Automatic License Plate Recognition (ALPR)

Sina Ghasempour

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
February, 2015
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Serhan Çiftçioğlu
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Hasan Demirel
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Erhan A. İnce
Supervisor

Examining Committee

1. Prof. Dr. Hasan Demirel
2. Prof. Dr. Hüseyin Özkaramanlı
3. Assoc. Prof. Dr. Erhan A. İnce

ABSTRACT

All around the world the traffic police would stop drivers on the road to check if their road tax has been renewed. Unfortunately this sometimes causes delays for the people stopped and may even slow down the general traffic. Furthermore, for the police involved a big portion of their time is wasted just doing these checks. In this work we propose a new strategy for automating the monitoring of road-tax for drivers on the road and propose to notify the ones who need to renew their road tax by a formal e-mail message.

The idea is to combine the license plate detection and recognition process with the driver's information stored in a database and an e-mail sending function. The first task is the acquisition of frontal images for approaching cars. Then the application will estimate the rotation angle on each frame and make it horizontal by reverse transforming it. Subsequently, the position of the license plate in the frame will be detected and it will be segmented out. Once the plate region is cropped out the next step will be to segment out the individual alpha numeric characters. This task is achieved by looking at a vertical projection profile. Later the characters are traced to obtain their associated stroke based (8-connectivity) chain codes. Finally features are extracted from characters and chain codes and character recognition is carried out based on the collected features.

Our hybrid license plate detection and recognition system was tested using 150 test images. With two-letter and three-digit notation of North Cyprus license plates this would mean around 750 characters. Simulation results have showed that our proposed system was 93.33% successful in detecting and segmenting out the license plates. For detected plates the character segmentation accuracy was 100% and the recognition accuracy was 96.42%. Finally our ALPR system was found 92.85% successful in recognizing vehicle license plates. Note that only the cases where all characters were correctly detected have been considered as correctly recognized plate.

After obtaining the plate number a database is searched to locate the owner of the vehicle. The name obtained will then be used to check the road tax statue of the driver and if necessary an automatic e-mail message will be sent out using the e-mail address of the driver. The message will be informing the driver about what he/she should do and also at the same time warn him/her about what would happen if he does not attend in a given time period.

Keywords: License Plate Detection, Character Segmentation, Vertical Projection Profile, Stroke Analysis, Chain Code, Feature Extraction.

ÖZ

Tüm dünyada trafik polisi sürücülerin yol vergisinin yenilendiğini kontrol etmek için yolda sürücüleri durdurmaktadır. Ne yazık ki bu bazen insanların gecikmelerine ve hatta genel trafiğin yavaşlamasına neden olmaktadır. Ayrıca, polislerin zamanının büyük bir kısmı sadece bu kontrollerde israf olmaktadır. Bu çalışmada yolda hareket halindeki sürücülerin yol vergisini plaka tanıma yardımı ile kontrol edecek ve vergi yenilemesi gereken sürücülere resmi bir e-posta mesajı ile bildirmde bulunacak bir sistem önerilmektedir.

Önerilen sistem plaka algılama, tanıma ve e-posta gönderme işlevlerini birleştirmektir. İlk görev yaklaşan arabanın ön görüntülerinin elde edilmesidir. Sonra uygulama her karede dönme açısını tahmin edecek ve ters dönüştürerek yatay yapacaktır. Daha sonra, çerçeve içinde plaka konumu algılanır ve dışarı bölümlere (gerekli yerlere) servis edilir. Plaka bölgesi kırpıldıktan sonra sıradaki işlem, bireysel alfa sayısal karakterler servis edilecek. Bu görev, bir dik projeksiyon profiline bakılarak elde edilir. Daha sonra karakterler ilişkili inme-tabanlı (8-bağlantı) zincir kodları takip edildir. Son olarak özelliklere dayalı, algılanan karakterler ve zincir kodları toplanir ve karakter tanıma çıkarılır.

Karma plaka algılama ve tanıma sistemimiz 150 test görüntüsü üzerinden test edilmiştir. Kuzey Kıbrıs plakalarındaki iki harf ve üç rakamlı gösterim ile bu yaklaşık tanınacak 750 karakter anlamına gelmektedir. Benzetim sonuçları önerdiğimiz sistemin plaka tespit ve bölütlemede %93.3 başarılı olduğunu göstermiştir. Yer tespiti başarı ile yapılan plakalar için karakter bölütleme doğruluğu

%100 ve bölütlenmiş karakterleri tanıma oranı ise %96.4 olmuştur. Son olarak otomatik araç plaka tanıma (OAPT) sisteminde plaka tanıma oranı %92.85 olarak bulunmuştur. Bir plakanın doğru tanındığını kabul etmek için tüm 5 karakterin de doğru bulunması şartı aranmıştır.

Plaka numarası elde edildikten sonra veritabanı aracın sahibini bulmak için aranır. Elde edilen isim daha sonra sürücünün yol vergisi durumunu kontrol etmek için kullanılır ve gerekirse bir otomatik e-posta iletisi sürücünün e-posta adresine gönderilecektir. Bu mesaj sürücüye ne yapması gerektiği hakkında bilgi verecek ve aynı zamanda, gerekli işlemleri yapmaması durumunda neler olacağı hakkında uyarılarda bulunacak.

AnahtarKelimeler: Plaka Bölgesi Kestirimi, Karakter Bölümleme, Dik Projeksiyon Profili, Çomak Analizi, Zincir Kodu, Öznitelik Özütleme.

DEDICATED

to my mother Fatemeh Ebrahimivand, my father Majid Ghasempour,

my sister Sahar and my brother Nima.

ACKNOWLEDGMENT

First I would like to thank my supervisor Assoc. Prof. Dr. Erhan A. İnce for guiding and helping me in my master study, for his patience and sharing kindly his knowledge with me.

I also wish to thank all the faculty members at the department of Electrical and Electronic Engineering: specially the chairman Prof. Dr. Hasan Demirel.

My sincere thanks go to my best friend Mr. Payam Alikhani who has always been supportive of me.

Last but not least, I would like to express my appreciation to my parents who have always given me support and guided me in the right direction.

TABLE OF CONTENTS

ABSTRACT iii

ÖZ v

DEDICATED vii

ACKNOWLEDGMENT viii

TABLE OF CONTENTS ix

LIST OF FIGURES xii

LIST OF TABLES xiv

LIST OF SYMBOLS AND ABBREVIATIONS xv

1 INTRODUCTION 1

1.1 License Plate Recognition System 1

1.2 Details of our Hybrid ALPR system 2

1.3 Difficulties and Obstacles of Development 4

1.3.1 The Influence of Rules and Regulations 4

1.3.2 The Effect of Climatic and Geography Conditions 5

1.4. Type of LPR Systems 5

1.4.1 Fixed Systems 5

1.4.2 Mobile Systems 6

1.5 LPR Application 7

1.5.1 Parking 8

1.5.2 Access Control 8

1.5.3 Tolling Control 9

1.5.4 Border Control 10
1.5.5 Stolen Cars 10
1.5.6 Traffic Control 10
1.6 Elements of LPR System 10
2 LITERATURE REVEIW 12
2.1 Introduction 12
2.2 Literature Review on LPR 12
3 LICENSE PLATE LOCALIZATION AND SEGMENTATION 19
3.1 Vehicle Images Acquisition 19
3.2 Locating the License Plate in the Acquired Frame 21
3.2.1 Tilt Correction of the Detected Foreground Object 22
3.2.2 Smoothing and Combining the Edge Images 26
3.2.3 Segmenting the Plate Region 27
4 FEATURE SELECTION AND LICENSE PLATE RECONITION 29
4.1 Preprocessing Steps for our Hybrid LPR System 31
4.1.1 Skeleton Correction 31
4.1.2 Thinning Algorithm 32
4.1.3 Characters Segmentation 33
4.2 Characters Tracing to Obtain Stroke Directions 33
4.2.1 Stroke Definition 34
4.2.2 Stroke tracing 34
4.3 Feature extraction 37

4.3.1 Stroke Simplification 37

4.3.2 Feature Points Selection... 39

4.4 Character Recognition 43

4.5 Automated Road Tax Checking 46

5 CONCLUSION AND FUTURE WORK... 48

5.1 Conclusions 48

5.2 Future Works 48

6 REFERENCES.. 50

LIST OF FIGURES

Figure 1.1: Block diagram of proposed ALPR system 2

Figure 1.2: Flowchart of license plate localization 3

Figure 1.3: Flowchart of character recognition and detection 4

Figure 1.4: Types of plates 5

Figure 1.5: Type of license plate recognition system 7

Figure 1.6: Parking 8

Figure 1.7: Access control 9

Figure 1.8: Tolling control 9

Figure 1.9: LPR system's units 11

Figure 2.1: A fully connected *n-h-m* three layer neural network 16

Figure 2.2: Tasks of a character recognition 18

Figure 3.1: Images acquired from fixed cameras 20

Figure 3.2: 2D Gaussian distribution function with $\sigma = 1$ 21

Figure 3.3: Gaussian filtering to reduce contrast 21

Figure 3.4: Detected edge image using Sobel operator 24

Figure 3.5: Radon line representation 24

Figure 3.6: Lines corresponding to peaks in the Radon transform 25

Figure 3.7: Tilt correction 26

Figure 3.8: Smoothing and removal of small components 27

Figure 3.9: Cropped license plate 28

Figure 4.1: Extracted skeletons for same characters 31

Figure 4.2: Skeleton correction 32

Figure 4.3: Thinning of license plate characters 32

Figure 4.4: Separating characters on XL file's sheets... 33

Figure 4.5: Stroke direction structure.. 34

Figure 4.7: Characters with more than one open end point 36

Figure 4.8: Characters without any open end point 36

Figure 4.9: Tracing algorithm 37

Figure 4.10: Extracted chain codes for different size of characters 38

Figure 4.11: Main directions and normalized chain codes 39

Figure 4.12: Locating intersections.. 40

Figure 4.13: Null insertion 41

Figure 4.14: Locating open end points... 41

Figure 4.15: Two layer classification.. 42

Figure 4.16: Euler numbers for sample characters.. 43

Figure 4.17: Extracted features for Latin character classification 44

Figure 4.18: Extracted features for classification of numerals 45

Figure 4.19: Driver selection 47

LIST OF TABLES

Table 4.1: Characters classification table.. 44

Table 4.2: Performance Evaluation of different tasks required by ALPR 46

.

LIST OF SYMBOLS AND ABBREVIATIONS

Σ	Standard deviation
Ρ	Smallest distance from the line to the origin
Θ	Angle between the normal to the line and *x*-axis
ANN	Artificial neural network
ANPR	Automatic Number Plate Recognition
AVI	Automatic Vehicle Identification
CPR	Car Plate Recognition
LPR	License Plate Recognition
DDE	Dynamic data exchange
DLL	Dynamic link library
GPS	Geographic Positioning Systems
IR	Infra-Red
ID	Identity
IMAP	Internet Message Access Protocol
SMTP	Simple Mail Transfer Protocol
OCR	Optical Character Recognition
POP	Post Office Protocol
PDE	Partial differential equation
RTR	Road Tax Recognition
RGB	Red Gray Blue

Chapter 1

INTRODUCTION

1.1 License Plate Recognition System

Currently more than half a billion cars are traveling on the roads. All the vehicles have license plates and these help differentiate between or identify vehicles. Due to the enormous wave of vehicles it is evident that the human resources even in a small scale would not be sufficient to check all vehicles without the use of computers and signal processing techniques. Automation in this area is of high significance and should be considered [1].

Though the applications of automatic license plate detection have emerged in the last decade or so, the technology has been present for nearly 45 years. In the late 1970s, researchers for the United Kingdom's police scientific development branch have manufactured the first working license plate recognition system and began deploying it by the beginning of the 1980s.

The application areas for automatic license plate recognition include traffic monitoring, parking management, accident reporting, identifying drivers that cause traffic signal violations or drive in excess of the speed limit, for toll collection or to identify uninsured motorist.

Generally most license plate recognition (LPR) systems are expected to comply with the following goals:

- fast processing speed
- ability to recognize plate numbers from images with noise
- ability to work for tilted plates
- ability to work for different font styles and sizes

In the literature some other names for automatic license plate recognition exists. Some of these include: automatic vehicle identification (AVI), car plate recognition (CPR), automatic number plate recognition (ANPR) and car plate reader (CPR).

1.2 Details of our Hybrid ALPR system

The most critical techniques will be used in this discussion such as pre-processing techniques, plate location techniques, characters recognition and detection techniques, car owner identification and automatically sending E-mail techniques as shown in Figure 1.1.

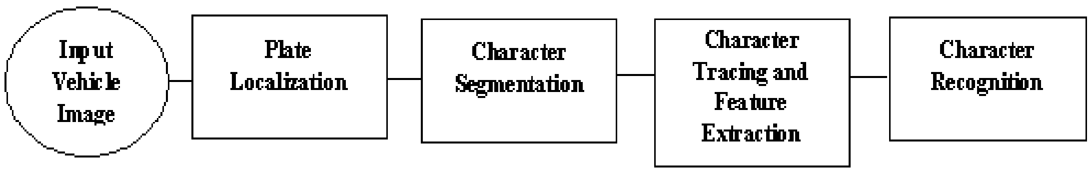


Figure 1.1: Block diagram of proposed ALPR system

The structure of the whole system will be illustrated in the chapter three. The task of license plate location and recognition from an image file, goes through a few processes as illustrated by the flowchart in Figure 1.2.

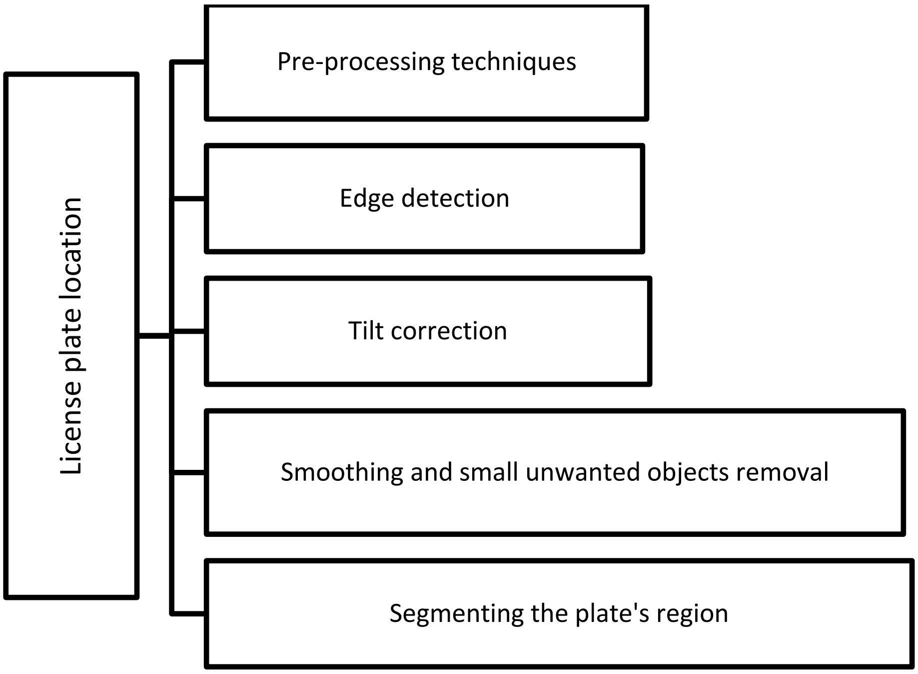


Figure 1.2: Flowchart of license plate localization

Our offered algorithm for characters recognition will be illustrated in the chapter four include of four steps; preprocessing techniques, characters tracing, features extractions and characters detection. Moreover, the preprocessing techniques and feature extraction include other stages. Finally, according to characters recognition's results, car owners will be identified and the system will send one email to inform car owners in shortest period of time. The outline for this research is adapted for characters recognition as shown by flowchart in Figure 1.3.

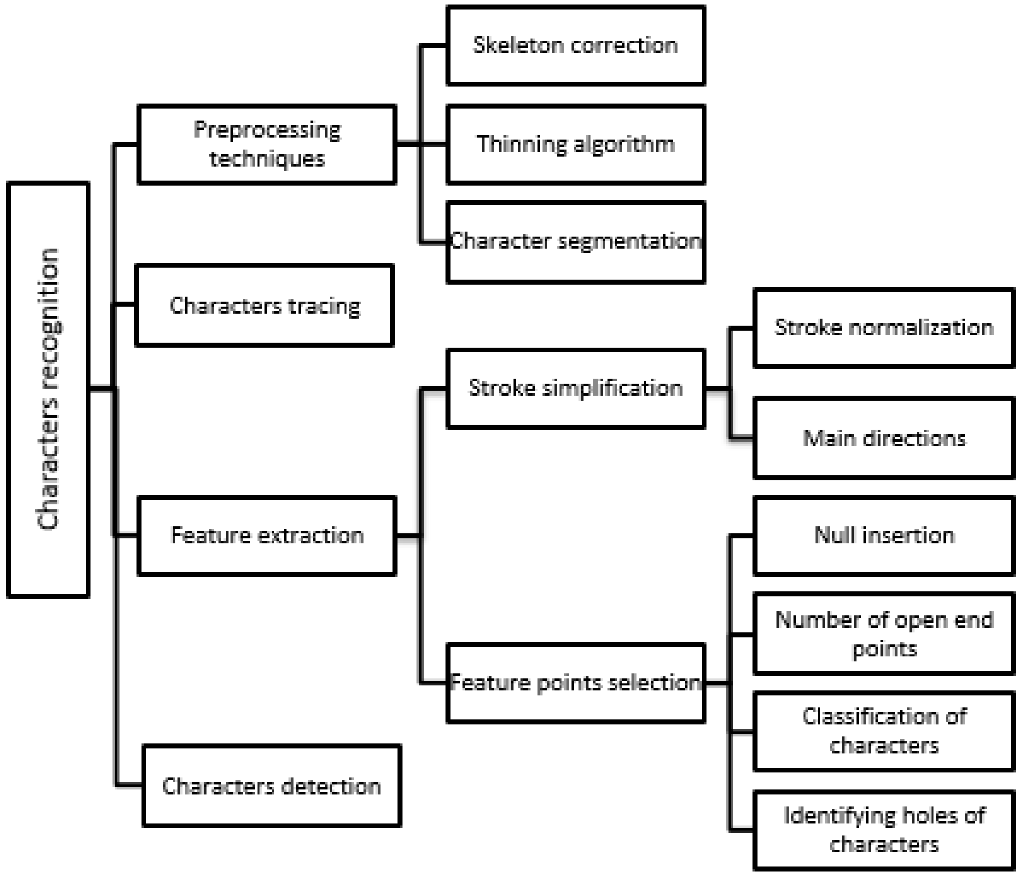


Figure 1.3: Flowchart of character recognition and detection

1.3 Difficulties and Obstacles of Development

In this view, there are many related parameters such as political and cultural condition, rules and regulations, climatic and geography conditions, science and technology and etc. which are affected to license plate recognition speed and accuracy. Here some mentioned parameters will be described.

1.3.1 The Influence of Rules and Regulations

Due to varying rules and regulation in different countries, license plates may have different locations and the system needs to collect knowledge and information about the character's distribution, font styles, size, color, spacing between characters etc. Various types of plates can be seen in Figure 1.5.

Figure 1.4: Types of plates

Common location and characters will be solved a wide range of research problems but researchers yet are facing with various types of plates with different designs and colors and different combinations of characters in this field.

1.3.2 The Effect of Climatic and Geography Conditions

Obviously, speed and accuracy of recognition's result is related to image quality. There is no doubt that one of the affecting parameter in the quality of the recorded images is atmospheric and environmental factors such as ambient light in the night and day, angle of sunlight, fog, humidity, rain and dust. It is expected that cameras with high performance regarding to geography condition can be solved this problem.

1.4. Type of LPR Systems

LPR system is divided into fixed and mobile types. Here, recognizes differences between the systems and their strengths and weaknesses are described.

1.4.1 Fixed Systems

LPR systems used in a fixed location are designed to be static and are not to be moved from where they are installed. The advantage of this method is its speed and simplicity of design regarding and the disadvantages is expense of sophisticated devices. The working principle of a fixed LPR system is shown in flowchart provided in Figure 1.5 (a).

1.4.2 Mobile Systems

Mobile systems are designed to have an officer instantly see the results of plates being read. This type of system allows for the mounting of cameras in a variety of positions [2]. It is clear this type is economical and due to high volume of procedures and algorithms has far less speed rather than previous approaches. The working principle of a mobile system is depicted by the flowchart in Figure 1.5 (b).

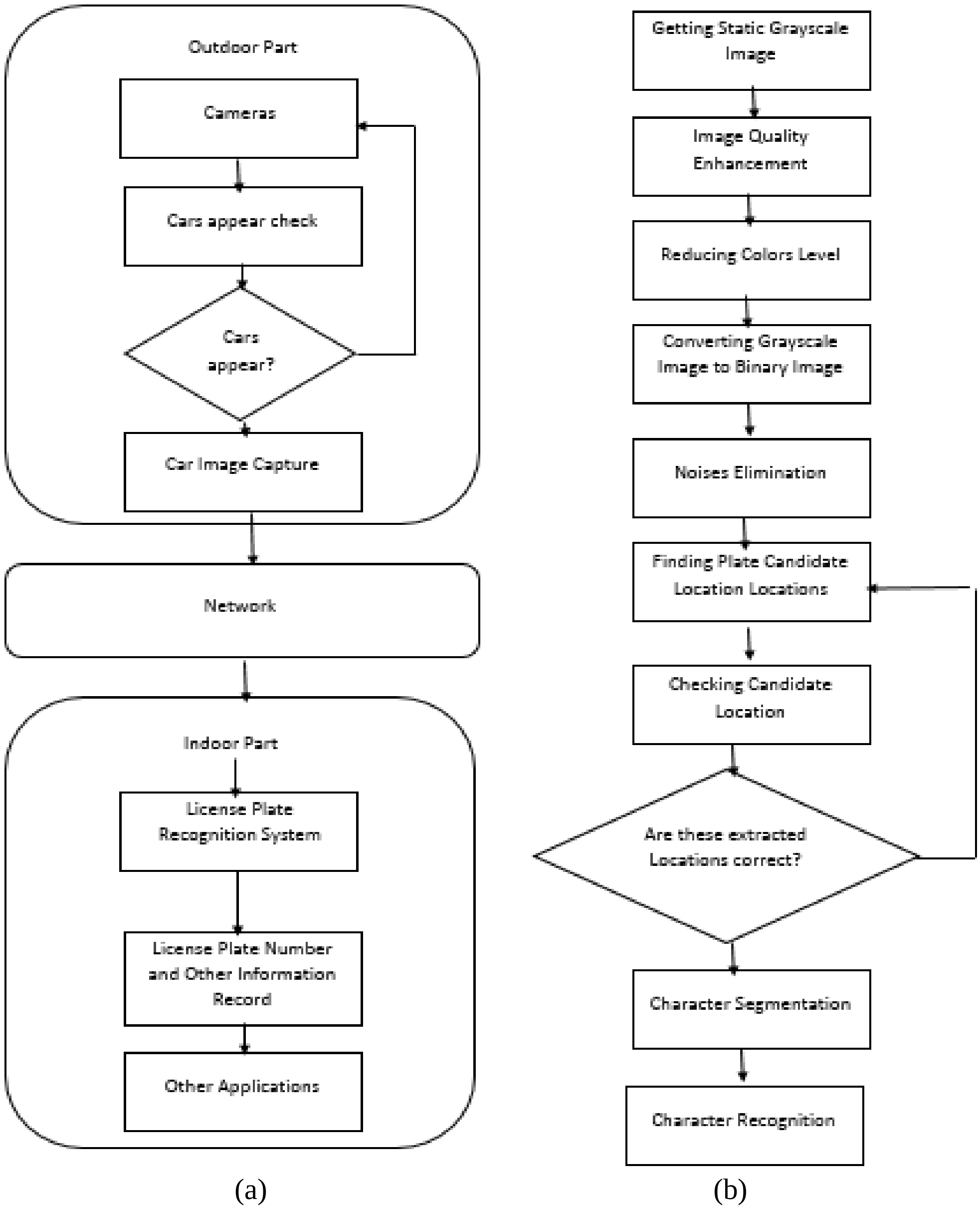


Figure 1.5: Type of license plate recognition system
(a) Flowchart of fixed system, (b) Flowchart of mobile system

1.5 LPR Application

The LPR systems are installed in many places such as toll gates, parking and entrance of secured buildings and etc. These systems are useful because it can mechanize car park managing, increase the security of car park operator, eliminate the usage of parking tickets, recover traffic flow during peak hours, and detect speeding cars on highways. Automatic plate recognition systems also control the following scenarios: country borders, traffic monitoring, law enforcement, traffic management, extensive parking, highways tax, stolen vehicle identification and vehicles tracking. Some of more practical applications will be expressed in the following.

1.5.1 Parking

LPR system recognizes the license plate when the cars enter to the parking and also calculate the recording time during park. This procedure saves time and costs and also increase the accuracy and efficiency of the acceleration process so that attract customers as well. In Figure 1.6, a vehicle enters to parking. The car plate are recognize and if the car owner has been charged parking fees, automatically the gate is opened soon after payment. (Maybe owner has a monthly permit)

Figure 1.6: Parking

1.5.2 Access Control

LPR system identifies car plate number and events are logged on a database so that entrance gate is opened for authorized members in the secured area. In Figure 1.7, entrance gate raise for the authorized vehicle automatically assisting to security guard.

Figure 1. 7: Access control

1.5.3 Tolling Control

The LPR system is used to calculate the travel fee in the toll-road gates as can be seen in the Figure 1.8. In this system, plates are read when vehicles enter the toll lane and present a pass card. The information of vehicles is retrieved from the database and it is compared with the pass information. In case of fraud, operators will be notified.

Figure 1.8: Tolling control

1.5.4 Border Control

The license plates information will be documented into core database of countries borders and they are monitored the national borders crossings. It can limit the national borders crossing and also this specific installation will insure the borders on the total state. This will be accustomed to track almost all national borders crossings.

1.5.5 Stolen Cars

The list of stolen vehicle will be employed to notify on a passing 'black' vehicle. This 'black list' instantly is updated and provided quick alarm system towards the police force. This LPR system is implemented for the roadside and it perform a new real-time match between the passing vehicle and the checklist. Using these calculations, burglars will be found and then police staffs will be notified about detected vehicles for ceasing the stolen car.

1.5.6 Traffic Control

Traffic congestion is often a problem in highway. Traffic management is substantial for road controlling so that sort of web digital cameras, timers, sensors are situated in the roads which are captured traffic jam information.

1.6 Elements of LPR System

The process starts while a sensor detects the presence of a vehicle and signals the system cameras to record an image of the passing vehicle. The image is passed on to a computer connected with a wireless network where software running on the computer extracts the license plate number. Recognized license plate is cross checked against record in a database and vehicle’s tax is found out. As shown in Figure 1.9 a license plate recognition systems needs to have four main elements; a light source to illuminate the license plate, a video camera to capture images of passing vehicles, a computer with image processing software, and wireless network [3].

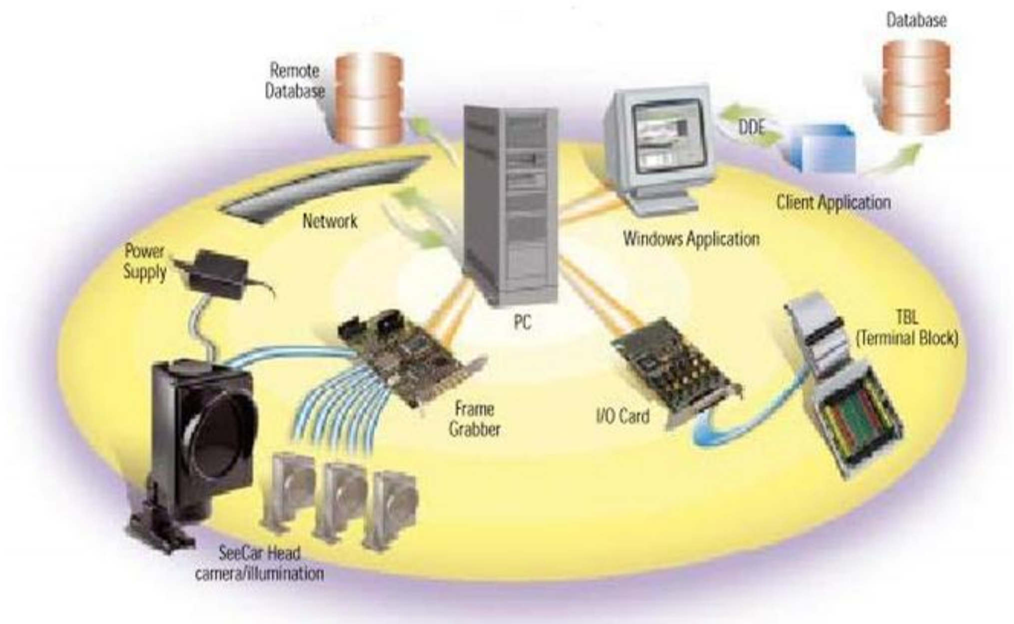


Figure 1.9: LPR system's units

Chapter 2

LITERATURE REVEIW

2.1 Introduction

Automatic license/number plate recognition is a specific application of optical character recognition. Typically employed by law enforcement agencies, the uses for automatic license plate recognition have grown tremendously since its introduction. Automatic license plate recognition may be used to cite individuals who violate traffic signals or drive in excess of the speed limit, as a method of electronic toll collection, to place a suspect at a scene, or identify uninsured motorist (when combined with a database search). Also there is a challenge in its application because different countries have different types of license plates where design, color, font style and size of fonts vary and Latin characters and numerals have different combinations. Section 2.2 that follows will give insight about different license plate recognition methods developed by researchers over the years.

2.2 Literature Review on LPR

In [4], the vector quantization is used to extract the textural properties of the regions. Character area has been determined by examining these features. In [5], it is assumed that the plate is white with black writing so areas of the plate will have high contrast. The plate is separated from the candidate regions so that with using mathematical morphology, dihedral areas with high contrast has been determined by taking the geometric plate features such as size, ratio of length to width.

References [6] [7] are the works that have used of the image gradient. In [3], they tried to determine the corners of the plate with edge detection and using geometric features of plate. In [6] the main effort is to highlight the edges of the plate. The edges that cannot be relevant to the plate are deleted with Gaussian filter. In the next step, due to the most of the plate edges are vertical so the non-vertical edges are removed and finally the plate is determined an area with a high density by sliding a window approximately in the same size of plate.

In [8] [9], the color information is used to determine plate. In [8], the eight neighbors of each pixel color have been given to a multi-layer perceptron in neural network so that the color of each pixel is classified into one of eight categories specified color. Then the image is scanned line by line and the plate is determined where pixel color sequences have certain features. Geometric characteristics of plate are used for accurate determination. In [9], vehicle area is specified with the difference between two consecutive images and plate candidate regions are identified with using the background color. Then, mathematical morphology operators are used and other areas are removed and the plate finally can be obtained accurately with taking the geometric characteristics of the plate.

Hough transform for line detection can be beneficial. In this case, the license plate with its adjacent lines is showed. Difficulty of this method is its high processing volume and time consuming. Histogram analysis for noisy images and images with rotated plated is not helpful. Morphological operations in real-time systems are used little because they have relatively time consuming [10]. Other methods have been proposed based on spectral light information of plate. In these methods, light condition is very important and problematic [11]. In [12], the proposed method are extracted the vertical edges after image improvement by Sobel operator and then noise borders of background is removed so the system is ultimately started to find on the remaining area making use of a rectangular window.

In [13], the genetic algorithm is used to find the plate. This method has a good performance, especially in images with different lighting conditions. In [14], it is used fuzzy logic and by % 90/9 precision could to recognize Taiwan vehicles which have a variety of plates.

There are many applications that deal with complex background images. Several plates extracted from complex background images are very difficult. In [15], a method is provided that the plate is extracted from images with complex background. In this study, the histogram equalization techniques for finding the threshold value have been used to improve image quality. After improving image quality, pixels differences for per big area is calculated and smoothed edges of the shapes using dilation technique. Finally, the number of pixels, the ratio of width to length and ratio of black pixels is checked. Eventually, plates are extracted as output.

The method presented in [16] [17] is the scanning row-column and count the number of changing in the color of the plate. This method does not work well for plate with angel because the corners of the plate image changes color level lower than threshold so that it will remove part of the plate. In this study, the lines which their above or below light changing intensity are lower than the threshold amount will be considered as part of the local candidates of plate.

Each plate candidate for the probable location should be assessed for accuracy. So the additional shape on image that cannot be part of plate is removed. Elimination of these shapes is done by checking the length, width, area and ratio of black spots [15] [18]. In [19], a well-known strategy for isolating the plates are introduced which include one considering model what is termed viral distribution such as edge detection, separation and clustering region.

Zheo and Gao have presented a method that extracts plates with complex background [20]. They used histogram equalization technique for finding light intensity threshold for improvement of image. After improving image quality, a big difference between pixels in each zone is calculated. Zhou and Gao have smoothed the image edges with dilation technique. Finally, the number of pixels, the ratio of width to length and the ratio of black pixels of the image are studied. Other than that, the algorithm is ignored regions that are not contained plates. Zhou and Gao have assumed that the plate is located near the middle of the picture. When the plate is placed on lateral borders of the plate, the plate will not be easily extracted. Zhou and Gao accuracy is about 80.7 percent.

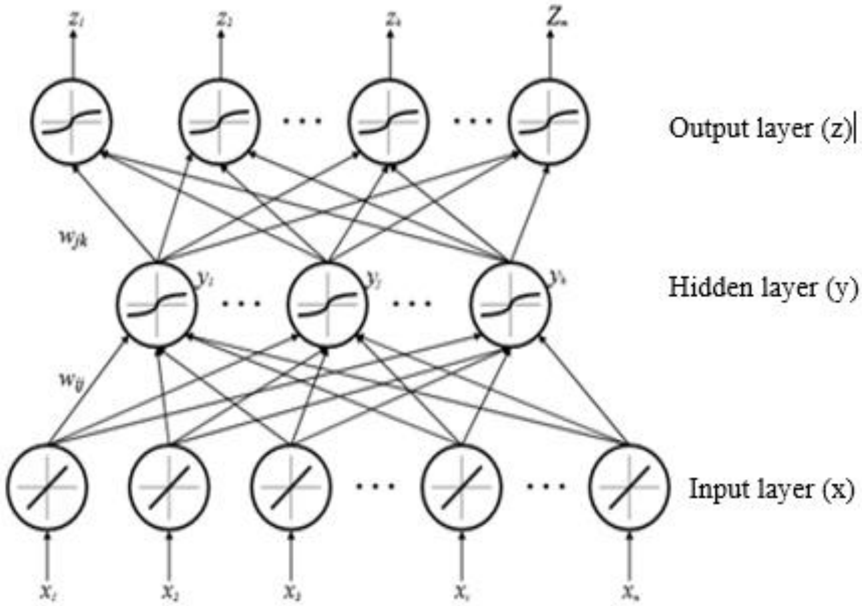


Figure 2.1: A fully connected *n-h-m* three layer neural network

Maro and Chacon have used the PCNN neural network to find candidate plates areas from static images which containing only one plate [21]. Then they used statistics and edge detection to find plate areas.

In [22], it proposes the main characteristics of Arabic writing, offering the importance of the sub-word structure of the Arabic word, showing the statistical results proving this phenomenon, and suggesting a new procedure for Arabic OCR system. The newly offered method suggests the treatment of the sub-word as the basic block in the recognition of Arabic characters. The size of the sub-word should be treated as a decisive factor in the method of recognition of the characters contained in the sub-word. The method of approximate stroke sequence matching is described and then applied to an example of unknown character and compared with two standard characters.

In [23], it gives an exclusive definition for the character similarity in order to its shape. What it is termed is approximate stroke sequence string matching. It exchanges two-dimensional of image data into one-dimensional data. Next, it implements a modified approximate string matching technique to measure the edit distance between them. Many features being are currently used on the handwriting analysis by practitioners or the document examiners. The form or shape is an important one for characterizing individual handwriting as it is quite consistent with most writers in normal undisguised handwriting.

In [24], it presents a new algorithm to extract text regions from color images that can be used for multi-segment characters such as Japanese and Chinese characters. The principle of the model is that the segmentation problem should be supported by a feedback obtained from recognition-based stroke analysis. The proposed model is a combination of competitive learning of neural network and multilayer perceptron (MLP) that it is trained by back propagation algorithm.

Performance of multi-methods in recognition does not always have an acceptable result in terms of processing time. The combination of proposed three steps in the recognition approach so that only improved the recognition rate but not the time complexity. This was due to the neural-based OCR process running on a sequential computer [25].

Many researchers applied a few methods for the recognition of characters like template matching, feature extraction, geometric approach, neural network, support vector machine, hidden Markov and Bayes net [26] [27].

The character recognition research was focused fundamentally on the shape recognition techniques. Although an upper limit in the recognition rate was realized, it was not sufficient in many practical applications. There are numbers of active research areas which can be isolated from the broader field of handwritten character recognition. This is true since the task of recognizing and classifying the characters from an image file, goes through few processes as generally showed by Figure 2.2. The image file of a handwritten character will have to undergo the process of pre-processing, feature extraction and recognition. Each of the phases plays an equally important role in the system [28] [29].

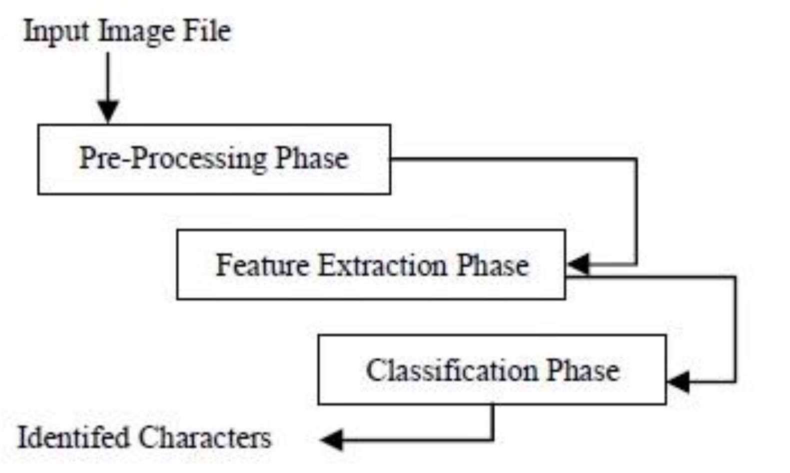


Figure 2.2: Tasks of a character recognition

Chapter 3

LICENSE PLATE LOCALIZATION AND SEGMENTATION

In this thesis, detection and segmentation of car plates and recognition of characters on each segmented plate will be carried out. This section of this thesis provides the simulation details for license plate localization. The first task is the acquisition of frontal images for approaching cars. Then the application will estimate the rotation angle on each frame and make it horizontal by reverse transforming it. Subsequently, the position of the license plate in the frame will be detected and it will be segmented out. Once the plate region is cropped out the next step will be to segment out the individual alpha numeric characters.

3.1 Vehicle Images Acquisition

For automated systems, vehicle images must be acquired through the use of cameras that have been installed on the side of the road or on bridges overlooking the lane(s) in the direction of the incoming traffic. Depending on in which direction the image has been taken (front or side), the license plate recognition system might have to carry out orientation analysis to estimate the rotation angle in the detected license plates and reverse transform the image of the segmented plate before proceeding to character segmentation. This step is important for proper segmentation of characters and if avoided can seriously degrade the character recognition performance. Figure 3.1 (a) and (b) depicts two sample-images acquired one from the front and one from the side.

(a) (b)

Figure 3.1: Images acquired from fixed cameras
(a) Image acquired from a bridge , (b) Image acquired from side of road

Since the acquired images might have different sizes, resolutions and illuminations, the LPR system must apply some preprocessing before the analysis stage. Once the grayscale image is obtained from the color image, we use a Gaussian filter to blur the face of the image. The main purpose of this step is to minimize the contrast (due to light and shadows) in a particular frame. Gaussian filtering an image implies convolving the image with a 2-D function denoted by (3.1) and as depicted in Figure 3.2. *G* is kernel matrix and σ is the standard deviation of the distribution.

$$\mathrm{G(x, y)} = \frac{1}{2\pi\sigma^2} e^{\frac{x^{2+}y^2}{2\sigma^2}} \qquad (3.1)$$

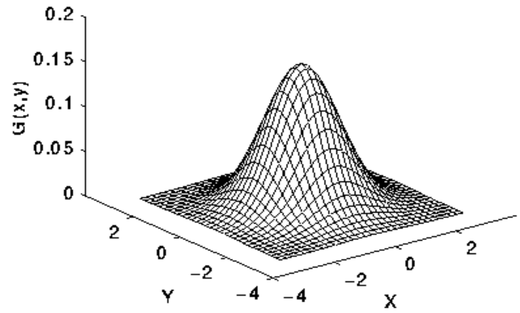


Figure 3.2: 2D Gaussian distribution function with σ =1

Two sample images obtained by convolving the 2D Gaussian kernel with the image are depicted in Figure 3.3. Images on the left are the original inputs and ones on the right are the filtered outputs.

(a) (b)

Figure 3.3: Gaussian filtering to reduce contrast
(a) Input image (b) Gaussian filtered output

3.2 Locating the License Plate in the Acquired Frame

In automatic license plate recognition systems, the very first thing the system has to do is to determine the location of a vehicle's license plate in an acquired frame.

However, before segmenting the plate number it is required to determine any rotation angle through the use of Hough transform and reverse transforming the image to correct the orientation of the image. After the image has been leveled edge detection, using the Sobel operator has to be carried out and then morphological operations will be used to eliminate components with improper aspect ratios, small areas and the ones touching the borders. Eventually, the system will crop part of the image where the license plate is and proceed to segmentation of the individual characters. For segmentation of the characters firstly the image will be converted to a binary form and then the algorithm will trace the columns of the binary image. It will sum the number of white pixels in each column by tracing the image from left to right and determine the position of columns with zero or few pixels. Since in practice there are either very few pixels or nothing in-between the characters, these minimums will help us to segment the individual characters. Once the characters are extracted, feature analysis and some other processing steps will be used to recognize the extracted characters and the plate number. The sub-sections that follow provide details for the different tasks described above.

3.2.1 Tilt Correction of the Detected Foreground Object

If no tilt correction is applied characters extracted from detected plate regions can lead to errors during the LPR process. To avoid such errors, before plate region detection the system has to estimate the tilt angle and reverse transform the image around its center point. In this work, estimation of the tilt angle is carried out by first obtaining and edge image and then using Hough transform to estimate the longest line and its angle with the horizontal axis.

In image processing, an edge is defined as a sudden change in image brightness and edge detection refers to the process of recognizing these changes (that is the fast interruptions in an image). These interruptions are unexpected fluctuations in the pixel values that can describe borders of objects in an image. Typically, techniques of edge detection require convolving the image with a function, which is able to sense the large gradients in an image. There are various operators used for edge detection. In this study, the Sobel operator has been employed to get the edge image. The kernel size chosen was (3×3) and is as depicted in (3.2).

$$\nabla f = \begin{matrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{matrix} \;,\; Gx = \frac{\partial f}{\partial x} = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \;,\; Gy = \frac{\partial f}{\partial y} = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix} \tag{3.2}$$

Figure 3.4 shows the edge image obtained by convolving the test image with the ′Sobel′ operator.

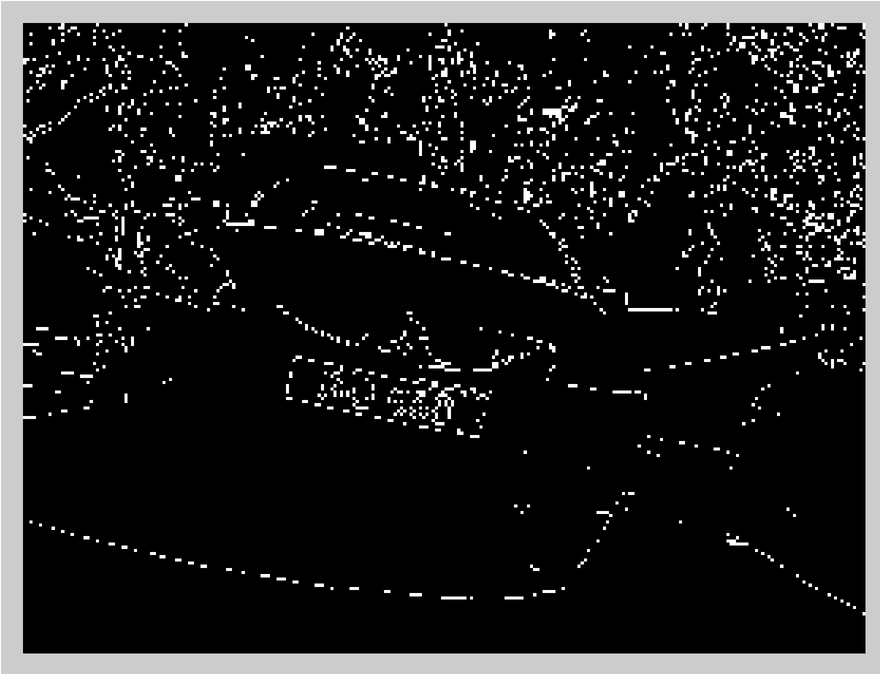


Figure 3.4: Detected edge image using Sobel operator

Radon transform is related to a common used computer vision transformation known as the Hough transform. The Radon function implements a form of the Hough transform to detect straight lines. A popular representation of the Radon transform expresses lines in the form of:

$$\rho = x * \cos\theta + y * \sin\theta \tag{3.4}$$

Where, θ is the angle and ρ the smallest distance to the origin of the coordinate system (refer to Figure 3.5).

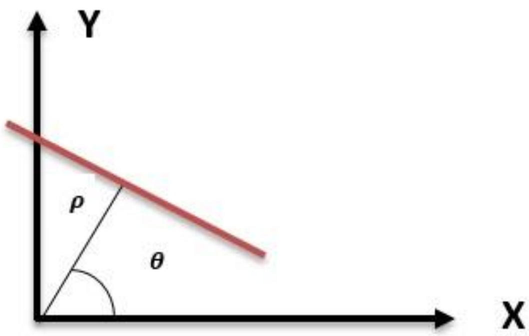


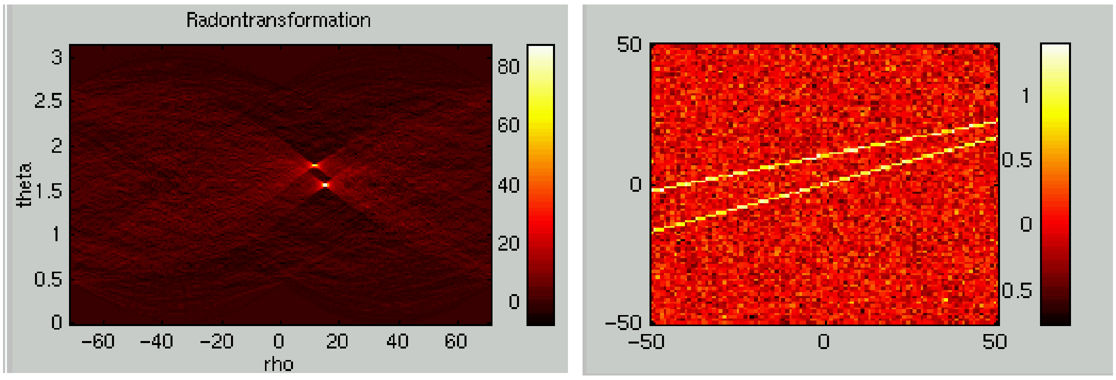
Figure 3.5: Radon line representation

Radon transform for a set of (ρ, θ) values is the line integral through the image $g(x, y)$.

The processing steps to determine the straight lines are as follows:

1. Compute a binary edge image using the edge function,
2. Compute the Radon transform of the edge image,
3. Find the locations of strong peaks in the Radon transform matrix

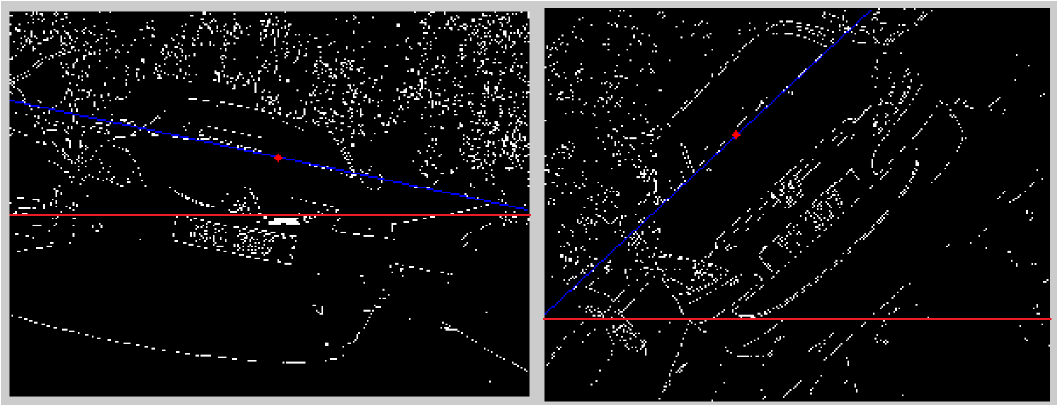
Locations of these peaks correspond to the location of straight lines in the original image. Figure 3.6 depicts the Radon transform and the two lines corresponding to the two peaks of the transform [30].



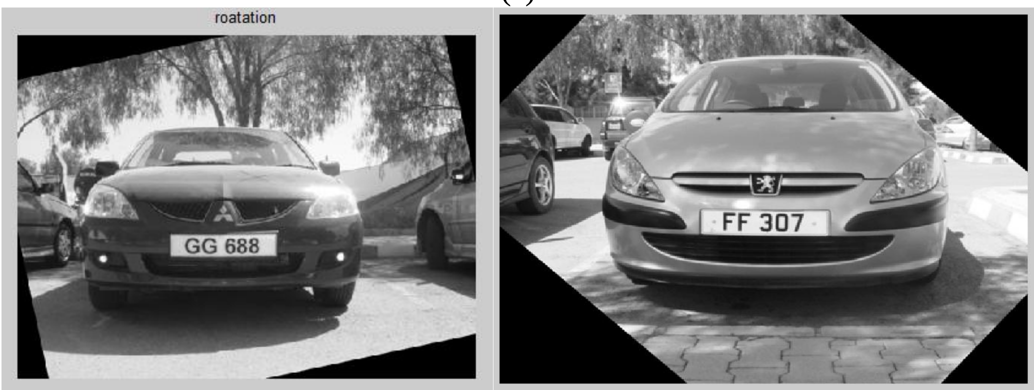
(a) (b)

Figure 3.6: Lines corresponding to peaks in the Radon transform
(a) Radon transform, (b) Corresponding two lines

After applying the Radon transform to the edge image of Figure 3.6 and selecting the longest line among the detected lines we have superimposed this longest line in blue color onto the edge image (refer to Figure 3.7 (a)). Eventually, the system would determine the angle between the horizontal (red line) and the longest line (blue line) and correct the tilt angle as shown in Figure 3.7 (b).



(a)



(b)

Figure 3.7: Tilt correction
(a) Longest line detection (b) Tilted images correction

3.2.2 Smoothing and Combining the Edge Images

To find the candidate plate regions in a given frame, we first convolving the image with the Prewiit operator in horizontal and vertical directions to obtain the horizontal and vertical edge images as depicted in Figures 3.8 (a) and (b). Afterwards, the edge images are smoothed to merge the edge pixels that are close to each other. Finally, some morphological operations are used to remove components with improper aspect ratios or to fill the holes in the remaining detected regions.

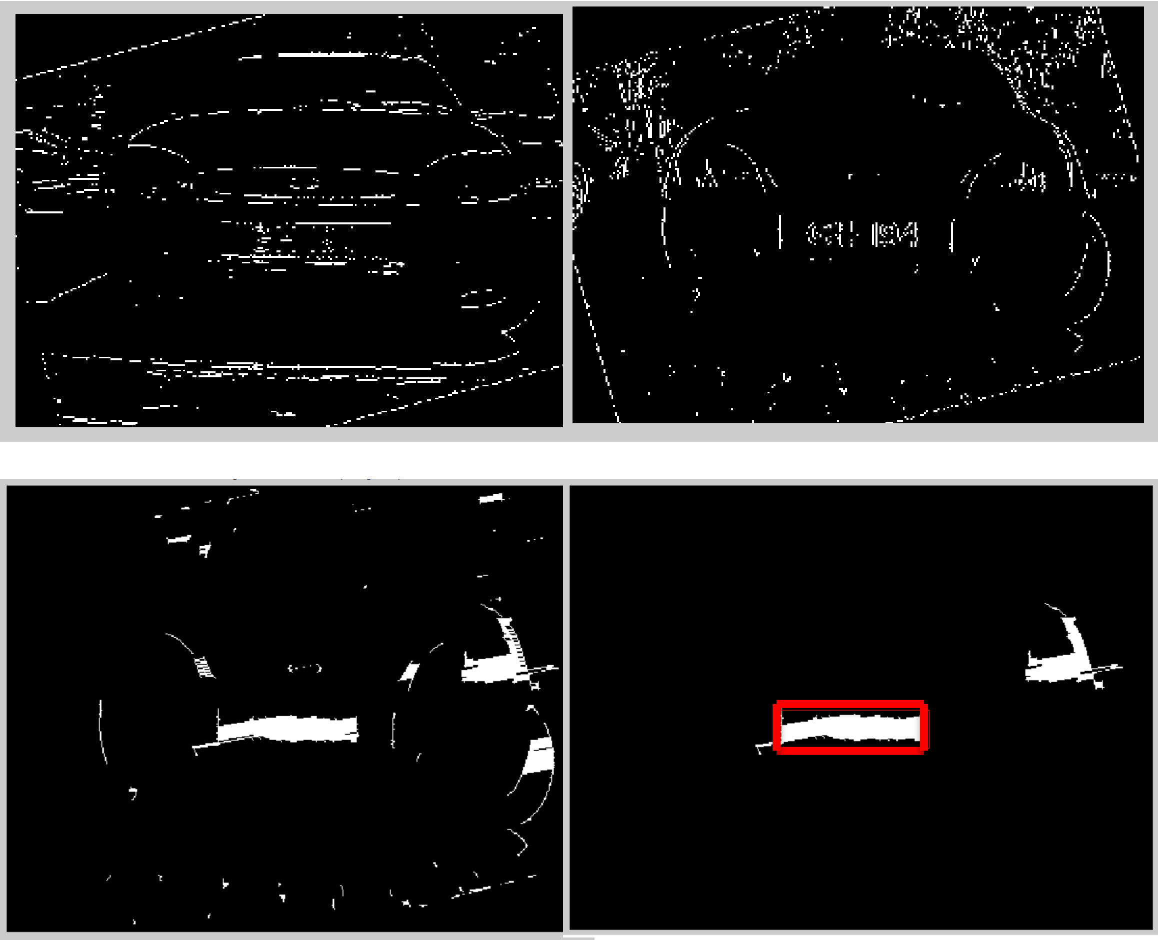
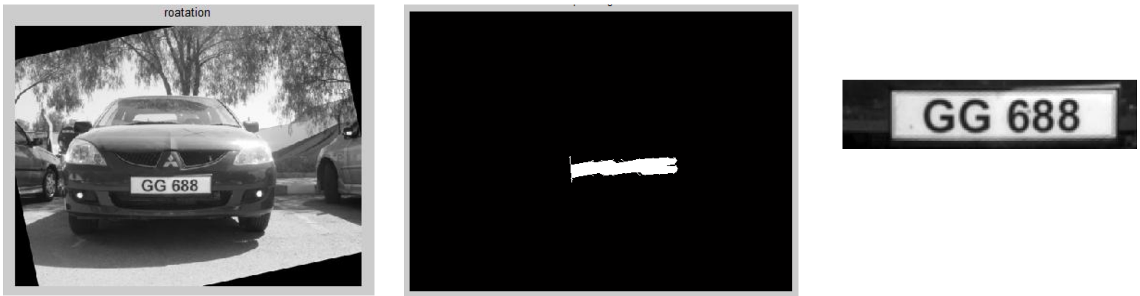


Figure 3.8: Smoothing and removal of small components
(a) Horizontal edge image , (b) Vertical edge image,
(c) Smoothed and combined edge images, (d) Removal of small-unwanted objects.

It can be seen from Figure 3.8 (c) that after removal of small components only two candidate regions will remain. We then select the correct plate region by considering the aspect ratios (TRNC plates are rectangular shaped and have dimensions 520 mm × 110 mm) of the two candidate regions. Note that in the figure 3.8(d) the correct plate region is boxed by a red color rectangle. This candidate region corresponds to the plate GG688 in Figure 3.6 (b).

3.2.3 Segmenting the Plate Region

As depicted in Figure 3.8 once the final plate region (binary mask) is determined the RGB image can then be cropped at the coordinates dictated by the mask and the plate region is obtained as shown in 3.9 (c). Our hybrid license plate detection and recognition system was tested using 150 test images. Simulation results have showed that our proposed system was 93.33% successful in detecting and segmenting out the license plates.



(a) (b) (c)

Figure 3.9: Cropped license plate

(a)Tilt corrected RGB image, (b) Binary mask, (c) Cropped license plate

Chapter 4

FEATURE SELECTION AND LICENSE PLATE RECONITION

There are mainly two reasons for using stroke analysis while doing character recognition. First reason is that, the characters on the license plates produced by different shops may not be conforming to a single standard and different font styles and sizes can be observed. It has been shown in the literature that [31], recognition based on stroke analysis would be invariant to these style and font size changes and can still recognize characters through a process of summarizing. Secondly, style and font size invariant recognition system would eliminate the need for retraining the system.

The proposed algorithm in this study includes three main steps namely:

1) Preprocessing technique, 2) Character tracing, 3) Features extraction and 4) Character recognition. The preprocessing steps include elimination of spurious pixels using some morphological operations, thinning of corrected characters and segmentation of individual characters. After segmentation of characters a tracing algorithm locates the starting point (point to start tracing the character from) [32], for each character and obtains the chain-code related to the specific character. Then, features of the chain codes (intersections, summarized chain codes within main directions, number of open end pixels, number of holes, etc.) are extracted and finally the system carries out character recognition by comparing the features.

Intersections which are also referred to as junctions are locations where the chain code goes in more than one direction in a 4 or 8 connected neighborhood. An intersection point is a pixel with has more than two neighbors in 4 or 8-conectivity and must be obtained after thinning. Intersections in a character or number of open-end pixels for characters can be used as some distinguishing features in the recognition stage.

To reduce the processing time of chain codes generally first and ambiguity check [28] is administered and any irregularity in the pixel shapes is corrected, then chain code normalization is applied to simplify chain codes of different lengths. Stroke directions which has a frequency of unity and which are not in any of the main directions can also be eliminated.

In North Cyprus the standard notation on civilian license plates is two alpha-characters (upper case) and three numeric characters from 0-9. The only exceptions to this rule are the license plates with taxis and rental cars. License plates for rental cars are black on red background and has three alpha characters. For taxis there must be also three characters but the left most character must be the character ‘T’. In order to improve the speed of processing and rate of recognition a two level classifier can be used. The first classifier is for Latin alpha characters and the second is for numerals.

Finally, the system uses features extracted from chain codes and characters and carries out recognition based on minimum distance and correlation.

4.1 Preprocessing Steps for our Hybrid LPR System

Before character recognition some preprocessing steps must be carried out on both the extracted plate regions and individual characters. In this thesis, the preprocessing steps include elimination of spurious pixels using some morphological operations (for skeleton correction), thinning of corrected characters and finally segmentation of individual characters. What follows gives brief explanations for the different preprocessing steps used in this study.

4.1.1 Skeleton Correction

It is possible that a particular character which is extracted from two different license plates has different character skeletons as depicted in Figure 4.1. In order to assure a good recognition efficiency these irregularities in the character skeletons have to be fixed. If this step is skipped then the chain codes for same character extracted from various different plates would be different from each other and this can lead to a reduction in efficiency of plate recognition.

Figure 4.1: Extracted skeletons for same characters

There are some morphological operations that can be used to obtain precise and regular skeletons for characters. Spurious pixel removal is one such morphological operation that helps remove spur pixels by setting pixels to zero if it has only one neighbor. Similarly, when only a single pixel is missing in a (3×3) block this pixel can be set to 1 to obtain a more regular skeleton for the character. Figure 4.2 shows an example where missing single pixels have been set to 1 using the morphological operation 'spur' in MATLAB.

Figure 4.2: Skeleton correction

4.1.2 Thinning Algorithm

In order to obtain chain codes for license plate characters, extracted characters need to be traced. Since locating an initial starting point on raw characters can be difficult generally a thinning algorithm is used to cut the points around the contour layer by layer which eventually obtains skeletons with a single character width. Figure 4.3 shows the result of thinning operation applied to two different car plates.

Figure 4.3: Thinning of license plate characters

4.1.3 Characters Segmentation

Character segmentation process seeks to analyze gaps among of the characters in order to separate them by recognize the columns and rows with values of zero. As can be seen in figure 4.4, there are sum the number of white pixels in each column by tracing the image from left to right and determine the position of columns with zero or few pixels. Since in practice there are either very few pixels or nothing in-between the characters, these minimums help us to segment the individual characters.

The system uses vertical projection profile analysis to segment the individual characters on each plate. Eventually, characters are placed on five XL sheets, one for each character. Figure 4.4 shows the character *H* which has been segmented and is stored in sheet-1 of the excel file.

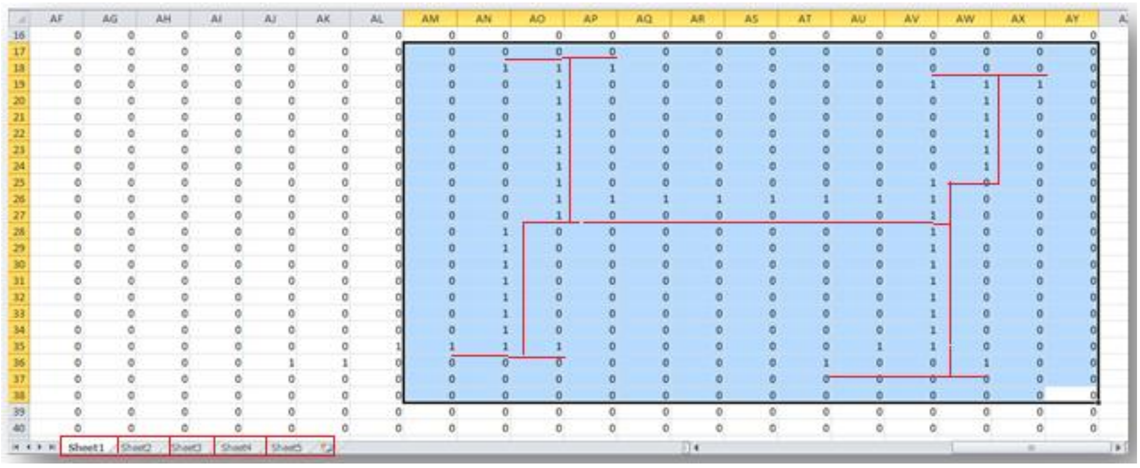


Figure 4.4: Separating characters on XL file's sheets

4.2 Characters Tracing to Obtain Stroke Directions

It is possible to trace the extracted characters to obtain stroke directions where the directions give a description of object's borders. In the literature a sequence of strokes is referred to as a chain code. Chain codes were first introduced by Freeman in 1961 and are known as Freeman Chain Codes (FCCs). There are two alternatives for obtaining chain codes. These include using either 4-connectivity or 8-connectivity while performing the tracing. In this study we adopt the 8-connectivity while extracting the chain codes for different characters.

4.2.1 Stroke Definition

A stroke as defined in [23] is a series of pixel directions obtained through n-connectivity tracing. The tracing of the contour aims at transforming the border of the characters into a string of codes to extract the features of the characters. The coding scheme starts by identifying the position of an initial pixel and continues identifying the relative positions of the successive pixels on the contour. Figure 4.5 shows the stroke directions for the case of 8-connectivity.

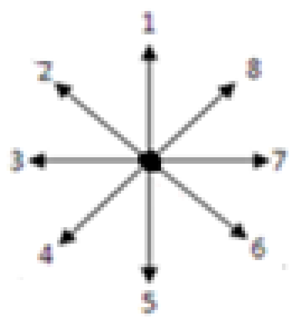


Figure 4.5: Stroke direction structure

4.2.2 Stroke tracing

In this step, the system starts to scan the character in a left to right and top to bottom manner to obtain the chain code for a given character. During the scan process, all pixels which have a value of 1 are located and their *x* and *y* coordinates stored in a matrix as depicted in Figure 4.6. For the pixels with a value of 1, the system also records the number of neighboring pixels and their stroke directions in the 4^{th} and 5^{th} rows of the *N*-matrix respectively. For instance, for the first pixel that has a value of

1 and has two neighbors, 5^{th} row of table lists the direction of neighbors as ‘64’. Here 6 denote the direction of the first neighbor and 4 is the direction of the second neighbor. In 6^{th} and 7^{th} rows, the number of the destination pixel in each direction has been provided. For example for the 1^{st} pixel that has a value of 1 there are two neighbors. The first neighbor that is in direction 6 is the pixel 2 as specified by 6^{th} row and the neighbor in direction 4 is the pixel 3 as specified by 7^{th} row.

Number of pixels	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Coordinate of pixels	16	17	17	18	18	19	19	20	21	22	23	24	25	26	27	28
	9	8	10	7	11	6	11	11	11	11	11	11	11	11	11	11
Number of neighbors	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	1
Direction paths	64	48	62	48	52	58	51	51	51	51	51	51	51	51	51	51
Number of pixels for 1^{st} direction	2	1	1	2	3	4	5	7	8	9	10	11	12	13	14	15
Number of pixels for 2^{nd} direction	3	4	5	6	7	0	8	9	10	11	12	13	14	15	16	0

Figure 4.6: Sample of *N*-matrix

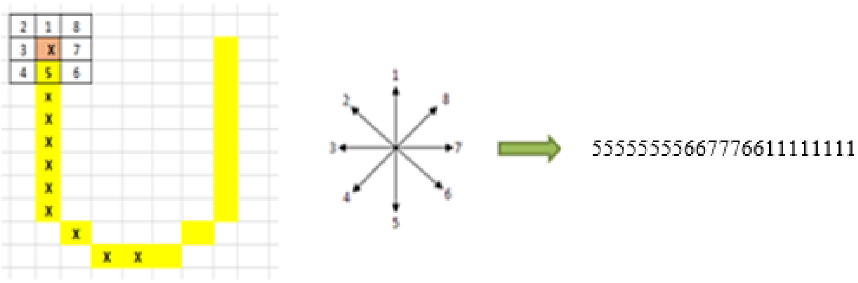
Note that in the *N*-matrix given in Figure 4.6 we only have two pixels that have single neighbors. These pixels are candidates for being a start point, that is to say the point from which scanning will start. Figure 4.7 shows possible start points marked in red on four different characters (’F’, ‘6’, ‘G’). For most characters the actual starting point is the first pixels found with a single neighbor while scanning in a left-to-right and top-to-bottom manner.

Figure 4.6: Characters with more than one open end point

Some characters like the ones shown in Figure 4.8 do not have any pixels with only one pixel neighbor. In these cases, while tracing in a left-to-right and top-to-bottom manner the first pixel which is detected to have a value of 1 is usually assumed to be the starting pixel.

Figure 4.7: Characters without any open end point

After determining the main starting pixel, the system would scan its neighborhood for finding other pixels which has values of 1. Pixels with value of 1 are specified and their directions are noted. Once a pixel is scanned it location is marked with an 'X' as shown in Figure 4.9 (a) to avoid repetition of scanning the same pixels. During the tracing process, there are some pixels which have more than one neighbor. For these pixels a trace is performed separately for each connecting neighbor. When tracing is complete the chain codes for the particular character would be ready. Figure 4.9 (b) depicts the extracted chain code for the character U. The algorithm will be finished when all pixels with value of 1 are traced.



(a) (b)

Figure 4.8: Tracing algorithm
(a) Character tracing (b) Extracted chain code

4.3 Feature extraction

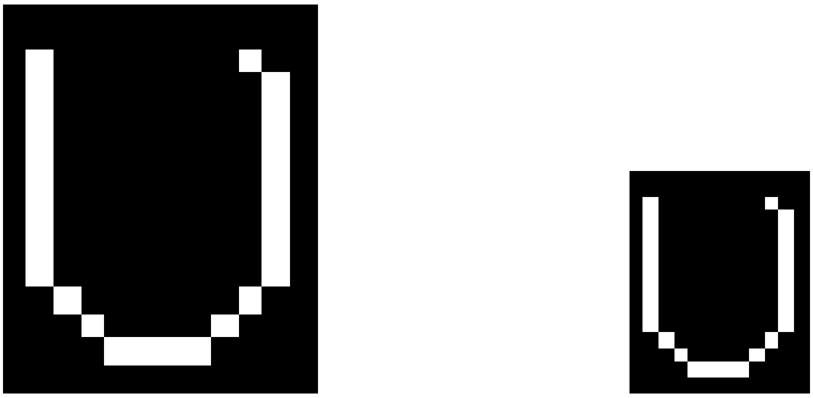
Feature extraction is the step where relevant feature that can be used for recognition are determined and extracted. Selection of the right feature is vital in order to achieve best results in the license plate recognition study [34]. The selected character features and feature extraction are two crucial issues. To achieve this goal, the following algorithm can be used to simplify the chain codes previously obtained using the tracing step. The processing will end after extraction of relevant features.

4.3.1 Stroke Simplification

There are multiple reasons for wanting to simplify stroke sequences. The main one is the need to reduce the processing time in order to have a system that operates in real-time. Secondly due to different character sizes and fonts some characters would be represented with a longer chain code or same character could end up having more than one chain code. For stroke sequences obtained using 8-connectivity if we choose 4 main directions we can re-form a chain code that becomes insensitive to slanted characters.

4.3.1.1 Stroke Normalization

In this step, the algorithm tries to reduce the redundancies present in the chain codes and obtains a new chain code which in the literature is referred to as the normalized chain code. The name given to stroke simplification is stroke normalization. As can be seen from Figure 4.10 the chain codes extracted from two different size ‘U’ characters can be different. Unless something is done this could lead to an ambiguity in the recognition stage.



555555555555555666777787811111111111111 55555677781111

Figure 4.9: Extracted chain codes for different size of characters

The solution is to perform normalization to the sequence of stroke directions for both characters and obtain a simplified code which is the same. For the characters shown in Figure 4.10 the simplified chain code will be {5, 6, 7, 8, 1}.

4.3.1.2 Main directions (4-connectivity directions)

Stroke directions which are extracted while tracing the characters are obtained using 8-connectivity (8 border pixels) direction pattern. Out of these 8 directions only four are known as main directions. Since these main directions have enough information embedded in them that can be used for recognition purposes it is not necessary to use all directions and hence the chain code can be simplified by selecting only stoke conforming to the main directions. In addition, main direction is often a major contributor to the efficiency of the feature extraction process for the recognition of characters with different fonts and style (i.e. italic etc.). During the simplification process any stroke directions that occur just once and are not conforming to the main directions could also be eliminated.

For the character ‘G’ depicted in Figure 4.11-(a) the chain code (sequence of stroke directions are :

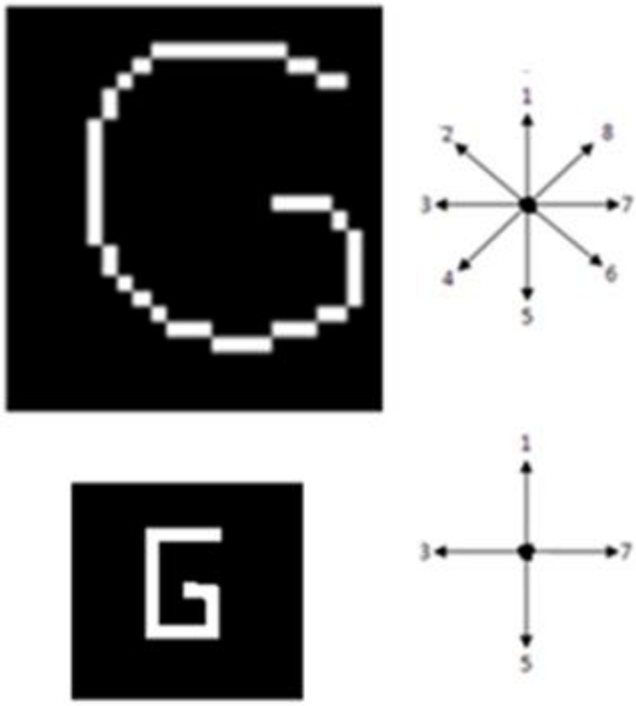
222333334454555556566667767778777878111122333

After normalization the stroke pattern becomes:

234567123

Finally, when only the main directions are selected the sequence obtained becomes:

35713



(a)

(b)

Figure 4.10: Main directions and normalized chain codes

Figure 4.11-(b) depicts the new ‘G’ character based only on the four main directions.

4.3.2 Feature Points Selection

Selection of feature points that will enhance the efficiency of character recognition is a critical step. Therefor after character tracing and simplification of stroke sequences, it all comes down to extracting appropriate features that can be used for recognition. In this study we have used the number of intersection points and number of open end points as major features that can be used in the character identification process.

4.3.2.1 Null Insertion

When a character is being traced those pixels that have more than two-pixel neighbors can be selected as intersection points. Also for characters that have two neighbors at the start point we can consider this points as intersections (refer to character 'O' and '8' in Figure 4.12).

Figure 4.11: Locating intersections

While tracing a character when we encounter an intersection there would be more than one path which the scanning can continue with. These pixels which are the neighbors in different directions are known as node pixels. While creating the chain code for a character we can use '0' whenever we encounter a node pixel. Figure 4.13 depicts the insertion of '0's into the chain code whenever a node pixel is encountered. Note that at an intersection when tracing selects one of the two direction and scanning in one direction is complete another '0' will be inserted since we have a second path which we need to trace (refer to character 'F' and its chain code).

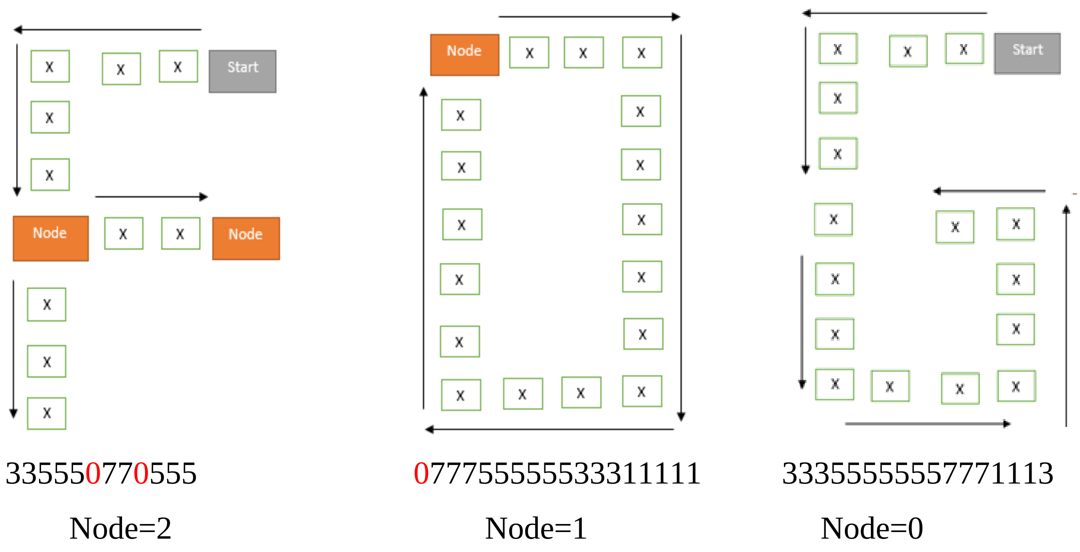


Figure 4.12: Null insertion

4.3.2.2 Number of Open End Points

An open-end point is considered to be a pixel which has only one neighbor and when a character is traced al pixels and the number of neighbors it has is determined and stored in the *N-matrix* which was previously introduced in Section 4.2.2. Hence to determine how many open end points a character has we just need to find the number of entries with the value of 1 in 4^{th} row of the *N*-matrix. Figure 4.14 shows some sample characters where the open end points have been marked in red color.

Figure 4.13: Locating open end points

4.3.2.3 Classification of Characters

In order to improve the speed and rate of recognition in this thesis we have used a dual character classification. The first classifier was for Latin characters and the second was for numerals. This approach was adopted since the civil number plates in North Cyprus as two letter on the left and three digits on the right. Figure 4.15 shows the designation on a number plate according to the dual classification.

Figure 4.14: Two layer classification

Classifying Latin letters such as 'B', 'S' and 'O' separately from '8','5' and '0' also help avoid making wrong decisions since these Latin characters resemble the numeric characters. Because our system can avoid these ambiguities its recognition rate will be improved.

4.3.2.4 Identifying Holes of Characters

Characters can also be classified regarding the number of holes they have. As can be seen from Figure 4.16 some characters such as 'A, P, R, D, O, Q' have only one hole and others such as 'B, 8' have two. On the other hand characters such as '7, 1, 2 and 3' o not have any holes.

Our system checks how many holes a character has using the 'Euler' function under the MATLAB platform. The possible values from the Euler function are 0, 1 and -1.

The value 0 indicates that the character has only one hole whereas the value 1 implies two holes. When the function returns -1 this indicates that the character has no holes. We have used the Euler classification in this thesis as an extra layer of classifier to help improve the character classification results.

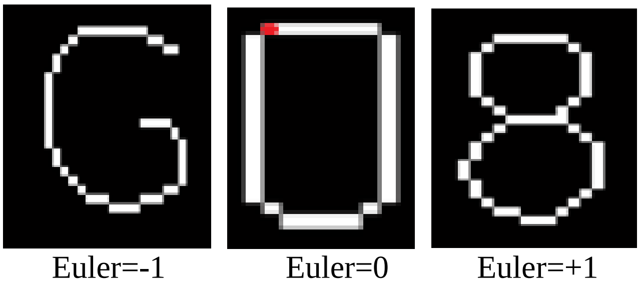


Figure 4.15: Euler numbers for sample characters

4.4 Character Recognition

The task of feature extraction and selection [35] is determination of a group of most effective features that could be used for classification. As mention earlier, significant features of the characters which our study have used include open end points, number of intersections, number of node points at each intersection and number of holes. Based on these features the Latin and Numeral characters can be classified as depicted by Table 4.1.

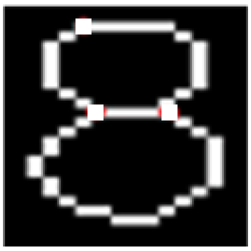
Table 4.1: Characters classification table

<table>
<tr><td>Open End Point</td><td colspan="2">0</td><td>1</td><td colspan="5">2</td><td>3</td><td colspan="2">4</td></tr>
<tr><td>Hole</td><td>1</td><td>2</td><td>1</td><td colspan="2">0</td><td colspan="3">1</td><td>0</td><td colspan="2">0</td></tr>
<tr><td>Node</td><td>1</td><td>6</td><td>1</td><td>0</td><td>2</td><td>5</td><td>3</td><td>5</td><td>2</td><td>3</td><td>4</td></tr>
<tr><td></td><td>O
D
0</td><td>B
8</td><td>P
9
6</td><td>C
G
I
J
L
N
S
U
V
Z
1
2
5
7</td><td>3</td><td>4</td><td>A
R</td><td>Q</td><td>E
F
M
T
W
Y
3</td><td>K
X</td><td>H</td></tr>
</table>

Figures 4.17 and 4.18 depicts classification of a Latin character (‘G’) and the numeral (‘8’).

Chain Code	: 3232324334454555556566776778787871112333
Normalized Stroke Pattern	: 35713
Stroke Pattern with main directions	: 35713
Number of Node-pixels	: 0
Number of open end pixels	: 2
Euler Number	: -1

Figure 4.16: Extracted features for Latin character classification



Chain Code

:04455566604444566776777888811122203333088811122333

Normalized Stroke Pattern : 05057103013

Stroke Pattern with main directions: 05057103013

Number of Node-pixels : 4

Number of open end pixels : 0

Euler Number : +1

Figure 4.17: Extracted features for classification of numerals

Our hybrid license plate detection and recognition system was tested using 150 test images. With two-letter and three-digit notation of North Cyprus license plates this would mean around 750 characters. As depicted in Table 4.2 the license plate localization accuracy of our proposed system was 93.33%. For detected plates the characters could always be segmented. From 700 characters 675 were correctly recognized and this brings the character recognition accuracy to 96.42%. Finally, the accuracy of plate recognition was 92.85%. Please note that only the cases where all characters were correctly detected have been considered as a recognized plate.

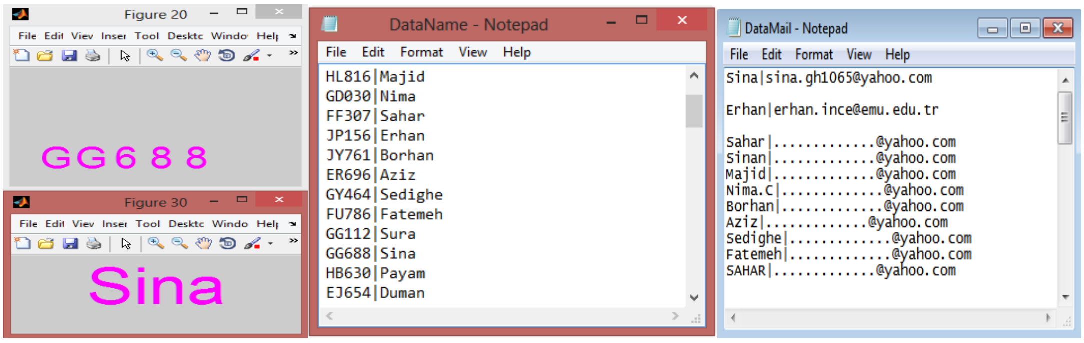
Table 4.2: Performance evaluation of different tasks required by ALPR

	Correct	Incorrect	Percentage accuracy
License plate localization	140 /150	10/150	93.33 %
Character segmentation	700/700	0/700	100 %
Character recognition	675/700	25/700	96.42%
Plate recognition	130/140	10/140	92.85%

4.5 Automated Road Tax Checking

Automated road tax checking stands as one of the most important concern within LPR systems. Nowadays, in overall the world, the road tax of the vehihcles for the aim of repair and maintenance of roads, is collected by police forces. By installing the automatic LPR system, the system permanently will be able to control and identify vehicle owners on the streets. In this project, we propose a new strategy for automating the checking of road tax for different drivers on the road and propose to notify the ones who need to renew their road tax by a formal e-mail message.

The idea is to combine the license plate detection and recognition process with the driver's information stored in a database and an e-mail sending function. Here, after obtaining the plate number as shown in Figure 4.19 (a), a data name file as shown in Figure 4.19 (b) will be searched to locate the owner of the vehicle. The name obtained will then be used to check the road tax statue for the driver and if necessary an automatic e-mail message will be sent out using the e-mail address of the driver as shown in Figure 4.19 (c). The message will be informing the driver about what he should do and also at the same time warn him about what would happen if he does not attend in a given time period.



(a) (b) (c)

Figure 4.18: Driver selection
(a) Results of Character and Owner Recognition, (b) Data Name File, (c) Data Mail File

Send mail function sends message to recipients with the specific subjects. The system has to set user email address and SMTP server information with the setpref function. The setpref function describes two mail-related preferences. Email address sets user email address that will show on the message. SMTP server sets user outgoing SMTP server address, which can be almost every email server that supports the Post Office Protocol (POP) or the Internet Message Access Protocol (IMAP). Sending email function and related message are depicted as follow.

```
sendmail('recipients','subject','message','attachments')
setpref('Internet','E_mail','useraddress@userserver.com');
setpref('Internet','SMTP_Server','mail.server.network');
```

Dear Sina Ghasempour,
Our automated license plate recognition system has reported that your road tax for 2014-2015 has not yet been renewed. You should renew your road tax within a week. If you don’t comply within the given time framework, you will be charged $400.
Best Regards
Polis Security Team

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusions

This thesis proposes a hybrid automatic license plate recognition system that could be coupled with a database search and E-mailing application to automatically monitor road tax status of drivers on highways. The backbone of the proposed method is based on tracing segmented characters to obtain chain-codes and then to extract distinguishing features that could be used for character recognition. In this study along with the extracted features Euler number of Latin and numeral was also used.

Chain-codes obtained through stroke analysis provide the flexibility that recognition analysis would be invariant to style and font size changes and can recognize characters from plates from different vendors.

For 150 vehicle images containing Turkish Cypriot car plates our proposed hybrid ALPR system was able to correctly localize 140 plates and from the localized plates could recognize 675 characters from a total of 700. This brings the character recognition accuracy to 96.42%. Finally, the accuracy of plate recognition was 92.85%.

Since our aim in this thesis was to develop a system that could aid automated check of road-tax status for drivers on the road our system would use the recognized plate number to retrieve details about the driver making use of the registered plate number. If necessary our system would generate and sent a notification e-mail to the driver asking them to pay their fine by a given deadline.

Finally, it is fair to say that if at start a plate is not localized correctly, the processes that follow would be considered unsuccessful and hence character recognition and plate recognition rates would be lower. Therefore it is believed that if the plate localization process is further improved the two recognition rates could be made even higher than current values.

5.2 Future Work

As future work the database of images can be further extended to test the robustness of the proposed hybrid automatic license plate recognition system. Lack of standardization for logos and their location on the plate, as well as inconsistent and multiple alphanumeric template or patterns, the proposed system can be extended to work in different countries which may have different letter and digit combinations.

In the future, the proposed system will be combined with different types of transportation sources, including stationary and moving cameras, GPS (Geographic Positioning Systems) devices, and historical databases and providing coherent and integrated information to users via a web-based interface. Each of the attempts started in this work may be enhanced further taking advantage of the tool built.

REFERENCES

[1] Hov, G., Zhao, J., Liu, M., 2006, "A license plate recognition method based on tophat-bothat changing and line scanning", *Journal of Physics*, vol.48, no.1, pp.431-436.

[2] Wu, H-C., Tsai, C.S., Lai, C-H., 2004, "A license plate recognition system in e-Government", *Int. J. on Information & Security*, vol.15, no.2, pp.199-210.

[3] Alharaki, O. O., Zeki, A. M., 2012, "Image Recognition Technique of Road Tax Sticker in Malaysia", *IEEE Int. Conf. on Advanced Computer Science Applications and Technologies*, pp. 397-401.

[4] Zunino R., Rovetta S., 2000, "Vector quantization for license-plate location and image coding", *IEEE Transactions on Industrial Electronics*, vol. 47, pp. 159-167.

[5] Wei Hsieh, Jun., Hao Yu, S., Sheng Chen, Y., 2002, "Morphology-based License Plate Detection from Complex Scenes", *Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02)*, pp. 30176-30179.

[6] Su Kim, D., Ii Chien, Sung., June 2001, "Automatic Car License Plate Extraction Using Modified Generalized Symmetry Transform and Image Warping ", *Proceedings of IEEE International Symposium on Industrial Electronics (ISIE'01)*, vol.3, pp. 2022-2027.

[7] Zheng, D., Zhao, Y., Wang, J., November 2005, "An Efficient Method of License Plate Location", *Pattern Recognition Letters*, no. 15, pp. 2431-2438.

[8] Wei, W., Wang, M., Huang, Z., 2001, "An automatic method of location for number-plate using color features", *Proc. of Int. Conf. on Image Processing (ICIP'2001)*, vol. 1, pp. 782-785.

[9] Gang, Z.W., Jiang, H.G., Xing, J., 2002, "A Study of Locating Vehicle License Plate Based on Color Feature and Mathematical Morphology", *6th Int. Conf. on Signal Processing*, vol. 1, pp. 748-751.

[10] Kim, S., Kim, D., Ryu, Y., Kim, G., 2002, "A Robust License-Plate Extraction Method under Complex Image Conditions", *IEEE*, pp. 216-219.

[11] Lih Chang, S., Shien Chen, Li., Chung, Yun-Chung., Chen, Sei-Wan., March 2004, "Automatic License Plate Recognition", *IEEE Transactions on Inteligent Transportation Systems*, vol. 5, no. 1, pp. 42-53.

[12] Zheng, D., Zhao, Y., Wang, J., 2005, "An efficient method of license plate location", *ELSEVIER*, *Pattern Recognition Letters*, vol. 26, 2431– 2438.

[13] Jun, X., Sidan, D., Duntang, G., Qinhong, S., 2004, "Locating car license plate under various illumination conditions using genetic algorithm", *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no.1.

[14] Chang, S. L., Chen, L. S., Chung, Y. C., Chen, S. W., March 2004, "Automatic license plate recognition", *IEEE Transactions on Intelligent Transportation System*, vol. 5, no. 1, pp. 42- 53.

[15] Gao, D.S., Zhou, J., August 2000, "Car License Plates Detection from Complex Scene", *International Conference on Signal Processing, Beijing*.

[16] Rao, Sainath, M., Nagaraju, K., August 2009, "A Neural Network Based Artificial Vision System for License Plate Recognition", *Proceedings of National Conference on Recent Developments in Computing and Its Applications*.

[17] Broumandnia, A., Fathi, M., Jan 2005, "Application of pattern recognition for Farsi license plate recognition", *ICGST Journal*, vol. 5, no. 2.

[18] Maro, I., Chacon, M., Alejandro Zimmerman, S., July 2003, "License Plate Location based on a Dynamic PCNN Scheme", *IEEE International Symposium on Computational Intelligence in Robotics and Automation*.

[19] Acosta, Beatriz Díaz., 2004, "Experiments in image segmentation for automatic US license plate recognition", PhD diss., *Virginia Polytechnic Institute and State University*.

[20] Da.Shan Gao, Jie Zhou, August 2000, "Car License Plates Detection from Complex Scene", *International Conference on Signal Processing, Beijing*, pp. 1409-1414.

[21] Maro, I., Chacon, M., Alejandro Zimmerman, S., July 2003, "License Plate Location based on a Dynamic PCNN Scheme", *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 972-976.

[22] Abandah, G., Khedher, M., 2004, "Printed and handwritten Arabic optical character recognition–initial study", *a report on research supported by the Higher Council of Science and Technology*. Amman, Jordan.

[23] Cha, Sung.Hyuk, Y.C. Shin, Sargur N. Srihari., 1999, "Approximate stroke sequence string matching algorithm for character recognition and analysis", Document Analysis and Recognition, *Proceedings of the Fifth International Conference on*, IEEE.

[24] Solera.Ureña, R., Padrell.Sendra, J., Martín.Iglesias, D., Gallardo.Antolín, A., Peláez.Moreno, C., & Díaz.de-María, F., 2007, "SVMs for automatic speech recognition: a survey", *In Progress in nonlinear speech processing, Springer Berlin Heidelberg*, pp. 190-216.

[25] C. Shyang.Lih, C. Li.Shien, C. Yun.Chung et al., 2004, "Automatic license plate recognition", *Intelligent Transportation Systems, IEEE Transactions*, vol. 5, no. 1, pp. 42-53.

[26] Indira, B., et al., 2012, "Classification and Recognition of Printed Hindi Characters Using Artificial Neural Networks", *MECS IJ Image, Graphics and Signal Processing*, vol. 6, pp. 15-21.

[27] Jin, Lisheng, et al., 2012, "License plate recognition algorithm for passenger cars in Chinese residential areas", *Sensors*, vol. 12, no. 6, pp. 8355-8370.

[28] Arica, N., Yarman-Vural. F.T., 2001, "An Overview of Character Recognition Focused on Off-line Handwriting", *IEEE Trans. On Systems, Man, and Cybernetics*, vol. 31, no. 2, pp. 216-233.

[29] Plamondon, R., Srihari, S.N., 2000, "On-line and Off-line handwriting Recognition: A Comprehensive Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 63-84.

[30] K. Deb, A. Vavilin, Jo Kang-Hyun, 2010, "An Efficient Method for Correcting Vehicle License Plate Tilt", *in Granular Computing (GrC), 2010 IEEE International Conference on*, pp. 127-32.

[31] Syafiq bin Haslan, M., 2009, "Special Car Plate Recognition System", degree of Bachelor of Engineering, *Faculty of Electrical Engineering University Technology Malaysia*.

[32] Suliman, A., Nasir Sulaiman, M., 2010, "Chain Coding and Pre Processing Stages of Handwritten Character Image File", *Electronic Journal of Computer Science and Information Technology (eJCSIT),* vol. 2, no. 1.

[33] Sarukhanyan, H., Alaverdyan, S., Petrosyan, G., 2009, "Automatic Number Plate Recognition System", *Seventh International Conference on Computer Science and Information Technologies*.

[34] Nasien, Dewi, Habibollah Haron, Siti Sophiayati Yuhaniz., 2011, "The Heuristic Extraction Algorithms for Freeman Chain Code of Handwritten Character", *International Journal of Experimental Algorithms (IJEA), Publisher: Computer Science Journals (CSC Journals)*, pp.1-20, vol.1, no.1.

[35] Suliman, A., et al., 2011, "Chain Coding and Pre Processing Stages of Handwritten Character Image File." *Electronic Journal of Computer Science and Information Technology*, vol. 2, no.1.