Dedeoglu, 2004; Wei, Ji, & Wang, 2006).

Background subtraction aims to detect moving regions by using current frame to subtract a reference background frame pixel by pixel (Horprasert, Harwood, & Davis,

1999; Jimenez, 2010; Piccardi, 2004). The results where pixel value is greater than a predefined threshold will be detected as the foreground. The formula of Background subtraction is:

$$|f(x, y) - b(x, y)| > T \quad (2.1)$$

where  $f(x, y)$  and  $b(x, y)$  are the pixel values at same location  $(x, y)$  in same scene, respectively,  $T$  is the predefined threshold value. If the subtraction between pixels in current frame  $f(x, y)$  and background frame  $b(x, y)$  is above  $T$ . Then, the result is the detected moving object region which is also called foreground.

Two important aspects influence the performance of background subtraction. The first aspect is to create and update a proper and reference background image (known as background modelling) (Stauffer & Grimson, 1999). The second aspect is the subtraction between the current image and the background image. It has a significant effect on the performance of traditional background subtraction method when background is dynamic and illumination changes frequently (Kulchandani & Dangarwala, 2015).

Temporal differencing compares the pixel-by-pixel difference between two or three consecutive frames in a video sequence to find out the moving target (Lan, Guo, Liu, Sun, Aibibu & Ran, 2013; Lipton, Fujiyoshi, & Patil, 1998). The method is quite similar with Background Subtraction. The only difference is the subtractor (background image) is replaced by previous image. The formula of temporal difference is:

$$|f_i(x, y) - f_{i-1}(x, y)| > T \quad (2.2)$$

Where,  $f_i(x, y)$  is the pixel value of frame at time  $i$  and  $f_{i-1}(x, y)$  is the previous pixel value of frame at time  $i-1$ .  $T$  is the predefined threshold value.

Optical flow method (Barron, Fleet, & Beauchemin, 1994; Horn & Schunck, 1981) attempts to detect the object by calculating the vectors assigned to each pixel in the image. Each pixel in the frame will be set a vector value, which has a direction attribute. When the object appears at the footage, the vector field surrounding the moving objects

will indicate the difference. Then, the moving object can be detected by calculating the difference. If no object outcrops on the scenery, the all vector filed in the image will be smooth.

The comparison of pros and cons of above three methods were summarized in Table 2.2.

Table 2.2 The Comparison of Pros and Cons of Above Three Methods

|      | <b>Background Subtraction</b>                             | <b>Temporal Differencing</b>   | <b>Optical Flow</b>   |
|------|---|--|---|
| Pros | Easy execution, high detection accuracy, fast performance | Low computational cost, easy of implementation, insensitive to illumination        | Unresponsive to variable backgrounds, no extra training required                        |
| Cons | Sensitive to complex background, single-purpose.          | Cannot detect slow and fast moving objects, detected object tend to be incomplete. | Low anti-noise capacity; sensitive to illumination intensity, heavy calculation burden. |

## 2.3 Moving Object Tracking

Moving object tracking is another major component of event detection system. The accuracy of moving object tracking affects the accuracy of the overall event detection system largely (Liu et al., 2012). Video event detection can be considered as one important step in video analysis. Events are extracted from videos by analyzing the object tracks to recognize their behaviours. The task of moving object tracking is to assign consistent labels to target moving object in continuous frames of a video. To be more exact, to solve the object tracking problem is to answer the following three questions: (1) which object representation should be chosen for tracking? (2) which

features of the image should be selected? (3) how should the motion and appearance of object be modelled (Yilmaz, Javed, & Shah, 2006). The majority state-of-the-art object representation approaches, feature selection methods and similarity measure algorithms were concluded and summarized in Yilmaz 's method.

Numerous approaches for moving object tracking have been presented (Comaniciu & Meer, 1999). The approaches can be grouped broadly into three categories by the features they use. The three categories are point tracking, kernel tracking and silhouette tracking (Joshi & Thakore, 2012; Vilaplana & Marques, 2008). Exactly, point tracking means objects are represented by points (e.g. the object centroid) and the tracker establishes a correspondence between points in consecutive frames. The widely used methods such as Kalman filters, particle filters and joint probabilistic data association filters belong to this class (Huang & Hong, 2011; Weng, Kuo, & Tu, 2006). Kernel tracking refers to using the object shape and appearance to track the target moving object (Comaniciu, Ramesh, & Meer, 2003). By estimating the motion of the kernel in consecutive frames, target objects are labelled and tracked. Silhouette tracking techniques are used in applications that need defining with precision the object shape. The object is represented by a region or by contours and the tracking estimates the area of support of the object.

Kanade-Lucas-Tomasi (KLT) algorithm is computer vision tracking algorithm which is based on tracking the point features (Karasulu, 2010; Tomasi & Kanade, 1991). The translation motion model is utilized to calculate the residuals of two translated window gradations and seek the residual Sum of Square Difference (SSD) to match the same features in two successive frames. Three premises assumptions should be observed before using KLT algorithm in tracking problem. The three assumptions are: 1) brightness constancy 2) time continuity or small motion and 3) space consistency, similar motion between adjacent points. The tracking process will under the above three assumptions. Specifically, to find out whether connective two frames I and J in same local window  $w$  are the same is equivalent to fulfil the equation,



$$I(x, y, t) = J(x', y', t + \tau) \quad (2.3)$$

where,  $I(x, y, t)$  represent the image  $I$  with space variables  $x, y$  and time variable  $t$  whose values are discrete and suitable bounded. Whereas  $J(x', y', t + \tau)$  refers to the image  $J$  with two space variables at time  $t + \tau$ . The above formula can be defined as pixel at  $(x, y)$  moves to  $(x', y')$  a vector value  $(dx, dy)$ . Then, the pixel at time  $t + \tau$  can be represented as  $(x + dx, y + dy)$ . To solve the problem of matching current frame with its previous frame in window  $w$  is to search the minimum value in following equation,

$$\epsilon(d) = \epsilon(d_x, d_y) = \sum_{x=u_x-w_x}^{u_x+w_x} \sum_{y=u_y-w_y}^{u_y+w_y} (I(x, y) - J(x + d_x, y + d_y))^2 \quad (2.4)$$

Where,  $u$  is the velocity vector of pixel at  $(x, y)$ .

KLT algorithm is used in our thesis project due to the robustness and accuracy of tracking small moving objects. We apply KLT algorithm to track the plate number area in frames. Comparing to the car body, the plate number area is quite smaller. Further, the characters in plate consist of distinguished point features which are easily tracked by KLT tracking algorithm.

## 2.4 Plate Number Recognition

As our objective in this thesis is to design an event-driven traffic ticketing system, plate number recognition module is the most important part of this system. A desired automatic License Plate Recognition system (LPR) uses digital image processing techniques to locate and recognize the characters on the plate number and output the results as a text string or other type data formats that can be easily understood by operators (Parisi, Di Claudio, Lucarelli, & Orlandi, 1998). LPR system has been applied into various applications requiring the automatic control of the presence and identification of a motor vehicle by its plate number such as stolen vehicles observation, automatic electronic toll-collection (ETC), automated parking attendant, traffic ticketing

management, security control and others (Bailey, Irecki, Lim, & Yang, 2002). A plate number recognition system usually consists of five important components which have been identified as being common to all plate number recognition application (Bailey et al., 2002). The five common components are image acquisition, image pre-processing, plate number localization, and image segmentation and character recognition modules (C.-N. E. Anagnostopoulos, Anagnostopoulos, Psoroulas, Loumos, & Kayafas, 2008; Duan, Du, Phuoc, & Hoang, 2005; Shi, Zhao, & Shen, 2005). Fig. 2.1 shows the basic five components in a LPR system.

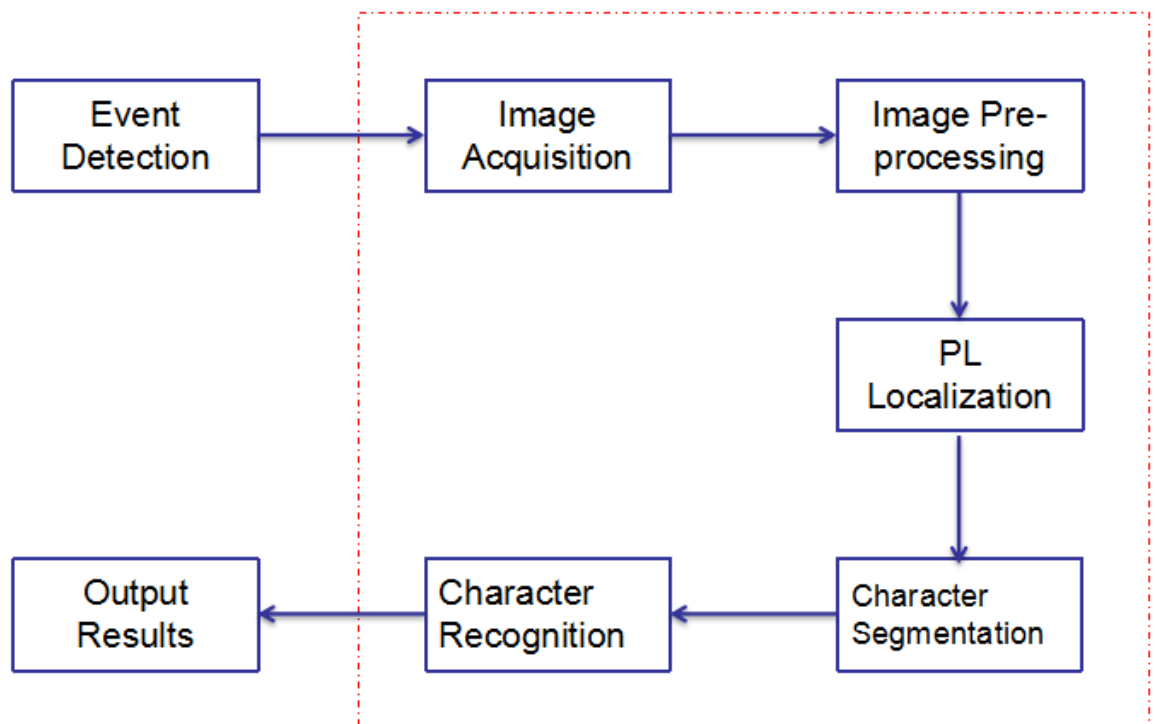


Fig. 2.1 The Basic Five Components in a LPR system

### 2.4.1 The Challenges of Recognizing Plate Numbers

Although the plate number recognition has been researched and studied for a long time, the study of this area is still very active. Researchers are trying to find out the best strategies and methods to reduce the false recognition rate. The false recognition is usually arisen by internal cause and external cause (Du, Ibrahim, Shehata, & Badawy, 2013). Internal cause refers to the problems due to the plate number itself. External

cause refers to the problems because of the outer environment (Kwaśnicka & Wawrzyniak, 2002). The details of each cause are summarized as follows.

(1) Internal cause,

(a) The rear location of vehicle's plate number: the plate number may hang at different positions in line with the different type of vehicles. For example, plate number appears at the middle of the two red-taillights in sedan cars, but at the left bottom of trucks.

(b) The colour of plate number: different plate may have different foreground and background colour. For example, black characters with white background, white characters with blue background and so on.

(c) The length of characters: the length of character in plate number may be various. Due to the requirement of customization, the number of characters in each plate number may be different from one another.

(d) The types of characters: the characters in plate may different in different countries. For example, the plate number only contains English alphabets and numeric characters in US, but contains Chinese or Korean character in China and Korean.

(e) The occlusion of plate number: plate may be obscured by dirt or dust. Sometimes the plate may be damaged.

(f) The inclination of plate number: plate may be tilted in the captured image due to the camera taking picture direction

(2) External causes,

(a) The complex background: complicated background will affect the plate number localization accuracy. For example, the background may contain patterns or structures, such as car body advertisement, headlights and radiator, are similar to plate. The more redundant background information is eliminated, the better results are increased (Enyedi, Konyha, Szombathy, & Fazekas, 2004).

(b) The motion of cameras: cameras may be moved slightly by wind or unexpected reasons when it is running, which leads to image blur problem.

(c) The position of cameras: the position of camera may lead to different size of plate number due to the camera distance and the zoom factor.

(d) Illumination: the pictures contained plate number may have different illumination due to the various lighting conditions and vehicle headlights.

## 2.4.2 The Plate Number Format

The plate numbers have many different types and formats. Take the New Zealand (NZ) plate number as example, the common type of NZ plate number is six white characters with black background. The size of common style plate is shown in Figure 2.2. The plate number has a 36 cm length and 13 cm height approximately. The prior knowledge of the ratio of width and height can be used as one important feature in plate number localization process (Gao, Wang, & Xie, 2007).



Fig. 2.2 The Size of a Common NZ Plate Number.

There are also some other types of plate such as red character with black background, red characters with white background and plate number with extra NZ flag symbol. Table 2.3 shows assorted types of NZ license plate found on vehicles in New Zealand.

Table 2.3 Different Types of NZ plate number

| Numbers of Character | Plate colour | Character colour | Extra symbol |
|----------------------|--------------|------------------|--------------|
| Six                  | White        | Black            | No           |
| Five                 | White        | Black            | No           |

|      |       |       |     |
|------|-------|-------|-----|
| Four | White | Black | No  |
| Six  | White | Red   | No  |
| Five | White | Red   | No  |
| Six  | Black | White | No  |
| ...  | ...   | ...   | ... |

Apart from the black characters with white background, the other types of plate number are rarely seen on most streets. So this thesis would focus on the plate numbers that the foreground is black characters with a white background.

### 2.4.3 Plate Number Localization

Plate number localization model plays a pivot role in plate number recognition system, which influences the whole system's accuracy (Patel, Shah, & Patel, 2013; Roomi, Anitha, & Bhargavi, 2011; Tarabek, 2012). If the position of plate number in a picture cannot be located initially, the following steps will not be continued. The plate number may occur anywhere within the image. To detect the precise location of plate number in the image, particular characteristic features of the plate number that should distinguish it from other regions in the image have to be determined. The candidate plate number would be located based on the features, then it is extracted and further analyzed to verify if it is the real plate number in the image (Megalingam, Krishna, Somarajan, Pillai, & Hakkim, 2010).

Moreover, regarding to accuracy, a good plate number localization method has the potential to implement a reliable plate number recognition system. Therefore, researchers devote themselves to consider different algorithms in plate number localization module so as to increase the recognition accuracy.

Extraction of a plate number from an image is not a trivial task. Two major factors have

an effect on the successful extraction of plate number, the quality of the image and the robustness of the extraction algorithms (Wu, Chen, Wu, & Shen, 2006). High resolution image of vehicle with a clear view of plate number has a better recognition precision than that of low resolution image. For instance, a clear and tidy image can be easily viewed by human, so as well to the computer. Further, if the image is captured at a fixed distance from the vehicle every time, the size of the plate number will be stable. Finally, the angle at which the image is captured is also crucial to the extraction process. If the captured image contains plate number without angle, no other method such as Hough transform would be added, which means time saving and better performance. The second factor is the plate number extraction algorithms. Several approaches have been used in past for extracting plate number and each of them varies in their performance time, complexity and success rates. Features used to locate interest of area are based on the properties of plate number itself on the properties of the character string present on the plate number. In image recognition domain, features play a significant role in determining the identification accuracy of a system. We can extract low level features and high level features from an image. Low level features are features that we can extract primary features easily without any further information such as shape information and spatial information. The low level features based detection techniques include edge detection, point's detection and motion detection. High level features concern more about the shapes information of a digital image stored in computer. The features such as the plate number shape, symmetry and aspect ratio are some geometrical properties that have been used by previous researchers. The following paragraphs will detail some of the previous plate number extraction techniques.

A primary characteristic found in images is edge information. Edge detection (Marr & Hildreth, 1980) is a powerful technique used in image processing to extract useful features from an image and is the basis for most method used in plate number extraction (Lin & Huang, 2007; Mahini, Kasaei, & Dorri, 2006). Edges represent boundaries in an image and are represented by sharp changes in intensity value from one pixel to another. Edge can provide more useful information about the object we are interested. Finding

the edge in an image preserves the basic structural details of the image removing other information about colour, intensity and contrast (Omachi & Omachi, 2009). The edge information can be divided into two types: horizontal and vertical edges (Yu & Kim, 2000). Horizontal and vertical edges can be obtained by computing the gradient between adjacent pixels in horizontal or vertical direction (Kim, Kim, Ryu, & Kim, 2002). The formulas calculating horizontal and vertical gradient are defined as equation (2.5) and (2.6):

$$G_h(i, j) = \text{abs}(f(i+1, j) - f(i, j)) \quad (2.5)$$

$$G_v(i, j) = \text{abs}(f(i, j+1) - f(i, j)) \quad (2.6)$$

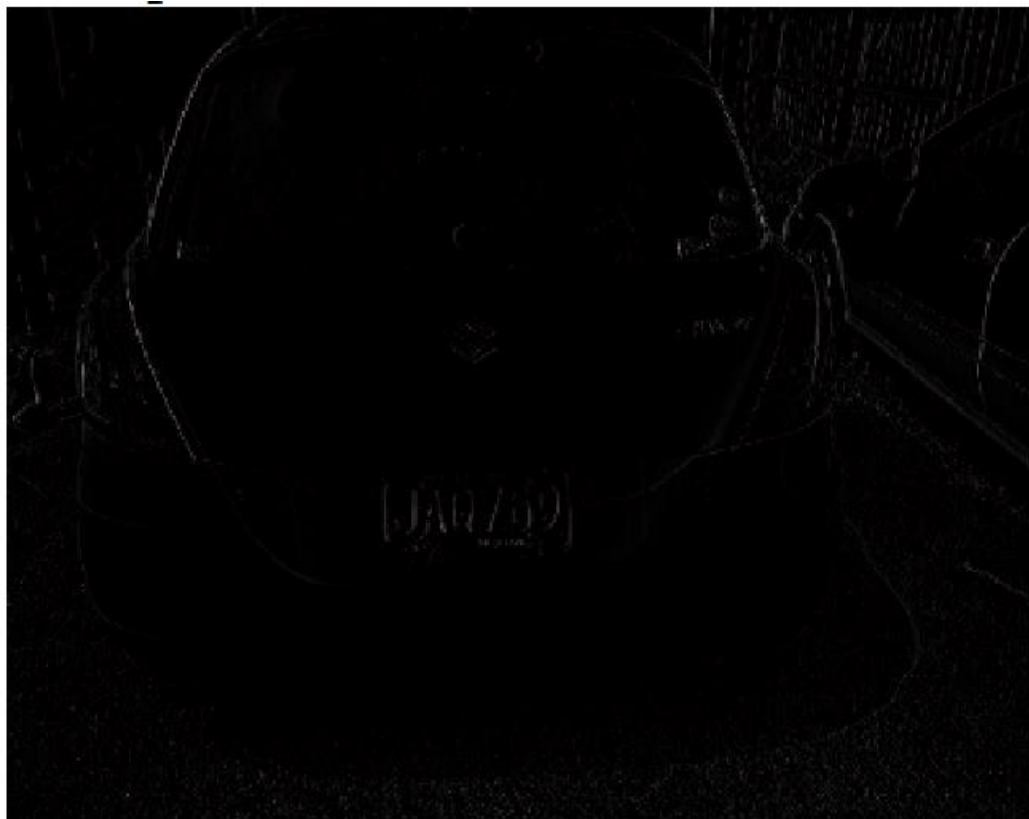
Figure 2.3 illustrates the results of applying the above equations to a gray scale vehicle image.



(a) Original Image



(b) Horizontal Gradient of an Image



(c) Vertical Gradient of an Image

Figure 2.3 Example of Horizontal and Vertical Gradient of an Image



The edge information can be enhanced by edge detectors. The commonly used edge detectors are ‘Sobel’, ‘Canny’ and ‘Roberts’ edge detector (Chong, Tianhua, & Linhao, 2013; Marr & Hildreth, 1980; Xu, Li, & Yu, 2004; Zheng, Zhao, & Wang, 2005). Specifically, these edge detectors are 3 \* 3 masks, which are used to filter the image to obtain the gradient image. Take the ‘Sobel’ as the example, the ‘Sobel’ in crosswise and lengthwise can be represented by matrix as,

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \text{ and } \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

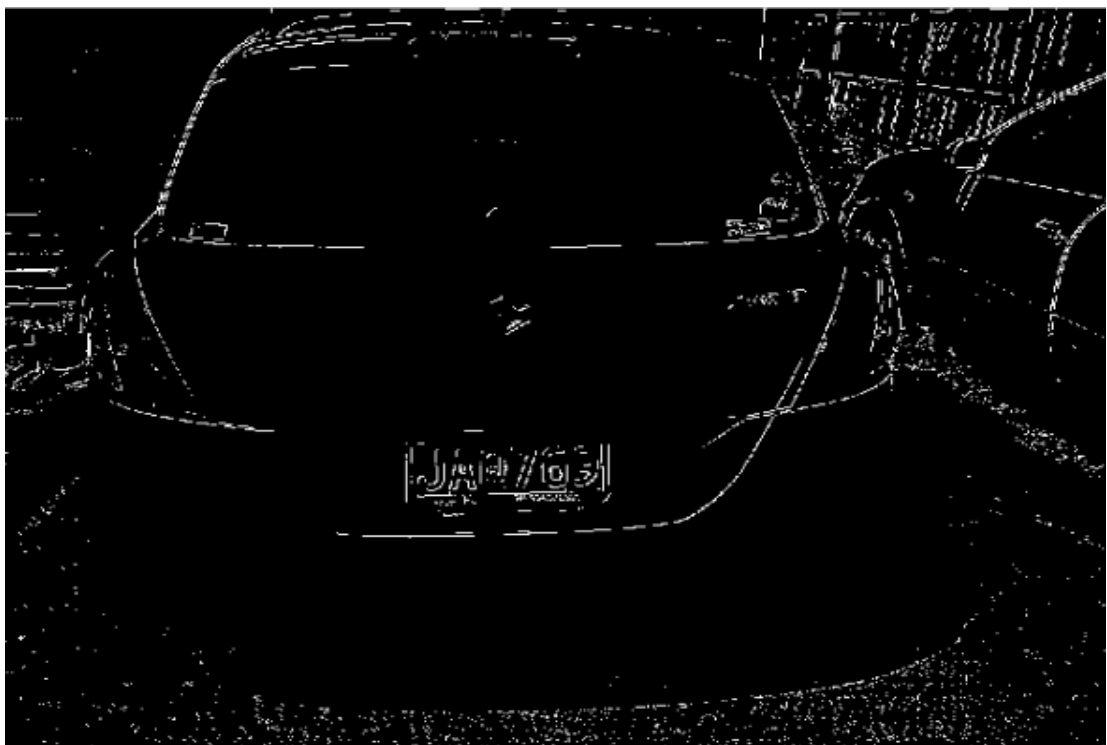
Then, the vertical or horizontal gradient can be calculated by using the 3\*3 matrix to apply a convolution in the original image. Figure 2.4 demonstrates the results of edge detection filter such as using ‘Sobel’, ‘Canny’ and ‘Roberts’ in vertical direction.



(a) Edge Detection by ‘Sobel’ Detector



(b) Edge Detection by 'Canny' Detector



(c) Edge Detection by Roberts Detector

Figure 2.4 Results of Vertical Edge Detection

Finding the edge of a vehicle image can help identify both plate number feature and the

character string feature. According to the edge information, other techniques can be proposed by its lines of edge. The plate number can be located by searching for a rectangle region with certain dimensions that correspond to the license plate (Lee, Kim, & Kim, 1994). The Hough transform is a popular technique to detect lines (horizontal or vertical lines) or curves (circle or oval) in an image and is used for finding significant features in images. Hough Transform is used in many LPR implementations for extracting plate number (Yanamura et al., 2003).

Spatial variance method can also be used to find out the plate area in input image. This type of method is quite similar to finding texts in image. This method can scan each line of the input image and choose the region with high variance as plate area (Cui & Huang, 1997). Figure 2.5 shows the spatial variance of two scan lines that traverses across the top of plate number and middle across the plate number.

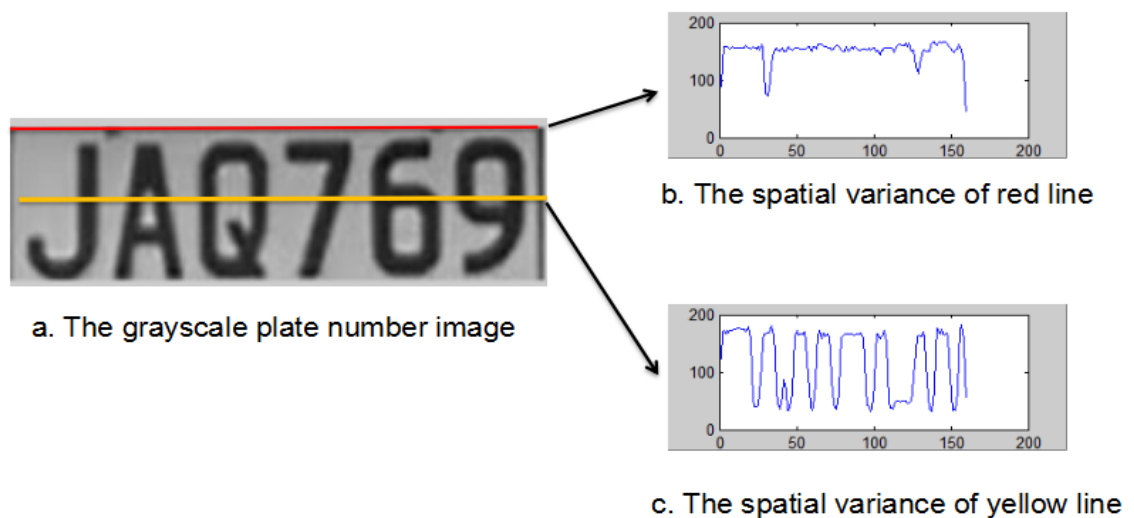


Figure 2.5 Spatial Variance of Two Scan Lines

As it is shown in Figure 2.5, the spatial variance of yellow line traversed across the characters has more ‘valleys’ than that by red line. The numbers of peak of wave or trough of wave can be employed as one characteristic to find out the position of plate number by scanning each line in the entire image. This type of method usually makes a counter to indicate the numbers of ‘valleys’ appeared in individual line. If the number of

a line satisfies a predefined threshold value, the line will probably be a candidate (Chen, Ren, Tan, & Wang, 2007). Then, other conditions are set to search the exact position of plate number.

The unique and distinguishable background colour of plate can be used to localize the plate area (Lee et al., 1994). The idea of extracting a plate region by its colour feature is that the colour combination of character and plate is distinct and only appear in a plate region. Due to the input image represented by a RGB palette which is not proper to represent tint, shade and tone, the original images should be transferred to other colour space such as Hue Saturation Lightness (HSL), Hue Saturation Value (HSV) and Hue Saturation Intensity (Oliveira & Conci, 2009; Wang, Zhou, & Geng, 2004). Then, Artificial Neural Network is trained to classify individual pixel in transferred image, which will determine whether the pixels belong to the initial specified background colour of plate or not. After the colour classification, candidate plate regions will be extracted by the horizontal and vertical colour histogram of the initial specified background colour of plate. Prior knowledge such as aspect ratio of the plate area will be used to verify the candidate plate region if the number of candidate plate regions is greater than one.

Mathematical morphology is a set of operations widely used in image analysis for segmentation, which is usually applied to LPR system (Haralick, Sternberg, & Zhuang, 1987; Suryanarayana, Mitra, Banerjee, & Roy, 2005). Morphology operations can preserve the basic shapes of an object while removing other redundant information from the image. Mathematical morphology operations such as dilatation and erosion will be processed on the result of edge detection. Image dilatation operation can connect the gaps between edges and create a connective region, whereas the image erosion operation removes small area. Both of the morphology operations need a structure element to determine the shape of the target that is wanted to explore. Figure 2.6 shows the procedure of detecting a plate number area in an image. Edge and morphology based method is simple and fast. However, they require the continuity of the edge.



Figure 2.6 The Detected Plate Number Area by Morphology Operations

#### 2.4.4 Plate Number Recognition Methods

A large number of character recognition techniques have been proposed in papers and even utilized in real-time plate number recognition system (C. N. E. Anagnostopoulos, Anagnostopoulos, Loumos, & Kayafas, 2006). According to the times of execution, these methods can be broadly divided into two main classes: iterative and non-iterative approaches (Chang, Chen, Chung, & Chen, 2004). All the recognition methods are based on the features extracted from plate number. By using either the statistical, syntactic or neural approaches, the features are classified (Ahmed, Sarfraz, Zidouri, & Al-Khatib, 2003).

Template matching is one very useful tool for intelligent perception process and pattern recognition task (Brunelli, 2009). Although this technique is often considered as a very basic and limited method in computer vision, it is still a powerful method related to many old and new techniques in the field (Brunelli, 2009). The definition of template

matching is that a technique in digital image processing to use a testing image matching each template (36 characters). The recognition procedure is to compare the testing image to each template and find out which one of templates match the testing image best. There are a slew of matching functions are used to measure the similarity between testing image and template. The common matching functions include: Sum of Square Difference (SSD) (Pentland, Moghaddam, & Starner, 1994), Normalized Cross Correlation (NCC) (Di Stefano, Mattoccia, & Tombari, 2005) and Mean Absolute Difference (Naito, Tsukada, Yamada, Kozuka, & Yamamoto, 2000).

The formula of SSD is:

$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2 \quad (2.7)$$

where  $R(x, y)$  is the return value which indicates how well the template  $T(x', y')$  matches the testing image  $I(x + x', y + y')$ ,  $x'$  and  $y'$  represent the row and column of template, respectively.

The formula of NCC is:

$$R(l, r) = \frac{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (x_{i+l, j+r} y_{ij})}{\sqrt{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (x_{i+l, j+r}^2)} \sqrt{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (y_{ij}^2)}} \quad (2.8)$$

where  $R(l, r)$  is the normalized correlation coefficient at the point  $(l, r)$ .  $N_1 \times N_2$  is the size of template.  $X_{i+l, j+r}$  and  $Y_{ij}$  are the gray value of pixels at  $(i+l, j+r)$  and  $(i, j)$  in template image and testing image respectively.

The formula of MAD is:

$$D(i, j) = \frac{1}{M \times N} \sum_{s=1}^M \sum_{t=1}^N |S(i + s - 1, j + t - 1) - T(s, t)| \quad (2.9)$$

where  $S(x, y)$  is the testing image and  $T(x, y)$  is the template image. The return value  $D(x, y)$  indicates the similarity between the two images. Template matching can be utilized on binary and on gray-level images (Ahmed et al., 2003). The quite long performance time is the major limitation of template matching techniques due to each time the testing

images should compare each blocks in template image. Moreover, the creation of templates also has a significant influence on the recognition accuracy.

Artificial Neural Network (C.-N. E. Anagnostopoulos et al., 2008) has been widely used in character recognition problems (Fahmy, 1994; Parisi et al., 1998; Raus & Kreft, 1995; Zhu, Huang, Xu, He, & Liu, 2011). Volna and Kotyrba (2013) developed a vision system for license plate recognition based on neural networks. The neural network is based on multilayer feed-forward back-propagation algorithm that using one hidden layer. 180 images taken under different lighting and weather conditions were used to test the proposed system. The average success rate reached at nearly 81%. Wei, Li, Wang, and Huang (2001) used neural network to classify the number plate from colour image. Then forth sub-network, three-layer MLPN networks and a MAX network, was used to realize character recognition. The recognition rate is about 95%. However, their system is not sensitive to variations of weather, illumination.

ANN is a connectionist massively parallel system which is similar to human neural system (Karnin, 1990). Commonly, a single neuron is a basic processing unit in ANN, an aggregation of numerous neurons is a neural network.

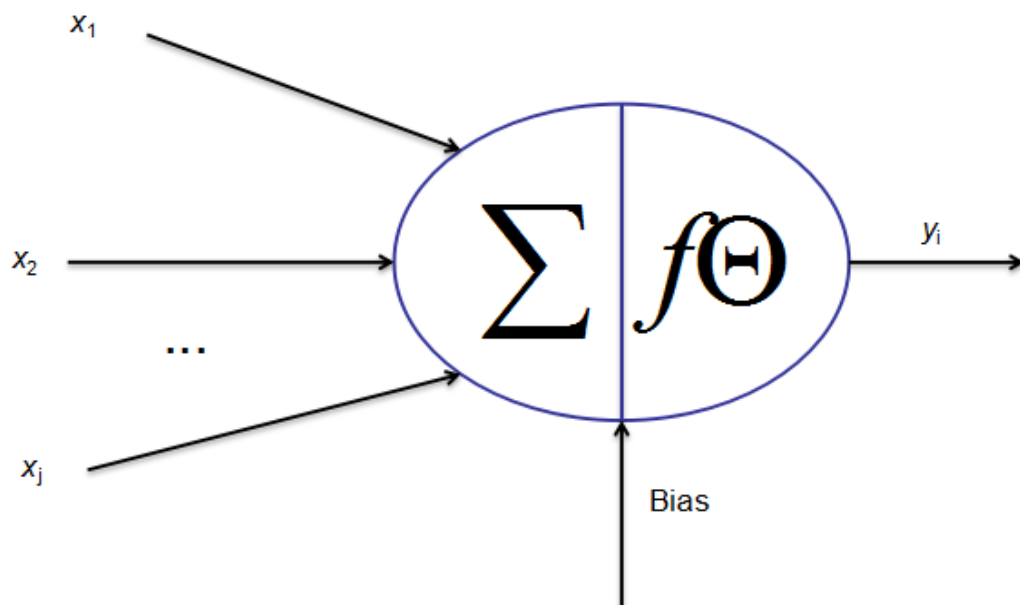


Fig.2.7. The Structure of a Neuron

As shown in Figure 2.7, the components of a neuron include an input vector  $x_i$ , corresponding weight vector  $w_{ij}$ , activation transfer function  $f(\cdot)$  and bias  $\Theta$ . The mathematical function is described in equation (2.10).

$$Y = f[\sum_{i=1}^n W_{ij} X_i - \Theta] \quad (2.10)$$

where  $W_i$  are weights,  $X_i$  are inputs,  $i$  is the number of sample,  $j$  is the quantity of element of input.

Among the components of neuron, the activation transfer function plays a key role of the ANN. The commonly used activation transfer function includes positive hard limit transfer function, symmetric hard limit transfer function, linear transfer function, logarithmic sigmoid transfer function and symmetric sigmoid function.

The learning algorithms of a neural network are grouped into two main categories, namely, supervised learning and unsupervised learning. The learning algorithm used in this thesis is Back Propagation (BP) Neural Network which is a supervised learning algorithm.

A BP-neural network is a multi-layer feed-forward network using error back propagation training algorithm, which consists of input layer, hidden layer and output layer. The number of neurons in the input layer and output layer are determined by the dimensions of samples and the classes of samples, respectively. Figure 2.8 is a structure of BP-neural network; four important mathematical functions of a BP-neural network are described as below.

The error function,

$$E = \frac{1}{2} \sum (d_l - y_l)^2 \quad (2.11)$$

The output function in hidden layer,

$$H_k = f(\sum_i M_{ik} x_i + \Phi_k) \quad (2.12)$$

The output function in output layer,



$$Y_j = f(\sum_k W_{kj} h_k + \Phi_j) \quad (2.13)$$

and weights adjusting function,

$$W_{ij}(n+1) = W_{ij}(n) + \alpha \delta_j x_i \quad (2.14)$$

In the equations (2.11)-(2.14),  $d$  and  $y$  means the target output and actual output of the ANN,  $H$  refers to the hidden layer output vector. The lowercase  $k$  means the index of neuron,  $\phi$  represents the bias,  $\alpha$  and  $\delta$  denote the learning rate and error term.

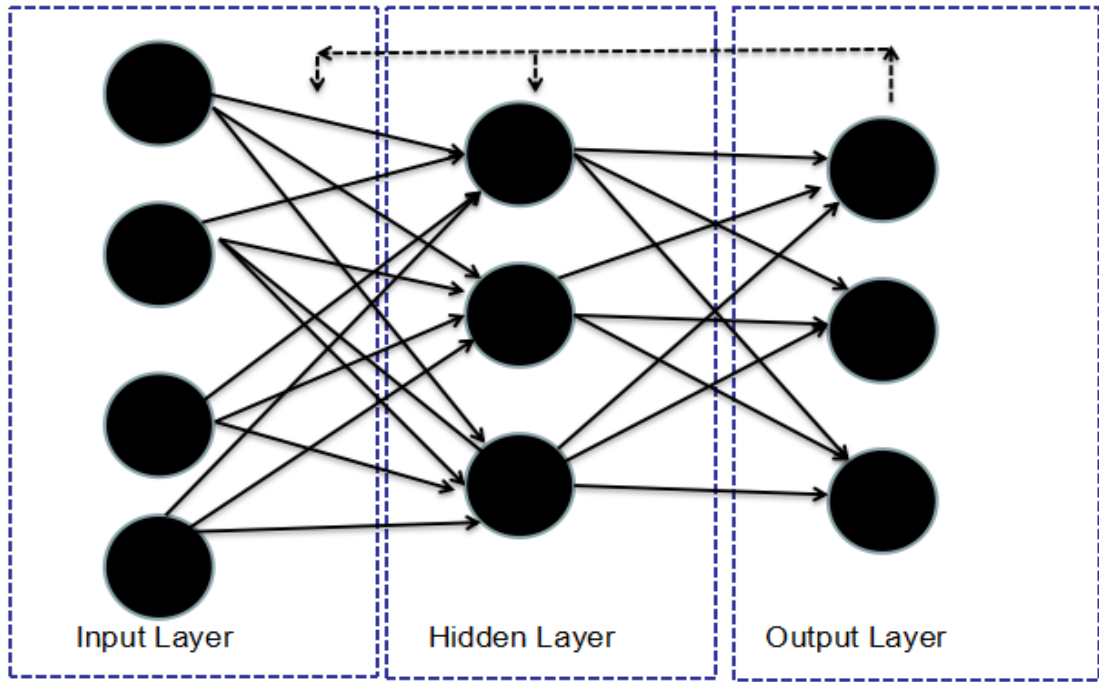


Fig. 2.8 The Basic Structure of BP-Neural Network

The procedure of implementing ANN to recognize the character can be described as, 1) create an input training data set (features extracted from character) and input target data set, 2) train the neural network and 3) simulate the trained neural network. Grid method is one feature extraction method to collect features from binary images. The thought of grid method is to divide a character image into  $n \times m$  grid, the number of white pixels in each grid and the total white pixel in the whole image are calculated as the features to represent the character. However, some of these features are not representative, especially the total number of white pixels in the entire image. The following Figure 2.9

demonstrates the feature, the total number of white pixels from the image, is not a proper feature.

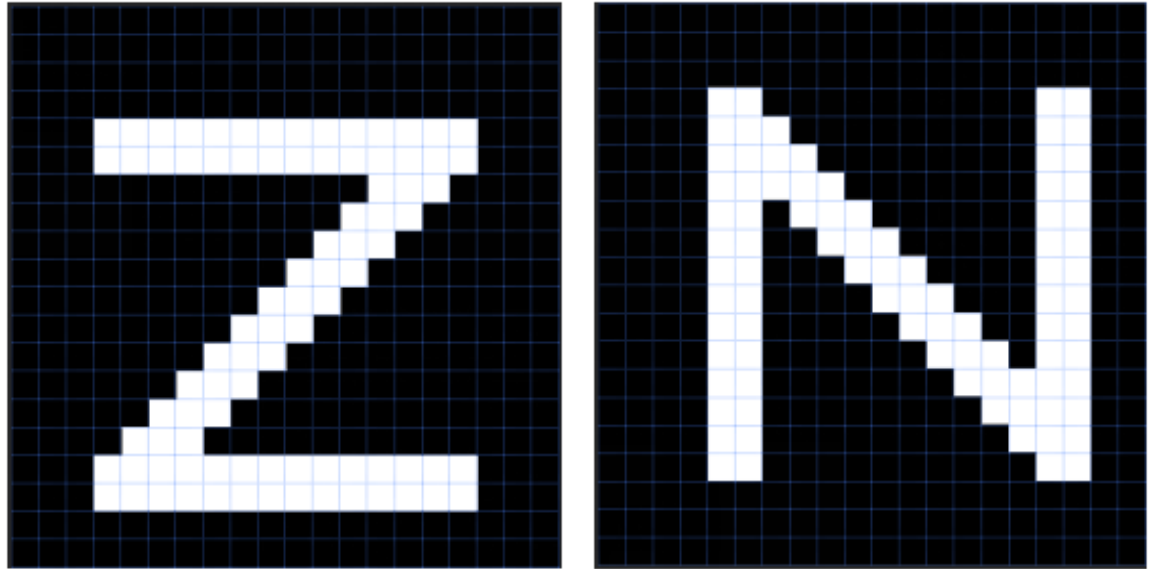


Figure 2.9 The Sample of Same Number of White Pixels in Character Z and N

The total number of white pixels in character “Z” and “N” are the same, which is not a good feature .The accuracy of recognition will be affected by this limitation. However, this method has advantages compared with other methods in binary image. The speed of extracting features is faster than feature extraction method used in frequency domain such as Fourier descriptor.

# Chapter 3

## Methodology

*This chapter explains research methods to satisfy the objectives of the research. This chapter mainly covers the details of research questions, data collection and relevant experimental measurement. To be more exact, related work of research background is introduced again. According to the former literature review, research question of this thesis is detailed. Data acquisition and measurements for the experiment are also described. Finally, the expected outcomes of this thesis are presented.*

### 3.1 The Related Work

The red light violation occurred at road junction leads to a great deal of serious road accident. The traffic violation has become a big threaten not only to the drivers themselves, but also to passing-by pedestrians. This situation is becoming quite common since the number of transport surged dramatically in the recent decades. On the other hand, the red light violation also causes traffic jam at rush hour during the weekdays (Klubsuwan, Koodtalang, & Mungsing, 2013). Therefore, it is important to detect a red light violation to avoid fierce traffic accidents and keep people safe. Red light violation detection system is an important component of intelligent transportation system which is being investigated globally (Wang, Meng, Zhang, Lu, & Du, 2013). To detect the red light violation is prerequisite for traffic ticketing management system as mentioned in Chapter 1 that the car running through the stop line will trigger the following actions such as plate number recognition, database retrieval and sending traffic tickets automatically.

Several technologies and methods have been applied to detect traffic violation. These methods can be generally grouped to physics-based and video-based method. The physics-based method, the traditional technology, usually needs help from physical devices to implement red light violation detection. The physical equipment and devices include induction coils, radar, laser, RFID, infrared ray, etc. Video-based method has become the main trend because it has many advantages such as low cost, long life service, easy installation, high accuracy of detection, robustness poly-functionality, etc. comparing with traditional technologies. In each frame of the traffic video, a detection areas fixed and located at the front of stop line are defined and setup initially in video-based method which is quite similar with placing inductive loop under the road. Moving object detection and tracking methods are used to detect moving vehicles and record the moving vehicles trajectory information in the rest area of frame. Normal events such as “car appear”, “speed down” and “car stop” will be extracted from the

trajectory information. For instance, using the current frame subtract the background frame will detect the foreground (vehicles in traffic surveillance scenario), if the foreground is detected, the “car appear” event is occurred. Similarly, “car stop” event will be extracted when using Frame differencing to compare the difference between continuous frames. After detecting the abnormal events, plate number recognition process will be added to event-driven traffic related applications.

### **3.2 The Research Question**

As mentioned in Chapter 1, this thesis aims to analyze computer vision technique used in video event detection and implement an event-driven traffic ticketing system. The core of a traffic ticketing system is plate number recognition. The rate of successful recognition will determine the performance of this application. So this thesis may focus more on plate number recognition module.

Through the systematic studying and analyzing the powerful computer vision techniques and methods, it is quite important to know the procedures of implementing a video-based event detection application. Therefore, the main research question of this thesis is designed as:

***Question:***

***What computing techniques should be used to implement an event-driven traffic ticketing management system?***

The core of a traffic ticketing system is plate number recognition, and the associated computing techniques which need to be evaluated and chosen before combining correct and proper techniques to implement plate number recognition. Our research will analyze and assess the computing techniques that are able to be used develop the desired system. Particularly, our research work will reveal the main techniques in computer vision related to event-driven application.

## 3.3 Data Acquisition

### 3.3.1 Data Collection for Experiments

Since the proposed event-driven traffic ticketing system is made by two main parts, two types of data are collected in this research project. The data for event detection module is a series of videos that contain vehicles running on road and twenty images contain white line are used to verify the line detection using Hough transform. The data for plate number recognition module are the pictures showing the rear view of vehicles. All the images or videos were captured by using 1200 pixels dimension Mi 4 Phone built-in camera.

The data for event detection module are two videos captured by mobile phone built-in cameras. The first video is a serial of frames related to the real traffic flow in an intersection. There are continuous 452 frames with a size  $360 \times 640$  in this video. This video is utilized to evaluate the line detection, foreground detection, and event trigger. The second video has 176 frames captured at nearly 30 meters on long driveway. The speed of the running car in this video is 15 kilometre per hour. The whole video has a little motion due to the camera is not fixed.

The data for plate number recognition module consists of 120 images of the rear views of various vehicles. 80 images were taken at Auckland different places under various lighting conditions such as sunny, cloudy, twilight, rainy and so on. The 80 images were true colour (RGB) image stored as “.jpg” format with the size  $1920 \times 1424$ . The reset 40 images randomly chosen from an online project (URL: <http://www.zemris.fer.hr/> ) are used to test the plate number localization method. Both of the images are the rear views of various vehicles including cars, trucks and buses. As described in the project online website, the test images were taken by using OLYMPUS CAMEDIA C-2040ZOOM digital camera and the size of images were  $640 \times 480$ .

### 3.3.2 Templates and Datasets for ANN

The 36 character templates, the numbers from '0' to '9' and the letters from 'A' to 'Z', used in our experiment are created by Adobe Photoshop CS6. The size of template is  $20 \times 40$  with a white foreground and black background.

The data set for Character recognition by ANN is comprised by input data set (p) and target data set (t). Inputs 'p' is a  $200 \times 360$  matrix, representing 360 samples of 200 elements. Specifically, each character (total 36) has ten images for training purpose in ANN. The individual character image is divided to 20 rows and 10 columns (200 grids) which will be transferred to a column vector to represent the character. The white pixels in each grid are calculated and normalized to the scope of 0 to 1 as the character's feature. Targets 't' is a  $36 \times 360$  matrix, representing 360 samples of 36 elements. 36 is the class number of characters. 360 samples are the same number as input 'p'. In the matrix 't', the rows' value determine the category of character. For example, the character 'A' in target 't' matrix will be represented as a column vector ( only contains one and zeros) that the row 11 is 1 and other rows are 0.

## 3.4 Evaluation Methods

Evaluation is essential to an experiment or a computer software system. The performance time should be one of the criteria in plate tracking in event detection module. In our experiments, we consider using feature points extracted from a small section in images to track the moving vehicles. Hence, the performance time of each feature extraction algorithms will be used as one measurement.

Precision and recall is another important criterion in pattern recognition and information retrieval. In precision (3.1) and recall (3.2) context, four terms should be defined and understood, which are: True Positives (*TP*), False Positives (*FP*), True Negatives (*TN*) and False Negatives (*FN*). Then the formulas to calculate precision, recall and accuracy are defined as,

$$Precision = \frac{TP}{TP + FP} \quad (3.1)$$

$$Recall = \frac{TP}{TP + FN} \quad (3.2)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.3)$$

In our plate number recognition, the accuracy of the recognition is employed. In this thesis, the numerator ( $TP+TN$ ) in equation (3.3) is defined as the number of plate numbers that are recognized correctly. Denominator ( $TP+TN+FP+FN$ ) in equation (3.3) refers to the plate numbers that are recognized both correctly and incorrectly. The accuracy formula is also expressed as:

$$Accuracy = \frac{\text{theNumberOfCorrectDetection}}{\text{theNumberOfCorrectDetection} + \text{theNumberOfIncorrectDetection}} \quad (3.4)$$

### 3.5 The Novelty of This Thesis

This thesis presents a new schema to implement an event-driven traffic ticketing system. The similar event detection from video stream systems can be developed step by step in accordance with the procedures provided in this thesis. The entire system proposed in this thesis consists of four modules, namely, (1) event detection module, (2) plate number recognition module, (3) database management module and (4) traffic ticket transmission module. The details of the underlying method and implementation in each module will be illustrated in Chapter 4.

Additionally, this thesis analyzes the core algorithms and approaches which are suitable for this system. The pros and cons of the moving object detection methods are also compared. The specific comparison and analysis have been shown in Chapter 2 Literature Review. Due to the importance and influence of the plate number recognition module, this thesis focuses on analyzing computer vision techniques with respect to implement plate number recognition module. A novel plate number localization



algorithm called Secondary Positioning (SP) method is presented in Chapter 4 of this thesis.

Further, this thesis also improves the traditional Template Matching method used in character recognition. Specifically, the character templates are grouped into four classes according to the symmetry characteristic of characters, which are illustrated in the next chapter. Results from Chapter 5 show the improved method has a better performance than traditional Template Matching method.

Last but not least, this thesis also provides a detail to implement character recognition by using Neural Network in MATLAB. The specific steps are illustrated in Chapter 4 of this thesis.

## Chapter 4

### Design and Implementation

*This chapter introduces the entire system schema and the details of implementing the functionalities in the each component of the system. Importantly, the novel plate number localization method called Secondary Positioning (SP) and improved Template Matching method is demonstrated in this Chapter. Some experimental results are also shown in this Chapter.*

## 4.1 Event Detection

### 4.1.1 Red Light Detection

Red Light Detection (RLD) is the first step of this thesis project. Because the event of running through the red light only occurs when the traffic light colour is red, the status of the traffic light determines whether event detection is triggered or not. The RLD program works under the condition that the traffic light turns to red in real scene. When the traffic light colour is turned to other colours instead of red, our program will be on status of sleeping. Owing to the camera is fixed in the traffic holder, the area containing the traffic lights will be also at an unchangeable position. Hence, the program will scan in a specified small region instead of the whole image. The algorithm of red light detection is illustrated in Algorithm 4.1.

---

**Algorithm 4.1. Red Light Detection**

---

**Input:** Frames of traffic video

---

**Output:** Trigger event detection module

---

Set global variable : *Global\_light\_status* =false;

**While** *Global\_light\_status* = false

    Read frame *Rgb\_image* from traffic video;

    Convert *rgb\_image* to *hsv\_image*;

    Variables: *H,S,V* is the components for *hsv\_image*;

    Set *x*=row *y*=column of the *rgb\_image*;

    Set *R\_count,Y\_count,G\_count* as the counter and initialize to 0;

**For** *i*=1:*x* **do**

**For** *j*=1:*y* **do**

**If** *H(i,j)&&S(i,j)&&V(i,j)* belongs to red **then**

*R\_count*++;

**Else if** *H(i,j)&&S(i,j)&&V(i,j)* belongs to Yellow **then**

*Y\_count*++;

**Else if** *H(i,j)&&S(i,j)&&V(i,j)* belongs to Green **then**

*G\_count*++;

**End if**

**End for**

**End for**

**If** *max(R\_count,Y\_count,G\_count)* is *R\_count* **then**

        Trigger event module

---

Set *Global\_light\_status*=true

**End if**

**End while**

---

To achieve an accurate detection, the original Red Green Blue (RGB) image is usually transformed to Hue Saturation Value (HSV) colour space. Because HSV colour space has more balanced colour-difference perception (an efficient for rare colour), which is quite suitable for image processing. RGB images are  $m \times n \times 3$  image arrays whose three planes contain the red, green, and blue components. HSV images contain the same array but whose three planes are hue, saturation and intensity value components. The equations of colour space transition from RGB to HSV colour space are defined as,

$$h = \begin{cases} 0, & \text{if } \max = \min \\ 60^\circ \times \frac{g-b}{\max-\min} + 0^\circ, & \text{if } \max = r \text{ \& } g \geq b \\ 60^\circ \times \frac{g-b}{\max-\min} + 360^\circ, & \text{if } \max = r \text{ \& } g < b \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ, & \text{if } \max = b \end{cases} \quad (4.1)$$

$$s = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max-\min}{\max}, & \text{otherwise} \end{cases} \quad (4.2)$$

$$v = \max \quad (4.3)$$

where,  $h \in [0^\circ, 360^\circ)$ ,  $s \in [0, 1)$ ,  $v \in [0, 1)$  and  $r, g, b$  are the three components in HSV and RGB colour space, respectively. *max* and *min* are the maximum value and minimum value among the values from  $r, g$ , and  $b$ . The process of converting RGB colour map to HSV colour map is easily implemented by MATLAB built-in function “*rgb2hsv*”.

The thought of detecting red light signal status is that: The region including traffic lights will be conversed to HSV colour space. Set three counters to calculate the number of corresponding three colours, red, green and yellow, in the HSV colour space. The colour of traffic light will be determined by the maximum value in the three counters.

### 4.1.2 Plate Area Detection and Tracking

Flowed by the red light detection is plate area detection and tracking. The event of “running through the red light” could be determined by the tracking trajectory of moving vehicles. To better handle the challenges of scale changes, rotation, and occlusion existing in object detection and tracking application, local features such as blobs, corners and edge pixels were often utilized (Cui, Li, Chen, & Li, 2011). According to the different local features, researchers have presented a large amount of feature detectors such as Harris feature detector and SURF feature detector for detection. Commonly, in computer vision, distinct features were extracted from a particular region, Region of Interest (ROI), in the entire image. In our scenario regarding to plate number area is considered as the ROI of images. The specific method to locate the plate number area in photo is described in 4.5.1 section. In this section, we mainly introduce the procedures of vehicles’ ROI detection and tracking. The procedures are described as below.

- 1) Apply our proposed plate number localization method to detect ROI.
- 2) Extract feature points from ROI.
- 3) Track the features in consecutive frames of the video.

Figure 4.1 and figure 4.2 illustrate the procedure of extracting feature points and locating the ROI automatically, and the result of tracking the ROI in the consecutive frame in the video, respectively.



(a) Original Image



(b) ROI Detection



(c) Extracted Feature Points

Figure 4.1 The Implementations of Locating ROI and Extracting Feature Points



Figure 4.2 The Tracking Results of a Number Plate

The plate area tracking method used in this thesis is Kanade-Lucas-Tomasi (KLT) algorithm. KLT algorithm is computer vision tracking algorithm which is based on

tracking the point features. The translation motion model is utilized to calculate the residuals of two translation window gradation and seek the residual Sum of Square Difference (SSD) to match the same features in two successive frames.

## 4.2 Plate Number Recognition

### 4.2.1 Plate Number Localization

The first step of our plate number localization method is to remove the complex background. The output of the first step is to find the small area that contains plate number between the left and right red taillights at the back of the car.

---

#### Algorithm 4.2 Plate Number First Scan

---

**Input:** Captured image with complex background

---

**Output:** Cropped vehicle with plate number image

---

```

Read image rgb_image from video frame;
Convert rgb_image to gray_image;
Set  $m=Width$ ,  $n=Height$ ;
Set BW_image=zeros( $m,n$ );
For  $i=1$  to  $m-1$  do
    For  $j=1$  to  $n-1$  do
        If hsv_image( $H,S$ ) belongs to red then
            BW_image( $i,j$ ) = 1;
        Else
            BW_image( $i,j$ ) = 0;
        End if
    End for
End for

```

Morphological operations were used to erode and dilate the binary image *BW\_image*. As a result, the binary image would contain two main areas: *left\_taillight\_area* and *right\_taillight\_area*;

Set  $(x1,y1)$  = *left\_taillight\_area*.centroid and  $(x2,y2)$ =*right\_taillight\_area*.centroid;

*Crop\_image*=*imcrop(rgb\_image,[x1,y1, (y2-y1), predefined\_value])*;

---

In plate number first scan, the RGB image will be transferred to HSV colour space initially. A binary image will be obtained by detecting the red pixels in HSV colour space in accordance with the prior knowledge of the Hue and Saturation value of red.



For example, the white pixel in binary image is calculated by comparing the Hue and Saturation value of red with a threshold value. After that, morphological operations such as closing and erosion are used to eliminate the noise (J. S. Lee, Haralick, & Shapiro, 1987). The small redundant areas will be omitted by using image opening operation.

Normally, the two red taillights areas will be displayed in filtered binary image. The two red taillights should appear at the same row. Hence, conditions are set to examine whether the detected area is demanded area. The condition in our experiment is whether the distance between two centroids of the detected areas is less than 25. If the value less than the predefined value, we consider the detected areas as two taillights in the same line. However, there are a lot of abnormal situations in binary image step. For example, the binary image possesses more than two connected components or only is comprised by one connected component. The program should be robust to the above conditions. Figure 4.1 illustrates the flowchart of how to crop the candidates (the area contains plate number).

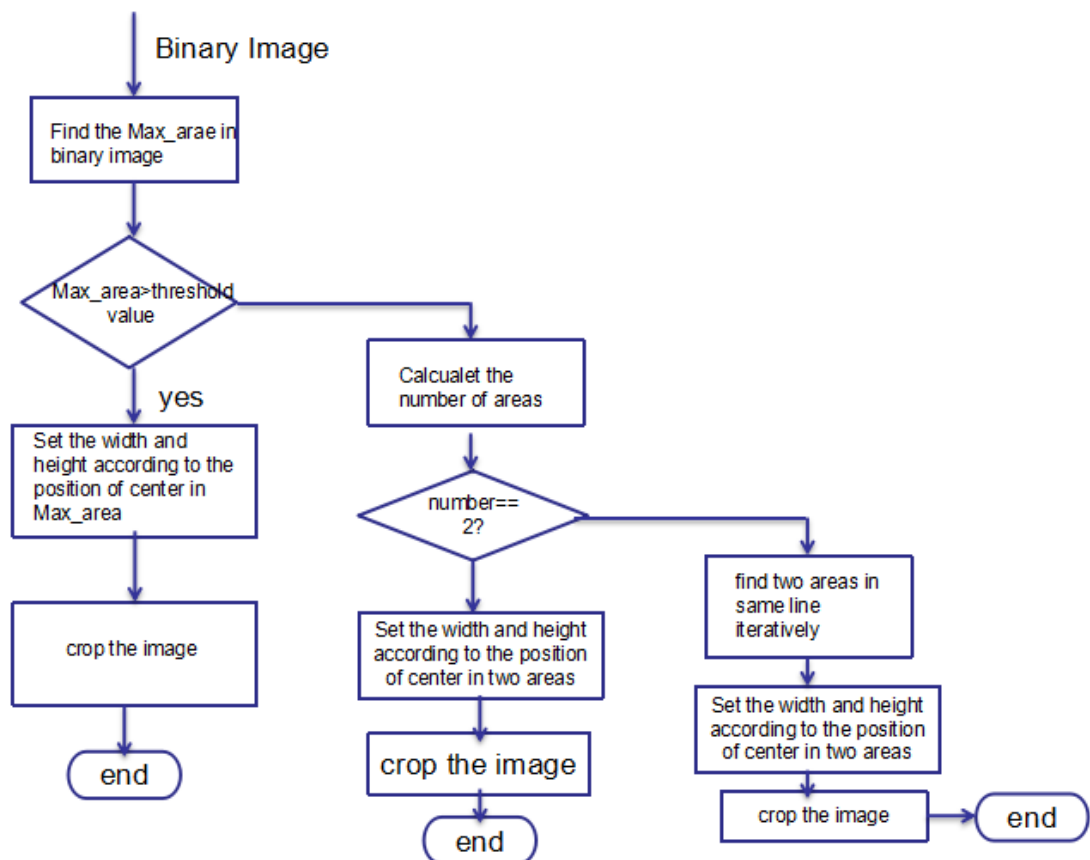


Figure 4.3 The Flowchart of How to Search the Candidates

In order to handle one complicated situation that the input image contains a car whose body colour is red, we design a condition to verify it. As we can see from Figure 4.4, the H component in HSV colour space is quite similar with the red taillight and the plate number is surrounded by red pixels. The fast and easy way to roughly crop the interest of area is to find out the biggest area when doing red colour detection in HSV colour space. It is shown in Figure 4.4 that the plate number appears in the middle bottom of the biggest detected red-pixel area. Figure 4.5 shows that the centroid of foreground is calculated as centroid  $(r, h)$ . Then, the coordinate of starting point in the cropped area is defined as  $row = r - r/2$ ,  $column = h - h/2$ . The width and height are calculated from,  $width = r$  and  $height = image\_height - h$ . Finally, the image contains plate number will be cropped by the above data.

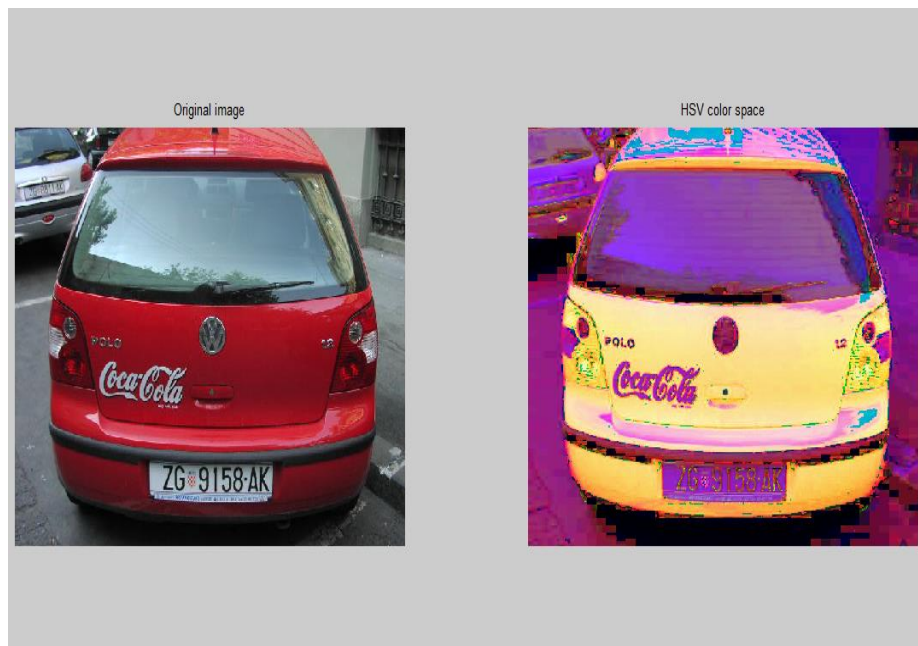


Figure 4.4 The Original Image and HSV Colour Space of a Red Car (Source from <http://www.zemris.fer.hr/>)

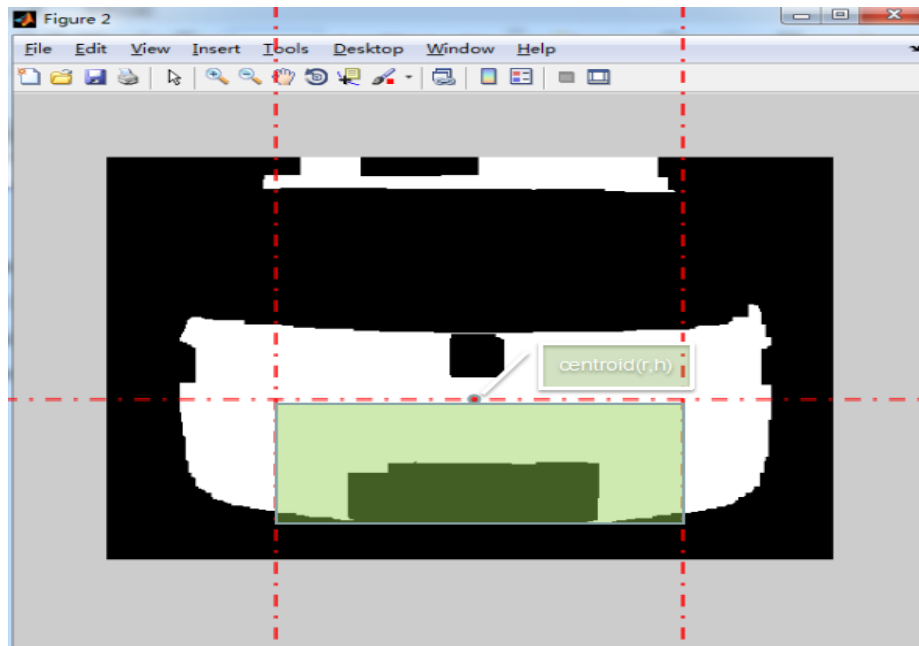
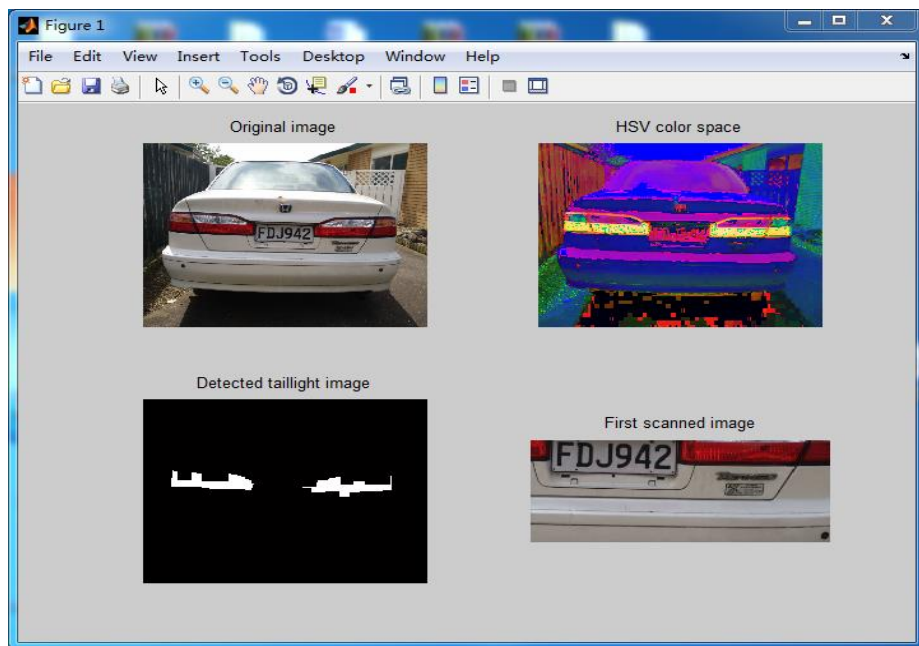
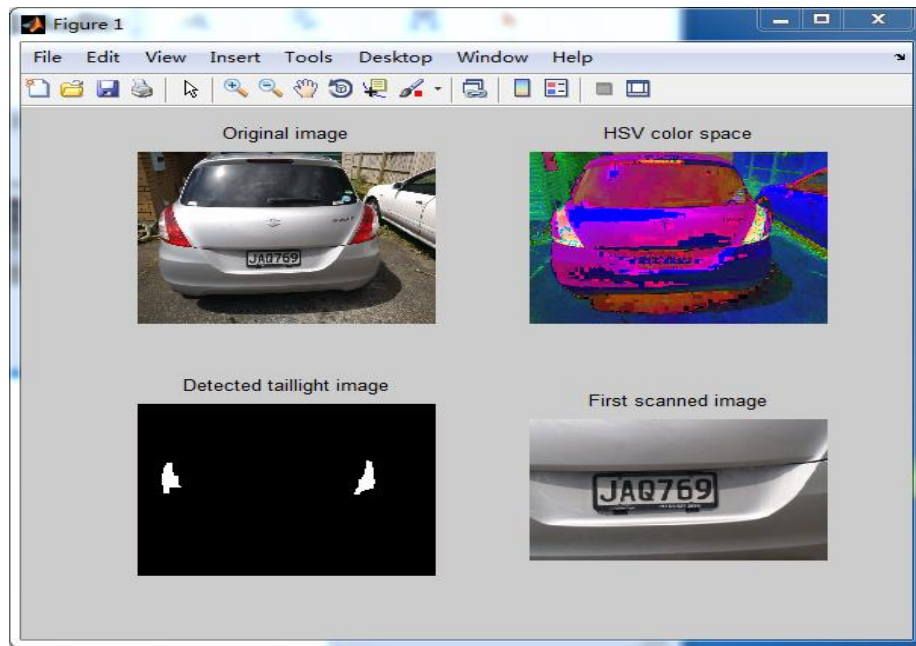


Figure 4.5 The Centroid of Detected Red-pixel Region

Another complicated situation is that the input image contains other objects whose colour is also red. As a result of detecting red pixels in HSV colour space, there are more than two foregrounds in binary image. To solve this problem, we design a double loop to search the two areas (two red taillights) in the same line. Figure 4.6 shows some results of the first scan.



(a) Example 1



(b) Example 2

Figure 4.6 The Results of The First Scan

The outcomes of plate number in the first scan are a small region of the original image which has removed complex background. The second scan will be carried out to locate the precise position of the plate number. Figure 4.7 shows the flow diagram of second scan.

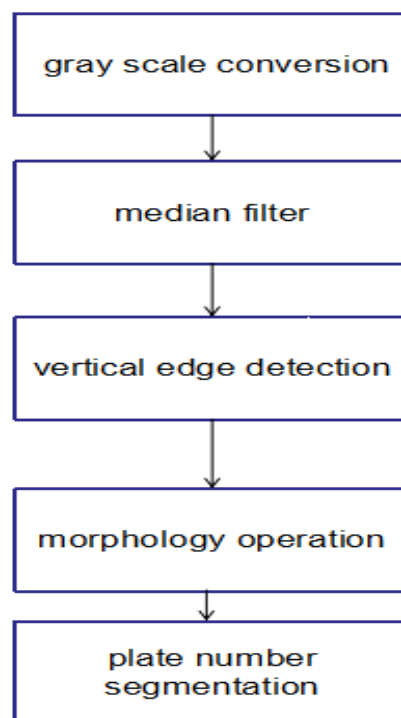


Figure 4.7 The Flow Diagram of The Second Scan.

Considered the cropped image contains valuable vertical edge information of characters in plate number, the followed algorithm will use edge detection to search the specific plate number area. ‘Sobel’ detector is used to detect the vertical edge in gray scale image. Then, morphological operations are used to enhance the plate number area and remove noises. The specific algorithm of second scan is listed in algorithm 4.3. Figure 4.8 presents the results of plate number second scan.

---

**Algorithm 4.3 Plate number second scan**

---

**Input:** Cropped vehicle image with plate number

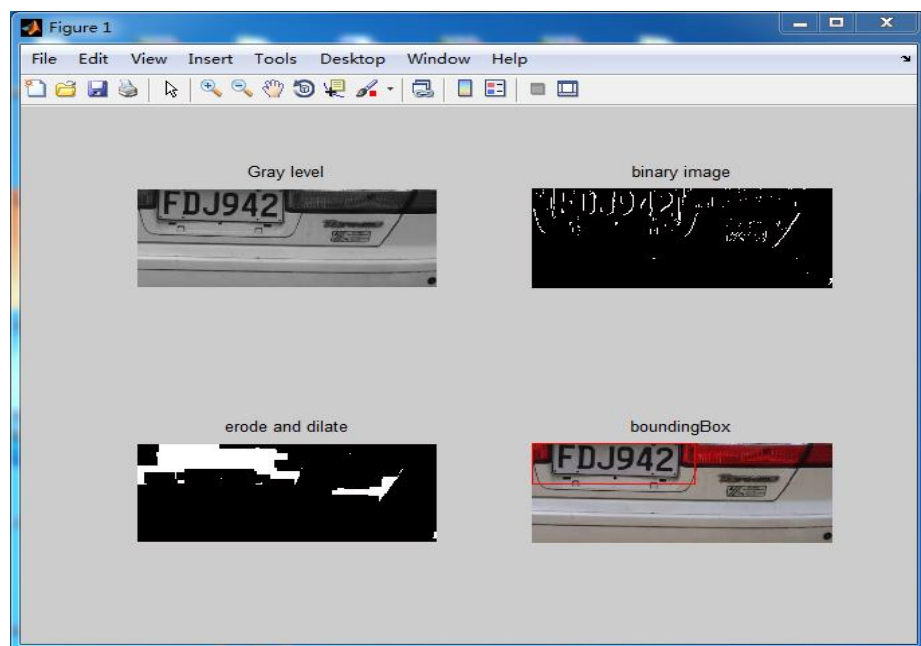
---

**Output:** Plate number image

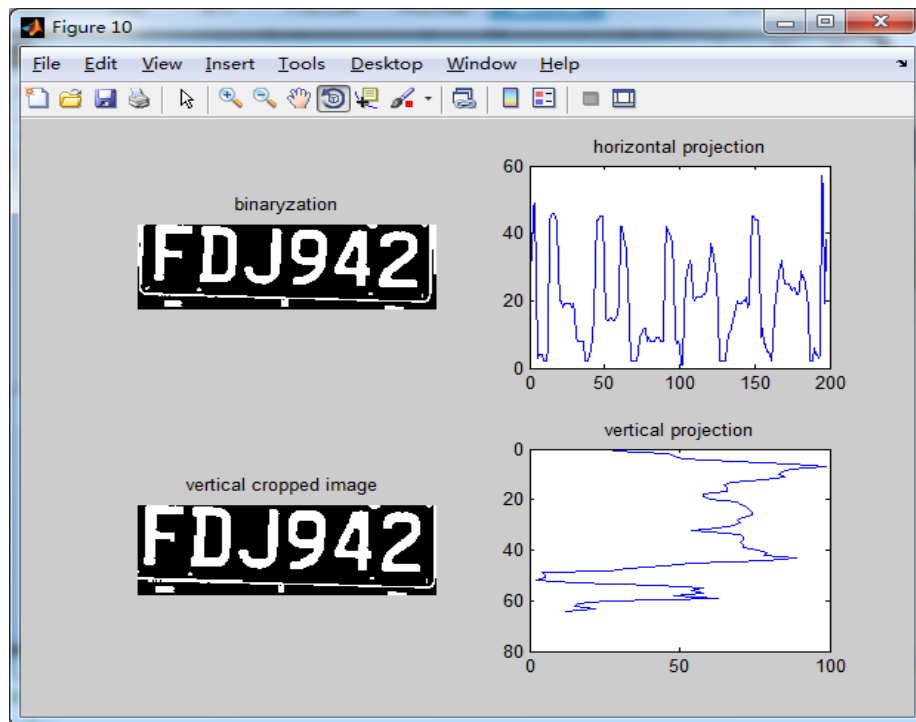
---

Set  $I$  as cropped vehicle image;  
 Convert  $I$  to gray level  $I_{gray}$ ;  
 $I_{gray} = \text{medfilt2}(I_{gray}, [6 \ 6])$ ;  
 Set  $BW\_image = \text{edge}(I_{gray}, 'sobel', 0.08, 'vertical')$ ;  
 Morphological operations were used to enhance plate number area( $BW\_image$ ) and remove noise;  
 Set  $stats = \text{regionprops}(BW\_image, 'BoundingBox')$ ;  
 Set  $plate\_number\_area = \text{imcrop}(I, stats.BoundingBox)$ ;  
 Set  $plate\_number\_binary = \text{im2bw}(plate\_number\_area)$ ;  
 Horizontal projection was used to specify the specific *left* and *right* of  $plate\_number\_binary$ ;  
 Vertical projection was used to specify the specific *top* and *bottom* of  $plate\_number\_binary$ ;  
 Plate number image =  $\text{crop}(I, [left \ top \ right - left \ top - bottom])$ ;

---



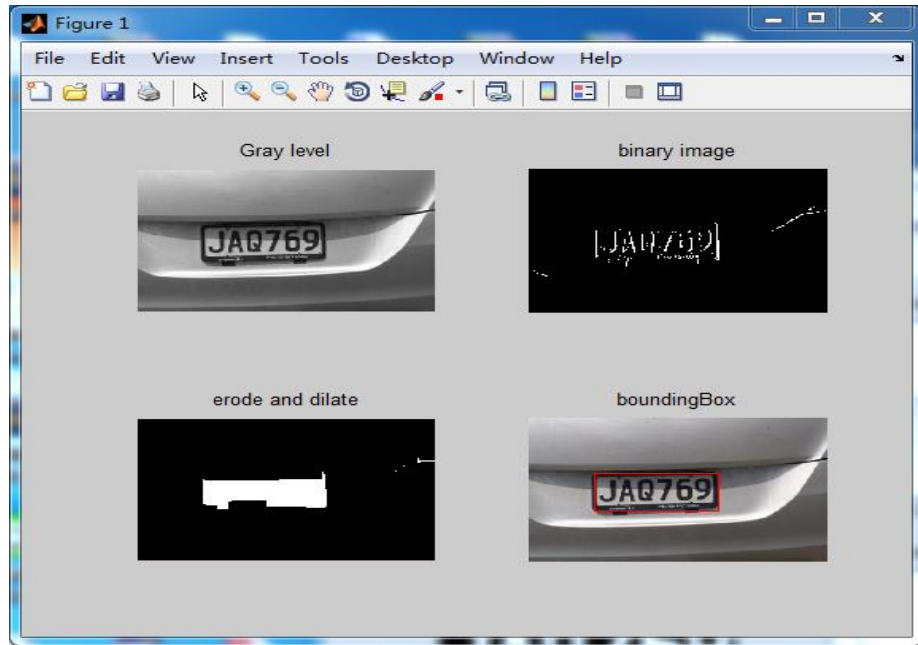
(a) Example 1: Rough Locating from The Second Scan



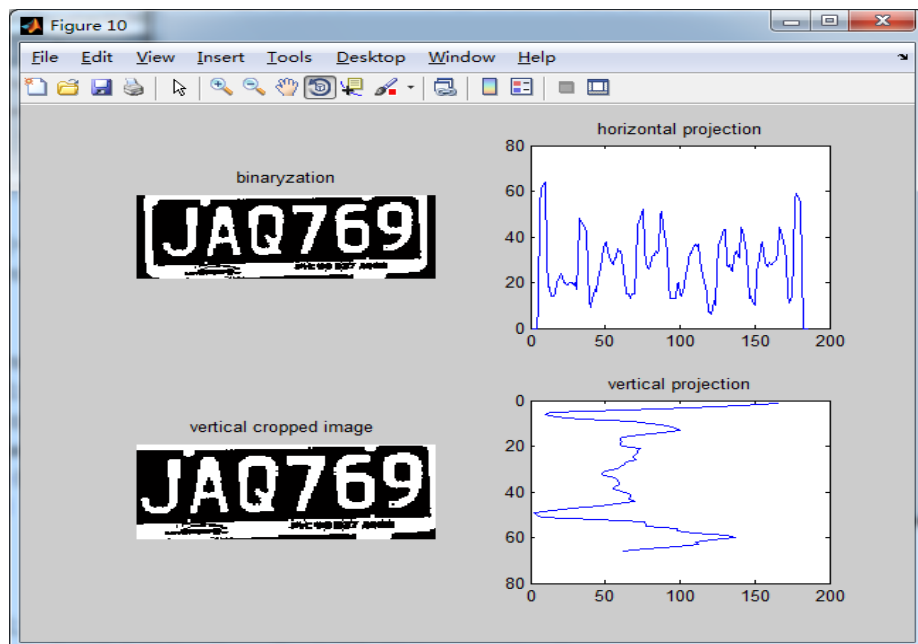
(b) Example1: Using Projection to Amend Plate Number Image



(c) Example1: Final Cropped Plate Number



(d) Example 2: Rough Locating from Second Scan



(e) Example2: Using Projection to Amend Plate Number Image



(f) Example 2:Final Cropped Plate Number

Figure 4.8 Results of The Second Scan of Plate Number

## 4.2.2 Character Segmentation

The cropped pictures that contain plate number area will be amended before the next process. The more extra space surrounding the characters in image is to be omitted, the better accuracy result will be achieved. After removing the noises in the image, image binarization will change the colour image to a binary one with white foreground and black background. Segmentation will be carried out by the vertical projection of images (Liu, Xiang, & Xu, 2013). The number of “valleys” appeared in the vertical projection can be used to identify how many characters in a plate number. As shown from Figure 4.9 (a) and 4.10 (a), five “valleys” exist between six adjacent characters. Algorithm 4.4 describes the steps of character segmentation.

---

### Algorithm 4.4 Character Segmentation

---

**Input:** Plate number image

---

**Output:** Segmented character image

---

*Plate\_number\_image*=imread('plate number image');

---



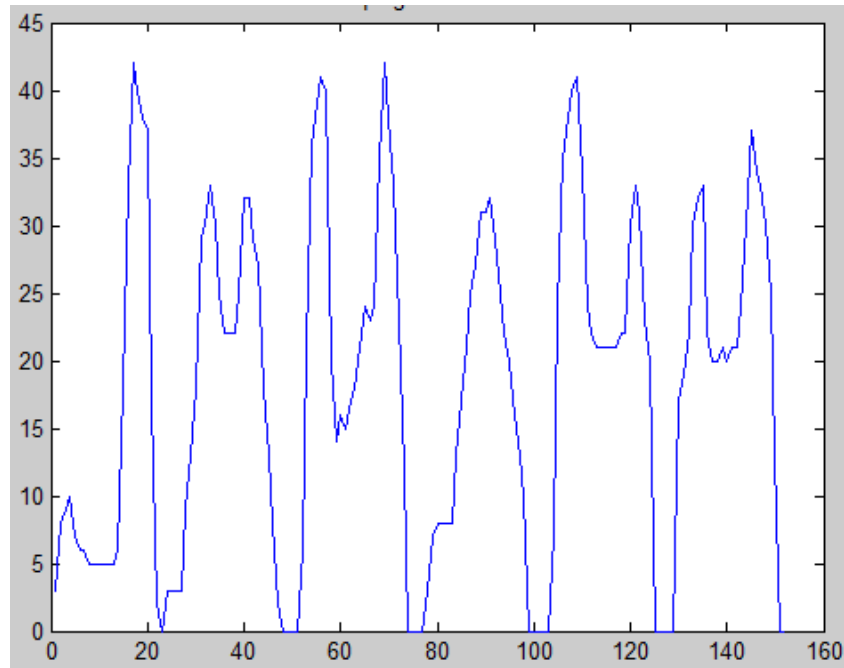
transfer *Plate\_number\_image* to binary image *BW\_image*;  
 morphological operations were used to remove noises;\n  
 the horizontal and vertical projection of filtered *BW\_image* were used to adjust the late  
 number area;

```

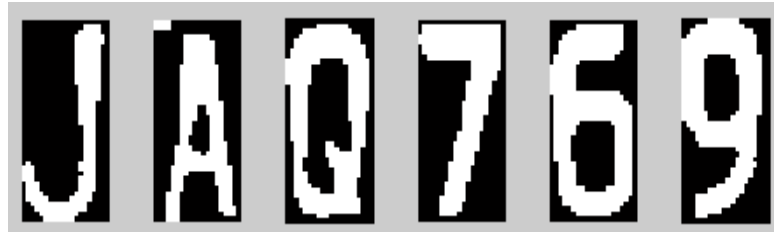
set cut_num = 5;
for count=1:cut_num do
  get the horizontal projection y1 from BW_image;
  [m,n]=size(BW_image);
  set the left as the left boundary of BW_image is 0;
  get the right from the minimums in min(y1(10:40));
  the width of the first character is obtained from equation: width=right-left;
  word=imcrop(BW_image,[0 0 width m]); // crop the first character from
  word=characterAmend(word); // remove the noise and redundant space
  imwrite(word, strcat(num2str(count),'.bmp')); // store the character in disk
  BW_image=imcrop(BW_image,[right 0 n-(right-left) m]);
  if (count==5) then
    word=image;
    word=characterAmend(word);
    Imwrite(word,strcat(num2str(num+1),'.bmp'));
  end if
end for

```

---

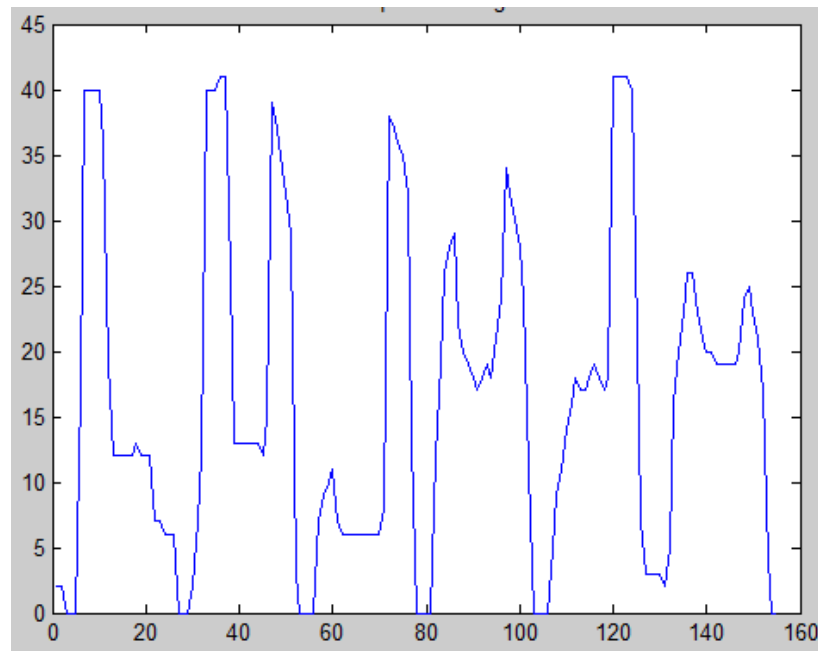


(a) Projection Result of the Second Scan of Plate Number



(b) The Cropped Characters

Figure 4.9 Results of Example 1



(a) Projection Result of the Second Scan of Plate Number



(b) The Cropped Characters

Figure 4.10 Results of Example 2

### 4.2.3 Character Recognition

Two character recognition methods, Template Matching and Neural Network, are implemented in this thesis. The correlation coefficient is used to measure the similarity

between the template image and testing image (Comelli, Ferragina, Granieri, & Stabile, 1995). The algorithm is designed as,

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \quad (4.4)$$

where  $\bar{A}$  and  $\bar{B}$  are the means of the matrices. The testing image compares with template images one by one, from number 0 to 9 and characters 'A' to 'Z'. After all the calculated correlation coefficients having been stored in an array, the highest value in the array will determine what result of the testing image is (See Figure 4.11).

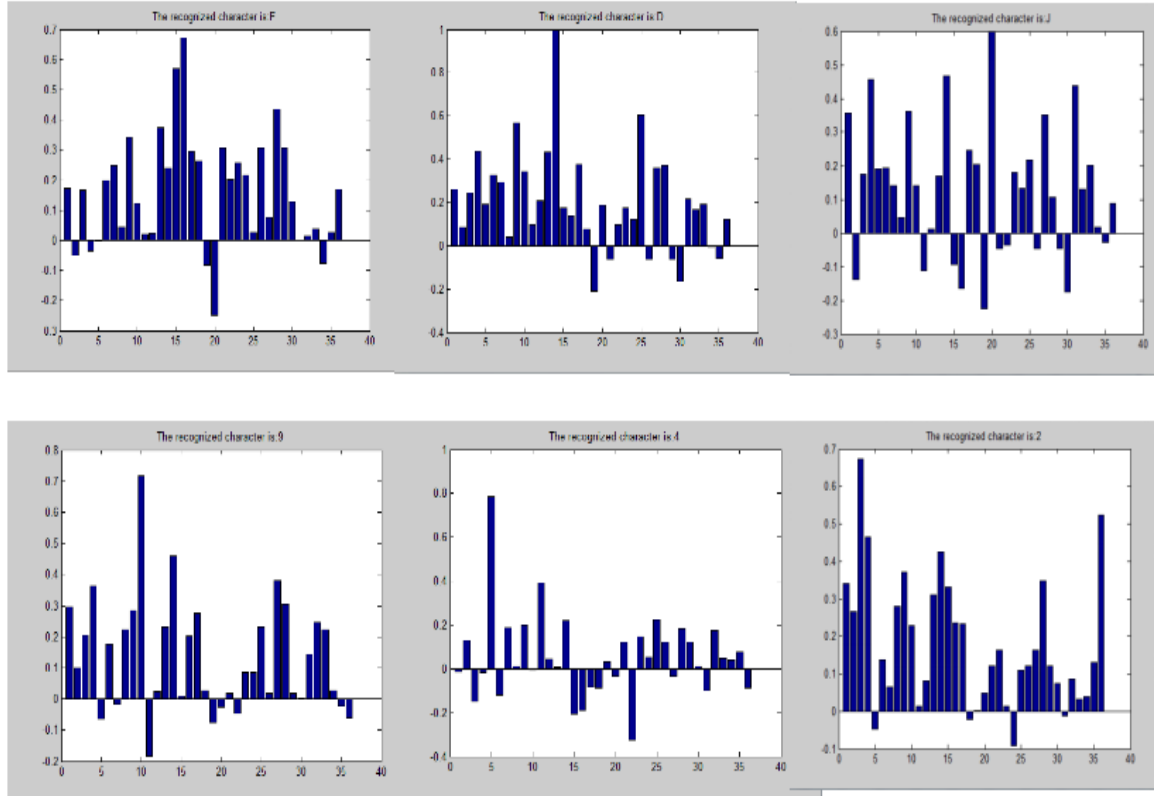


Figure 4.11 Results of Character Reorganization

Apart from using Correlation Coefficient to measure the similarity between testing image and template, the traditional criteria, pixels subtraction, is also utilized in our experiments.

We also improve the Template Matching by adding a character classification. The characters are divided into four categories according to their symmetry characteristic as

shown in Table 4.1.

Table 4.1 Categories of Plate Number Characters

| Categories | Symmetry                            | Letters                             |
|------------|-------------------------------------|-------------------------------------|
| Array_A    | Left vs Right and Top vs Bottom     | 0,1,2,5,8,9,B,D,H,I,M,N,O,Q,R,V,X,Z |
| Array_B    | Left vs Right and Top not vs Bottom | 7,A,T,U,W,Y                         |
| Array_C    | Left vs not Right and Top vs Bottom | 3,6,C,E,G,K,P,S                     |
| Array_D    | None                                | 4,F,J,L                             |

It is shown in Table 4.1, all the letters have been classified to four groups in accordance with their symmetry. The way to distinguish the category of particular letter is to analyze its vertical and horizontal projection. The letters have been normalized into same size 40\*20 before character recognition. The left and right symmetry can be determined by comparing the maximum values in font half (0 to 10) and rear half (10 to 20) of the projection line. Figure 4.12 shows the horizontal projection line of letter 'H'. The acquisition of top and bottom symmetry use the same method but using vertical projection line of the letter.

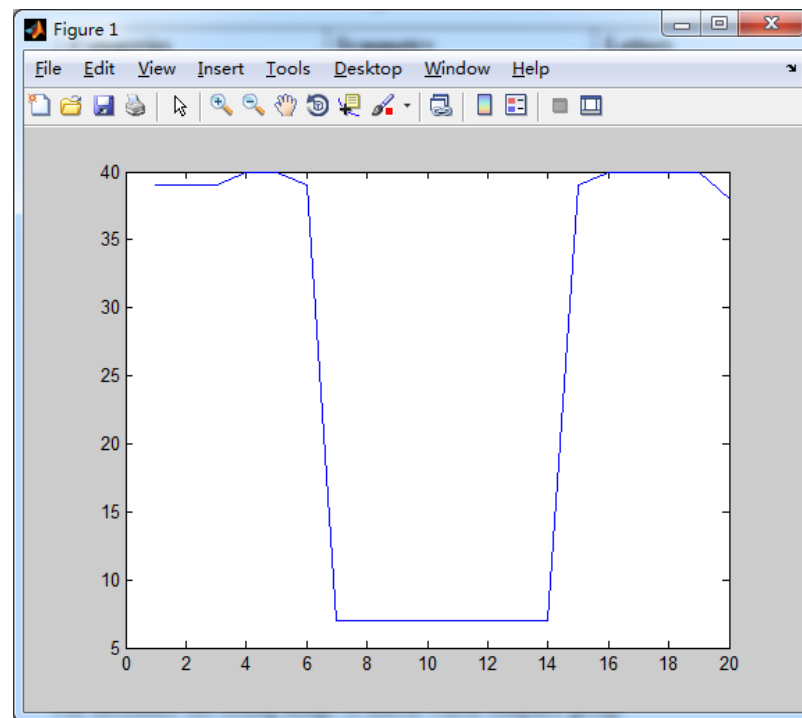


Figure 4.12 The Horizontal Projection Line of Letter 'H'

Unlike the traditional Template Matching method that all the templates are store in the same file. The templates from the improved Template Matching method will be stored in four different files with respect to their symmetry. Before using the testing character image to match the templates, the testing image will be verified its category. Then, the program will determine the testing image to match which template group. The traditional Template Matching will compare the testing image with all template images. However, in improved Template Matching method, the testing image only compare with the template images whose symmetry is same as the testing image. Consequently, the comparison times will be reduced.

Artificial Neural Network (C.-N. E. Anagnostopoulos et al. , 2008) as the second method is also tested in our experiment. The network should be well trained before character recognition. The procedure of creating a BP-neural network is comprised by preparing the input and output datasets, creating a BP-neural network and simulating the obtained neural network. A two-layer neural network with hidden and output neurons is used in our system. The parameters of our BP-neural network are defined as follows. We use 200 features extracted from each character by using Grid method, the number of neurons for input layer is 200. In the target matrix, we define the rows as the classes of the training data set. From 1 to 10, the value of row represents the plate number 0 to 9 and the rest of rows represent the English alphabets. When using BP-neural network to simulate a testing data, the value of row of one element will be found as 1 while the rest rows and columns are zeros. According to the result of BP-neural network, the test data will be recognized.

## **4.3 The Database Design**

In our traffic ticketing management system, the following step is to search the vehicle owner information in accordance with the registered plate number in database. This subsection will give a brief description about the database designed for this system.

The first consideration of establishing a database is to consider the requirement analysis. In our scenario, the primary entities such as owners of vehicle, vehicles and registration information should be stored in our database. The attributes of the owner entity are made by personal details such as registered driver license number, name, gender, e-mail address, mobile number .etc. Among these attributes in owner entity, the driver license is selected as the primary key, which is unique and can determine the vehicle owner. The vehicle entity may consist of plate number, the model of the car, the year built, the colour, status .etc. as its attributes. The primary key will be its plate number. The registration information should record the driver owning how many vehicles. The relationship between the owner entity and vehicle entity is one to many, which means the owner can possess more than one car.

According to the previous analysis, three tables are designed for our database. The tables are comprised by Driver table, Vehicle table and Event Registration table. Table 4.2, 4.3 and 4.4 show the details of the three tables.

Table 4.2 Driver Table in the Database






















| Column Name   | Datatype  | NOT NULL  | AUTO INC | Flags  |
|---|---|---|----------|--|
|  driver_license    |  VARCHAR(12) |  |          | <input type="checkbox"/> BINARY  |
|  name              |  VARCHAR(45) |  |          | <input type="checkbox"/> BINARY  |
|  sex               |  VARCHAR(6)  |  |          | <input type="checkbox"/> BINARY  |
|  age               |  INT(10)     |  |          | <input checked="" type="checkbox"/> UNSIGNED <input type="checkbox"/> ZEROFILL |
|  Email             |  VARCHAR(45) |  |          | <input type="checkbox"/> BINARY  |
|  phone             |  VARCHAR(45) |  |          | <input type="checkbox"/> BINARY  |
|  registration_date |  DATE        |  |          |  |

Table 4.3 Vehicle Table in the Database





























| Column Name  | Datatype  | NOT NULL                            | AUTO INC | Flags                           | Default Value  |
|--|---|-------------------------------------|----------|---------------------------------|--|
|  plate_number |  VARCHAR(12) | <input checked="" type="checkbox"/> |          | <input type="checkbox"/> BINARY |  NULL |
|  model        |  VARCHAR(45) | <input checked="" type="checkbox"/> |          | <input type="checkbox"/> BINARY |  NULL |
|  year         |  DATE        | <input checked="" type="checkbox"/> |          |                                 |  NULL |
|  color        |  VARCHAR(45) | <input checked="" type="checkbox"/> |          | <input type="checkbox"/> BINARY |  NULL |
|  status       |  VARCHAR(45) | <input checked="" type="checkbox"/> |          | <input type="checkbox"/> BINARY | 'normal'   |
|  wof          |  VARCHAR(45) | <input checked="" type="checkbox"/> |          | <input type="checkbox"/> BINARY | 'pass'   |

Table 4.4 Violation Event Table in the Database

| Column Name   | Datatype  | NOT NULL                            | AUTO INC                            | Flags  |
|---|---|-------------------------------------|-------------------------------------|--|
|  id              |  INT(10)       | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> UNSIGNED |
|  plate_number    |  VARCHAR(12)   | <input checked="" type="checkbox"/> |                                     | <input type="checkbox"/> BINARY              |
|  driver_license |  VARCHAR(12)  | <input checked="" type="checkbox"/> |                                     | <input type="checkbox"/> BINARY              |
|  place         |  VARCHAR(45) | <input checked="" type="checkbox"/> |                                     | <input type="checkbox"/> BINARY              |
|  time          |  DATETIME    | <input checked="" type="checkbox"/> |                                     |  |
|  event         |  VARCHAR(45) | <input checked="" type="checkbox"/> |                                     | <input type="checkbox"/> BINARY              |

The functions of database connection and data operation have been integrated in MATLAB. Three ways to connect a database are provided in MATLAB. In this thesis, a JDBC driver is used for database connectivity. The syntax of database connection is defined as:

*Conn*=database (*instance*, *username*, *password*, *driver*, *databaseurl*);

where the syntax returns a database connection object *Conn*. *Instance* is the name of database need to be accessed. *username* and *password* are the logging user's name and password. *driver* means the name of JDBC driver provided by database company. For example, the MYSQL database is adopted in this thesis project, so the diver should be "com.mysql.jdbc.Driver". The last parameter is *databaseurl* that refers to the details of connected database. Take our database connection in this thesis as an example, the

format should be,

```
Conn=database('trafficTicketManagement','root','root','com.mysql.jdbc.Driver','jdbc:
mysql:localhost:3306/trafficTicketManagement');
```

The data query and acquisition are implemented in MATLAB by using the commands “*cursor=exec(conn,sqlStatments)*” and “*fetch(cursor).Data*” to get the data.

## 4.4 Information Notification

The final step of this thesis project is information notification. The violator’s email address can be retrieved from database by the recognized plate number. Sending e-mail is not hard in MATLAB. There is a built-in function called *sendmail* in MATLAB. The format of *sendmail* is that,

sendmail (TO, SUBJECT, MESSAGE, ATTACHMENTS)

where, TO is a string specifying a email address where the email need to be sent to.

SUBJECT is the title of email. The parameters MESSAGE and ATTACHMENTS

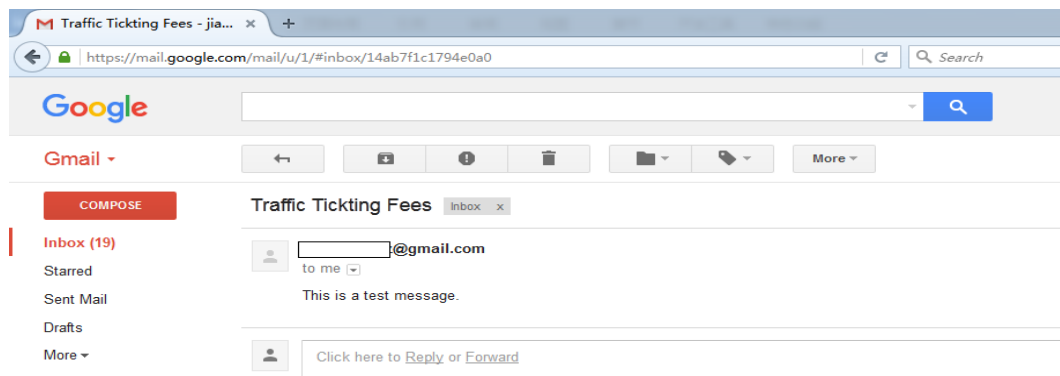
provide the traffic fines information sent to violators. Before using the function

sendmail, two preferences “Internet: SMTP\_Server” and “Internet: E\_mail” should be specifying initially. Figure 4.13 shows the results of information notification.

```
myaddress = [ ]@gmail.com';
mypassword = [ ]';
setpref('Internet','E_mail',myaddress);
setpref('Internet','SMTP_Server','smtp.gmail.com');
setpref('Internet','SMTP_Username',myaddress);
setpref('Internet','SMTP_Password',mypassword);
props = java.lang.System.getProperties;props.setProperty('mail.smtp.auth','true');
props.setProperty('mail.smtp.socketFactory.class',...
    'javax.net.ssl.SSLSocketFactory');props.setProperty('mail.smtp.socketFactory.port','465');
sendmail(Email_address, 'Traffic Tickting Fees', 'This is a test message.');
```

(a) Implementation Code





(b) Result of Sending Email

Figure 4.13 Results of Information Notification

## Chapter 5

### Experimental Results and Discussion

*In this chapter, discussion and analysis with respect to the outcomes from experiments are presented. Specifically, the comparisons of moving tracking algorithms between Kanade-Lucas-Tomasi (KLT) and Continuously Adaptive Mean Shift (CAMShift) algorithm are conducted in plate tracking section. Edge-based and the proposed plate number locating methods are also compared in this chapter. The outcomes from character recognition methods implemented by template matching and BP neural network are presented. Finally, some limitations are also concluded through the analysis of the outcomes of the experiments.*

## **5.1 Introduction**

The previous chapters have introduced and concluded the numerous approaches and algorithm. This chapter will show the results from our experiments. All the experiments are conducted on an Intel Core i7-3537U CPU 2.50 GHZ laptop using MATLAB R2013a.

## **5.2 Event Detection Module**

### **5.2.1 Event Detection**

Figure 5.1 shows the results of detecting the lines from images. 16 out of 20 images are detected correctly. Figure 5.2 illustrates the results of event detection module. The stop line is detected by Hough transform before moving vehicles detection and alarm making. The location of a stop line is detected for the event triggering purpose. Connected components analysis was used to detect the moving vehicle in the current footage. The condition for triggering an event is the rectangle enclosed a car passing the stop line. We conduct our experiments and detect the cars driving through the line correctly, the precision is around 79% as shown in Table 5.1.

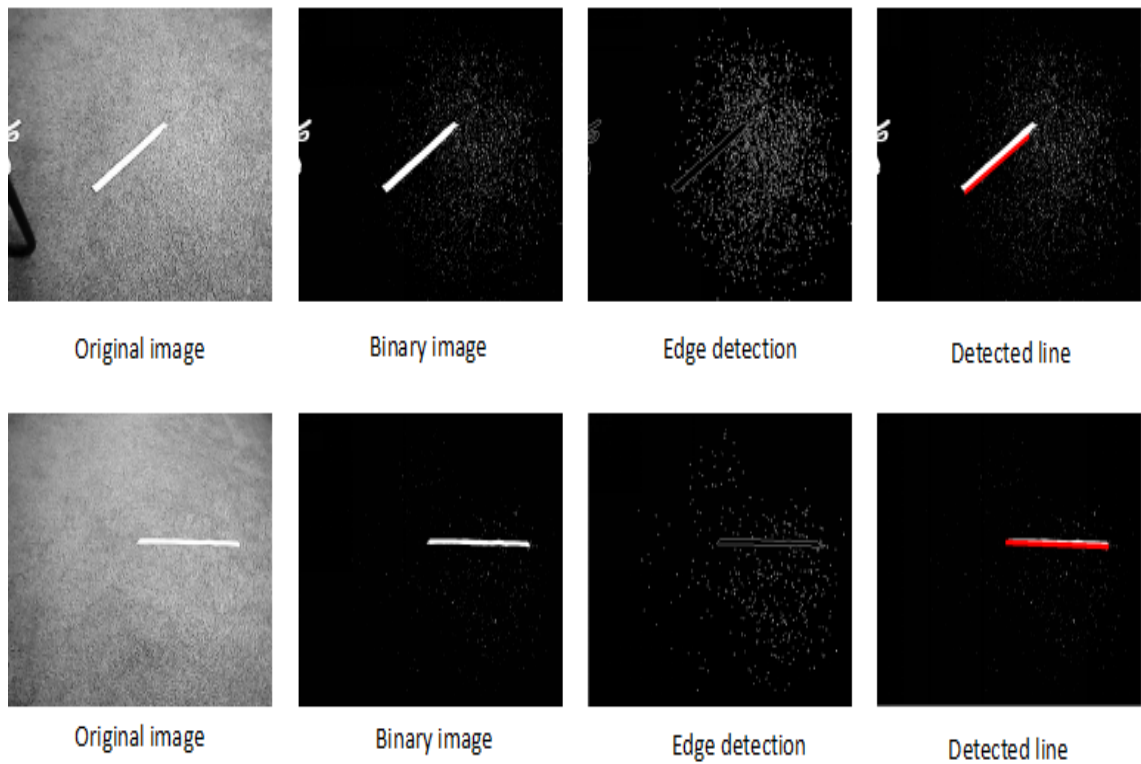


Figure 5.1 Results of Straight Line Detection

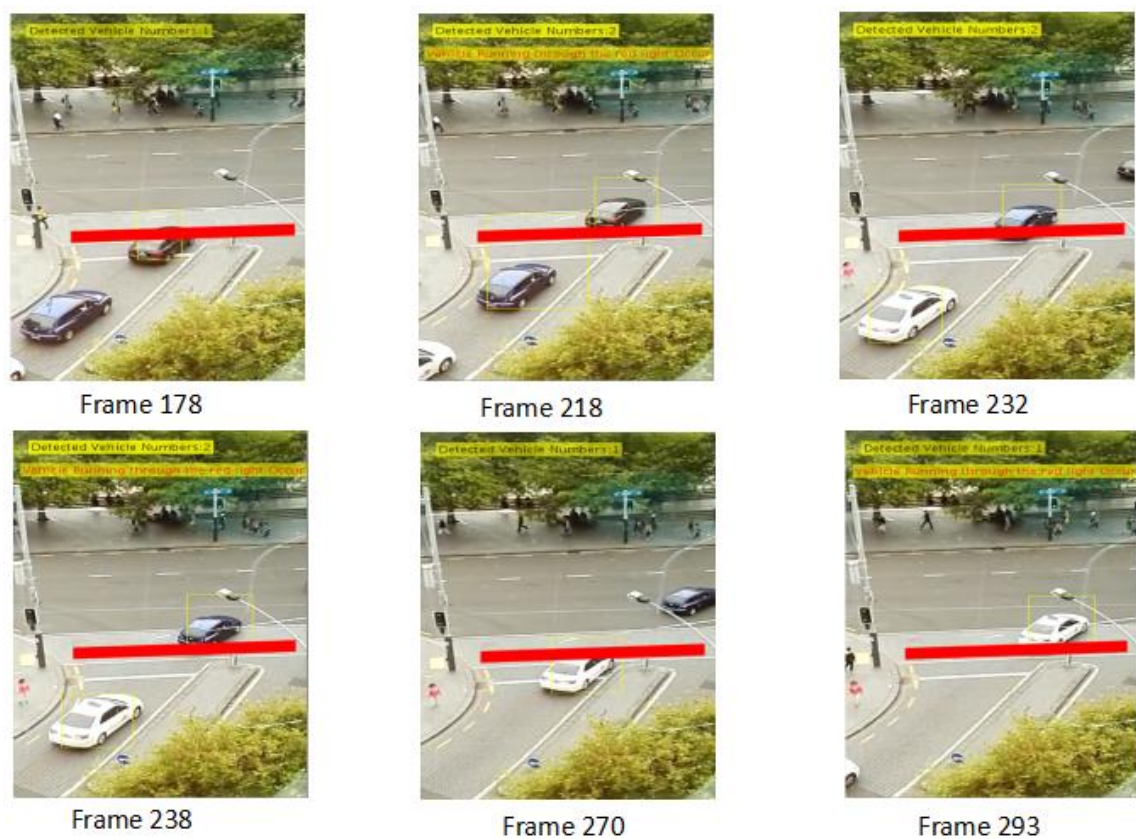


Figure 5.2 Results of Stop Line Detection and Event Detection

Table 5.1 Results of Foreground Detection

|                 | Car 1 | Car 2 | Car 3  |
|-----------------|-------|-------|--------|
| Detected Frames | 85    | 115   | 130    |
| Ground Truth    | 110   | 145   | 160    |
| Precision       | 77.3% | 79.3% | 81.25% |

### 5.2.2 Plate Tracking

Table 5.2 Results of Plate Number Tracking

| Algorithms | Features       | Frame Numbers | Correct Numbers | Accuracy | Feature Extraction Time (Approximately) |
|------------|----------------|---------------|-----------------|----------|---|
| KLT        | Harris         | 20            | 19              | 95%      | 0.2452s                                 |
| KLT        | MinEign        | 20            | 18              | 90%      | 1.1338s                                 |
| KLT        | FAST           | 20            | 20              | 100%     | 0.1183s                                 |
| CAMshift   | Gray-Histogram | 20            | 10              | 50%      | 3.8321s                                 |

Table 5.2 presents the results of plate tracking in event detection module. The object tracking algorithms, KLT and CAMshift, are compared in this thesis. In terms of performance accuracy, KLT is better than CAMshift. The accuracy of KLT ranges from 90% to 100%, which is quite higher than 50% from CAMshift. The first 20 frames from the testing video are used to conduct this experiment. The plate number from the first 20 frames is still readable and recognizable. Although the KLT algorithm tracking the plate number in the rest of video is correct, the latter frames are not taken into account in our experiments. Figure 5.3 and Figure 5.4 provide the examples of plate number tracking. Figure 5.3 illustrates the result of tracking in original video frames. In order to demonstrate the details of tracking, only the true tracking area as the plate number is

shown in Figure 5.4.



Figure 5.3 An Example of Plate Tracking

The event of running through the red light can be found from the trajectory of tracking plate. We define a “stop line” in the frames to simulate the real-world testing as shown in Figure 5.5. When an event happens, it can be detected if the car drives across the line.

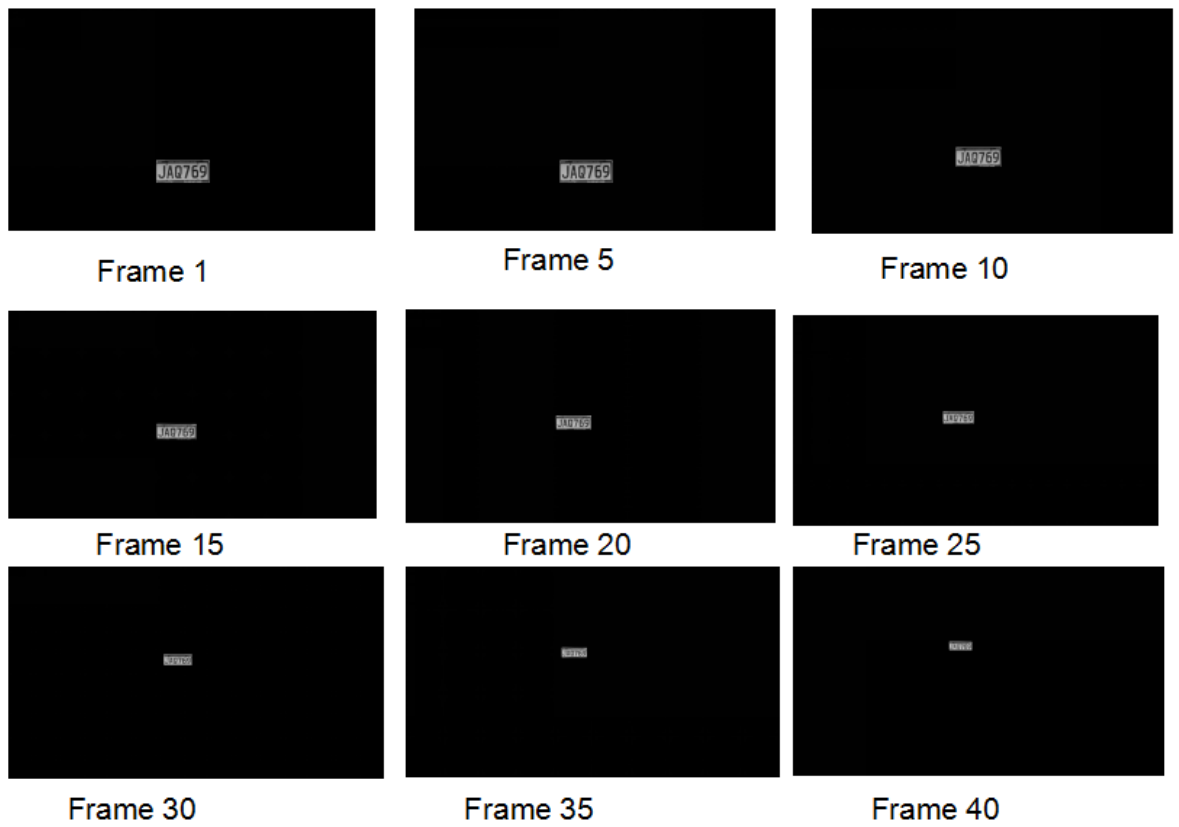


Figure 5.4 The Results of Plate Tracking Removing the Background

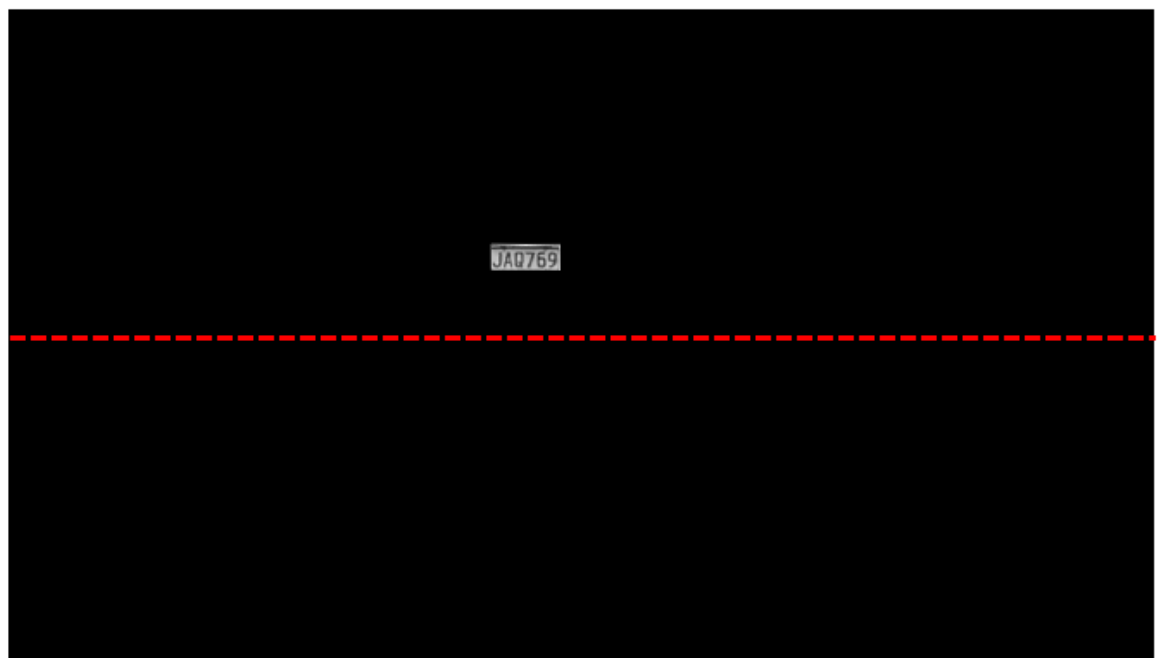


Figure 5.5 A Plate Number Goes Through the Red Traffic Light



## 5.3 Plate Number Recognition Module

### 5.3.1 Plate Number Localization

Table 5.3 Results for Edge Detection and the Proposed Method

| Methods                 | Successes | Accuracy |
|-------------------------|-----------|----------|
| Vertical Edge Detection | 75        | 62.5%    |
| The Proposed Method     | 91        | 75.8%    |

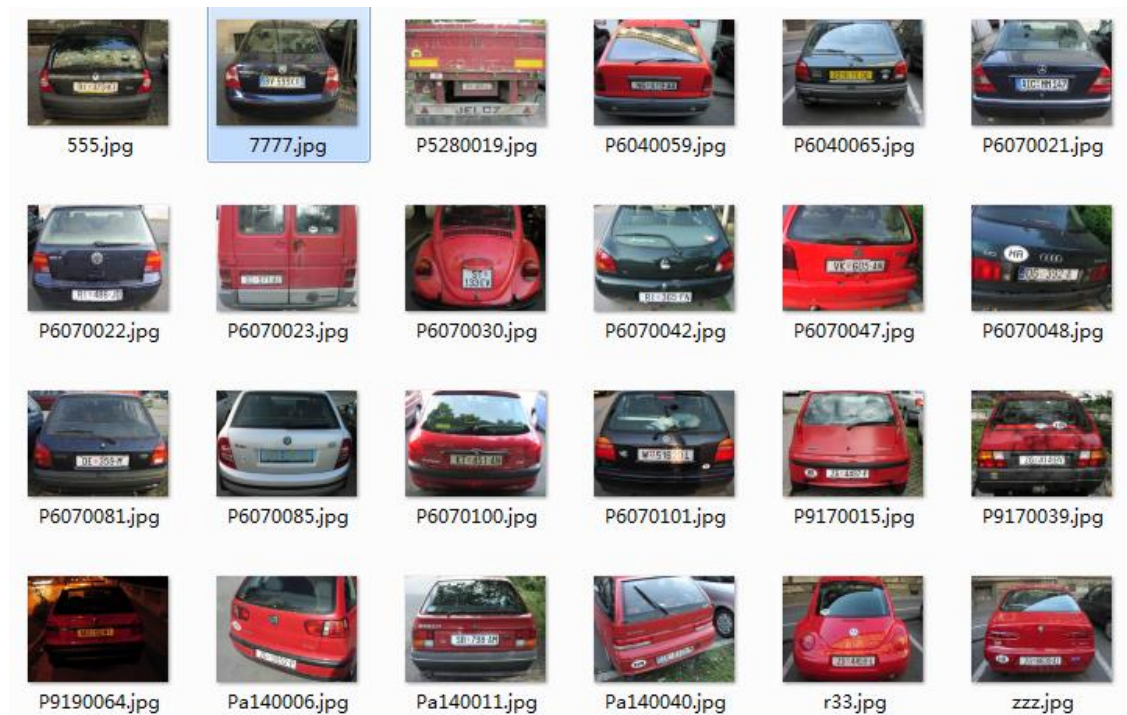


Figure 5.6 Samples from Website: <http://www.zemris.fer.hr/>

Table 5.3 presents the information regarding to plate number locating techniques. Vertical edge detection and our proposed method SP are compared. 91 testing images out of a total 120 images are localized correctly by our proposed method, presenting a 75.8% accuracy. Among the 29 cases of failure plate number extraction, 12 cases failed at the first scan. Figure 5.6 demonstrates some challenged situations of testing images.



The proposed method performs better than vertical edge detection when the input image has character disturbance in car body. Vertical edge detection method is insensitive to images that have a massive of vertical edge information. Due to the proposed method extract plate number from two red taillights, it can have a better result in the situation that images contain complicated edge information.

### 5.3.2 Character Recognition

Table 5.4 Results from Character Recognition

| <b>Techniques</b> | <b>Methods/Features</b> | <b>Successes</b> | <b>Accuracy</b> |
|-------------------|-------------------------|------------------|-----------------|
| Template Matching | Correction Coefficient  | 58               | 72.5%           |
| Template Matching | Pixels Subtraction      | 44               | 55%             |
| Neural Network    | Pixel Counting          | 60               | 75%             |

Results from this thesis for character recognition are shown in Table 5.4. A total of 80 images are tested in our thesis project by Template Matching and BP Neural Network with different parameters or features. The similarity measurements used in TM are correction coefficient and pixels subtraction. Pixel counting features are extracted from character as training data in the process of neural network creation. Accuracy from both techniques reach over 70%. A comparison with the Template Matching (72.5%) used in character recognition reveals that Neural Network has a better accuracy (75%). TM using correction coefficient has a higher success rate than pixels subtraction. 58 plate number images are recognized correctly without any loss by correction coefficient and 44 images are successfully recognized by pixels subtraction.

## 5.4 Discussions

Experiments and results were demonstrated previously in this Chapter. We firstly conducted the experiments related to event detection module. Detecting “run across the stop line” event was verified in Section 5.2.1. In order to tackle the problem that how to detect the stop line from footage rather than predefining the line at the road, Hough transform was used to detect the line automatically. We extract the foreground and make the alarm to trigger the next module. Comparing with the specifying the stop line in the frame, detecting lines using Hough transform is more practical. However, another considerable issue is that the first frame used to detect the line should be clear, which means the stop line cannot be covered by objects such as vehicles or pedestrians. Frame differencing has a better resistance than Background Subtraction when the videos contain motion. Section 5.2.2 shows the results of plate tracking using different method. We only made a comparison about the performance time of different feature extraction among KLT and CAMshift. The experiments related to plate number recognition module was illustrated in Section 5.3. This section is made by two subsection 5.3.1 and 5.3.2. The former subsection compares the proposed method and edge detection method in plate number localization. Our proposed method has a better accuracy than the edge based method. However, edge based method has a slight faster processing speed than our proposed method, which is due to our proposed method consists of more procedures to locate the plate area. The final subsection gives the information involved with the plate number recognition. Template Matching using Correction Coefficient has a significant advantage in recognition result than the pixel subtraction method. That because the Correction Coefficient is used to justify the similarity between two matrixes (the image stored format in Matlab is matrix). If the structures of character have a higher similarity, the recognition using Correction Coefficient will fail.

## Chapter 6

### Conclusion and Future Work

*In this thesis, we successfully proposed and discussed the framework of an event-driven traffic ticketing system. The corresponding approaches to each module in this system have been implemented and verified in this thesis. In this chapter, we conclude the whole thesis and recommend the future work.*

## 6.1 Conclusion

In the first chapter, we firstly introduced the significant role of video event detection in our society and the corresponding computer techniques. The motivation and structure were also detailed respectively in this chapter. As our goal was to study the computer vision technique and develop an event detection application, a novel schema of event-driven traffic ticketing system was proposed in this thesis. The whole system consists of four primary modules, event detection module, plate number recognition module, database management module and traffic tickets notification module. Followed by the structure created in first chapter, the second chapter conducted a detailed literature review. The-state-of-the-art of moving object detection and license plate recognition methods were summarized in this Chapter. In addition, the considered techniques and algorithms were evaluated for achieving the objectives of this thesis. After reviewing multiple papers, the research question and methodology were present in Chapter 3. The specific experiment design and implementation were introduced in Chapter 4. This chapter was made up of two important modules, event detection and plate number recognition, in the entire system. In event detection module, foreground detection method was used to detect the passing vehicles. Then, a combination of points features extraction and KLP was used to track the captured vehicle. After analyzing whether the trajectory of moving vehicle meets the probability of a predefined event occurrence, the state signal will be sent to trigger the plate number recognition module. In the module, a novel dual scan method (Secondary Positioning) was introduced to localize the plate number area. A rough area contains plate number will be cropped by the first scan, which search the red pixels in HSV colour space. The specific plate number was extracted by the horizontal and vertical projection of its edge information in the second scan. Then, the character recognition processes will be conducted by template matching. The system will send traffic ticket notification information to the owner of the plate number registered in the database of the system. The other two

modules are implemented in MYSQL database and MATLAB. Chapter 5 showed our experimental results and discussion about the involved methods.

Additionally, this thesis analyzes the core algorithms and approaches which are suitable for this system. Due to the importance and influence of the plate number recognition module, this thesis focuses on analyzing computer vision techniques with respect to implement plate number recognition. The given methods and algorithms in this thesis answer the research question of this thesis. Further, this thesis also improves the traditional Template Matching method used in character recognition. Specifically, the character templates are grouped into four classes according to the symmetric characteristic of characters. Results show the improved method has a better performance than traditional one.

## **6.2 Limitations and Future Work**

Although the whole event traffic ticketing management system has been implemented successfully, there are some limitations should be improved in future. They are,

- (1) We used smart phone built-in cameras to capture experimental videos and images instead of using professional CCTV devices.
- (2) In plate number recognition module, only the plates that contain six characters are considered in our experiment. However, the plate numbers in real life may contain different numbers of characters. Plate numbers which less than six characters cannot obtain a good character segmentation result from our proposed system.
- (3) The proposed character template classification method limits the input testing image. All the testing images should be modified to fulfill the criterion of character classification, which leads to a bad evenness.
- (4) The image rotating problem was not considered in this thesis. In our present project, we only stressed on important theories in computer vision that were applied to this proposed system therefore a few aspects cannot be considered

seriously.

Some suggestions for the future work are as follows:

(1) In future, we should take the vehicle images from both front and back sides into account instead of only considering the plate number existing on back of the vehicles. No matter what the plate number appears in image, the program will recognize them fast and correct.

(2) The training data set used for neural network should be expanded. The future training data set should be expanded to 100 different formats of images for each character at least.

(3) The features extracted from individual segmented character image should be more robust to factors such as rotations, illuminations and variances, etc.

# References

- Ahmed, M. J., Sarfraz, M., Zidouri, A., & Al-Khatib, W. G. (2003). *License plate recognition system*. IEEE International Conference on Electronics, Circuits and Systems, 898-901, doi: 10.1109/ICECS.2003.1301932.
- Aköz, Ö., & Karşlıgil, M. E. (2010). *Video-based traffic accident analysis at intersections using partial vehicle trajectories*. Signal Processing and Communications Applications Conference (SIU), 499-502, doi:10.1109/SIU.2010.5651544
- Anagnostopoulos, C. N. E., Anagnostopoulos, I. E., Loumos, V., & Kayafas, E. (2006). A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems*, 7(3), 377-392.
- Anagnostopoulos, C.-N. E., Anagnostopoulos, I. E., Psoroulas, I. D., Loumos, V., & Kayafas, E. (2008). *License plate recognition from still images and video sequences: A survey*. Intelligent Transportation Systems, IEEE Transactions on, 9(3), 377-391.
- Bailey, D. G., Irecki, D., Lim, B., & Yang, L. (2002). *Test bed for number plate recognition applications*. doi:http://doi.ieeecomputersociety.org/10.1109/DELTA.2002.994684.
- Barron, J. L., Fleet, D. J., & Beauchemin, S. S. (1994). Performance of optical flow techniques. *International journal of computer vision*, 12(1), 43-77.
- Brunelli, R.(2009). *Template Matching Techniques in Computer Vision: Theory and Practice*. John Wiley & Sons, Ltd. doi:10.1002/9780470744055.fmatter
- Bunyak, F. (2005). *Moving object detection and tracking for event-based video analysis*. University of Missouri/Rolla.
- Burges, C. J. (1998). *A tutorial on support vector machines for pattern recognition*. Data mining and knowledge discovery, 2(2), 121-167.
- Castellanos, R., Kalva, H., Marques, O., & Furht, B. (2010). *Event detection in video*

- using motion analysis*. Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams, 57-62, doi:10.1145/1877868.1877884
- Castello, P., Coelho, C., Del Ninno, E., Ottaviani, E., & Zanini, M. (1999). *Traffic monitoring in motorways by real-time number plate recognition*. Proceedings of International Conference on Image Analysis and Processing, 1128-1131, doi: 10.1109/ICIAP.1999.797753.
- Chang, S.-L., Chen, L.-S., Chung, Y.-C., & Chen, S.-W. (2004). *Automatic license plate recognition*. IEEE Transactions on Intelligent Transportation Systems, 5(1), 42-53.
- Chen, H., Ren, J., Tan, H., & Wang, J. (2007). *A novel method for license plate localization*. Fourth International Conference on Image and Graphics, 604-609, doi: 10.1109/ICIG.2007.175.
- Chong, J., Tianhua, C., & Linhao, J. (2013). *License Plate Recognition Based on Edge Detection Algorithm*. Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 395-398, doi: 10.1109/IIH-MSP.2013.105.
- Comaniciu, D., & Meer, P. (1999). *Mean shift analysis and applications*. Proceedings of the Seventh IEEE International Conference on Computer Vision, 1197-1203, doi: 10.1109/ICCV.1999.790416.
- Comaniciu, D., Ramesh, V., & Meer, P. (2003). *Kernel-based object tracking*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5), 564-577.
- Comelli, P., Ferragina, P., Granieri, M. N., & Stabile, F. (1995). *Optical recognition of motor vehicle license plates*. IEEE Transactions on Vehicular Technology, 44(4), 790-799.
- Cui, L., Li, K., Chen, J., & Li, Z. (2011). *Abnormal event detection in traffic video surveillance based on local features*. International Congress on Image and Signal Processing (CISP), 362-366, doi: 10.1109/CISP.2011.6099933.
- Cui, Y., & Huang, Q. (1997). *Automatic license extraction from moving vehicles*. Proceedings of International Conference on Image Processing, 126-129,



doi:10.1109/ICIP.1997.632014.

- Dedeoglu, Y. (2004). *Moving object detection, tracking and classification for smart video surveillance.*( Master's thesis, Bilkent University). Retrieved from <http://www.academia.edu/download/30911531/10.1.1.59.2055.pdf>
- Di Stefano, L., Mattoccia, S., & Tombari, F. (2005). *ZNCC-based template matching using bounded partial correlation.* Pattern recognition letters, 26(14), 2129-2134.
- Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2013). *Automatic license plate recognition (ALPR): A state-of-the-art review.* IEEE Transactions on Circuits and Systems for Video Technology, 23(2), 311-325.
- Duan, T. D., Du, T. H., Phuoc, T. V., & Hoang, N. V. (2005). *Building an automatic vehicle license plate recognition system.* Paper presented at the Proc. Int. Conf. Comput. Sci. RIVF, 59-63. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.101.8984&rep=rep1&type=pdf>.
- Elhabian, S. Y., El-Sayed, K. M., & Ahmed, S. H. (2008). *Moving object detection in spatial domain using background removal techniques-state-of-art.* Recent patents on computer science, 1(1), 32-54.
- Enyedi, B., Konyha, L., Szombathy, C., & Fazekas, K. (2004). *Strategies for fast license plate number localization.* Proceedings of 46th International Symposium on Electronics in Marine, 579-584.
- Fahmy, M. M. (1994). *Automatic number-plate recognition: neural network approach.* Proceedings of Vehicle Navigation and Information Systems Conference, 99-101, doi: 10.1109/VNIS.1994.396858.
- Gao, Q., Wang, X., & Xie, G. (2007). *License plate recognition based on prior knowledge.* IEEE International Conference on Automation and Logistics, 2964-2968, doi: 10.1109/ICAL.2007.4339089.
- Haralick, R. M., Sternberg, S. R., & Zhuang, X. (1987). *Image analysis using mathematical morphology.* IEEE Transactions on Pattern Analysis and Machine Intelligence, (4), 532-550.
- Horn, B. K., & Schunck, B. G. (1981). *Determining optical flow.* Technical symposium

east, 319-331, doi:10.1117/12.965761

- Horprasert, T., Harwood, D., & Davis, L. S. (1999). *A statistical approach for real-time robust background subtraction and shadow detection*. IEEE ICCV, 1-19.  
Retrieved from <http://vast.uccs.edu/~tboult/frame/Horprasert/>
- Huang, S., & Hong, J. (2011). *Moving object tracking system based on camshift and Kalman filter*. International Conference on Consumer Electronics, Communications and Networks (CECNet), 1423-1426, doi: 10.1109/CECNET.2011.5769081
- Jiang, F. (2012). *Anomalous Event Detection From Surveillance Video*.
- Jimenez, R. A. C. (2010). *Event detection in surveillance video*. Florida Atlantic University. ( Master's thesis, Florida Atlantic University, Boca Raton, Florida).  
Retrieved from [http://fau.digital.flvc.org/islandora/object/fau%3A2924/datastream/OBJ/download/Event\\_detection\\_in\\_surveillance\\_video.pdf](http://fau.digital.flvc.org/islandora/object/fau%3A2924/datastream/OBJ/download/Event_detection_in_surveillance_video.pdf)
- Joshi, K. A., & Thakore, D. G. (2012). *A survey on moving object detection and tracking in video surveillance system*. IJSCE, ISSN, 2231-2307.
- Karasulu, B. (2010). *Review and evaluation of well-known methods for moving object detection and tracking in videos*. Journal of aeronautics and space technologies, 4(4), 11-22.
- Karnin, E. D. (1990). *A simple procedure for pruning back-propagation trained neural networks*. IEEE Transactions on Neural Networks, 1(2), 239-242.
- Ke, Y., Sukthankar, R., & Hebert, M. (2010). *Volumetric features for video event detection*. International journal of computer vision, 88(3), 339-362.
- Kim, S., Kim, D., Ryu, Y., & Kim, G. (2002). *A robust license-plate extraction method under complex image conditions*. Paper presented at the Proceedings of 16th International Conference on Pattern Recognition, 216-219, doi:10.1109/ICPR.2002.1047833.
- Klubsuwan, K., Koodtalang, W., & Mungsing, S. (2013). *Traffic Violation Detection Using Multiple Trajectories Evaluation of Vehicles*. International Conference on Intelligent Systems Modelling & Simulation (ISMS), 220-224, doi: 10.1109/ISMS.2013.143.

- Kulchandani, J. S., & Dangarwala, K. J. (2015). *Moving object detection: Review of recent research trends*. Paper presented at the 2015 International Conference on Pervasive Computing (ICPC), 1-5, doi:10.1109/PERVASIVE.2015.7087138.
- Kumar, P., Vaidehi, V., & Chandralekha, E. (2013). *Video traffic analysis for abnormal event detection using frequent item set mining*. International Conference on Recent Trends in Information Technology (ICRTIT), 551-556, doi: 10.1109/ICRTIT.2013.6844262.
- Kwaśnicka, H., & Wawrzyniak, B. (2002). *License plate localization and recognition in camera pictures*. Symposium on Methods of Artificial Intelligence, 243-246. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.104.2678&rep=rep1&type=pdf>.
- Lan, J., Guo, M., Liu, X., Sun, X., Aibibu, T., & Ran, B. (2013). *Real-time detection algorithm for moving vehicles in dynamic traffic environment*. IEEE International Conference on Electro/Information Technology (EIT), 1-6, doi: 10.1109/EIT.2013.6632662.
- Lee, E. R., Kim, P. K., & Kim, H. J. (1994). *Automatic recognition of a car license plate using colour image processing*. IEEE International Conference on Image Processing, 301-305, doi: 10.1109/ICIP.1994.413580.
- Lee, J. S., Haralick, R. M., & Shapiro, L. G. (1987). *Morphologic edge detection*. Robotics and Automation, IEEE Journal of, 3(2), 142-156.
- Lee, K. M., & Kwon, J. (2012). *A unified framework for event summarization and rare event detection*. Paper presented at the 2012 IEEE Conference on Computer Vision and Pattern Recognition, 1266-1273, doi:<http://doi.ieeecomputersociety.org/10.1109/CVPR.2012.6247810>.
- Li, W., Chen, S., & Wang, H. (2009). *A rule-based sports video event detection method*. International Conference on Computational Intelligence and Software Engineering, 2009.
- Lin, C.-C., & Huang, W.-H. (2007). *Locating license plate based on edge features of intensity and saturation subimages*. Second International Conference on Innovative Computing, Information and Control, 227-227, doi:

10.1109/ICICIC.2007.372.

- Lipton, A. J., Fujiyoshi, H., & Patil, R. S. (1998). *Moving target classification and tracking from real-time video*. Fourth IEEE Workshop on Applications of Computer Vision, 8-14, doi: 10.1109/ACV.1998.732851.
- Liu, C., Hu, C., Liu, Q., & Aggarwal, J. (2013). *Video event description in scene context*. Neurocomputing, 119, 82-93.
- Liu, J., Xiang, J., & Xu, C. L. (2013). *Plate Character Segmentation Algorithm Based on Projection Feature and Prior Knowledge*. Advanced Materials Research, 677-681, doi:10.4028/www.scientific.net/AMR.790.677
- Mahini, H., Kasaei, S., & Dorri, F. (2006). *An efficient features-based license plate localization method*. International Conference on Pattern Recognition, 841-844, doi: 10.1109/ICPR.2006.239.
- Marr, D., & Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London B: Biological Sciences*, 207(1167), 187-217.
- Megalingam, R. K., Krishna, P., Somarajan, P., Pillai, V. A., & Hakkim, R. U. (2010). *Extraction of license plate region in Automatic License Plate Recognition*. International Conference on Mechanical and Electrical Technology (ICMET), 496-501, doi:10.1109/ICMET.2010.5598409.
- Morris, B. T., & Trivedi, M. M. (2008). *A survey of vision-based trajectory learning and analysis for surveillance*. Circuits and Systems for Video Technology, IEEE Transactions on, 18(8), 1114-1127.
- Naito, T., Tsukada, T., Yamada, K., Kozuka, K., & Yamamoto, S. (2000). *Robust license-plate recognition method for passing vehicles under outside environment*. Vehicular Technology, IEEE Transactions on, 49(6), 2309-2319.
- Oliveira, V., & Conci, A. (2009). *Skin Detection using HSV colour space*. Retrieved from <http://tpf-robotica.googlecode.com/svn-history/r18/trunk/Vision/59928.pdf>
- Omachi, M., & Omachi, S. (2009). *Traffic light detection with colour and edge information*. IEEE International Conference on Computer Science and Information Technology, 284-287, doi: 10.1109/ICCSIT.2009.5234518.
- Papageorgiou, C. P., Oren, M., & Poggio, T. (1998). *A general framework for object*

- detection*. International Conference on Computer vision, 555-562, doi:10.1109/ICCV.1998.710772
- Parisi, R., Di Claudio, E., Lucarelli, G., & Orlandi, G. (1998). *Car plate recognition by neural networks and image processing*. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 195-198, doi:10.1109/ISCAS.1998.703970
- Patel, C., Shah, D., & Patel, A. (2013). *Automatic number plate recognition system (anpr): A survey*. International Journal of Computer Applications, 69(9).
- Pentland, A., Moghaddam, B., & Starner, T. (1994). *View-based and modular eigenspaces for face recognition*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 84-91, doi: 10.1109/CVPR.1994.323814
- Piccardi, M. (2004). *Background subtraction techniques: a review*. IEEE international conference on Systems, man and cybernetics, 3099-3104, doi: 10.1109/ICSMC.2004.1400815.
- Piciarelli, C., Micheloni, C., & Foresti, G. L. (2008). *Trajectory-based anomalous event detection*. IEEE Transactions on Circuits and Systems for Video Technology, , 18(11), 1544-1554.
- Polana, R., & Nelson, R. (1994). *Detecting activities*. Journal of Visual Communication and Image Representation, 5(2), 172-180.
- Rashid, M. M., Musa, A., Rahman, M. A., Farahana, N., & Farhana, A. (2012). *Automatic parking management system and parking fee collection based on number plate recognition*. International Journal of Machine Learning and Computing, 2(2), 94.
- Raus, M., & Kreft, L. (1995). *Reading car license plates by the use of artificial neural networks*. Proceedings of the 38th Midwest Symposium on Circuits and Systems, 538-541, doi:10.1109/MWSCAS.1995.504495.
- Roomi, S., Anitha, M., & Bhargavi, R. (2011). *Accurate license plate localization*. International Conference on Computer, Communication and Electrical Technology (ICCCET), 92-97, doi:10.1109/ICCCET.2011.5762445

- Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). *Earthquake shakes Twitter users: real-time event detection by social sensors*. Proceedings of the 19th international conference on World wide web, 851-860, doi:10.1145/1772690.1772777
- Shaikh, S. H., Saeed, K., & Chaki, N. (2014). *Moving Object Detection Using Background Subtraction*: Springer.
- Shi, X., Zhao, W., & Shen, Y. (2005). *Automatic license plate recognition system based on colour image processing*. Computational Science and Its Applications–ICCSA 2005: Springer.
- Siskind, J. M., & Morris, Q. (1996). *A maximum-likelihood approach to visual event classification*. 4th European Conference on Computer Vision Cambridge, 347-360, doi:10.1007/3-540-61123-1\_152
- Stauffer, C., & Grimson, W. E. L. (1999). *Adaptive background mixture models for real-time tracking*. Paper presented at the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, doi:10.1109/CVPR.1999.784637.
- Suryanarayana, P., Mitra, S. K., Banerjee, A., & Roy, A. K. (2005). *A morphology based approach for car license plate extraction*. Annual IEEE on INDICON, 24-27, doi:10.1109/INDCON.2005.1590116
- Tarabek, P. (2012). *A real-time license plate localization method based on vertical edge analysis*. Federated Conference on Computer Science and Information Systems (FedCSIS), 149-154.
- Tomasi, C., & Kanade, T. (1991). *Detection and tracking of point features*: School of Computer Science, Carnegie Mellon Univ. Pittsburgh.
- Vallejo, D., Albusac, J., Jimenez, L., Gonzalez, C., & Moreno, J. (2009). *A cognitive surveillance system for detecting incorrect traffic behaviors*. Expert Systems with Applications, 36(7), 10503-10511.
- Vilaplana, V., & Marques, F. (2008). *Region-based mean shift tracking: application to face tracking*. IEEE International Conference on Image Processing, 2712-2715, doi:10.1109/ICIP.2008.4712354
- Volna, E., & Kotyrba, M. (2013). *Vision system for licence plate recognition based on neural networks*. International Conference on Hybrid Intelligent Systems (HIS),

140-143, doi:10.1109/HIS.2013.6920470

- Wang, X., Meng, L.-M., Zhang, B., Lu, J., & Du, K.-L. (2013). *A video-based traffic violation detection system*. Proceedings of 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), 1191-1194, doi:10.1109/MEC.2013.6885246
- Wang, X., Zhou, M., & Geng, G. (2004). *An Approach of Vehicle Plate Extract Based on HSV Colour Space*. Computer Engineering, 17.
- Wang, Y., & Kankanhalli, M. S. (2015). *Tweeting Cameras for Event Detection*. Proceedings of the 24th International Conference on World Wide Web, 1231-1241, doi:10.1145/2736277.2741634
- Wasserkrug, S., Gal, A., Etzion, O., & Turchin, Y. (2012). *Efficient processing of uncertain events in rule-based systems*. IEEE Transactions on Knowledge and Data Engineering, 24(1), 45-58.
- Wei, W., Li, Y., Wang, M., & Huang, Z. (2001). *Research on number-plate recognition based on neural networks*. Proceedings of the 2001 IEEE Signal Processing Society Workshop, 529-538, doi:10.1109/NNSP.2001.943157
- Wei, Z., Ji, X., & Wang, P. (2006). *Real-time moving object detection for video monitoring systems*. Journal of Systems Engineering and Electronics, 17(4), 731-736. doi:10.1016/S1004-4132(07)60007-3
- Weng, J., & Lee, B.-S. (2011). *Event Detection in Twitter*. ICWSM, 11, 401-408.
- Weng, S.-K., Kuo, C.-M., & Tu, S.-K. (2006). *Video object tracking using adaptive Kalman filter*. Journal of Visual Communication and Image Representation, 17(6), 1190-1208.
- Wu, H.-H. P., Chen, H.-H., Wu, R.-J., & Shen, D.-F. (2006). *License plate extraction in low resolution video*. International Conference on Pattern Recognition, 1, 824-827, doi:10.1109/ICPR.2006.761
- Xie, L., Sundaram, H., & Campbell, M. (2008). *Event mining in multimedia streams*. Proceedings of the IEEE, 96(4), 623-647.
- Xu, J.-F., Li, S.-F., & Yu, M.-S. (2004). *Car license plate extraction using colour and edge information*. Proceedings of 2004 International Conference on Machine

- Learning and Cybernetics, 96(4), 623-647, doi:10.1109/JPROC.2008.916362
- Yan, W., & Weir, J. (2011). *Visual Event Computing*. Retrieved from <http://http://bookboon.com/>.
- Yanamura, Y., Goto, M., Nishiyama, D., Soga, M., Nakatani, H., & Saji, H. (2003). *Extraction and tracking of the license plate using Hough transform and voted block matching*. Intelligent Vehicles Symposium, 243-246, doi: 10.1109/IVS.2003.1212916
- Yilmaz, A., Javed, O., & Shah, M. (2006). *Object tracking: A survey*. Acme computing surveys (CSUR), 38(4), 13.
- Yu, M., & Kim, Y. D. (2000). *An approach to Korean license plate recognition based on vertical edge matching*. IEEE International Conference on Systems, Man, and Cybernetics, 4, 2975-2980, doi:10.1109/ICSMC.2000.884453
- Zelnik-Manor, L., & Irani, M. (2001). *Event-based analysis of video*. Proceedings of the Computer Vision and Pattern Recognition, 2, 123-130. doi:10.1109/CVPR.2001.990935
- Zelnik-Manor, L., & Irani, M. (2006). *Statistical analysis of dynamic actions*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9), 1530-1535.
- Zhang, Z., Huang, K., Tan, T., & Wang, L. (2007). *Trajectory series analysis based event rule induction for visual surveillance*. IEEE Conference on Computer Vision and Pattern Recognition. 1-8. doi:10.1109/CVPR.2007.383076
- Zheng, D., Zhao, Y., & Wang, J. (2005). *An efficient method of license plate location*. Pattern recognition letters, 26(15), 2431-2438. doi:10.1016/j.patrec.2005.04.014
- Zhou, Y., Yan, S., & Huang, T. S. (2007). *Detecting anomaly in videos from trajectory similarity analysis*. 2007 IEEE International Conference on Multimedia and Expo, 1087-1090. doi:10.1109/ICME.2007.4284843
- Zhu, G., Huang, Q., Xu, C., Rui, Y., Jiang, S., Gao, W., & Yao, H. (2007). *Trajectory based event tactics analysis in broadcast sports video*. Proceedings of the 15th international conference on Multimedia, 58-67. doi:10.1145/1291233.1291250
- Zhu, Y., Huang, H., Xu, Z., He, Y., & Liu, S. (2011). *Chinese-style Plate Recognition Based on Artificial Neural Network and Statistics*. Procedia Engineering, 15,



3556-3561. doi:10.1016/j.proeng.2011.08.666