IDENTIFICATION OF SAUDI ARABIAN LICENSE PLATES

BY

Syed Adnan Yusuf

A THESIS PRESENTED TO THE

DEANSHIP OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE

MASTER OF SCIENCE

IN

INFORMATION AND COMPUTER SCIENCE

KING FAHD UNIVERSITY
OF PETROLEUM & MINERALS

January, 2005

Dedication

This thesis is dedicated to my beloved

family

and

all my teachers.

ACKNOWLEDGMENT

In the name of Allah, Most Gracious, Most Merciful

All praise and glory to Almighty Allah (SWT) who gave me courage and patience to carry out this work. Peace and blessing of Allah be upon last Prophet Muhammad (Peace Be upon Him).

Acknowledgement is due to King Fahd University of Petroleum and Minerals for supporting this research.

My unrestrained appreciation goes to my advisor, Dr. Muhammad Sarfraz, for all the material and moral support he has given me throughout the course of this work and on countless other occasions. I cannot simply imagine how the things would have materialized without his guidance. I am thankful for his lenient nature and overlooking patience. I also wish to thank my thesis committee members, Dr. Wasfi Ghassan Al-Khatib, Dr. Mayez Al-Mohamed, Dr. Hussein Al Muallim and Dr. Abdul Waheed M. Abdul Sattar for their help support and contribution. Lastly I thank Dr. Muhammad Farrukh Khan for his help and encouragement.

I also acknowledge my friends and colleagues with whom I spent excellent and fruitful company. Especially I would like to thank my friends Akhtar Ghazi, Jameel, Zeeshan and Adnan Shahab for their help and support. Finally, the ever helpful and always present companion of mine, I don't know how I could I have managed almost everything in this university without Aiman Rasheed, May Allah help him in all his future prospects and intentions.

Special thanks are due to senior faculty members of our community, especially Faisal Alvi and Ajmal Khan.

Finally I wish to express my gratitude to my family members for being patient with me in all their hardships that they incurred in my long absence and offering words of encouragements to spur my spirit at the moments of depression.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THESIS ABSTRACT

Name: Syed Adnan Yusuf

Title: Identification of Saudi Arabian License Plates

Major Field: Information and Computer Science

Date of Degree: December 2004

An Automatic License Plate Recognition (ALPR) System is one kind of Intelligent Transport System. The system's scope in research and development is considerably broad because of a lot of areas still unexplored. The work has been commercialized significantly and systems already exist for Chinese, Korean, US and EU license plates. However, little exploration is being done in the plates bearing Arabic script.

The thesis addresses the problem of an ALPR system for Saudi Arabian License Plates. A general system consists of four main stages: Image acquisition, License plate area extraction, Character block segmentation and isolated character recognition. In the thesis we have proposed techniques based on local car features to reduce the area of search for license plate candidates, image enhancement using intensity adjustment. The algorithms for extraction phase are based on color edge detection based techniques and candidate plates are extracted using a Fuzzy compactness operator and template matching. The segmentation phase stage is performed using a Hybrid fuzzy c means and Image projection profile based technique. Finally the results obtained using Principal Component Analysis (PCA).

The proposed system has been implemented under Matlab 7.1 environment on an Intel PIV dual processor machine. The performance of the system has been investigated on true color images of 852 vehicles captured under various illumination conditions and tilts. The system proves an overall accuracy of 93% showing the system is quite efficient.

CHAPTER 1

INTRODUCTION

With the rapid development of Intelligent Transport Systems, automatic identification of vehicles using license plates has played an important role in a lot of applications. The LPR (License Plate Recognition) technology uses image enhancement, feature extraction and classification techniques to search, locate and identify license plate in an image, segment and recognize characters present and classify the vehicle on the basis of a database working with regional car records present at a backend server. The system can also be made available on a range of networks to harmonize records of vehicles moving on regional basis. The area has already explored intermediate level expects of enhancing image quality and reducing irrelevant information from an image while still leaving enough information for a system to successfully recognize license plate area, characters present, segmented individual characters, define rules based on area or country specific standards and finally the identification of vehicle on the basis of information retrieved. This chapter will first focus on introducing a number of systems at work worldwide. Next, we will go for a number of phases that are commonly processed in a license plate recognition system. Next, we will present the structure of the proposed license plate recognition system. Finally, the objectives of the work are stated. The chapter ends with a brief review of the rest of the thesis.

1.1. LPR Applications

LPR technology has been used around the world for a variety of applications including:

Security Imaging System Applications:

- International Border Control
- Security and Access Control
- Military Base Surveillance
- Industrial and Nuclear Plant Security

Traffic Management Applications:

- Traffic Law Enforcement
- Parking Lot Access Control
- Port and Shipping Traffic Management
- Electronic Toll Collection Enforcement

A Commercial LPR System generally:

1. Reads the alphanumeric code on license plates
2. Operates accurately in most weather conditions
3. Captures license plates on vehicles traveling at highway speeds
4. Works 24 hours a day, 7 days a week in all conditions

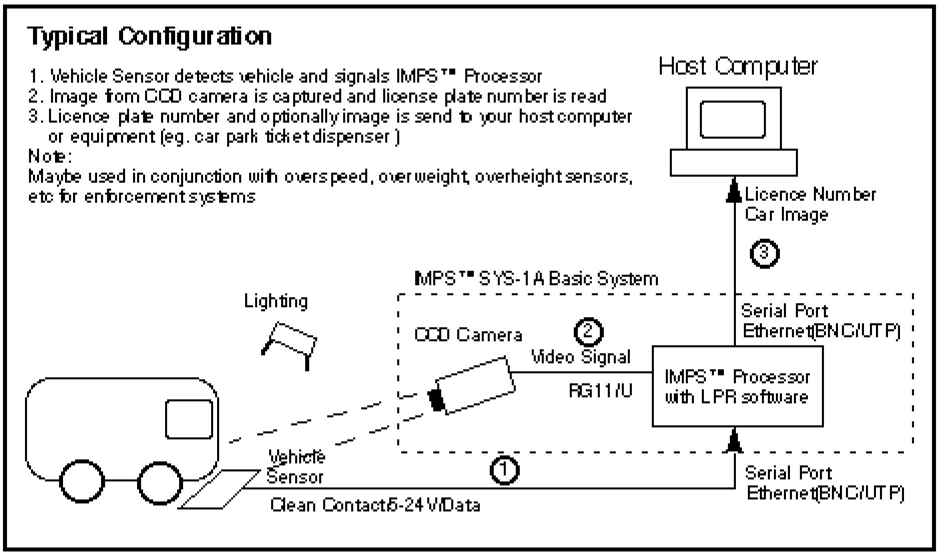
Figure 1-1 Toll Collection

Figure 1-2 Restricted Access

Figure 1-3 Automated Container Identification

The system has its application in many areas where one wants to restrict car speed, monitor traffic flow at signals, record parking statistics or identify car thefts. It provides an automated environment to detect and identify a vehicle presence through its license plate. The detection of vehicle is applied to advisory of congestion, occupancy management of parking areas and surveillance of illegally parked vehicles. The classification is utilized for the electronic toll collection system (ETC) and to display available parking spaces to vehicles [1]. The identification is also employed for managing container transport facilities, monitoring and analysis of travel time, and security system such as observation of stolen vehicles and monitoring of unauthorized vehicles entering parking areas.

There are multiple commercial license plate recognition systems available [66], [67]; however a majority of the systems available are for foreign plates only. Applications such as SeeCar [68], Perceptics [69], and Pearpoint [70] are available in the United States and are able to identify license plates in all states. There are commercial products for European and some of the Asian countries too. The most popular United States application is SeeCar by Hi Tech Solutions. This application detects and reads vehicle license plates for parking, access control, traffic surveillance, law enforcement and security applications. In addition to these commercial products, some of the research projects include: ESPRIT 5184 LOCOMOTIVE [9], software architecture thru DLL's [10]. Until now, to our knowledge, there is no software available for the recognition of Saudi Arabian license plates. The proposed work would be the first attempt towards the development of such an LPR system. Although there is some research work done in LPR [2] but this work only addresses the issue of static number plates with constraints and limitations pertaining to highly tilted, overly illuminated images and those plates with almost same car color compared to the license plates.



Typical Configuration

1. Vehicle Sensor detects vehicle and signals IMPS™ Processor
2. Image fom CCD camera is captured and license plate number is read
3. Licence plate number and optionally image is send to your host computer or equipment (eg. car park ticket dispenser)

Note:
Maybe used in conjunction with overspeed, overweight, overheight sensors, etc for enforcement systems

Figure 1-4: Integrated Multi-pass System by OPTASIA

1.2. Components in a Typical LPR System

LPR system normally contains the following components

- Camera – Takes either the front or rear image of a vehicle.
- Illumination – A light source that could bring up the license plate.
- Frame Grabber- An interface card establishing a connection between the imaging device and the PC allowing the software to read the image information.
- Computer – A personal computer running over Windows or Linux operating system. Runs the LPR application that controls the system flow, reads the

images, analyzes and identifies the license plate(s) and interfaces the front-end application over a network with a central server or database.

- Software – The application program.
- Hardware – Various interface cards, network hubs, servers and control switches.
- Database – The events are recorded either at a central server or locally at a back-end database.
- Generally, Image-Based vehicle recognition is categorized into detection, classification, and recognition.

1.3. Proposed System Structure

The system presented is designed to read License Plate System from the rear and front of the car though it is advised that the whole process be carried out with the snaps taken from the back of the car. Following are a few reasons to support this issue:

1. Drivers who want to conceal their car plate normally try to hide their car front behind the back of the front car by driving too close to it.
2. Normally the cars do not contain information at the back that could confuse a license plate extracting mechanism such as labels, stickers, etc.
3. Plates at the back of the car are normally cleaner and clearer.
4. Backlights do not hinder a system's recognition capabilities as does the front lights.

5. In some countries there is no concept of front car plates (for example MI, United States).

The input to the system is an image or frame sequence acquired through a digital or CCD (Charge Coupled Devices) camera. The input probably contains a scene containing license plate(s). The output of the system is the sequence of recognized characters present on the license plate. The system is conventionally standardized into four main modules, viz. Image Acquisition, License Plate Extraction, Character Segmentation and License Plate Identification. The structure of the system is shown in Fig 1.5.

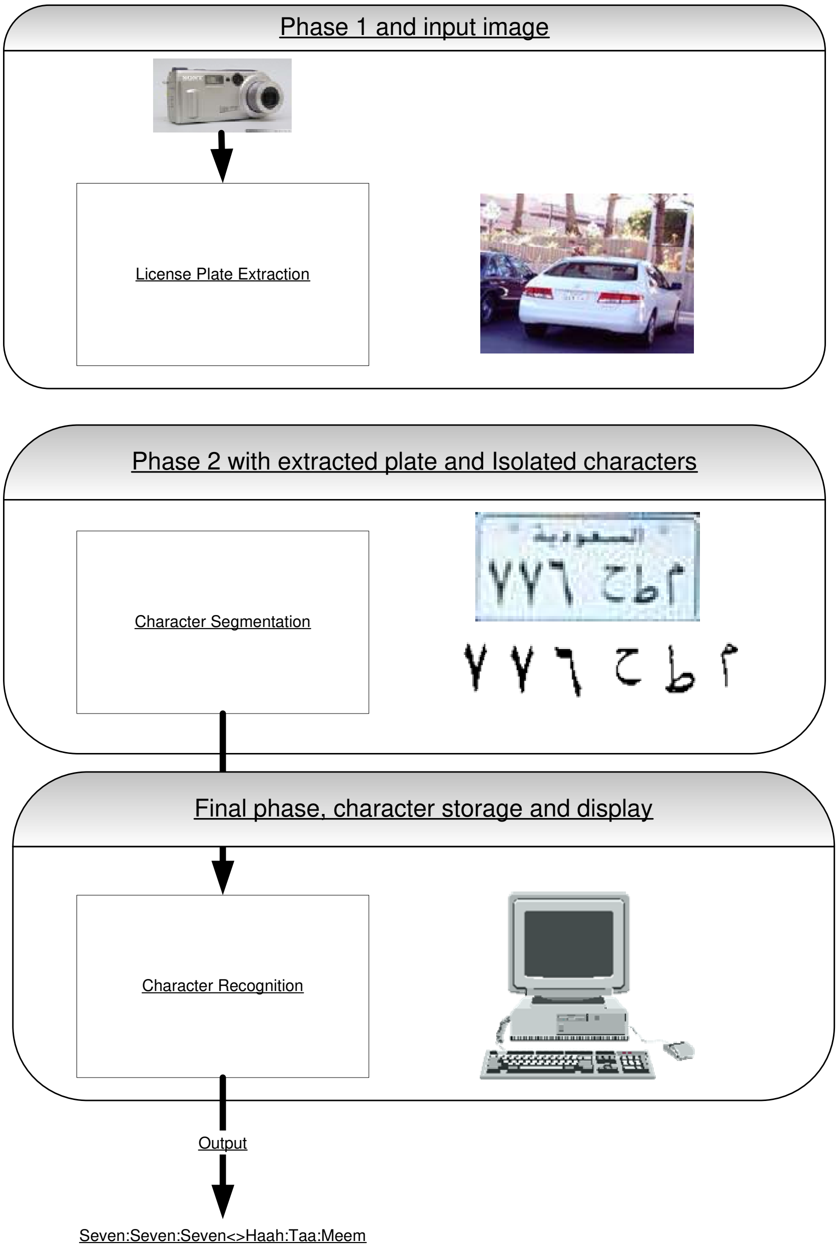

Phase 1 and input image
License Plate Extraction
Phase 2 with extracted plate and Isolated characters
Character Segmentation
Final phase, character storage and display
Character Recognition
Output
Seven:Seven:Seven<>Haah:Taa:Meem

Figure 1-5: The typical structure of a standard LPR system

1.4. Image Acquisition

This is the first phase in the LPR system. There are a number of ways of acquiring an image and transferring it to a computer for further processing:

1. Using an analogue camera and a scanner.
2. Using a digital camera
3. Using a video camera and a frame grabber capture card to select a frame

The first method using a conventional analogue camera is clearly not appropriate for the LPR system as it is time consuming, tedious and impractical. The second method, i.e., using a digital camera is more convenient, cost effective and reliable. The third one uses a video camera with a frame grabber which is used in real life system to make the system automated and is suitable for real time processing. Figure 1.6 shows a sample result of this stage, the figure shows a front end of the vehicle.

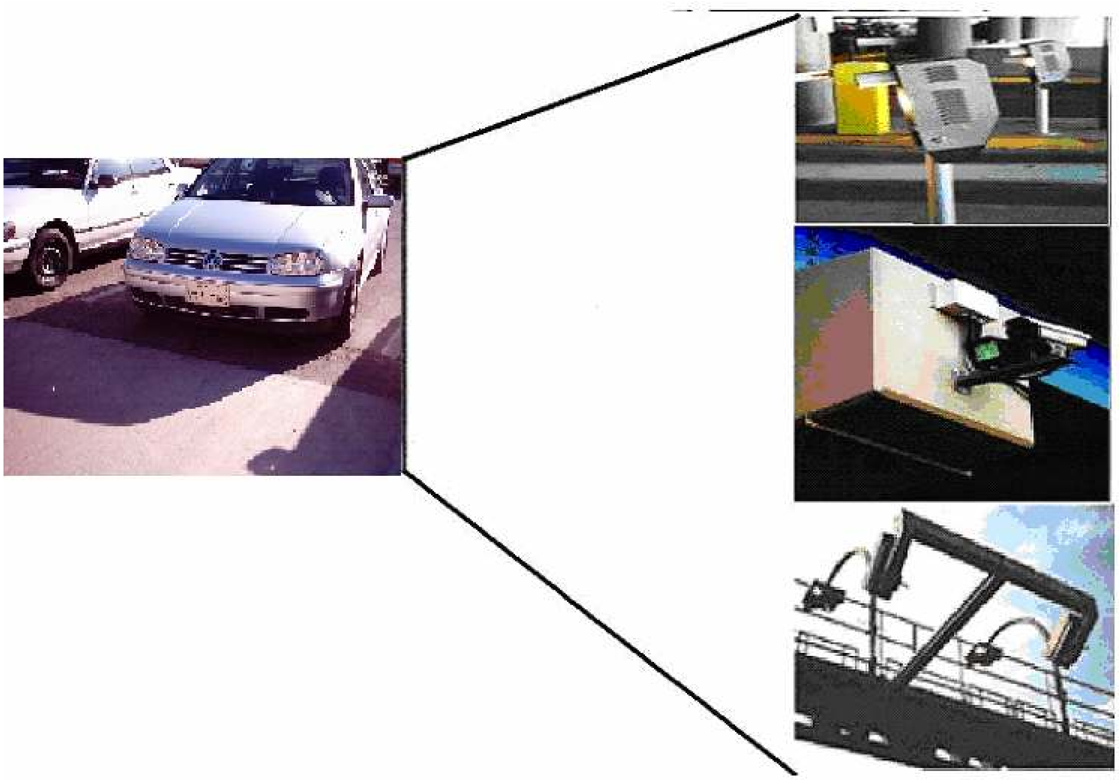


Figure 1-6: Typical High-Speed Image Capturing Device

In our proposed system we use a high resolution Kodak DX-3600 digital camera. The images are taken at a resolution of 900×600.

1.5. License Plate Extraction

The extraction of license plate area is a key step in an LPR system, which influences the accuracy as well as the computational cost of the system significantly. This phase extracts the region of interest (ROI) from the image acquired. The proposed approaches are as follows:

1. Image Enhancement and Intensity Adjustment using Histogram Equalization.
2. Probable license plate presence detection using horizontal Sobel mask.
3. License Plate Area Localization using
    - Edge Map based on RGB Vector Angle [3] based Fuzzy Technique
    - Edge Map based on RGB Euclidean Distance [4] based Fuzzy Technique
4. License Plate Candidate Selection on the basis of Fuzzy Compactness and Template Matching.

These steps will be discussed in detail in the Section 6.2.

1.6. License Plate Segmentation

The phase of License Plate Segmentation takes the ROI and attempts to divide it into individual characters. The final output divides the ROI into six sub-images, sequenced from left to right of the license plate, each containing a single isolated character/numeral. Since a standard Saudi Arabian license plate consists of six characters, with 3 letters and 3 numerals. The proposed approach discusses segmentation using two approaches; Fuzzy C-Means clustering and a Hybrid approach based on Fuzzy c means and pixel projection profiling. This will be elaborated in Section 6.3.

1.7. License Plate Recognition

The last phase in the LPR system is to recognize the isolated characters. The sub images of the six characters generated in the License Plate Extraction phase are fed to a recognition module. The techniques we have investigated are based on a module investigating feature components using Principle Component Analysis (PCA).

1.8. Objective

The work presented in this thesis aims at the following aspects.

- Study of the Intelligent Transport Systems related to Car Recognition,
- An analytical as well as research based survey of latest image processing techniques used in image enhancement and object recognition,

- Development/Enhancement of techniques for all the phases involved in the LPR,
- Performance comparison of the present techniques with those proposed,
- Identification of the performance bottlenecks of such systems,
- Identification of the recovery procedures that might help improve success rates in all the phases, and
- Building a system that could operate under various environments, lighting conditions and accidental skews with closely matching accuracy and speed.

1.9. Thesis Organization

The thesis presents a complete system for the recognition of Saudi Arabian license plates. The structure of the thesis document is as follows:

- Chapter 2: A literature review of the previous research in this area. A survey of the techniques used in the four phases of the LPR system followed by an overview of the latest LPR systems available worldwide. The chapter also discusses a number of commercialization issues for such a system as well as a number of applications in image processing industry.
- Chapter 3: A brief introduction to the state-of-the-art image processing techniques used in to enhance the performance and accuracy of such applications. The techniques specifically include color edge detection, region growing, morphological operations, normalization and contrast adjustment.

- Chapter 4: Some insight into shape analysis and realization of geometrical figures in fuzzy domain. The analysis includes features of fuzzy image processing.
- Chapter 5: The chapter addresses the issue of image intensity and enhancement in images affected by noise distortion.
- Chapter 6: The chapter discusses the conceptual layout, design and implementation of the system proposed.
- Chapter 7: The chapter contains experimental analysis, results and the performance evaluation of the results obtained at different stages.
- Chapter 8: Conclusion about the work done and its future prospects.

CHAPTER 2

LITERATURE REVIEW

License Plate Recognition (LPR) is considered a major area in the field of Intelligent Transport Systems (ITS). The system still needs significant improvements in order to be to be used in a totally unrestricted, natural environment. The limitations and future possibilities of LPR systems were discussed in the United States Department of Transportation's Field Operational Test (FOT) [3] carried out under the auspices of Federal Highway Administration for LPR/Vehicle Imaging and is narrated as follows: *"LPR continues to demonstrate its limitations as clearly as its usefulness. During the course of the WI/MN OOS FOT, several hundreds of reads were attempted with limited success. Of 3,460 attempted reads, 1,413 were successful in correctly interpreting the license plate information, for a success rate of 40.8 percent. Unsuccessful reads fall into two categories: "no reads" and "bad reads." Reasons cited for "no reads" included missing, damaged, or dirty license plates. No reads accounted for 27 percent of the unsuccessful attempts. "Bad reads" were attributed to misinterpretation of the license data, often caused by different styles and colors of various state plates. "Bad reads" accounted for 32 percent of the unsuccessful reads. Excluding unreadable plates, the success rate was 56 percent."*

Although there has been a lot of research in the area of License Plate Recognition, the work has a nation-wide scope only. One reason being the lack of standardization between different license plates. Because of different standards adopted worldwide, license plates differ in shape, size, character locations, logos, etc. Most of the research is done in Korean, Chinese, Dutch and English License Plates. This section gives an overview of the research carried out so far in this area and the techniques employed in developing an LPR system. The system mainly consists of Image Acquisition, Extraction, Segmentation and Recognition Phases.

Automatic license plate recognition has an important role in numerous applications such as Unattended Parking Lots [11] [12], security control of restricted areas [13], Traffic law enforcement [14] [15], congestion pricing [16], and automatic toll collection [17].

Typically an LPR process consists of two main stages:

1. Locating license plates, and
2. Identifying license numbers.

2.1. Image Acquisition

After taking a snap of the vehicle, Image Acquisition proves to be the first step in an LPR system. There are a number of ways to capture images and a lot of work has been in the image acquisition of license plates. Yan et. al. [18] used an image acquisition card that converts video signals to digital images based on some hardware-based image preprocessing. Naito et al. [19], [20] developed a sensing system, which uses two CCDs (Charge Coupled Devices) and a prism to split an incident ray into two lights with different intensities. The main feature of this sensing system is that it covers wide illumination conditions from twilight to noon under sunshine, and this system is capable of capturing images of fast moving vehicles without blurring. Salgado et. al. [9] used a Sensor Subsystem having a high resolution CCD camera supplemented with a number of new digital operation capabilities. Comelli et al. [20]

used a TV camera and a frame grabber card to acquire the image for the developed vehicle license plate recognition system.

2.2. License Plate Extraction

The second phase is extraction and is an important phase in any LPR system. Here we discuss some of the techniques used in the literature on license plate extraction. The orientation of the license plate is determined by the local features of the license plates such as Histogram Stretching [21], Size and Aspect Ratio comparison [1] [22], Bounding Box matching, Fuzzy Segmentation, Vertical Edge Matching [24],. Character features include blob analysis [25], Aspect ratio of characters [26], the distribution [27] and alignment of characters [28]. In reality a universal, small and easy to detect features are would suffice.

Hontani et al. [25] proposed a method for extracting characters without prior knowledge on their position and size in the image, which is based on scale shape analysis and scale shape analysis, is in turn based on the assumption that characters have line-type shapes locally and blob-type shapes globally. In the scale shape analysis the given images are blurred by Gaussian filters at various scales and larger size shapes appears at larger scales. To detect these scales the idea of principal curvature plane is introduced. By means of normalized principal curvatures, characteristic points are extracted from the scale space x-y-t. The position (x, y) indicates the figure positions and the scales t indicates the inherent characteristic sizes of corresponding figures. All these characteristic points enable to extract figures from the given images that have line-type shapes locally and blob-type shapes globally.

Kim et al. [29] used two neural network-based filters and a post processor to combine the two filtered images in order to locate the license plates. The two neural networks used are vertical and horizontal filters, which examine small windows of vertical and horizontal cross sections of an image and decide whether each window contains a license plate. Cross-sections have sufficient information for distinguishing a plate from the background. Lee et al. [30] and Park et al. [31] devised a method to extract Korean license plate depending on the color of the plate. A Korean license plate is composed of two different colors – one for characters and other for background and depending on these they are divided into three categories. In this method a Neural Network is used for extracting color of a pixel by HLS (Hue, Lightness and Saturation) values of eight neighboring pixels and a node of maximum value is chosen as a representative color. After every pixel of input image is converted into one of the four groups, horizontal and vertical histogram of white, red and green (i.e. In case of Korean Plates, as it contains white, red and green colors) are calculated to extract a plate region. To select a probable plate region horizontal to vertical ration of plate is used. Dong et al. [32] presented histogram based approach for extraction phase. Kim G. M [33] used Hough transform for the extraction. Mei Yu et al. [24] discuss a vertical edge based matching technique for locating the license plate region.

2.3. Segmentation and Recognition

The License Plate candidate(s) determined in the locating stage are examined for the presence of the plate characters or character signature. The major tasks involved in this stage are character segmentation and identification. Character separation in the past is being done by techniques such as projection profiles [34] [9], morphology [27], connected components [22] and blob coloring.

Core issues that make a significant difference in the localization of license plates and affect the recognition rate in the later stages are the factors changing Illumination Conditions such as Fog, Background Light Sources, Dirty Plates, Overly Illuminated Plates, etc. An image with low background light and a white number plate can be extracted by using a regular or K-means region-growing algorithm [35]. This technique generates poor results for cars with white number plates and lighter backgrounds and requires closer limits to threshold values on the conventional region growing algorithms. Increasing the number of seeds to achieve a better region growing appears to be computationally expensive. A morphology based technique is suggested for such conditions [36]. Again, the technique fails for overly illuminated images and a need arise to use image enhancement techniques for the recovery of distorted characters [37]. In Gray-Scale images, edges are typically modeled as brightness discontinuities. These discontinuities are employed by most edge detectors using some form of difference operator on neighboring pixels [38]. These discontinuities greatly reduce license plate area locating accuracy if the car and license plate color is similar and there is no visible line of demarcation among the two. On the other hand these techniques generate unwanted edges in much greater numbers.

In the literature Segmentation and Recognition steps are combined and discussed commonly under the recognition phase, some of the previous work in the recognition of characters is as follows: Cowell et al. [39] discussed the recognition of individual Arabic and Latin characters; this approach identifies the characters based on the number of black pixels rows and columns of the character and compares these values to a set of templates or signatures in the database. Cowell et al. [40] discusses the thinning of Arabic characters to extract essential structural information of each character which may be later used for the classification stage. Mei Yu et al. [24] Naito et al. [19] uses template matching, Hasen et al. [41] discusses a statistical pattern recognition approach for recognition but is found to be inefficient.

Here we discuss some literature which incorporates different techniques for different phases of the LPR system. Mei Yu et al. [42] proposed two simple approaches land mark based method and BS & Edge methods for vehicle detection and shadow rejection. Based on these two methods, vehicle counting, tracking, classification and speed estimation are achieved. AbdelMalek et al. [43] used MCR expression for recognition of Arabic text without segmentation using a structural pattern recognition approach. Lotufo et al. [17] proposed automatic number plate recognition using optical character recognition techniques. Johnson et al. [44] proposed knowledge guided boundary following and template matching for automatic vehicle identification. Fahmy [45] proposed BAM neural network for reading number plates. Fuzzy logic and neural network was used by Nijhuis et al. [22]. Choi [46] and Kim et al. [31] proposed the method based on vertical edge using Hough Transform to extract license plate. Lee et al. [30] used neural network for color extraction and template matching to recognize characters. Kim et al. [47] used genetic algorithm based segmentation to extract the plate region but this method was found to be time consuming.

2.3.1. Segmentation using Blob-coloring and NN Recognition

Botha et al [48] proposed a PC Based Number Plate Recognition. The system thresholds gray-level images of cars using Niblack algorithm. The reason for using Niblack algorithm is because of its robustness against shadows in the images and other image defects. The algorithm calculates a local Binarization threshold by calculating the local standard deviation and mean and then adding the mean to the product of a predefined weight constant and standard deviation:

$$T(x, y) = w \times \sigma(x, y) + \mu(x, y) \tag{2-1}$$

Where T is the threshold at pixel (x, y), w is the weight, and $\sigma(x, y)$ and $\mu(x, y)$ are the standard deviation and mean of the local neighborhood of the pixel (x, y) respectively.

Botha et al [48] used a set of rules for Digit location. The technique iterates through all the pixels in an image and checks at every digit if there is a candidate digit at the current position. Draghici [49] used expected alphanumeric size as a criterion for a candidate digit. The assumption can be considered if the distance between the car and the number plate remains constant. The same author considered another issue of "Pixel" coverage checking. ON pixels in an area were counted and percentage calculated. If the 'ON' percentage appeared 15% or above the position was classified as a potential alpha-numeric character, if not it's disqualified.

Plate area location was detected by using the digit like entities detected in the previous stage. The attempt was to determine the geometries (i.e. location and size) of all candidate plate areas.

The approach analyzed contiguous regions. Blob-coloring is a region growing method which operates on binary images. It labels pixels which form 8-connected contiguous regions (“blobs”) each region receiving a unique label. It thus “colors” the “blobs”.

2.3.2. Usage of Morphological Operations for License Plate Detection

Mathematical Morphology or Image algebra is defined as the study of shape or form. It is the study of shape using the tools of set theory. It is an area of images that uses set-theoretic operations of images.

Mathematical Morphological operations were used method by Arregui et al [43] to detect the license plate location. A shape recognition technique was used by Cimmins et al [51] on performing hit and miss transforms using different sizes of every possible character as a structuring element. The approach proved expensive in terms of very expansive computation. The results proved inaccurate because of distortion in the images due to noise, position of the camera and uncertainty of background images.

The binary morphological operations involved Binarization followed by a closing operation to merge the characters thus leaving a white rectangular image at the location of the license plate. Unwanted white regions in the image are eliminated by an opening operation. In addition, contours of these white regions can be extracted and a series of parameters such as area, perimeter compactness, and center of gravity are calculated.

Poon et al [27] Gray level transitions are determined by taking the first order difference of the digitized image and in the location of the license plate region they form a cluster. In order to decide a region from other regions, a justifiable method based on density, amplitude and width of the cluster is used. The algorithm uses the features that may be present rarely in other regions besides the license plate.

2.3.3. Usage of a Real-Time Character Classification and Recognition

The system proposed and tested by Bailey et al [52] used a real-time IVP-150 TMS320C50 digital signal processor based Video Image Processing System. The preprocessing phase of the system involved brightness correction, contrast correction and noise filtering. Next, a reliable thinning algorithm will be applied to simplify the image into lines of only one pixel thickness while retaining the shape and character properties and features of the image. The system used a rule based features of the characters. The system provides an advantage of real-time digital signal processing and provides real-time recognition results.

The system can work with speed detector cameras and red light cameras on the road. The same algorithm can be used for handwritten character recognition applications.

The system has a few problems:

1. Thinning is a tricky process because important features of the image need retaining.

2. The features of the characters may not be consistent, especially if there is a significant tilt present.

3. Improper image capturing result in non-distinct characters.

2.4. Commercially available LPR Systems

2.4.1. DPL Surveillance Equipments

DPL-LPRS-3000 [53] License Plate Recognition System is designed to work at a video input feed of 120 fps and can handle 40 Million car inputs.

2.4.2. Performance Measures:

Its special features include usage of massive parallel networks that utilize Neural Networks for classification eventually claiming an accuracy of 100%.

2.4.3. Algorithms Used

The algorithm used by them basically interprets a line filtering mechanism based on adaptively threshold edge maps to pertain the position of the license plates. It has three separate image device triggering mechanisms, namely, manual, external sensor based and software based internal motion detection modules. The system can handle 4 plates simultaneously and handles the following tasks:

1. Saves the License Plate Image
2. Shows Notification Message.

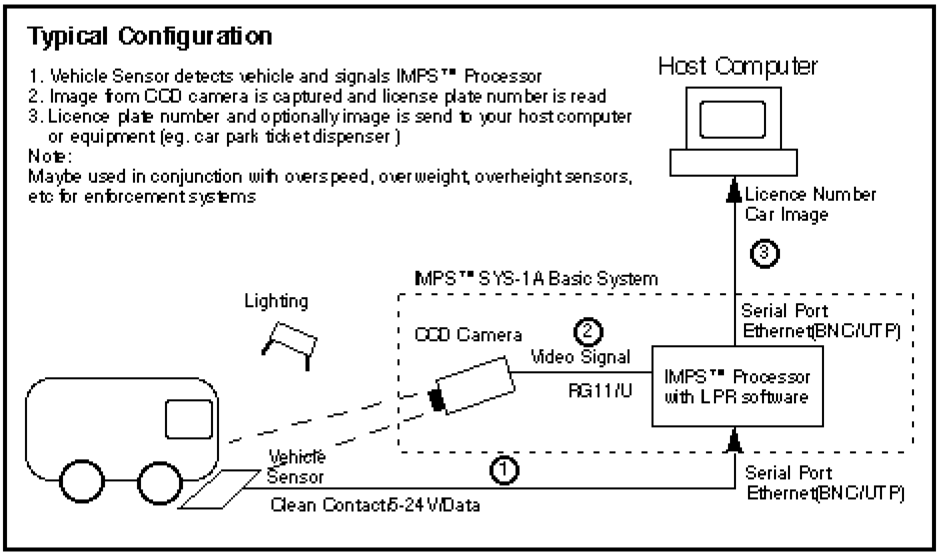
3. Supports an informative Sound Base.
4. Activates an I/O device
5. Manually edits License Plate’s ID.

The system is able to handle images taken under high speed, hostile and ambient environment and severe weather conditions. All this is achieved using most of the software logic built-in (integrated) over hardware chips.

The system is able to recognize correctly at a skew of 0~30 degrees.

2.4.4. IMPS (Integrated Multi-pass System)

The system [54] works in China, Hong Kong, Malaysia and South Africa. The algorithm used a best first breadth-wise search algorithm in a combination of template and neural network-based classifiers, fuzzy logic and a number of image processing and enhancement technologies.



Typical Configuration

1. Vehicle Sensor detects vehicle and signals IMPS™ Processor
2. Image from CCD camera is captured and license plate number is read
3. Licence plate number and optionally image is send to your host computer or equipment (eg. car park ticket dispenser)

Note:
Maybe used in conjunction with overspeed, overweight, overheight sensors, etc for enforcement systems

Figure 2-1: The vehicle sensing system uses standard inductive loop sensors or internal software triggering (day and night).

2.4.5. Performance Measures

The vehicle detection rate for this system is 95%. For the Singaporean system the identification accuracy is 99.7%. The response time ranges from 0.4 to 2 seconds.

2.4.6. Image Format and Snapping Conditions

The system uses single field image from standard monochrome CCTV (CCIR) cameras. The image size is 768×288. The system is able to handle horizontal and vertical angles of 45 degrees. Artificial lighting is a requirement and must be mounted near and above the camera. IR Cameras are required to bypass high headlight glare present in the case of front license plates at night.

2.5. Perceptics

The company [55] is involved with the design and development of Security, Traffic Management and Computer Vision based systems.

2.5.1. Performance Measures

The system converts the array of information (Image) into an ASCII string containing the plate number. The image size is roughly 512×512. The recognition library presents the following advantages.

- Can integrate into existing application based on VB/VC++ environment.
- Has a multiple letter/digit check providing with a high recognition rate.
- Covers a wide range of plate size (80-300; recommended 150 pixels side-to-side) and deals with very small plates (down to ~80 pixels per plate on typical plates)
- Deals with a wide range of contrast images (recommended: 50 gray levels font to background)
- Rotation allowed ± 30 degrees (depends on plate type; recommended smallest
- angle)
- Fast response - Pentium 1900 MHz at typical25msec per image to return the recognition string. This allows the user's application to utilize most of the PC resources.
- Can operate over standard PCs.
- Is adaptable to different country standards.
- Available in 32bit versions, for use with standard compilers

2.6. Zamir Recognition Systems (Ltd)

The company [56] introduced its all-in-one Insignia 4 License Plate Recognition System on March 02, 2004. The system consists of a Lane Controller (LC) housing an imaging sensor that enables a 24 hour all weather functionality. The system works in the domains of Photo-optical Instruments, Entrance and Parking Control Systems, Vehicle Detection, Infrared Detection and Number Plate Recognition.

2.7. Real Time Traffic Sign Recognition (TSR)

The first works related to the road sign recognition have been published in Japan in 1984. The aim was to try various computer vision methods for the detection of the objects in outdoor scenes. At the beginning of nineties, several groups interested in the subject emerged. We may found there solutions focused on particular road sign type together with more general systems. There were reported various approaches for the sign detection (employing of edges, color segmentation, correlation etc.) in these works. The classification step has been - in most cases - solved by the use of neural network.

The compilation of [72] from CRIM (Canada) the most valuable information source describing systems before 1995. Presently, there exist several groups, involved in the road sign recognition, in the world. Without doubt, the most advanced system is the German TSR (Traffic Sign Recognition System) developed at the University Koblenz-Landau in cooperation with Daimler-Benz.

Following are the specific features present in the developed application:

2.7.1. Real time application

The CSC-TSR, a fast, robust color image evaluation system for the detection of traffic signs on European highways, is installed in a driving car. The real-time ability is achieved with parallel implementation on a TIP-system *(Parsytec Transputer Image Processing)* with PowerPC processors *(Motorola MPC).*

2.7.2. Cooperation

The CSC-TSR has been developed in cooperation with *Daimler Benz* within the European PROMETHEUS project (Programme for a European Traffic with Highest Efficiency and Unprecedented Safety).

It has been integrated into the autonomous car VITA II.

2.7.3. Performance

- Database of approx. 40.000 traffic scenes
- Analysis of 3 images per second
- Recognition rate of 98 %

2.7.4. Image Processing with TIP system

- Parallel architecture developed by Parsytec.
- Special BUS-system for online image-transfer.
- Special processing nodes for graphical in-/output. Color Frame Grabber (CFG), Color Graphics Display (CGD).

- Processing nodes with T805-Transputer (VPU).

- Integration of MPC processing nodes with Motorola PowerPC 601 processor (MPC).
- Integration of Texas-Instruments C40 processing nodes.

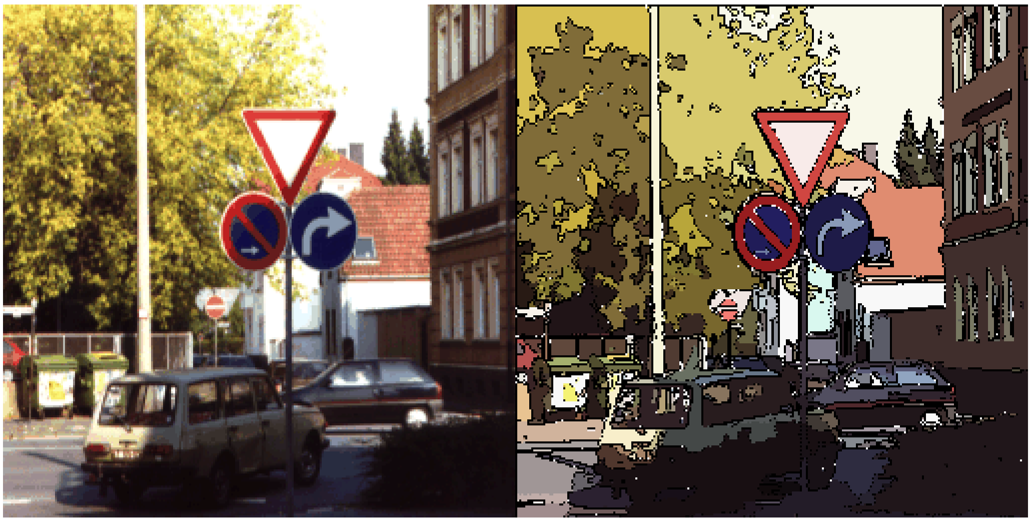


Figure 2-2: Example of a CSC-segmented scene, where all found objects (trees in the CSC data structure) are presented by the outer border and the involved pixels in the mean color of the object. [72]

The group's work related to the issue of color segmentation. The areas addressed are as following:

1. CSC (Color Structure Code)
2. Split and Merge
3. Recursive Histogram Splitting

The results of these algorithms are available over a number of real time road images as shown in Fig. 2-4 and Fig 2-5.

2.7.5. Color Structure Code (CSC)

The CSC is an inherently parallel hierarchical color segmentation method that can operate on distributed data on an optional number of processors. The CSC operates on a hierarchical overlapping hexagonal topology. This leads to inherently parallel algorithms and combines the advantages of local region growing (simplicity and quickness) and global techniques (robustness and accuracy). An example of such a 3x3 image grid on a processor is shown in Fig. 2-3.

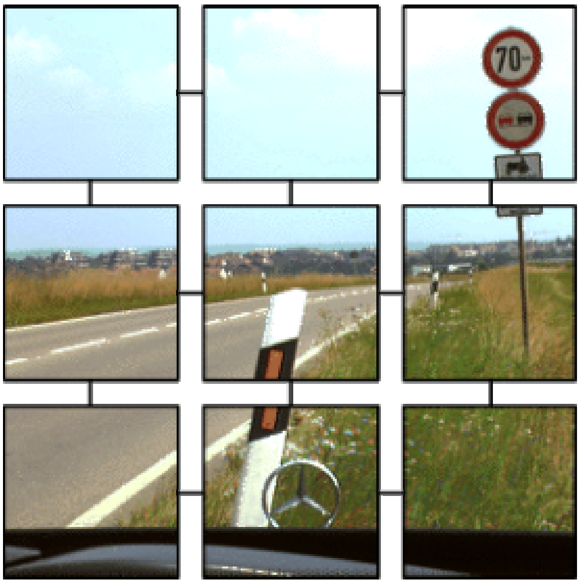


Figure 2-3: An image on a 3x3 processor grid.

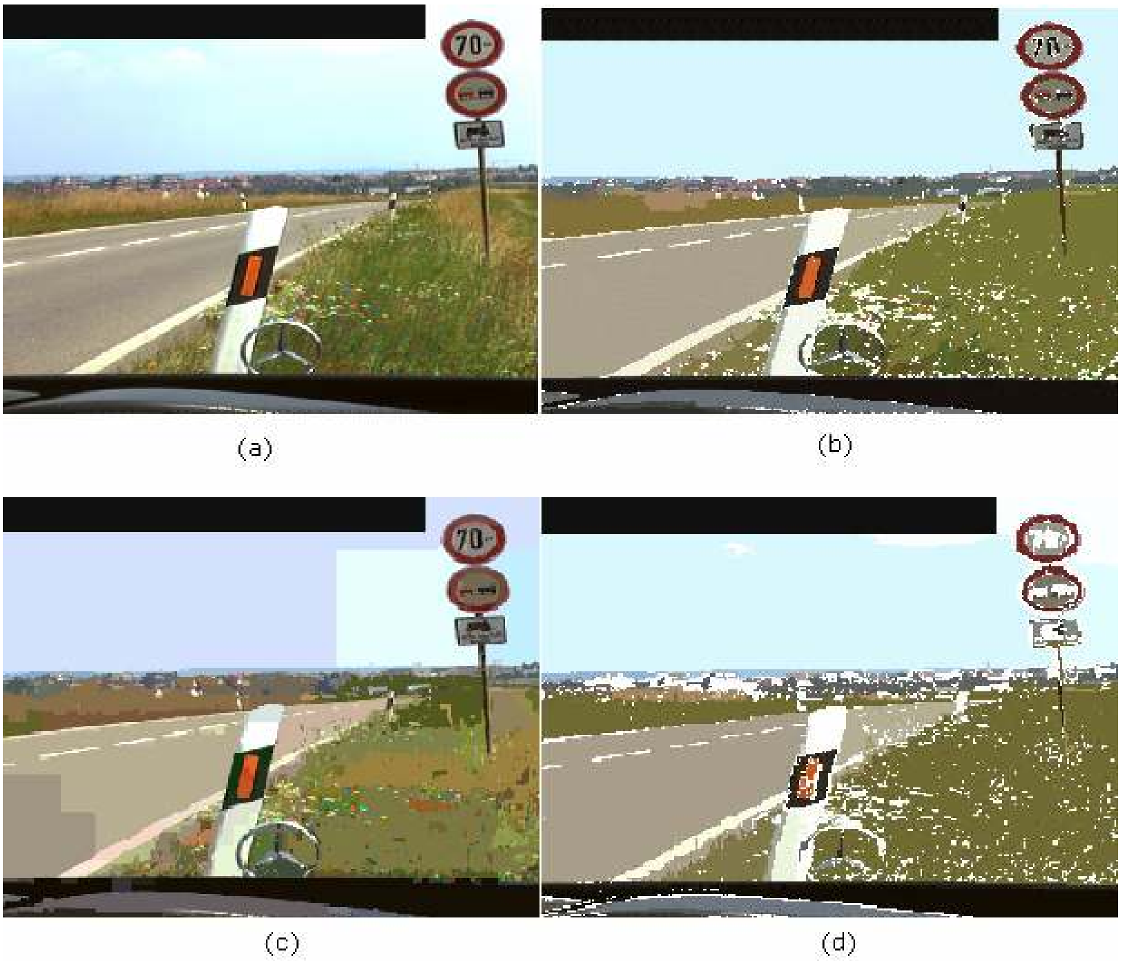


Figure 2-4: (a) An original image. (b) CSC Algorithm. (c) Split and Merge. (d) Recursive Histogram Splitting

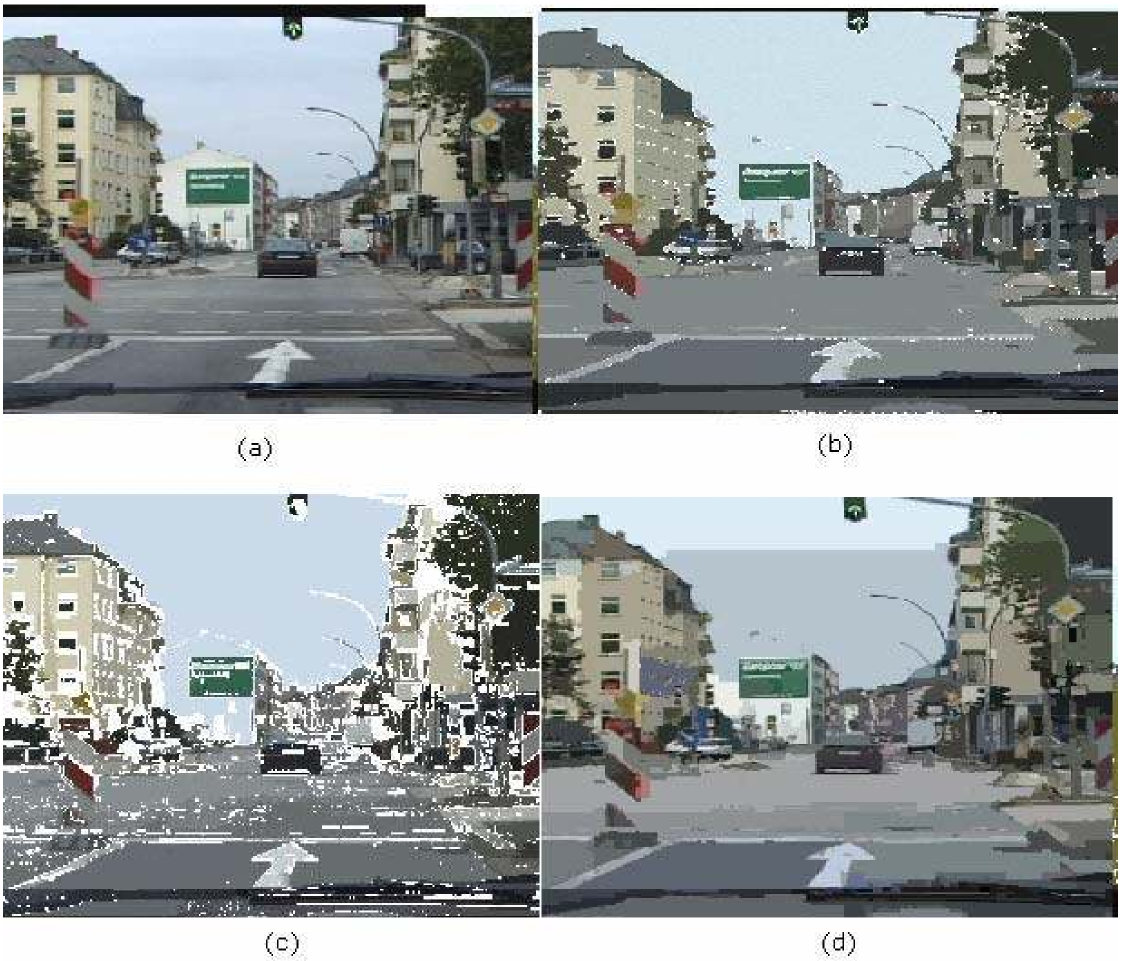


Figure 2-5: (a) An original image. (b) CSC Algorithm. (c) Recursive Histogram Splitting (d) Split and Merge.

2.8. Iris Recognition

The common procedures now employed for proving one's identity—the use of passwords, personal identification numbers, picture IDs and so forth—have given way in many situations to automated biometric analysis. One such system takes advantage of the detailed patterns within a person's iris, which make it possible to identify someone using nothing more than an infrared image of the eye and a suitably programmed computer to process the information. The algorithm for iris recognition discerns whether two images taken at different times are of the same iris. The scheme encodes iris patterns compactly, so that comparisons can be made extremely quickly and tests against large numbers of candidate images can be performed in a reasonable time when searching for a match.

2.8.1. Iris Scan

Iris scan biometrics employs the unique characteristics and features of the human iris in order to verify the identity of an individual. The iris is the area of the eye where the pigmented or colored circle, usually brown or blue, rings the dark pupil of the eye as shown in Fig. 2-6.

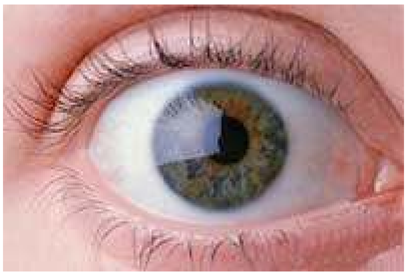


Figure 2-6: Human Iris

The iris-scan process begins with a photograph. A specialized camera, typically very close to the subject, no more than three feet, uses an infrared imager to illuminate the eye and capture a very high-resolution photograph. This process takes only one to two seconds and provides the details of the iris that are mapped, recorded and stored for future matching/verification. Eyeglasses and contact lenses present no problems to the quality of the image and the iris-scan systems test for a live eye by checking for the normal continuous fluctuation in pupil size.

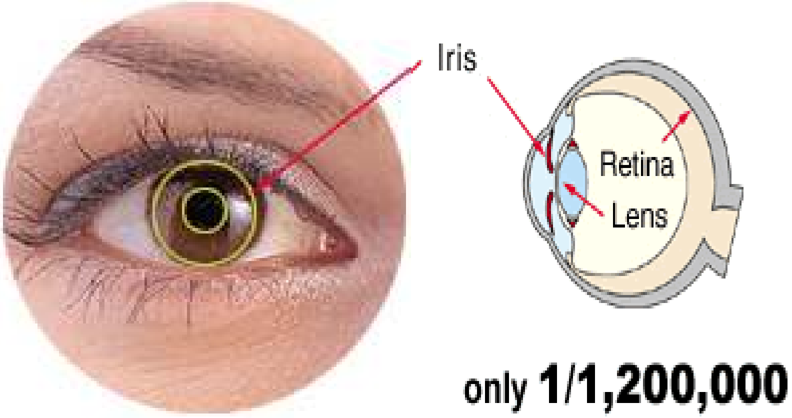


Figure 2-7: Identification technology based on individual Iris Patterns

The inner edge of the iris is located by an iris-scan algorithm which maps the iris' distinct patterns and characteristics.

The Iris processing algorithm presents a series of directives that direct a biometric system how to interpret a specific problem. The samples are matched on the basis of template feature comparison present in a database and present the closest match on the basis of state-of-the-art pattern recognition technologies.

Iris' are composed before birth and, except in the event of an injury to the eyeball, remain unchanged throughout an individual's lifetime. A typical Iris pattern is very complex and carries at least 200 unique spots that can be used as specific feature sets. The fact that an individual's right and left eyes are different and that patterns are easy to capture, establishes iris-scan technology as one of the biometrics that is very resistant to false matching and fraud.

2.8.2. Iridian's iris-recognition technology

Iris recognition is of course based on the visible qualities of the human iris (see Figure 5). Visible characteristics include rings, furrows, freckles, and the iris corona. Iridian's iris-recognition technology converts these visible characteristics into an IrisCode, a template stored for future verification attempts. From the 11-mm diameter iris, Daugman's algorithms provide 3.4 bits of data per square millimeter. This information density means that each iris can have 266 unique spots—compared to 10 to 60 unique spots for traditional biometric technologies [73].

The first step in scanning an iris is locating it with a dedicated monochrome camera no more than three feet from the eye. After the camera situates the eye, the search algorithm locates the outer and inner edges of the iris and then proceeds to analyze it. The system uses 2D Gabor wavelets [74]—transforms used typically in visualization applications—to filter and map iris segments into hundreds of vectors. The wavelets assign values drawn from the orientation and spatial frequency of select areas of the iris and they then form an 'IrisCode'. According to Daugman, the equal-error rate (the point at which the likelihood of a false accept and false reject are the same) is one in 1.2 million for IrisCodes.

The false acceptance rate for iris recognition systems is 1 in 1.2 million, statistically better then the average fingerprint recognition system. The real benefit is in the false-rejection rate, a measure of authenticated users who are rejected. Fingerprint scanners have a 3 percent false-rejection rate, whereas iris scanning systems boast rates at the 0 percent level.

2.8.3. Technical Scope and Usage

Iris-scan technology has been piloted in ATM environments in England, the US, Japan and Germany since as early as 1997. In these pilots the customer's iris data became the verification tool for access to the bank account, thereby eliminating the need for the customer to enter a PIN number or password. When the customer presented their eyeball to the ATM machine and the identity verification was positive, access was allowed to the bank account. These applications were very successful and eliminated the concern over forgotten or stolen passwords and received tremendously high customer approval ratings.

Airports have begun to use iris-scanning for such diverse functions as employee identification/verification for movement through secure areas and allowing registered frequent airline passengers a system that enables fast and easy identity verification in order to expedite their path through passport control.

Other applications include monitoring prison transfers and releases, as well as projects designed to authenticate on-line purchasing, on-line banking, on-line voting and on-line stock trading to name just a few. Iris-scan offers a high level of user security, privacy and general peace of mind for the consumer.

A highly accurate technology such as iris-scan has vast appeal because the inherent argument for any biometric is, of course, increased security.

2.8.4. Benefits of Iris Recognition Technology

- The iris is a thin membrane on the interior of the eyeball. Iris patterns are extremely complex.
- Patterns are individual (even in fraternal or identical twins).
- Patterns are formed by six months after birth, stable after a year. They remain the same for life.
- Imitation is almost impossible.
- Patterns are easy to capture and encode

2.8.5. Technology Comparison

Table 2-1: Biometric Application Stats for Industrial Applications

Method	**Coded Pattern**	**Misidentification rate**	**Security**	**Applications**
Iris Recognition	Iris pattern	1/1,200,000	High	High-security facilities
Fingerprinting	Fingerprints	1/1,000	Medium	Universal
Hand Shape	Size, length and thickness of hands	1/700	Low	Low-security facilities
Facial Recognition	Outline, shape and distribution of eyes and nose	1/100	Low	Low-security facilities

Signature	Shape of letters, writing order, pen pressure	1/100	Low	Low-security facilities
Voiceprinting	Voice characteristics	1/30	Low	Telephone service

2.9. Face Recognition

Face Recognition technology basically analyzes core features of a human face that do not change much under various aspects such as glasses, hairstyles, facial expressions, beards, etc, therefore, all face-recognition technologies share certain commonalities, such as emphasizing those sections of the face that are less susceptible to alteration, including the upper outlines of the eye sockets, areas surrounding the cheekbones, and sides of the mouth.[75] Facial-scan technology works well with standard PC video capture cameras and generally requires cameras that can capture images at least at 320 × 240 resolution and at least 3 to 5 frames per second. More frames per second, along with higher resolution, will lead to better performance in verification or identification, but higher rates typically aren't required for basic one-to-one verification systems that compare your face scan to a template you've previously stored on the verifying system.

2.9.1. Face Recognition Process

As with all biometric technologies, sample capture, feature extraction, template comparison, and matching define the process flow of facial-scan technology. The sample capture process will generally consist of 20 to 30 seconds during which a facial-recognition system will take several pictures of the subject's face. Ideally, the series of pictures will incorporate slightly different angles and facial expressions to allow for more accurate searches. After entering a subject's general face scan, the system will typically extract the subject's distinctive features and create a graphic template.

The exact algorithm any given commercial system uses to create and then later verify the templates is typically a closely guarded secret. The template is much smaller than the image from which it's drawn. Whereas quality facial images generally require 150 to 300 Kbytes, templates will only be approximately 1 Kbytes.

Visionics, one of the most prominent biometric vendors, uses an even smaller 84-byte template to help accelerate one-to-many searches.

Authentication follows the same protocol. Assuming your user is cooperative; he or she stands or sits in front of the camera for a few seconds and is either verified or rejected. This comparison is based on the similarity of the newly created template against the template on file. One variant of this process is the use of facial-scan technology in forensics. The templates come from static photographs of known criminals and are stored in large databases. The system performs a one-to-many search of these records to determine if the detainee is using an alias. If the database has only a handful of enrollees, this kind of search isn't terribly processor intensive. But as databases grow large, into the tens and hundreds of thousands, this task becomes more difficult. The system might only narrow the search to several likely candidates and then require human intervention at the final verification stages.

Another variable in identification is the dynamic between the target subjects and capture device. Standard verification typically assumes a cooperative audience, one consisting of subjects motivated to use the system correctly. Facial-scan systems, depending on the exact type of implementation, might also have to be optimized for uncooperative subjects. Uncooperative subjects are unaware that a biometric system is in place, or don't care, and make no effort to be recognized. Facial-scan technologies are more capable of identifying cooperative subjects.

2.10. Summary

The optimal LPR systems operate in a hybrid environment that utilizes the techniques of both software and hardware logics. Most of these industrial applications perform computationally expensive tasks over logic built on hardware. A standard frame-grabbing device comes with operations supporting Color Edge Detection, RGB to HSI conversions and Region Growing based Image Segmentation. Such real time systems are also provided with Pipe-lined or Parallel processors to further support speedy image retrieval, handling and processing. Off course, developing an efficient algorithm that does the job without employing expensive hardware devices would be a significant contribution to the respective Research and Development sector. The proposed approaches tested and implemented image enhancement and preprocessing techniques to adjust the image to a level suitable enough for the plate extraction algorithm to work precisely and accurately even at darker environments. It was also required that such techniques wont have an effect on the images taken at clearer surroundings. Again, utilizing the availability of processors good enough to support true-color (RGB) images should also provide more flexibility towards processing images distorted by speed, dust, dirt, rust, murkiness, glare, etc. In this domain, detection of sharp discontinuities in image parameters could be exploited more perfectly.

Respecting and keeping these ideas in our mind and the exploration done by previously done work we presented an insight to the latest industrial work going and the areas asking future work.

Biometrics technology has come a long way from simpler forms of systems security. While biometric proponents stress the strength of their proprietary technologies or biometrics in general, no system is ever completely secure. Bruce Schneier once pointed out that all computer security is like putting a wooden stake in front of your house and hoping that trespassers will run into it. [76] Contrary to what many biometric proponents would have us believe—that biometric security outclasses traditional forms of security—all biometric systems are, after all, another form of computer security with its own set of strengths and weaknesses.

Biometrics effectively trade some amount of privacy and cost effectiveness for ultimate convenience—and these systems are certainly no less secure than standard ‘pass-wording’ systems. Pass-wording systems are cheap. Complex biometric scanning equipment is usually expensive. But biometrics seems to be where the industry is headed. Aside from the Orwellian connotations, biometrics systems offer an enormous amount of convenience to users. And, in the present political climate, it's hard to counter the argument that we should adopt biometric systems simply as additional layers of security on top of traditional pass-wording systems.

CHAPTER 3

COLOR EDGE DETECTION

The use of color in image processing and pattern classification application is provoked by two main factors. The ability of color to describe a scenic detail in much clearer form making object extraction a far easier task. Second, Humans are able to differentiate between thousands of colors no matter how minute their intensity differences and saturation details are as compared to just a handful of (256) colors present in Gray Scale composition.

The chapter gives a basic introduction of color image processing techniques used to detect edges, segment Region of Interests (ROI) and promote Region Growing. It will also touch briefly the effect of Edge Operators on an Image used in real time image applications.

A typical gray scale image ranges from an intensity level of ‘0’ to ‘256’ where ‘0’ depicts an off-pixel of black color and ‘256’ a purely white pixel. On the other hand a true color image contains three basic color components namely Red, Green and Blue (RGB) that ranges from ‘0’ to ‘256’. The combination of these colors produces different shades of intensities, saturations and hue in a color that are more easily perceivable by a human eye as shown in Fig. 3.1.

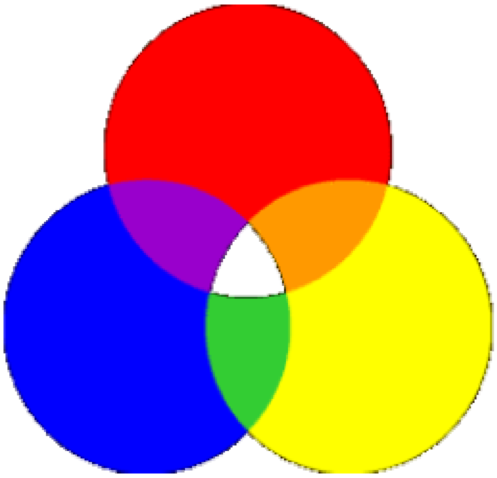


Figure 3-1 A method of combining pigments

3.1. Color Edge Detection

Edge detection is an important process in low level image processing. Research shows that more or less 90% of edges are same in gray values and color images [57]. Still there are 10% edges left that are not detected in intensity images. These 10% edges may significantly affect the edge information that could be utilized in a close extraction or segmentation process. An edge typically is a sharp discontinuity in an image due to brightness or intensity variations. There are numerous approaches of various computational complexities to edge detection in color images. To realize an edge various masks know as edge operators are scanned over image ROI to mark such discontinuity areas. In general masks of 3×3 size are enough but larger sizes can also be used with multi-resolution techniques to avoid computational overheads. A number of significant edges are only missed at areas where two different objects with different hues but similar intensities. Such objects are treated as a combined object in gray value images.

On the other hand, occasionally we require segmentation of objects having very minute geometrical shapes as shown in the Fig. 3.2 (a). If the color information of these objects is lost due to noise or the objects present on both the sides of these objects are of different intensities but similar hues, situation becomes more challenging.

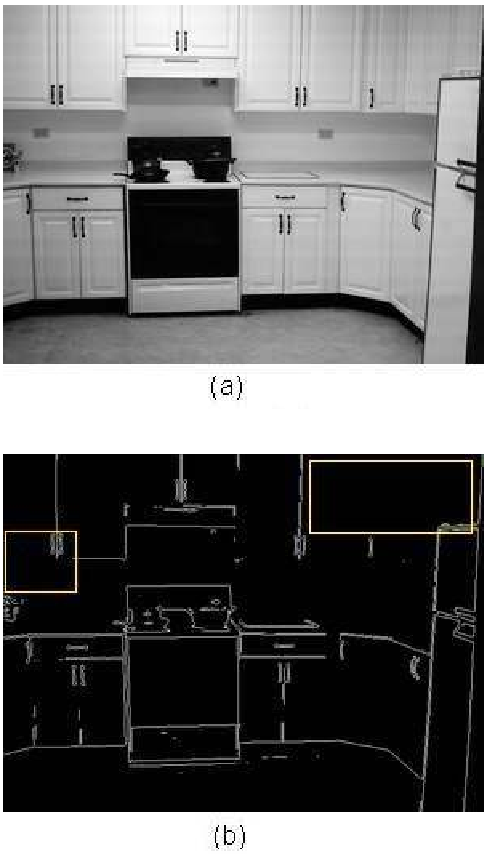


Figure 3-2: (a) The gray-scale image. (b) An edge map using a 3×3 Sobel Operator. Areas with similar color but different intensities are lost.

Most edge detection schemes find maxima in the first derivative of the image function or zero crossing of the second derivative. The main problem in extending the same approach to color images arises because the image function in such images is vector valued. Combination of separate RGB gradients also remains a significant issue. The simplest approach appears to apply Sobel masks to the three color channels (Red, Green and Blue) independently and combine the image using logical operation.

3.2. Performance

The performance of such edge detectors greatly depend upon the application at hand. Generally an algorithm using a space other than RGB is more computationally complex. Transformations such as HIS, CIE and LUV are very complex. On the other hand, RGB suffers from the high correlation among the three planes. However a reliable method to extract hue and saturation difference information directly from RGB is vector angle measure [59]. [58] Dony et al propose a method for obtaining chromaticity information for the purpose of intensity invariant segmentation directly from the RGB image. The methods used a modified Robert's operator to implement a vector angle measure in RGB images. [58] also compared the approach with results obtained from the Robert's operator over Euclidean Distance.

3.3. Edge Detection Techniques

3.3.1. Euclidean Distance

The approach [59] is to use Euclidean distance, E_D between adjacent color pixels to calculate an edge map,

$$E_D(c_1, c_2) = \|c_1 - c_2\| \tag{3-1}$$

Where $\|\bullet\|$ is a L_2 vector norm. For an RGB space coordinate system, $C = [r\ g\ b]^T$ the distance is calculated as

$$E_D(c_1,c_2)=\left[(R_1-R_2)^2+(G_1-G_2)^2+(B_1-B_2)^2\right] \qquad (3\text{-}2)$$

The effects of ED can be analyzed in RGB space as shown in Fig 3.3. Light Green and Dark Green only differ in intensity values and have same chromatic values. Dark Green and Dark Brown differ in both intensity as well as chromaticity values. Same is the case for any colors present in RGB space.

3.3.2. Vector Angle

An alternative defined in [59] is to use Vector angle measure defined as:

$$\cos\theta=\frac{\vec{c_1}^T\vec{c_2}}{\left|\vec{c_1}\right|.\left|\vec{c_2}\right|} \qquad (3\text{-}3)$$

Opposed to ED, VA is insensitive to intensity differences, but qualifies well hue and saturation differences. Using angle θ as an edge value has a drawback that the calculation of inverse cosine is expensive relative to simple Euclidean arithmetic. Furthermore, statistical analysis of values in angular coordinates is problematic [59]. For these reasons and since we are interested in Hue differences however small, the $\sin\theta$ was proposed in [58] and is defined as follows:

$$\sin\theta=\left(1-\left(\frac{\vec{c_1}^T\vec{c_2}}{\left|\vec{c_1}\right|.\left|\vec{c_2}\right|}\right)^2\right)^{1/2} \qquad (3\text{-}4)$$

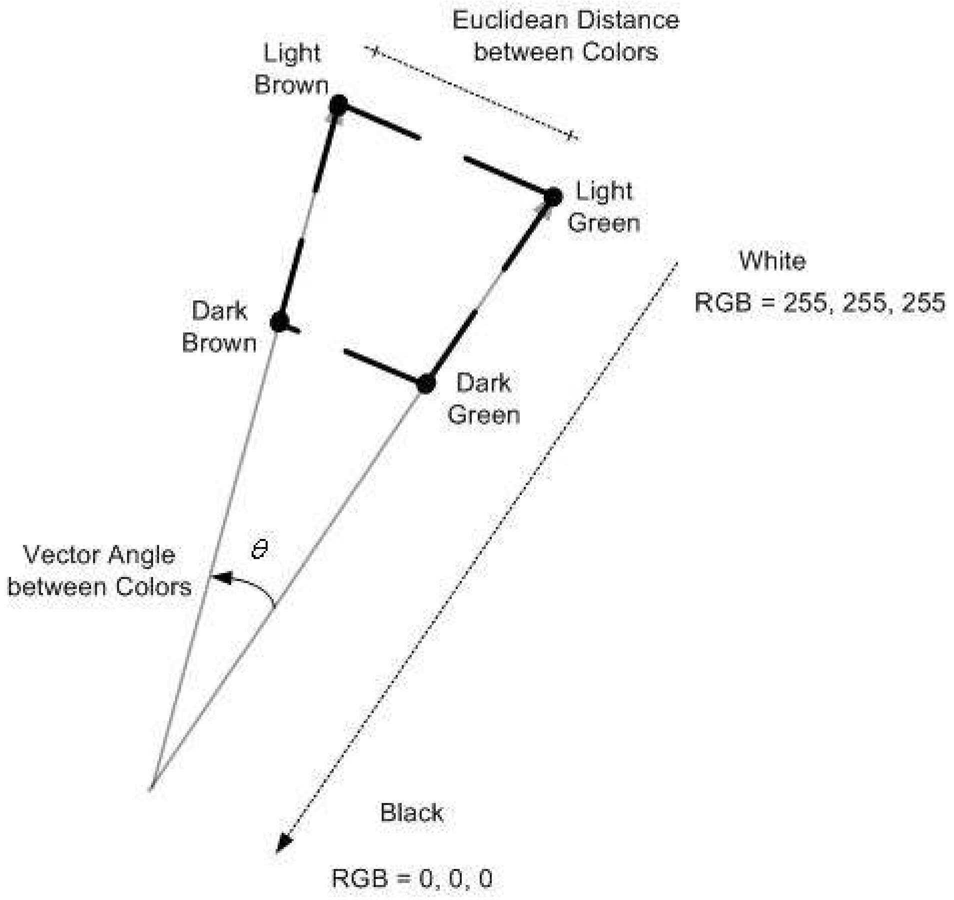


Figure 3-3: Vector Geometry between four color examples

3.4. Edge Operators

3.4.1. Difference Vector Edge Detectors

Difference Vector Edge Detector [60] [61], is a 3x3 operator that calculated the maximum gradient across the central pixel. The Euclidean distance version of this operator can be written as

$$E_{DV} = \max_{i=1,\ldots,4}\left(\left\|\vec{v}_i(x,y) - \vec{v}_{4+i}(x,y)\right\|\right) \tag{3-5}$$

Where i represents one of the first four (out of a possible eight) positions around the central pixel. This gives the four directional gradients across the pixel.

The vector angle version of the Difference Vector Edge Detector is characterized by

$$S_{VG} = \max_{i=1,\ldots,8}\left[\sqrt{1-\left(\frac{\vec{v}_i^T(x,y).\vec{v}_o(x,y)}{\left|\vec{v}_i^T(x,y)\right|\left|\vec{v}_o(x,y)\right|}\right)^2}\right] \tag{3-6}$$

3.4.2. Vector Gradient Edge Detectors

The Vector Gradient Edge Detector is an edge operator that computes the maximum distance in the desired metric between the central pixel and the 8-connected pixels adjacent to it.

The Euclidean distance version of this operator is defined as follows:

$$E_{DV} = \max_{i=1,\ldots,4}\left(\left\|\vec{v}_i(x,y)-\vec{v}_o(x,y)\right\|\right) \tag{3-7}$$

Where i is the counter representing one of the eight neighboring pixels.

The vector angle version of this operator is written as

$$S_{VG} = \max_{i=1,\ldots,8}\left[\sqrt{1-\left(\frac{\vec{v}_i^T(x,y).\vec{v}_o(x,y)}{\left|\vec{v}_i^T(x,y)\right|\left|\vec{v}_o(x,y)\right|}\right)^2}\right] \tag{3-8}$$

3.4.3. Comparison of Vector Angle and Euclidean Distance

Color spaces that use polar coordinate systems such as HSI use the HUE component directly in the space as one of the coordinates. Such spaces produce more robust results in terms of quality but have a negative effect on the systems performance due to their computational complexity. This is primarily due to the non-linear transformation involving sinusoids to convert the raw RGB data to HSI [62]. For high speed applications such as Color OC readers, Hand Sign Language Recognition, Intelligent Transport Devices, Check Reading, etc this extra computational overhead and delay may not be feasible. Processing based on raw RGB coordinates would have more advantage. For such applications operators based on 2×2 masks are enough.

3.5. Summary

The chapter gave a brief introduction to color edge detection. The shortcomings of gray scale image processing mainly arise when an objective of intensity invariant image processing is concerned. RGB is an expansive domain where we can exploit the presence of both color and intensity information. A lot of research work has been done in this area. For real time image processing applications, a trade off between quality and performance must be maintained. Application using masks greater than 5×5 and processing image resolutions of 640×480 may not give optimal results in terms of performance but a standard image using a 3×3 mask does the job with better efficiency.

CHAPTER 4

FUZZY IMAGE PROCESSING

Fuzzy image processing is not a unique theory. It is a collection of different fuzzy approaches to image processing. Nevertheless, the following definition can be regarded as an attempt to determine the boundaries. *"Fuzzy image processing is the collection of all approaches that understand, represent and process the images, their segments and features as fuzzy sets. The representation and processing depend on the selected fuzzy technique and on the problem to be solved."[63]*

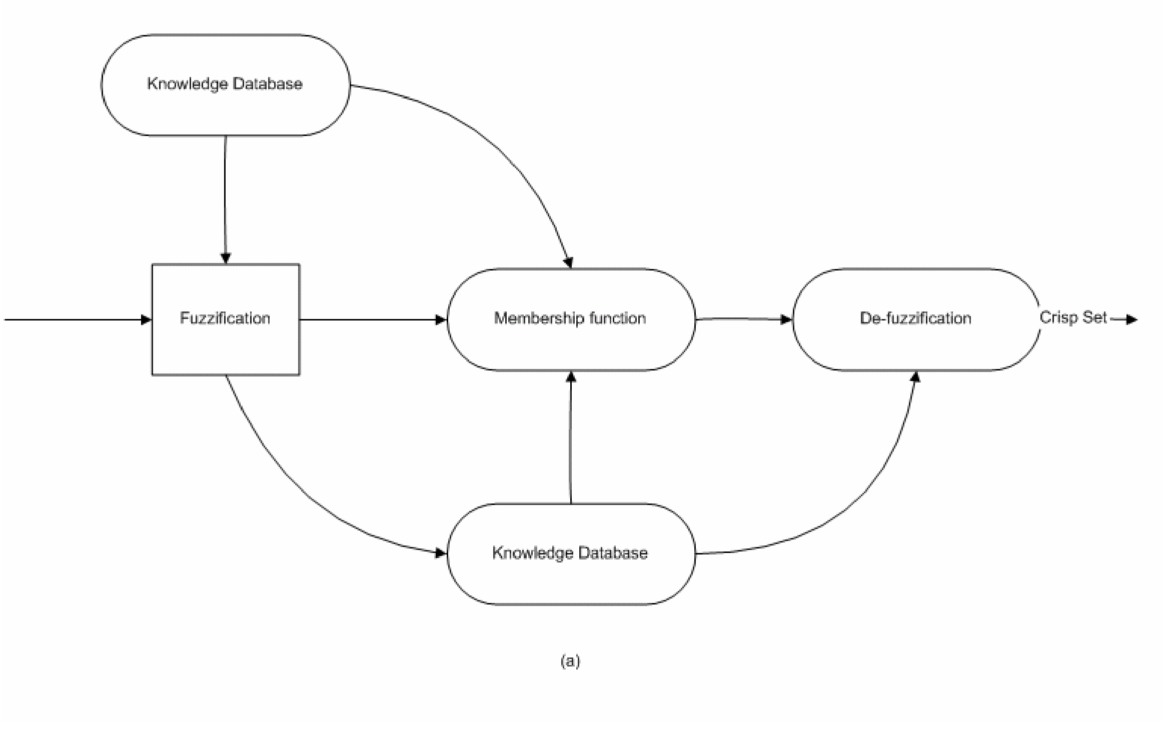
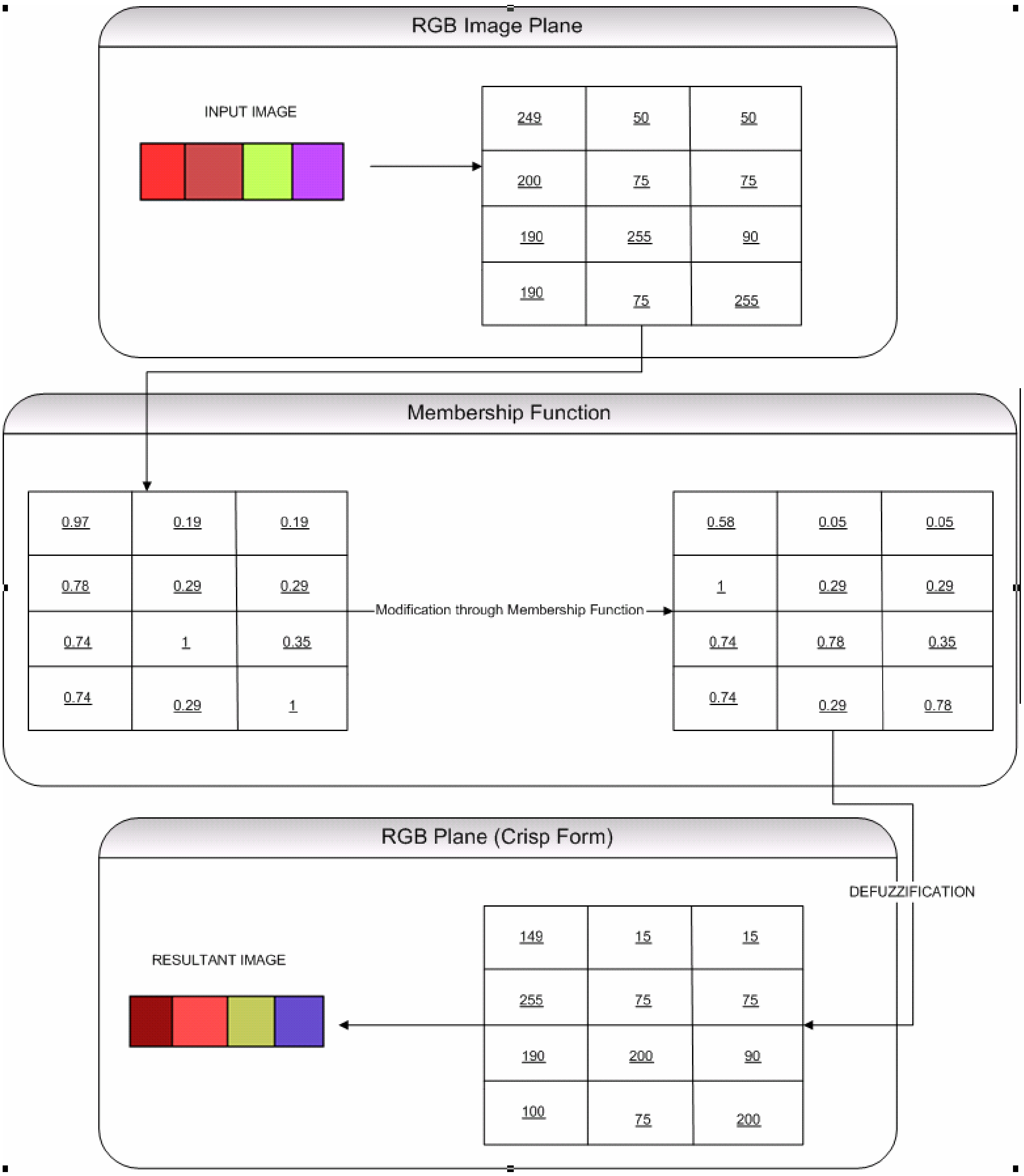


Figure 4-1: General Structure of a Fuzzy System



249	50	50
200	75	75
190	255	90
190	75	255

0.97	0.19	0.19
0.78	0.29	0.29
0.74	1	0.35
0.74	0.29	1

0.58	0.05	0.05
1	0.29	0.29
0.74	0.78	0.35
0.74	0.29	0.78

149	15	15
255	75	75
190	200	90
100	75	200

Figure 4-2: Pixel-Based RGB Color Modification using Fuzzy Image Processing

The three main steps in Fuzzy Image Processing are:

- Image Fuzzification
- Membership value modification through Membership Functions
- Image Defuzzification

We use the coding (Fuzzification), decoding (Defuzzification) of image data (in our case RGB intensity values) to process images with fuzzy techniques. The main power of fuzzy techniques lies in the middle step (Membership Function(s)) which is derived using a knowledge database as shown in Fig 4.2. This is done in the Fuzzification step. The overall procedure is somewhat akin to normalizing all the inputs according to a knowledge based parameter (in the case shown, it is 255). Applying the membership function (Not given in the figure above), and applying the Defuzzification mechanism on the modified membership values. The new values are also regarded as the Crisp set.

The overall working system can be further elaborated as follows. "... A pictorial object is a fuzzy set which is specified by some membership function defined on all picture points. From this point of view, each image point participates in many memberships. Some of this uncertainty is due to degradation, but some of it is inherent…In fuzzy set terminology, making figure/ground distinctions is equivalent to transforming from membership functions to characteristic functions." **1970, J.M.B. Prewitt.**

4.1. Fuzzy C-Mean Clustering

Clustering of numerical data forms the basis of many classification and system modeling algorithms. The purpose of clustering is to identify natural groupings of data from a large data set to produce a concise representation of a system's behavior.

In the iterations of the conventional c-means (Non-Fuzzy) algorithm, each data point is assumed to be the member of exactly one cluster. In pattern classification domain such memberships are rarely seen and a classifier bearing a feature vector of more than 2 dimensions is normally considered to be partially associated to one specific domain. This partial association can be efficiently described and presented on the basis of Fuzzy version of K-Means Clustering Algorithm known as *Fuzzy C Means Algorithm.* Fuzzy c-means is a data clustering technique wherein each data point belongs to a cluster to some degree that is specified by a membership grade. This technique was originally introduced by Jim Bezdek in 1981 [64] as an improvement on earlier clustering methods. It provides a method that shows how to group data points that populate some multidimensional space into a specific number of different clusters [65].

The memberships in this algorithm are equivalent to the probabilities $\hat{P}\left(\omega_i / x_j, \hat{\theta}\right)$, where θ is the parameter vector for the membership function. The fuzzy c means clustering algorithm seeks a minimum of a heuristic global cost function.

$$J_{fuzz} = \sum_{i=1}^{c} \sum_{j=1}^{n} \left[\hat{P}\left(\omega_i / x_j, \hat{\theta}\right)\right]^b \left\| x_j - \mu_i \right\|^2 \tag{4-1}$$

Where b is a free parameter chosen to adjust the blending of different clusters. If b is set to '0', J_{fuz} is merely a sum-of-squared error criterion with each pattern assigned to each cluster.

The probabilities of cluster membership for each point are normalized as

$$\sum_{i=c}^{c} \hat{P}\left(\omega_i / x_j\right) = 1, j = 1,..., n \tag{4-2}$$

Where, for, simplicity, we have not explicitly shown the dependence on $\hat{\theta}$. We let $\hat{P}_j$ denote the prior probability of $\hat{P}(\omega_j)$, then at the solution (i.e., the minimum of J_{fuzz}), we have

$$\partial J_{fuzz} / \partial \mu_i = 0 \qquad (4\text{-}3)$$

$$\partial J_{fuzz} / \partial \hat{P}_j = 0 \qquad (4\text{-}4)$$

The solution, therefore, is given as

$$\mu_j = \frac{\sum_{j=1}^{n} \left[\hat{P}(\omega_i / x_j)\right]^b x_j}{\sum_{j=1}^{n} \left[\hat{P}(\omega_i / x_j)\right]^b} \qquad (4\text{-}5)$$

And

$$\hat{P}(\omega_i / x_j) = \frac{\left(1/d_{ij}\right)^{1/b-1}}{\sum_{r=1}^{c} \left(1/d_{rj}\right)^{1/b-1}} \qquad (4\text{-}6)$$

And

$$d_{ij} = \left\| x_i - \mu_i \right\|^2 \qquad (4\text{-}7)$$

In general, the J_{fuzz} criterion is minimized when the cluster centers μ_j are those points that have high estimated probability of being in cluster j. Because the equations rarely have analytical solutions, the cluster means and point probabilities are estimated iteratively according to the following algorithm.

Algorithm 4-1: Fuzzy C-Means Clustering

```
Begin     Initialize $n, c, b, \mu_1, ..., \mu_c, \hat{P}\left( \omega_i / X_j \right), i = 1, ..., c; j = 1, ..., n$

          Normalize $\hat{P}\left( \omega_i / X_j \right)$

          Do recomputed $\mu_i$

          Re compute $\hat{P}(\omega_i / x_j)$

          until small change in $\mu_i$ and $P(\omega_i / x_j)$

return    $\mu_i, \mu_2, ..., \mu_c$

End
```

For early iterations, the means lie near the center of the full dataset because each point has a non negligible “membership” in each cluster. At later iterations the means separate and each membership tends toward the value ‘1.0’ and ‘0.0’.

CHAPTER 5

DIGITAL IMAGE ENHANCEMENT

In today's world of advanced technology where most remote sensing data are recorded in digital format, virtually all image interpretation and analysis involves some element of digital processing. Digital image processing may involve numerous procedures including formatting and correcting of the data, digital enhancement to facilitate better visual interpretation, or even automated classification of targets and features entirely by computer. In order to process remote sensing imagery digitally, the data must be recorded and available in a digital form suitable for storage on a computer tape or disk. Obviously, the other requirement for digital image processing is a computer system, sometimes referred to as an image analysis system, with the appropriate hardware and software to process the data. Several commercially available software systems have been developed specifically for remote sensing image processing and analysis.

For discussion purposes, most of the common image processing functions available in image analysis systems can be categorized into the following four categories:

- Preprocessing
- Image Enhancement
- Image Transformation
- Image Classification and Analysis

Preprocessing functions involve those operations that are normally required prior to the main data analysis and extraction of information, and are generally grouped as radiometric or geometric corrections. Radiometric corrections include correcting the data for sensor irregularities and unwanted sensor or atmospheric noise, and converting the data so they accurately represent the reflected or emitted radiation measured by the sensor. Geometric corrections include correcting for geometric distortions due to sensor-Earth geometry variations, and conversion of the data to real world coordinates (e.g. latitude and longitude) on the Earth's surface.

The objective of the second group of image processing functions grouped under the term of image enhancement is solely to improve the appearance of the imagery to assist in visual interpretation and analysis. Examples of enhancement functions include contrast stretching to increase the tonal distinction between various features in a scene, and spatial filtering to enhance (or suppress) specific spatial patterns in an image.

Image transformations are operations similar in concept to those for image enhancement. However, unlike image enhancement operations which are normally applied only to a single channel of data at a time, image transformations usually involve combined processing of data from multiple spectral bands. Arithmetic operations (i.e. subtraction, addition, multiplication, division) are performed to combine and transform the original bands into "new" images which better display or highlight certain features in the scene. We will look at some of these operations including various methods of spectral or band ratioing, and a procedure called principal components analysis which is used to more efficiently represent the information in multi-channel imagery.

Image classification and analysis operations are used to digitally identify and classify pixels in the data. Classification is usually performed on multi-channel data sets and this process assigns each pixel in an image to a particular class or theme based on statistical characteristics of the pixel brightness values. There are a variety of approaches taken to perform digital classification. We will briefly describe the two generic approaches which are used most often, namely supervised and unsupervised classification.

5.1. Preprocessing

Pre-processing operations, sometimes referred to as image restoration and rectification, are intended to correct for sensor- and platform-specific radiometric and geometric distortions of data. Radiometric corrections may be necessary due to variations in scene illumination and viewing geometry, atmospheric conditions, and sensor noise and response. Each of these will vary depending on the specific sensor and platform used to acquire the data and the conditions during data acquisition. Also, it may be desirable to convert and/or calibrate the data to known (absolute) radiation or reflectance units to facilitate comparison between data.

Variations in illumination and viewing geometry between images (for optical sensors) can be corrected by modeling the geometric relationship and distance between the areas of the Earth's surface imaged the sun, and the sensor. This is often required so as to be able to more readily compare images collected by different sensors at different dates or times, or to mosaic multiple images from a single sensor while maintaining uniform illumination conditions from scene to scene.

For many quantitative applications of remote sensing data, it is necessary to convert the digital numbers to measurements in units which represent the actual reflectance or emittance from the surface. This is done based on detailed knowledge of the sensor response and the way in which the analog signal (i.e. the reflected or emitted radiation) is converted to a digital number, called analog-to-digital (A-to-D) conversion. By solving this relationship in the reverse direction, the absolute radiance can be calculated for each pixel, so that comparisons can be accurately made over time and between different sensors.

5.2. Image Intensity Adjustment

Enhancements are used to make it easier for visual interpretation and understanding of imagery. The advantage of digital imagery is that it allows us to manipulate the digital pixel values in an image. Although radiometric corrections for illumination, atmospheric influences, and sensor characteristics may be done prior to distribution of data to the user, the image may still not be optimized for visual interpretation. Remote sensing devices, particularly those operated from satellite platforms, must be designed to cope with levels of target/background energy which are typical of all conditions likely to be encountered in routine use. With large variations in spectral response from a diverse range of targets (e.g. forest, deserts, snowfields, water, etc.) no generic radiometric correction could optimally account for and display the optimum brightness range and contrast for all targets. Thus, for each application and each image, a custom adjustment of the range and distribution of brightness values is usually necessary.

In raw imagery, the useful data often populates only a small portion of the available range of digital values (commonly 8 bits or 256 levels). Contrast enhancement involves changing the original values so that more of the available range is used, thereby increasing the contrast between targets and their backgrounds. The key to understanding contrast enhancements is to understand the concept of an image histogram. A histogram is a graphical representation of the brightness values that comprise an image. The brightness values (i.e. 0-255) are displayed along the x-axis of the graph shown in Fig 5-1(b). The frequency of occurrence of each of these values in the image is shown on the y-axis.

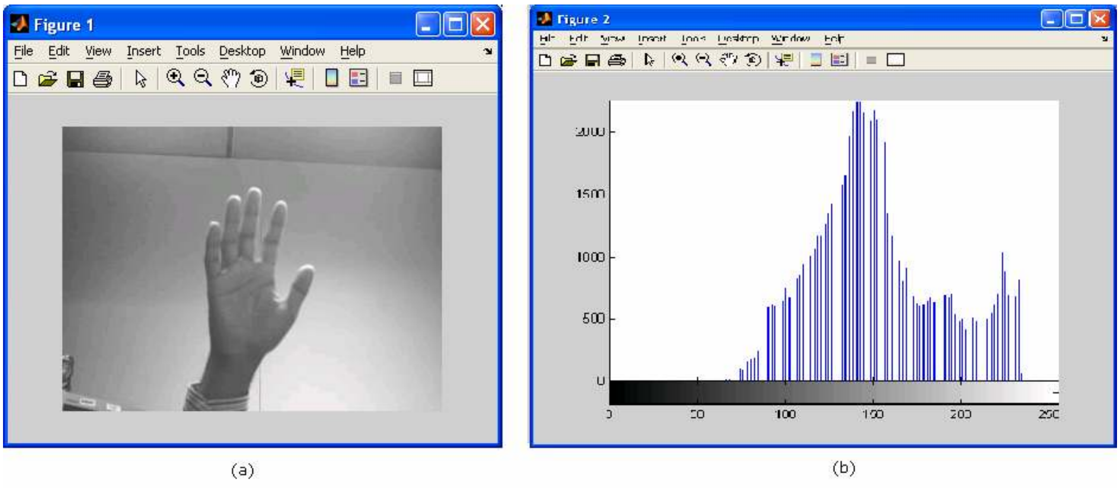


Figure 1
File Edit View Insert Tools Desktop Window Help
(a)
Figure 2
File Edit View Insert Tools Desktop Window Help
2000
1500
1000
500
0
0
50
100
150
200
250
(b)

Figure 5-1: (a) Gray scale profile of a hand. (b) Histogram profile of number of occurrences of every intensity over the map shown in (a).

By manipulating the range of digital values in an image, graphically represented by its histogram as shown in Fig. 5-1(b), we can apply various enhancements to the data. There are many different techniques and methods of enhancing contrast and detail in an image; we will cover only a few common ones here. The simplest type of enhancement is a linear contrast stretch. This involves identifying lower and upper bounds from the histogram (usually the minimum and maximum brightness values in the image) and applying a transformation to stretch this range to fill the full range. In our example, the minimum value (occupied by actual data) in the histogram is 84 and the maximum value is 153. These 70 levels occupy less than one-third of the full 256 levels available. A linear stretch uniformly expands this small range to cover the full range of values from 0 to 255. This enhances the contrast in the image with light toned areas appearing lighter and dark areas appearing darker, making visual interpretation much easier.

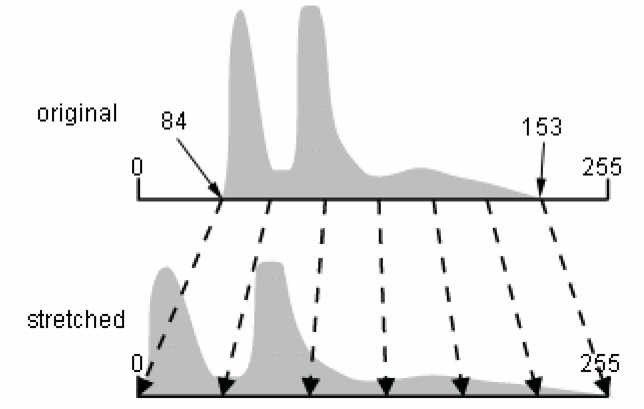


Figure 5-2: An example of stretching image gray scale limits

Fig 5.3 illustrates the increase in contrast in an image before (left) and after (right) a linear contrast stretch.

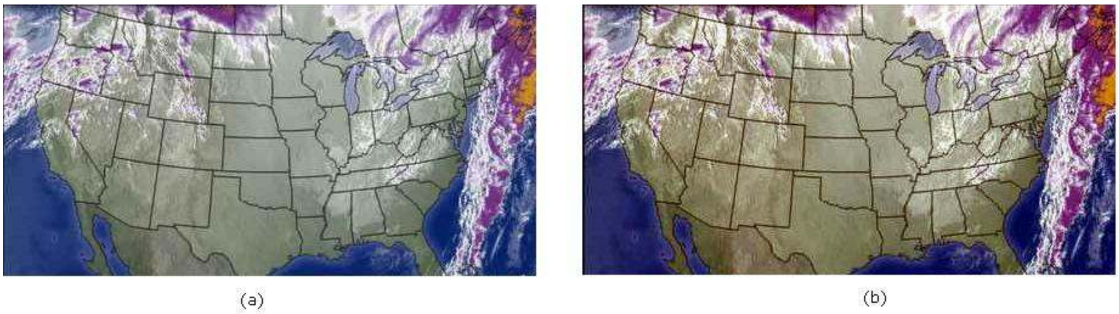


Figure 5-3: The images are manipulated specimen of satellite data. (a) Original satellite image. (b) An intensity enhanced sample of (a)

A uniform distribution of the input range of values across the full range may not always be an appropriate enhancement, particularly if the input range is not uniformly distributed. In this case, a histogram-equalized stretch as shown in Fig 5-4 may be better. This stretch assigns more display values (range) to the frequently occurring portions of the histogram. In this way, the detail in these areas will be better enhanced relative to those areas of the original histogram where values occur less frequently. In other cases, it may be desirable to enhance the contrast in only a specific portion of the histogram. For example, suppose we have an image of the mouth of a river, and the water portions of the image occupy the digital values from 40 to 76 out of the entire image histogram. If we wished to enhance the detail in the water, perhaps to see variations in sediment load, we could stretch only that small portion of the histogram represented by the water (40 to 76) to the full grey level range (0 to 255). All pixels below or above these values would be assigned to 0 and 255, respectively, and the detail in these areas would be lost. However, the detail in the water would be greatly enhanced.

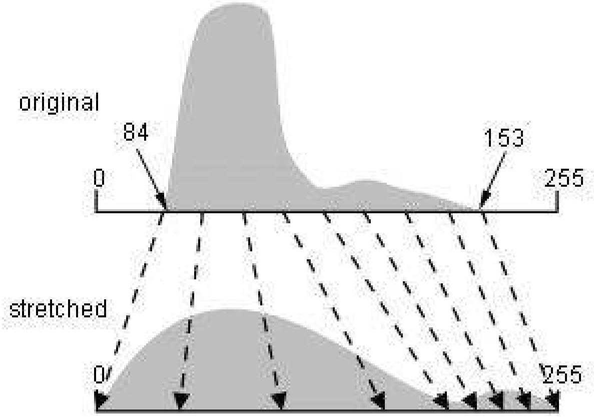


Figure 5-4: An example of image histogram stretching

5.3. Spatial Filtering

Spatial filtering encompasses another set of digital processing functions which are used to enhance the appearance of an image. Spatial filters are designed to highlight or suppress specific features in an image based on their spatial frequency. Spatial frequency is related to the concept of image texture. It refers to the frequency of the variations in tone that appear in an image. "Rough" textured areas of an image, where the changes in tone are abrupt over a small area, have high spatial frequencies, while "smooth" areas with little variation in tone over several pixels, have low spatial frequencies. A common filtering procedure involves moving a 'window' of a few pixels in dimension (e.g. 3x3, 5x5, etc.) over each pixel in the image, applying a mathematical calculation using the pixel values under that window, and replacing the central pixel with the new value. The window is moved along in both the row and column dimensions one pixel at a time and the calculation is repeated until the entire image has been filtered and a "new" image has been generated. By varying the calculation performed and the weightings of the individual pixels in the filter window, filters can be designed to enhance or suppress different types of features.

5.3.1. Low Pass Filtering

A low-pass filter is designed to emphasize larger, homogeneous areas of similar tone and reduce the smaller detail in an image. Thus, low-pass filters generally serve to smooth the appearance of an image. Average and median filters, often used for radar imagery, are examples of low-pass filters.

5.3.2. High Pass Filters

High-pass filters do the opposite and serve to sharpen the appearance of fine detail in an image. One implementation of a high-pass filter first applies a low-pass filter to an image and then subtracts the result from the original, leaving behind only the high spatial frequency information.

Directional, or edge detection filters are designed to highlight linear features, such as roads or field boundaries. These filters can also be designed to enhance features which are oriented in specific directions. These filters are useful in applications such as geology, for the detection of linear geologic structures.

5.4. Image Classification and Analysis

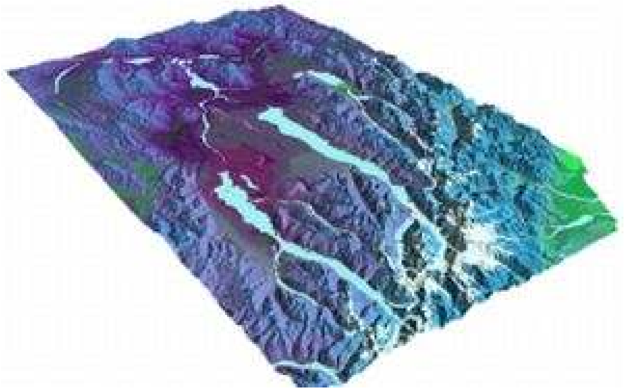


Figure 5-5: An image of the elevation map of the Waitaki basin (NZ). The lower right corner shows the coast with Alps classified as dark violet areas on the left

A human analyst attempting to classify features in an image uses the elements of visual interpretation to identify homogeneous groups of pixels which represent various features or land cover classes of interest. Digital image classification uses the spectral information represented by the digital numbers in one or more spectral bands, and attempts to classify each individual pixel based on this spectral information. This type of classification is termed spectral pattern recognition. In either case, the objective is to assign all pixels in the image to particular classes or themes (e.g. water, coniferous forest, deciduous forest, corn, wheat, etc.). The resulting classified image is comprised of a mosaic of pixels, each of which belong to a particular theme, and is essentially a thematic "map" of the original image.

When talking about classes, we need to distinguish between information classes and spectral classes. Information classes are those categories of interest that the analyst is actually trying to identify in the imagery, such as different kinds of crops, different forest types or tree species, different geologic units or rock types, etc. Spectral classes are groups of pixels that are uniform (or near-similar) with respect to their brightness values in the different spectral channels of the data. The objective is to match the spectral classes in the data to the information classes of interest. Rarely is there a simple one-to-one match between these two types of classes. Rather, unique spectral classes may appear which do not necessarily correspond to any information class of particular use or interest to the analyst. Alternatively, a broad information class (e.g. forest) may contain a number of spectral sub-classes with unique spectral variations. Using the forest example, spectral sub-classes may be due to variations in age, species, and density, or perhaps as a result of shadowing or variations in scene illumination. It is the analyst's job to decide on the utility of the different spectral classes and their correspondence to useful information classes.

Common classification procedures can be broken down into two broad subdivisions based on the method used: supervised classification and unsupervised classification.

5.4.1. Supervised Classification

In a supervised classification, the analyst identifies in the imagery homogeneous representative samples of the different surface cover types (information classes) of interest. These samples are referred to as training areas. The selection of appropriate training areas is based on the analyst's familiarity with the geographical area and their knowledge of the actual surface cover types present in the image. Thus, the analyst is "supervising" the categorization of a set of specific classes. The numerical information in all spectral bands for the pixels comprising these areas are used to "train" the computer to recognize spectrally similar areas for each class. The computer uses a special program or algorithm (of which there are several variations), to determine the numerical "signatures" for each training class. Once the computer has determined the signatures for each class, each pixel in the image is compared to these signatures and labeled as the class it most closely "resembles" digitally. Thus, in a supervised classification we are first identifying the information classes which are then used to determine the spectral classes which represent them.

5.4.2. Unsupervised Classification

Unsupervised classification in essence reverses the supervised classification process. Spectral classes are grouped first, based solely on the numerical information in the data, and are then matched by the analyst to information classes (if possible). Programs, called clustering algorithms, are used to determine the natural (statistical) groupings or structures in the data. Usually, the analyst specifies how many groups or clusters are to be looked for in the data. In addition to specifying the desired number of classes, the analyst may also specify parameters related to the separation distance among the clusters and the variation within each cluster. The final result of this iterative clustering process may result in some clusters that the analyst will want to subsequently combine, or clusters that should be broken down further - each of these requiring a further application of the clustering algorithm. Thus, unsupervised classification is not completely without human intervention. However, it does not start with a pre-determined set of classes as in a supervised classification.

5.5. Image Segmentation

Partitioning of an image into several constituent components is called segmentation. Segmentation is an important part of practically any automated image recognition system, because it is at this moment that one extracts the interesting objects, for further processing such as description or recognition. Segmentation of an image is in practice the classification of each image pixel to one of the image parts. If the goal is to recognize black characters, on a grey background, pixels can be classified as belonging to the background or as belonging to the characters: the image is composed of regions which are in only two distinct grey value ranges, dark text on lighter background. The grey level histogram, viz. the probability distribution of the grey values, has two separated peaks, i.e. is clearly bimodal. In such a case, the segmentation, i.e. the choice of a grey level threshold to separate the peaks, is trivial. The same technique could be used if there were more than two clearly separated peaks.

Unfortunately, signal and background peaks are usually not so ideally separated, and the choice of the threshold is problematic. A typical histogram, still bimodal, but with peaks not separated, is shown in the Fig. 5-6

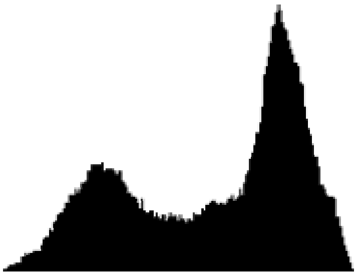


Figure 5-6: A vertical histogram of a binary image.

A variety of techniques for automatic threshold selection exists. A relatively successful method for certain applications is described in [71], where it is suggested that a modified histogram is employed by using only pixels with a small gradient magnitude, i.e. pixels which are not in the region of the boundaries between object and background.

In many cases, segmentation on the basis of the greyvalue alone is not efficient. Other features like colour, texture, gradient magnitude or orientation, measure of a template match etc., can be put to use. This produces a mapping of a pixel into a point in an *n*-dimensional feature space, defined by the vector of its feature values. The problem is then reduced to partitioning the feature space into separate clusters, a general pattern recognition problem that is discussed in the literature.

5.6. Summary

The chapter covered the essentials basics and methodlogies present in practice. The areas of image intensity adjustment, enhancement of image clusters of interest and suppression of unwanted details were discussed. Furthermore, classification techniques addressing supervised and unsupervised segmentation were explored.

CHAPTER 6

CONCEPTUAL MODEL

The main objective of an image processing system is to improve the quality of an image to a level where it could be easily processed by an image information retrieval system. Generally, if carefully done, this step takes care of much of the burden from a computer vision application whose main objective is to recognize a specific pattern, object or area from an image while keeping accuracy and performance.

A conceptual model of such a pattern recognition application can be considered as a black box in which an image in its raw form is input and at the output we get standard information of our objective. Consequently, such an application can be divided into following main sub-objectives or goals.

- Data Collection: Setting up a testing and training database covering all the cases that might encounter such an application in its real life situation.
- Sampling: Filtering out information that is not required, thereby, reducing the level of complexity incurred by the system at the training level.
- Segmentation: Isolating well defined set of blocks.
- Normalization: To make input invariant to all sorts of scaling, rotation and translation.
- Feature Extraction: To further reduce the input space by grouping the segments obtained on the basis of relevant features.
- Classification: To correctly classify the input as one of the output classes.

In the conceptual model of our system, data is collected by acquiring the image of the vehicle. The image is preprocessed in order to obtain the area of interest (i.e. License plate). The ROI is further re-sampled in order to filter-out unnecessary areas from the plate before commencing to the segmentation phase. The segmented blocks (plate characters) are then normalized and their features are extracted. These features are input to the classification stage to obtain the respective group or class of the individual blocks.

The chapter is organized in the following order. The first section deals with the collection of image data. The data is gathered by acquiring images of cars using a digital camera. The second section covers the extraction of the license plates. This process is further divided into four phases, viz., contrast adjustment, color-intensity based edge detection/ color saturation based edge detection, connected candidate search, compactness based shape filtering and template matching. The next section presents the segmentation of the license plate into individual characters based on character height and centroid similarity. Following the segmentation stage is the recognition of the segmented characters using the syntactic approaches. The chapter is concluded by a brief summary.

6.1. Image Acquisition

This is the first phase in the LPR system. An image is generally acquired in the following three ways.

1. Using a conventional analogue camera and a scanner
2. Using a digital camera
3. Using a video camera and a frame grabber (a frame averaging device for optimal frame selection)
4. Digital Camcorder with a Software/Hardware trigger.

The first method remains impractical for real time image processing applications because it is time consuming and tedious. Generally a digital camera can be used for research and development purposes which can later be exchanged by a camcorder and a frame grabbing device. The fourth case requires much programming overhead towards the programmer since the system also has to decide between a number of frames taken for the most suitable one for processing.

In the proposed system a high resolution camera is used to acquire the image. The image is processed in its true-color (24 bit) form. Conversion to gray-scale is avoided in any of the image preprocessing phases in order to obtain stronger edge information in the later stages. The true color image is shown in Fig. 6.1.

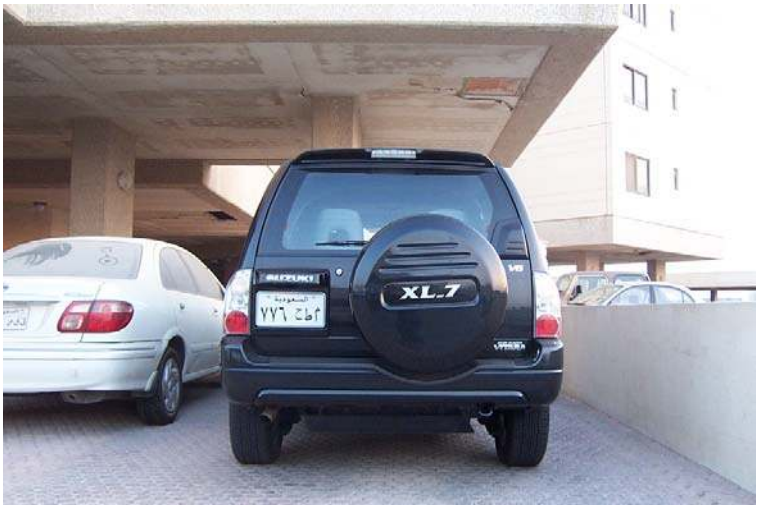


Figure 6-1: Original Image (A 16 million color (24 bit) image)

6.2. License Plate Extraction

License plate extraction is a key step in an LPR system. The objective of this phase is, given an input image, to produce a number of candidate regions with a significant degree of similarity with a license plate. In the adopted approach the extraction phase is divided into 4 phases which are explained in the following subsections.

6.2.1. Image Contrast Adjustment

Images taken in darker surrounds like rainy days, evenings or low light areas tend to stay short at their intensity profiles. The overall picture looks dim, which in turn leads to large saturation values through out the picture as shown in Fig 6.2.

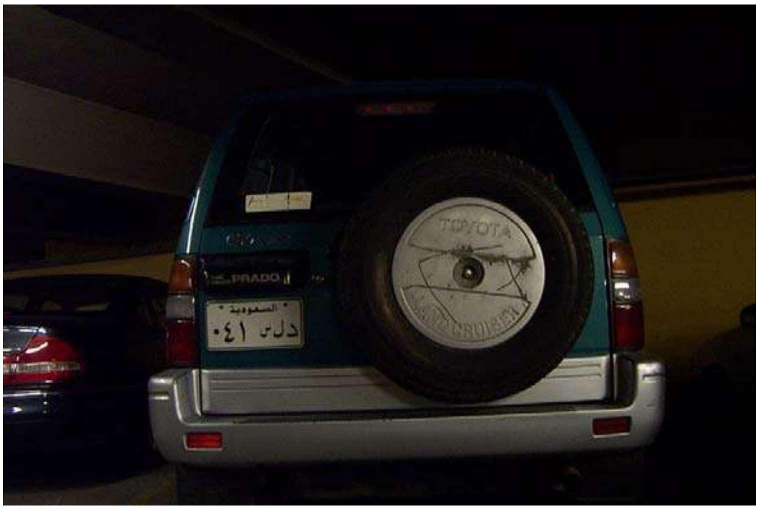
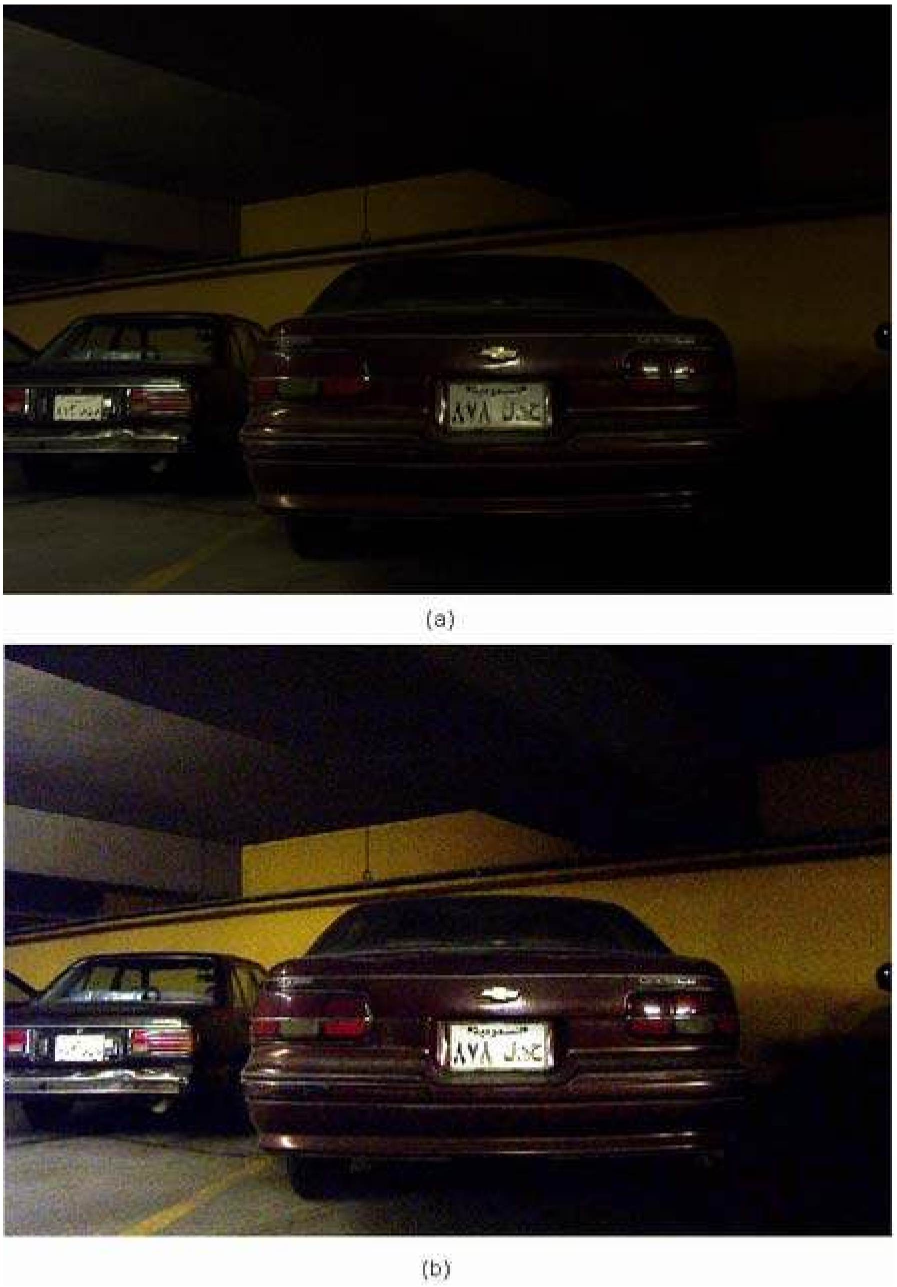


Figure 6-2: An image in low light

The presence of shadows and partial illumination in an image directly affects the ability of color edge detection or a region growing algorithm to detect a ROI accurately.

We tested a number of license plates taken at various lighting conditions to estimate the intensity variation due to the effect of shadows, low light and glare. The intensity of these images was adjusted such that a specific percentage of data at low and high intensities of the images was saturated. A saturation of 1% at both end of the image showed improvement in the image quality in darker surroundings. We simply map the low intensity to bottom intensity, and high intensity to the top. The values between low and high are, in general, mapped linearly to values between bottom and top. For example, the value halfway between low and high corresponds to the value halfway between bottom and top.

The tested images showed minimal variation in image quality in normal lightning conditions. Examples of two such images are shown in Fig 6.3 and Fig 6.4. The main objective in this section remains to enhance an image's quality and remove the effect of shadows and glare.



(a)

(b)

Figure 6-3: (a) Image taken at low lightning conditions. (b) Results of Image Contrast Adjustment

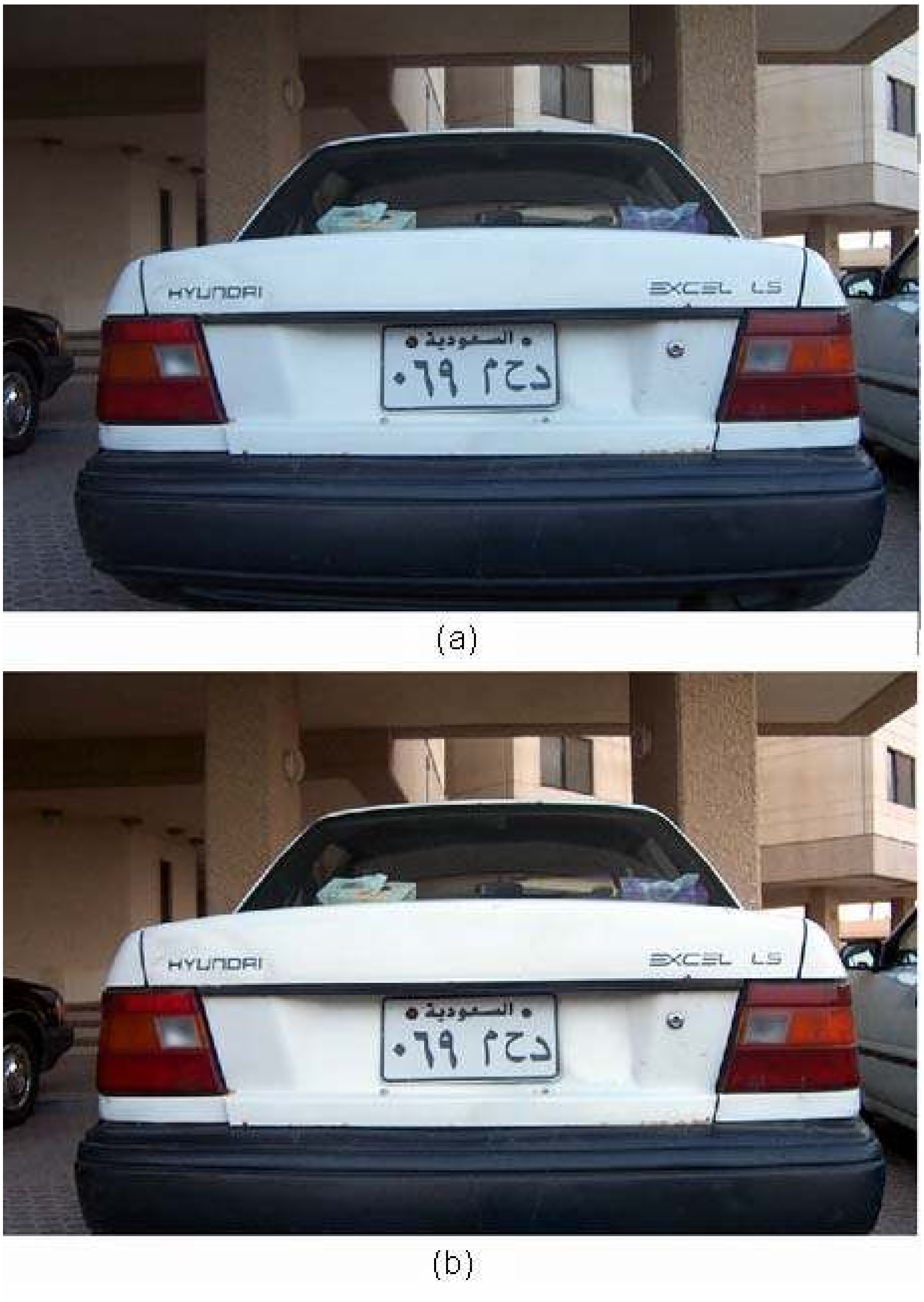


Figure 6-4: (a) Image taken in sufficient light. (b) Effect of contrast adjustment on the image.

6.2.2. Search Map Optimization

We consider the enhanced true color map as our search map. Our core objective is to locate or isolate an area with highest probability of the presence of the plates and characters. The presence of a car in an image is always marked by some local features that are normally not present in the entire image. Before commencing a plate extraction search over an entire image, it is better to look for some local features describing the presence of a car. The objective in this section is only to reduce the search space and chances of mistakenly processing a structure similar to a license plate in the background. If the two techniques fail to locate car presence successfully, the whole image is fed into the system for processing.

In the proposed approach an image is first search for the following features.

- Presence of maximum width horizontal edges
- Presence of car break lights

6.2.3. Horizontal edge based features

The number of horizontal edges in an image always remains greater than the vertical edges. This is mainly due to the presence of Windscreen joints, Bumpers, Headlights and other similar objects present in a car. A horizontal edge map of Fig 6.4 is shown in Fig 6.5 (a).

To enhance the horizontal edges we created a 2D special filter enhancing only the horizontal edges of the image. We used a 3×3 Sobel horizontal edge emphasizing filter h_s given below. The filter emphasizes the smoothing effect by approximating a vertical gradient. The result returned is used as a correlation kernel which is used with a two dimensional filter.

$$h_s = \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix} \qquad (6\text{-}1)$$

The two dimensional filter we are discussing here is our actual image intensity matrix $I(m,n)$, derived from the (Hue, Saturation, Intensity) HSI domain where m is the number of rows and n is the number of columns. The result of this horizontal edge emphasizing filter is shown in Fig 6.5 (a).

Algorithm 6-1: Local Edge-based Feature Search

```
        Input:
        1. A True Color (24 bit) Image Matrix
        2. A Sobel based correlation kernel
Begin
        Scale the image matrix $RGB(m \times n)$ resolution to $320 \times 240$.
        Perform $RGB$ to HSI conversion.
        Obtain the Intensity Map ‘$I(m \times n)$’ (Discard the other two).
        Convert the Map ‘$I$’to Binary Map ‘$BW$’using dynamic threshold.
        Use the special filter given in Eq.6-1 over 2D matrix ‘$BW$’.
        Obtain an 8-connected labeled matrix of filtered $BW$ as’ filBW ’.
        Sort all the regions (horizontal lines) present in filBW in descending
        order according to their widths.
        Select the first three lines (widest)
        If the lines cover more area underneath them in the image. Select that area for search starting from the very first line.
        If a license plate candidate is found, terminate the search
        Else
        Scan the image area above the lines
End
```

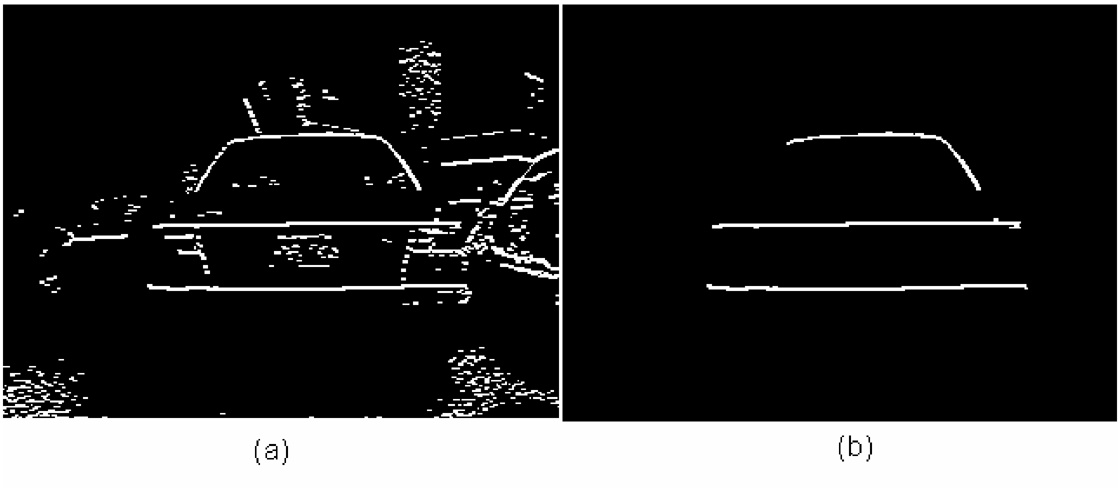


Figure 6-5: (a) A horizontal edge of image present in Fig 6.4 using Sobel Horizontal Edge Mask. (b) The first three edges found.

6.2.4. Break light shape and symmetry based features

Car break lights offer a number of unique features that are rarely common in all the other areas present in an image contain the back of a car. The break lights are prominent because normally the only rich color present at the back of the car is of break lights. The important characteristics of RED color are:

1. Its invariance to HUE in the Hue, Saturation, Intensity (HIS) domain.
2. Dissimilarity of break lights to any other objects in the image.

These characteristics contain some important local features in an image like color, shape, symmetry and distance.

Algorithm 6-2: Local Structure-based Search

```
Input       A True Color (24 bit) Image Matrix
Output      Feature vector for height, width and area of the regional components
Begin
            Scaled down the image ‘ I ’to 320x240 using a bi-cubic filter.
            Convert I  it to HSV domain as ‘ HSV ’
            Set a threshold to extract the upper most color domain
            Apply a ‘7x7’ median filter to remove salt & pepper noise
            Label ‘ HSV ’in a 4-connected regional neighborhood search as
            ‘ labeled ’
            Divide the matrix ‘ HSV ’into two halves
            Store the following values for all connected regions
                    i.   Regional Height
                    ii.  Vertical Width
                    iii. Area
```

Based on all the features, all the regions in the left halve are compared with those on the right. To realize fuzziness in these comparisons a fuzzy set termed as “similar to each other” is shown in Fig 6-6:

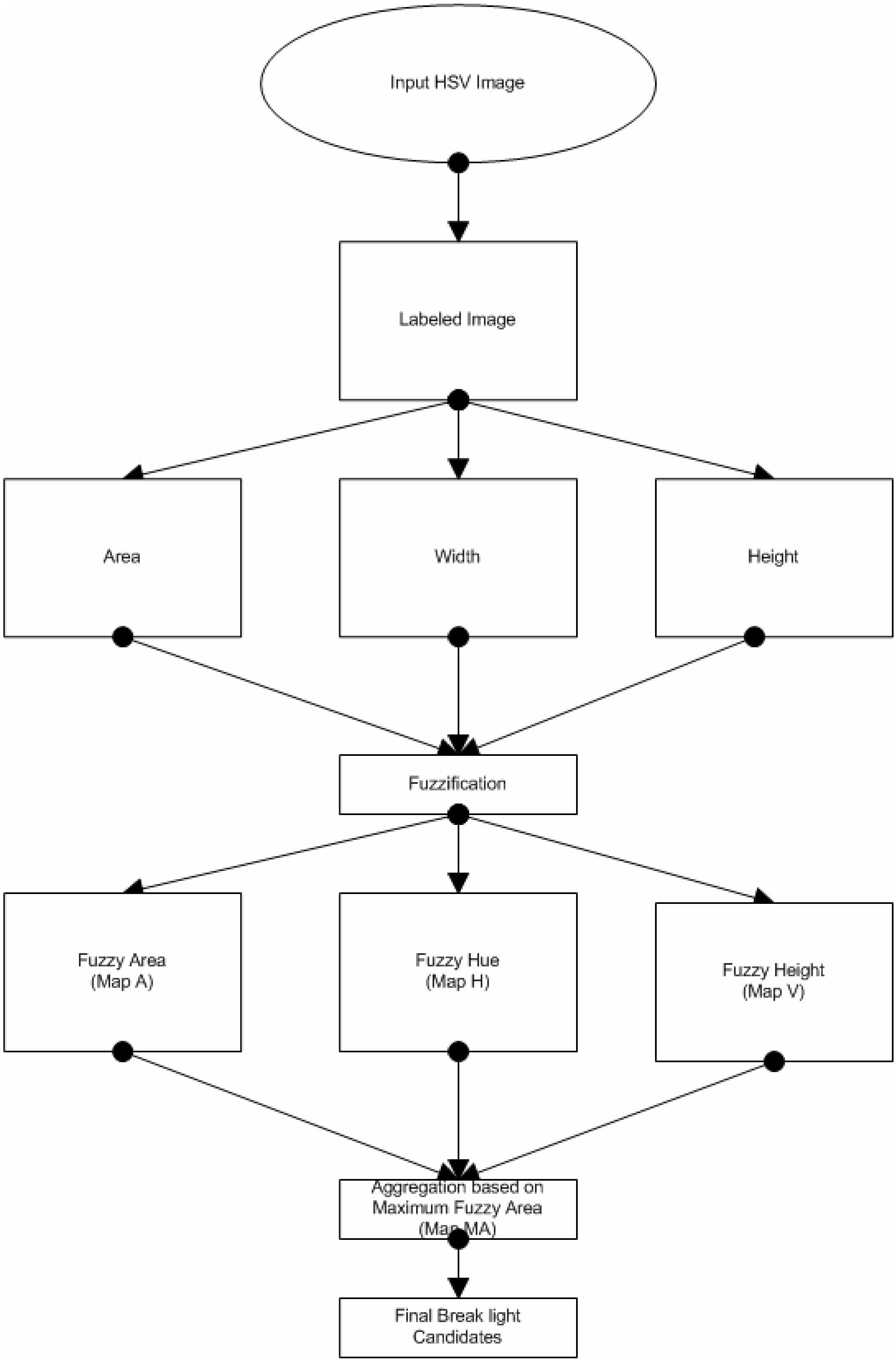


Figure 6-6: Flowchart for Break light pair locating module

Consider the universal set of areas of all regions in the image.

Suppose the object of comparison on the left has an area A_l and that on the right has an area A_r .

The measure of similarity between the two can be calculated as follows:

$$(weight)_{k,l} = \min(A_l, A_r) / \max(A_l, A_r) \tag{6-2}$$

$$membership_{k,l} = A_l \times (weight)_{k,l} \tag{6-3}$$

$$\mu(A_{k,l}) = (membership_{k,l}) \times \frac{1}{A_r} \tag{6-4}$$

Where 'k'is a region in the right half of the image and 'l'is a region in the left half of the image.

The two operators of Fuzzy Height and Vertical Width are calculated on the basis similar to that of Fuzzy Area given in Equations 6-2, 6-3 and 6-4 and will be defined as follows.

$$\mu(H_{k,l}) = (membership_{k,l}) \times \frac{1}{H_l} \tag{6-5}$$

$$\mu(V_{k,l}) = (membership_{k,l}) \times \frac{1}{V_r} \tag{6-6}$$

This results in a $k \times l$ membership matrix in which each region in the left half is similar to a region in the right half of the image by some degree set by the fuzzy membership functions defined in Equations 6-4, 6-5, 6-6. These membership functions result in three separate membership values ranging from 0 to 1 based on the closeness of the region with the one compared on the basis of the three maps discussed. The overall aggregate fuzzification as shown in Fig. is achieved by multiplying the membership values to achieve an ultimate membership as follows:

$$\mu(AHV_{k,l}) = \mu(A_{k,l}) * \mu(H_{k,l}) * \mu(V_{k,l}) \tag{6-7}$$

The final map, thus obtained, as shown in Fig 6-6; has the final break light candidates selected on the basis of maximum-area-pair.

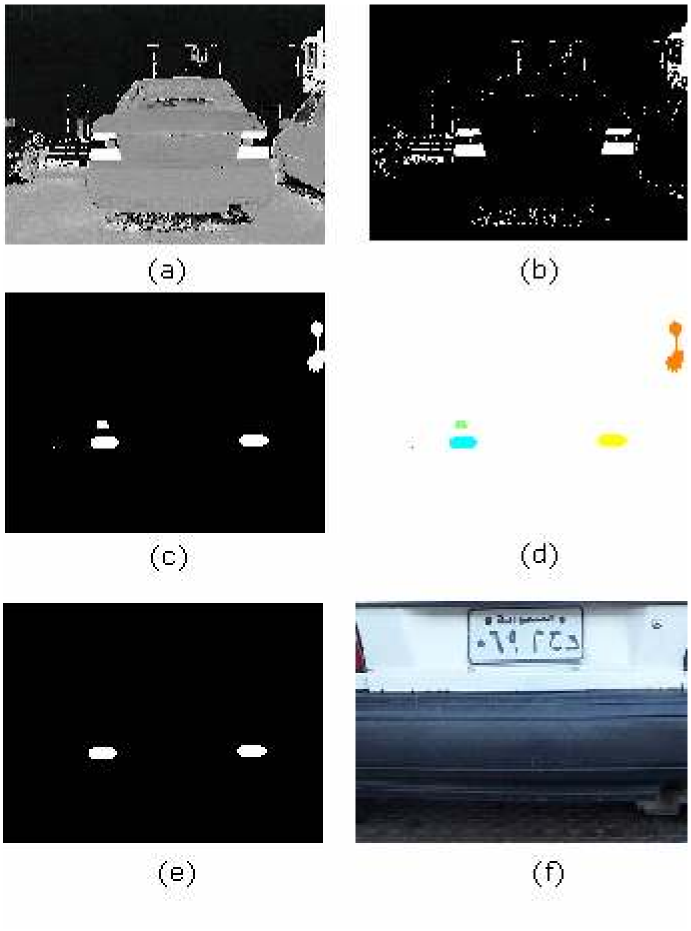


Figure 6-7: (a) HUE Map of an image shown in Fig 6-4 of resolution 320x240. (a). (b) Threshold Image for the top 20% rich values (RED). (c) Candidate area with similarity to break lights. (d) A color map of (c). (e) Final Break light pair. (f) Clipped Image for input of high resolution.

The two techniques based on extracting local features to minimize search area contributed a small part for

6.2.5. Color Edge Detection

From the enhanced true color image, its corresponding edge maps are calculated and modified on the basis of a fuzzy operator calculating the degree of edginess of a pixel in a neighborhood. The edge maps are calculated on the basis of two different techniques.

a. Euclidean Distance based technique.
b. Vector Angle based technique.

The two techniques have already been discussed in detail in Chapter 3. To ensure better performance, the techniques were employed using a simple 2×2 color mask.

6.2.6. Euclidean/Vector Angle Robert's Operator

A simple edge operator known as Robert's operator was used [58]. The operator uses a 2×2 neighborhood of the current pixel. The convolution masks used by Robert's Operator are as follows:

$$h_1 = \begin{bmatrix} h_1(0,0) & h_1(1,0) \\ h_1(0,1) & h_1(1,1) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, h_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad (6\text{-}8)$$

The edge magnitude is computed as follows

$$\left|g(i,j)-g(i+1,j+1)\right|+\left|g(i,j+1)-g(i+1,j)\right| \tag{6-9}$$

This operator can be generalized to multidimensional pixel values for ED and VA as follows:

$$R_{ED}=\max\begin{pmatrix}\left\|c(x,y)-c(x+1,y+1)\right\|,\\ \left\|c(x,y+1)-c(x+1,y)\right\|\end{pmatrix} \tag{6-10}$$

$$R_{VA}^{2}=1-\max\begin{pmatrix}\dfrac{c^{T}(x,y)c(x+1,y+1)}{\left\|c^{T}(x,y)\right\|.\left\|c(x+1,y+1)\right\|},\\ \dfrac{c^{T}(x,y+1)c(x+1,y)}{\left\|c^{T}(x,y+1)\right\|.\left\|c(x,y+1)\right\|}\end{pmatrix}^{2} \tag{6-11}$$

where $c(x,y)$ is the vector containing the multiple values of the pixel at coordinate (x,y), and $\left\|\bullet\right\|$ is the L_2 vector norm.

The masks thus obtained were enhanced using a membership function indicating the degree of edginess in each pixel's neighborhood. The membership function is measured as follows.

$$\mu_{edge}(g(x,y))=1-\frac{1}{1+\left(\sum_{N}\left|g(x,y)-g(i,j)\right| \big/ \Delta\right)} \tag{6-12}$$

Where $g(x,y)$ is the pixel whose edginess is to be determined and N is a predefined neighborhood (3×3 in our case) along which the edginess of the pixel is to be calculated. 'Δ' is the total number of pixel present in the neighborhood N (8 in our case). The membership function enhances the edginess of a pixel in a way that if there are a lot of edges present in a neighborhood, the membership value of the pixel $g(x, y)$ increases to '1'. The same value asymptotically reaches '0' if there are almost no pixels present in the neighborhood. The maps thus obtained are filtered based on their eight connected neighborhood.

6.2.7. Plate Candidate Selection

The Edge Maps thus obtained are searched for connected components similar to a license plate. The feature used is the regional compactness of each connected area found. It is compared with the compactness of the standard license plate image using Fuzzy Logic. Based on area and perimeter, the compactness of a fuzzy set can be determined as follows:

$$a(\mu) = \sum \mu$$

$$P(\mu) = \sum_{m=1}^{M} \sum_{n=1}^{N-1} \left|\mu_{m,n} - \mu_{m,n+1}\right| + \sum_{n=1}^{N} \sum_{m=1}^{M-1} \left|\mu_{m,n} - \mu_{m+1,n}\right| \tag{6-13}$$

$$Compactness(\mu) = {a(\mu)}\Big/{P^2(\mu)} \tag{6-14}$$

The value of compactness is invariant to distance and skew in the image and serves well in Fuzzy domain even if an extent of noise distortion is present. This resulted in a Compactness Map $\mu_{\hat{C}}(c)$ for all the candidate regions present.

$\hat{C}$ **Map**: Suppose that the object of interest has compactness $'c_i'$. Given an entry in map $\widehat{C}$, say $'c'$, the membership degree, $\mu_{\hat{C}}$ of the entry belonging to fuzzy set *'Like a Plate'* can be written

$$\mu_{\hat{C}}(c) = \exp(-a\left|c - c_i\right|) \tag{6-15}$$

According to the definition of a Fuzzy Map, a large entry indicates a high degree of possibility that the region belonged to a license plate. The candidate license plate(s) are extracted on the basis of threshold criteria finding out the minimum and the maximum entry present in the $\hat{C}$ Map. Let $\tilde{M}$ be any fuzzy map,

$$threshold = (\tilde{a}_{\max} + \tilde{a}_{\min})/2 \tag{6-16}$$

Where $\tilde{a}_{\max}$ and $\tilde{a}_{\min}$ are the maximum and minimum values in $\tilde{M}$. All the entries that are greater than the threshold are taken as potential candidates of being a license plate. The process of filtering unwanted regions normally ends up with more than one license plate candidates. This situation is handled by matching all such regions with a standard template as shown in Fig 6-8 (b).

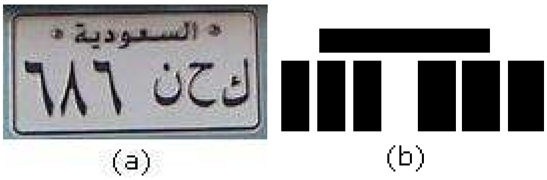


Figure 6-8: (a) Standard Saudi Arabian License Plate. (b) Saudi Arabian License Plate Template (I_T)

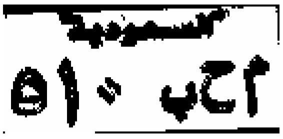
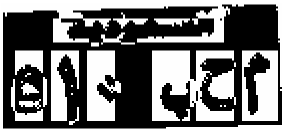
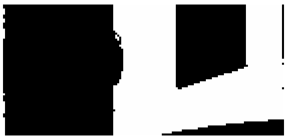
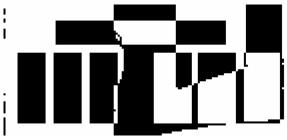
Candidate Image: **'I$_C$'**	**XOR Matrix:** $I_T \oplus I_C$	**Mismatch**
		3839
		4745

Figure 6-9: The white area in the XOR matrix displays the mismatch present between the actual and the template image.

Let I_T be the template image and I_C be the candidate image to be compared. The candidate image is first normalized to the template image size $[65\ 145]$ using Nearest-neighbor Interpolation. We measure the difference between the two matrices using the standard Hamming Distance (XOR). The operation on the binary image matrix of the two will give a HIT/MISS based matrix containing ones in the regions with dissimilarity and zeros in the regions containing similar pixels. Counting the number of MISS pixel positions give us the error code or the number of pixel that mismatched with the standard template. We simply select the candidate with lowest mismatch value. The whole extraction process starting from the search optimization based cropped image is shown in Fig 6.10(a) – Fig 6.11.

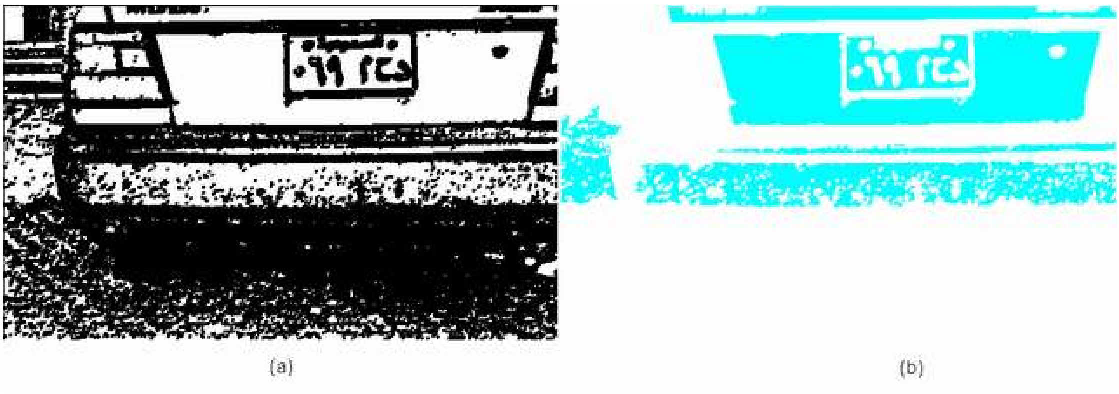


Figure 6-10: (a) Vector Angle based edge map of image shown in Fig 6-4(b). (b) 8-connected areas of the image shown in (a).

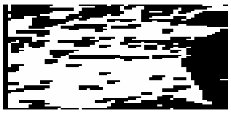
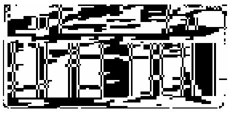
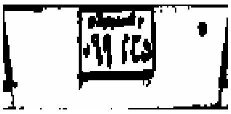
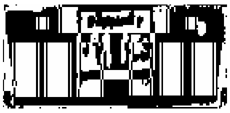
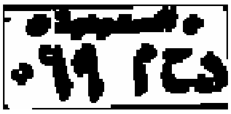
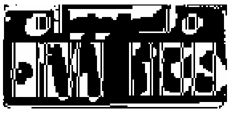
Candidate Areas I_C	**XOR Matrix:** $I_T \oplus I_C$	**Mismatch**
		4902
		4352
		2887
(a)	(b)	(c)

Figure 6-11: Candidate areas shown in column (a) are the normalized form equivalent in dimension to the template image shown in Fig 6.8(b).

6.3. Character Segmentation

The license plate segmentation is used to divide the final plate extracted into individual characters. The individual recognition of characters is only possible when the plate is divided into six isolated blocks each containing a single character. This is done because a standard Saudi Arabian license plate consists of 6 characters, with 3 letters and 3 numerals.

6.3.1. Plate Enhancement

Before commencing the segmentation process, the plate area is enhanced using a dynamic threshold based plate conversion. The gray scale image of the plate is converted into a binary image using a dynamic mask of size 15×15. The mask averages the gray scale intensity value in a 15×15 neighborhood for a pixel P_i and converts the pixel to black or white on the basis of this threshold.

RGB-to-Binary conversions, using a mask of such larger sizes, are not recommended at big images and are, therefore, not carried out in real time application. The approach proved very effective in removing noise and intensity distortions.

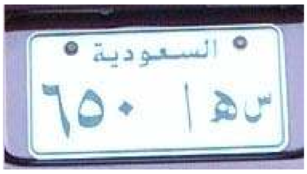
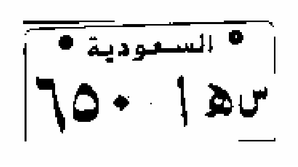
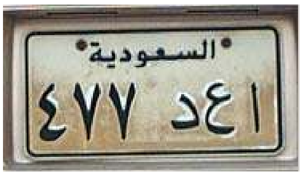
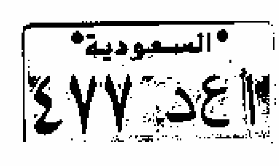
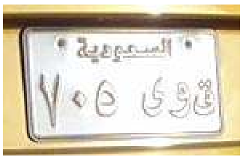
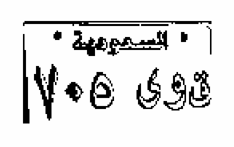
Original Plate	Dynamically Threshold
	
	
	

Figure 6-12: Plates taken under (a) Fog. (b) Rust. (c) A tilt of 45 degree and a Reflection

Note: The additional area visible around each plate shown in Fig 6.12 is only to compensate for the huge mask size.

The effect of lesser mask size is shown in Fig 6.13 (c).

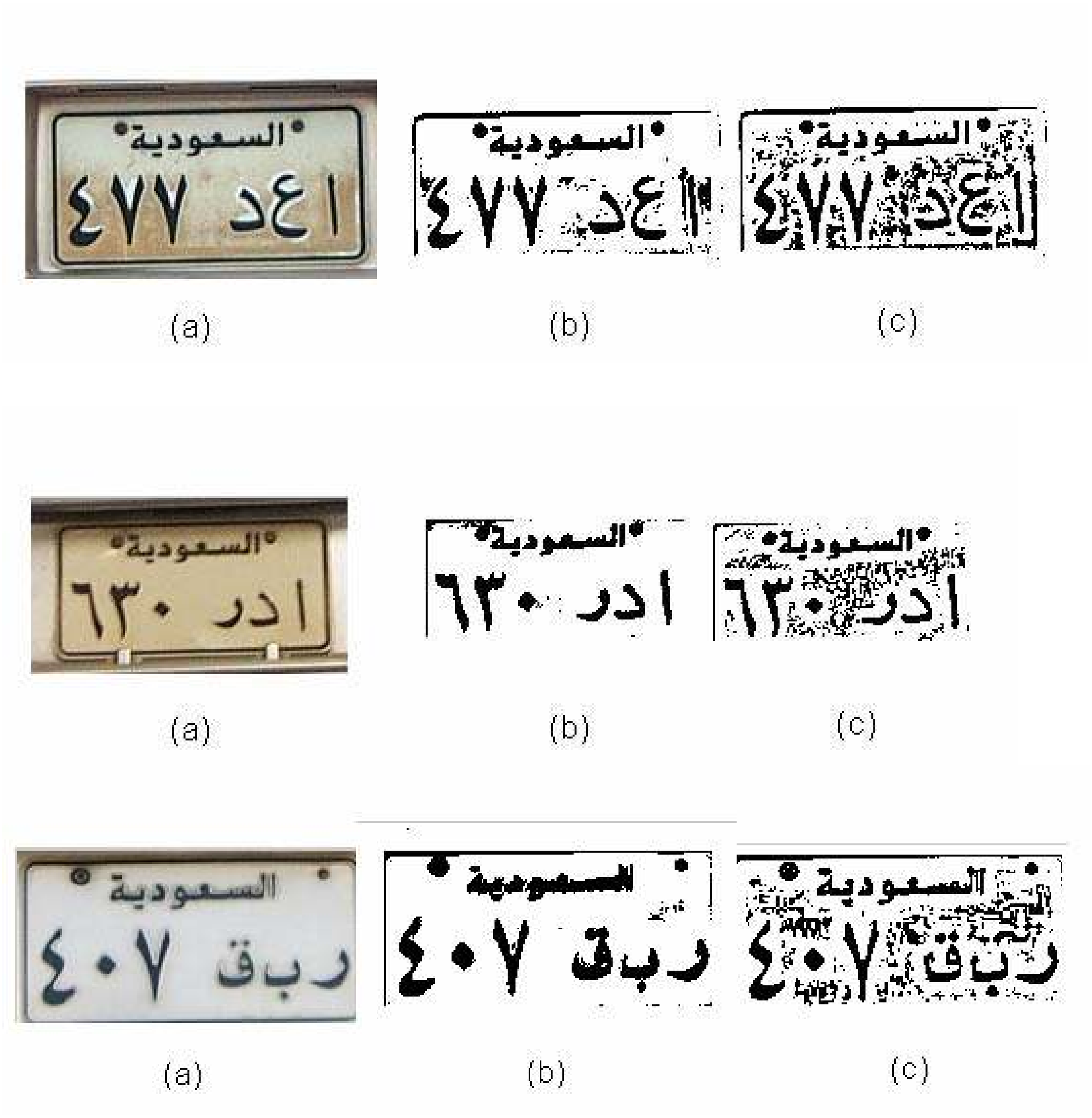


Figure 6-13: All three cases. (a) Actual Plate. (b) Threshold with a 15×15 mask

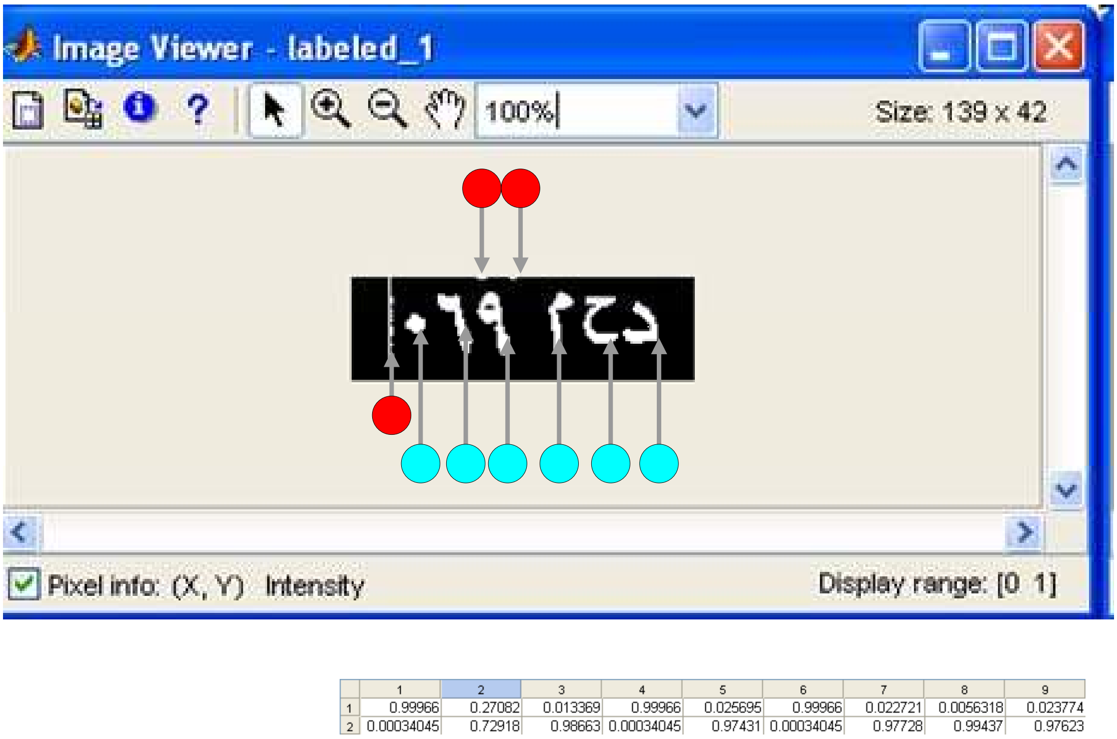
(c) Threshold with a 5×5 mask.

Before commencing to the character isolation phase, the upper part of the plate is cropped to remove السعودية. For the segmentation phase, the proposed strategy is based on separating every connected component present in the plate using a hybrid approach. The approach uses a fuzzy c means algorithm with the plate’s projection profile.

6.3.2. Fuzzy C Means Clustering

The features local to a license plate that differentiate between license plate characters and other objects such as bolts, monograms and state names can be utilized to classify valid characters and drop other areas before commencing an image histogram projection. This eliminates significant number of areas that affect the threshold of a plate's segmentation.

We basically adopted a hybrid approach of making use of bounding box search for extracting the local segments. A feature vector based on the centroid, vertical and horizontal width of the regions found in the candidate plate area was passed through a bi-cluster fuzzy c-means algorithm. The actual cluster among the two was decided on the basis of its average centroid height in the plate. The process is shown in Fig 6.14. If the number of segmented characters found is larger than the prescribed number (six), components are deleted one at a time starting with the smallest one until the above condition is satisfied.



	1	2	3	4	5	6	7	8	9
1	0.99966	0.27082	0.013369	0.99966	0.025695	0.99966	0.022721	0.0056318	0.023774
2	0.00034045	0.72918	0.98663	0.00034045	0.97431	0.00034045	0.97728	0.99437	0.97623

Figure 6-14: The figure shows a tilt-invariant clustering of license plate clusters using a fuzzy c means algorithm (cluster-2 candidates are shown in blue color).

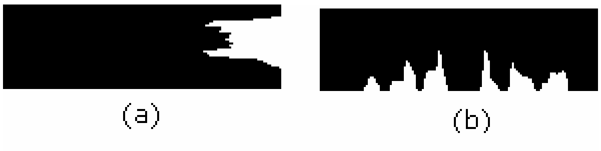


Figure 6-15: Horizontal (a) and Vertical (b) projection profile of the plate based on the selected cluster (cluster 2) from the image shown in Fig 6.14.

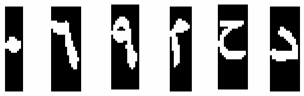


Figure 6-16: The final '6' characters of the license plate

6.4. Plate Recognition

Plate Recognition is the final phase in any LPR system. The phase deals with the recognition of the characters isolated in the phase of character segmentation. Respecting the fact that characters present on the license plate are of same font, size and shape, recognition is considered a simple task. Our work here deals with the offline recognition of the characters.

Because classification, basically, is a task of generating a group that generated the patterns, various classification techniques are useful depending on the type of application at hand.

There are basically three approaches in pattern recognition: Statistical, Syntactic and Neural [65]. Statistical Approach is based on decision making (i.e. probabilistic model), Syntactic deals with the structural description of the pattern and Neural is based on training the system with a large dataset of input and storage of weights that are used at the later stages in recognition of trained patterns.

6.4.1. Statistical Pattern Recognition

In statistical pattern recognition we address properties of patterns related to the probability densities. In this case the model for a pattern may be a single specific set of features, though; the actual pattern has been corrupted by some form of random noise. Focusing this major factor we effort was concentrated at utilizing an approach from this area for our recognition problem. The area of neural pattern recognition is considered a close descendant of statistical pattern recognition [65].

6.4.2. Syntactic Pattern Recognition

If the model consists of a set of crisp logical rules, the methods of syntactic pattern recognition are used. Basically a set of grammar describe the recognition module in such a case.

6.4.3. Recognition using Principal Component Analysis

To prevent human error in reading number plates, the characters chosen in the Saudi Arabian number plates are categorized based on the structure of the character and the characters bearing dots are kept as low as possible. Characters with similar features are avoided. This left a typical Saudi Arabian number plate with only 17 out of 26 Arabic alphabets and 10 numerals. The situation favors the recognition and classification strategies as well.

Since the standard set for the License Plate only used key features of Arabic characters to make reading easier for human beings, a classification technique making use of these key features would provide nearest results. A sudden human glance over a bunch of characters of a rapidly separating car uses structural features for identification such as the lengthy and straight structure of ا, the semi-rounded shape of ن or arced nature of ر enables one to discriminate them from other such characters on the foundation of their principal components.

Keeping the idea of such key features in mind, we tested the segmented characters using PCA (Principal Component Analysis). The basic structures (2D matrix) of all of these 17 characters and 10 digits were passed as feature vector for training to the PCA.

Definition: A set of variables that define a projection that encapsulates the maximum amount of variations in a dataset and is orthogonal (and therefore) uncorrelated to the previous principle component of the same dataset.

PCA is commonly used as a cluster analysis tool. It is designed to capture the variance in a dataset in terms *principle components*. In effect, we try to reduce the dimensionality of the data to summarize

To use PCA, five steps must be performed as follows:

1. Development of proper data
2. Calculation of Covariance Matrix
3. Calculation of Eigenvalues/Vectors
4. Selection of Eigenvectors
5. Data Mapping

If it is chosen to use R eigenvectors, then the dataset must include at least R samples from each class.

2. Calculation of Covariance Matrix

To hold the correlation within the training data, the covariance matrix, C_x is calculated as follows:

$$C_x = E\{(x - \mu_x)(x - \mu_x)^T\} \tag{6-17}$$

3. Calculation of Eigenvalues/Vectors

The eigenvalues, λ_j and eigenvectors, e_j are calculated as follows:

$$\det(C_x - \lambda_j I) = 0 \qquad (6\text{-}18)$$

The eigenvectors span an orthonormal space. This space will be used for the mapping of training data.

4. Selection of Eigenvectors

After the solution of the Eigenvalue problem, proper eigenvectors must be determined to be used.

5. Data Mapping.

Data mapping is done as follows

$$C = B^T (x - \mu_x) \qquad (6\text{-}19)$$

$$x = (x_1, x_2, ..., x_n)^T \qquad (6\text{-}20)$$

Where x denotes the output data.

B denotes the transformation matrix containing the chosen eigenvalues as column vectors.

x denotes the input data.

μ_x denotes the centroid of the training data.

CHAPTER 7

EXPERIMENTAL ANALYSIS AND RESULTS

The chapter describes the tests and analysis done on all the thesis work described in the previous chapter. The tests performed for various techniques used for the phases of extraction, segmentation and recognition will be discussed. Analysis of the following issues will be addressed in detail.

- **Image enhancement**
- **Edge Mapping and Detection**
- **Fuzzy Techniques used for**
  - Compactness Matching
  - Edge Enhancement, and
  - Segmentation
- **Mask based Averaging for plate enhancement**
- **Recognition**

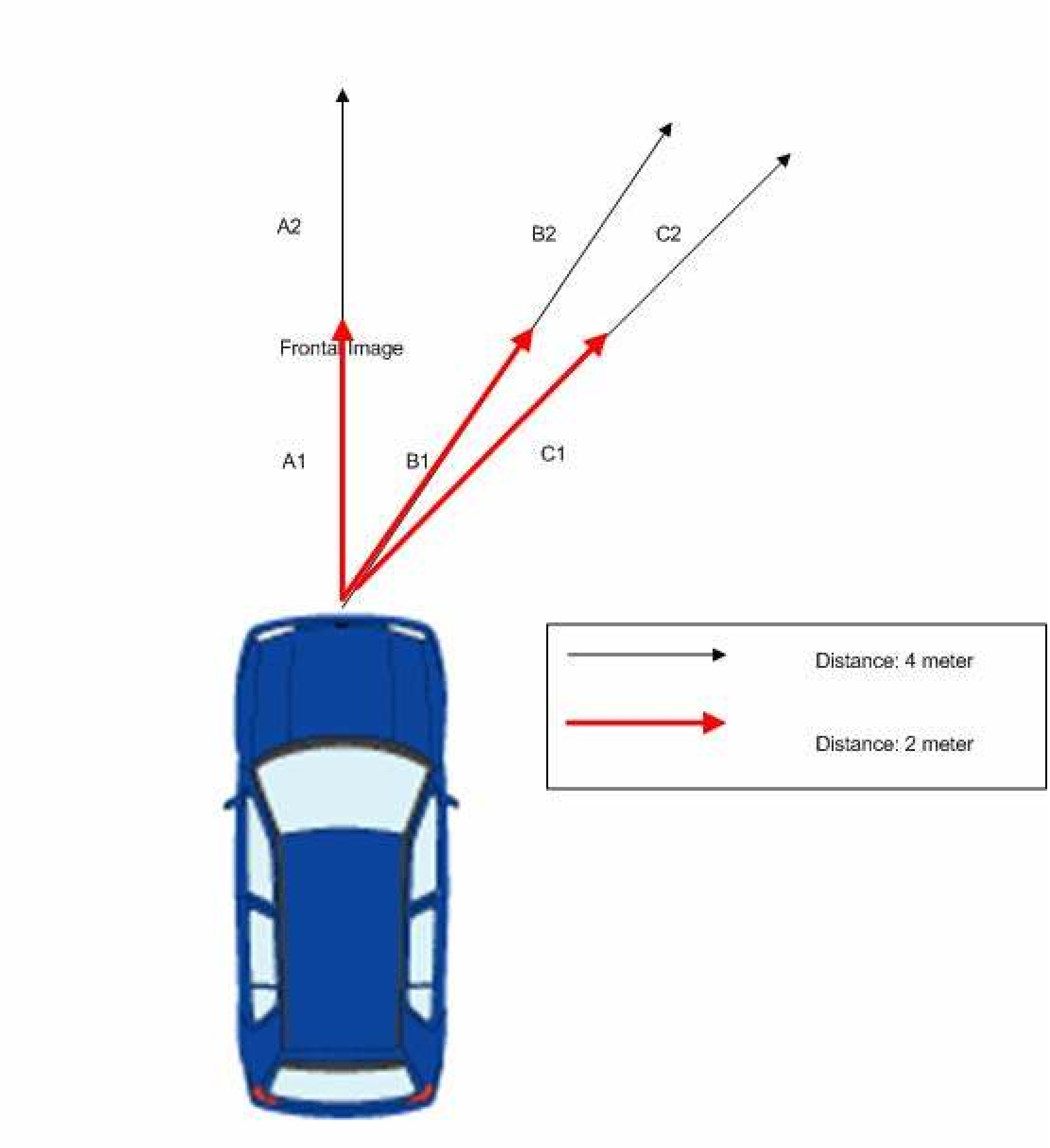


Figure 7-1: Figure displaying the various angles and distances at which the test samples were taken.

Table 7-1: Number of images taken at various measures

	2 meter	4 meter
0 degree skew	200	200

45 degree skew	198	202
60 degree skew	35	17
Total Number of Images: 852		

7.1. Image Enhancement

In the extraction phase the main object remains to enhance an image taken in considerable low light or glares due to the presence of luminance at an angle incident to the camera. A good enhancement technique helps in clearing areas with a balanced color presence (like a license plate) to enhance their contrast as compared to the other objects present with comparable uneven illumination.

7.1.1. Criteria of success

While testing this important factor must be kept in mind that the goal is to enhance the plate area only. Since we are saturating (mapping) intensities in the upper (whiter) 2% of the image, the license plate that generally appears to be among the whitest areas in an image, enhances its presence.

7.1.2. Testing

The image contrast adjustment module was tested basically at two sets of images. The first set consisted of images taken in a garage at around 10:00 PM and contained images either low in details due to the presence of lesser light, shadows, tilt or those affected by artificial light sources. The second set was taken in open environment under either direct sunlight or sufficient illumination. The angular and distance information was kept as shown in Fig 7.1.

The results were initially tested manually and observed for any apparent improvement. The phase had a significant effect over the forthcoming license plate extraction phase as it improved the rate of license plate recovery cases.

7.2. Extraction Test

In the extraction phase a test is performed and results are compared with the method proposed by Sarfraz et al [2]. Sarfraz et al [2] used Hough transform to detect vertical edges; filtering, vertical edge matching and B/W ratio matching for license plate extraction.

The proposed method is based on steps starting from the enhanced true color image: Color Edge Detection, Connected Area Blob Search, Candidate Search based on a Fuzzy Compactness Map and Template Matching for final extraction.

7.2.1. Criteria of success

The goal in this phase is to shorten the search for the license plate candidates by taking only those areas that are 8-connected based on the edge maps obtained. This leaves us with a limited set of regions in which at least one is probably containing the license plate. The criterion is relatively straight forward and was adopted on the basis of the following two reasons.

1. An 8-connected search doesn’t come up with regions that are computationally hard to process.
2. The regions are numerically less than those extracted on the basis of vertical edges and are, therefore, less expensive to process.

The situation does not emphasize on what is present in these connected regions neither is it extracting the characters when the plate candidate is finalized on the basis of the following reasons:

1. The edge maps (Vector Angle, Euclidean distance) detect more edges than regular color edge detection schemes. This generally results in oversized, inter-connected character pairs that are hard to segment. Though, the schemes proved its robustness in its own account.
2. Once the final candidate is decided, it is relatively easy to use some well defined technique (addressed in the thesis under section 6.4) to cater the character segmentation process.
3. Size and shape filtering is error prone to cars with reflective surfaces.

7.2.2. Test Data and Test Description

The images were taken as shown in Fig 6.1. The images were taken at varying lighting conditions including low light, direct sunlight and uneven intensities. The test set included plates with dirt, rust, skew, break and other noise distortions.

As mentioned earlier, the test was performed on all the images shown in Table 7.1. Generally a lower number of connected regions will make the task of finding the number plate easier.

7.2.3. Results

The proposed method based on Vector Angle based edge detector turned out to be a very robust way of finding the license plate. As Table 7-2 indicates, it almost always created an 8-connected license plate candidate entry in the edge map. In only 8 out of 852 test images it failed in locating a connected region. The failure cases are shown in Fig 7.7 and Fig 7.8 for vector angle based technique. This is because either the plate character got connected because of severely illuminated surface as shown in Fig 7.7 or the plate's black outline was unclear with a background of same hue and saturation as shown in Fig 7.8.

In the case of Euclidean Distance based edge map, the number of failures was a bit high. This was basically due to the fact that the edge detector only emphasized edges similar in color but different in intensity. The system failed for 12 out of 852 images in locating the 8-connected edge map.

The 8-connected labeled matrix, thus obtained, contained a number of candidate connected region. The regions were tested for a fuzzy compactness map that created a membership based on the similarity of compactness of the candidate region with that of a standard license plate. Those with membership values close to 1 (greater than

0.8), were selected as prospective license plates candidates. All the cases tested contained at least one actual license plate. The final candidate plate selection module compared these candidates with the template of a standard Saudi Arabian license plate (as discussed in the previous chapter). This, again, had an almost 100% outcome. The statistical results are shown in Table 7-3.

Table 7-2: Comparison with previous work.

	Method	Success Rate	Percent Error	No. of Regions
1.	Hough Transform	46/72	63.88%	50-100
2.	VA	844/852	99.06%	2-5
3.	ED	840/852	98.59%	2-8
4.	Vertical Edge	68/72	94.44%	20-30

Table 7-3: Success rate at candidate selection

	Method *(Candidate Selection)*	Success Rate (VA)	Success Rate (ED)
1.	Fuzzy Compactness Matching	842/844	838/840
2.	Template Matching	842/842	838/838

7.3. Segmentation Test

The section describes the test for isolation of characters using a hybrid approach based on fuzzy clustering and projection profiles. To enhance the plate image a dynamic mask based threshold was performed over the plate area. The method simply takes average value of a masked neighborhood for each pixel and sets the pixel's value on the basis of that threshold. The quality of the image increases with the size of the mask as shown in Fig 6.11. The image detail converges at a point with the mask increase.

The image, thus obtained, is processed to extract areas using fuzzy c means clustering. Only areas extracted in this phase are subject to the pixel profile projections (horizontal and vertical). On the extracted areas, horizontal pixel profile is found. The profile only has one wide peak formed by the character rows. The clipped plate on the basis of horizontal profile is shown in Fig 6.13 (a). The clipped plate is passed through a vertical pixel projecting profiler to obtain the final candidate characters. The tests on these methods are provided with the results.

7.3.1. Criteria of success

The purpose of this step is to divide a license plate into exactly six sub images, each containing one of the six characters. A successful isolation fills all of the following criteria:

- The plate must be divided into six sub images

- None of the six characters have significant loss of details.
- The order of the images has to be in correct order and matches the characters present on the plate.

7.3.2. Test Data and Test Description

The method for isolating the characters is a simple black box test. An input image is given to the license plate character isolation module and the success of the test simply depends upon the resulting outcome as discussed in the previous section. The test is performed on the images received from the successfully extracted plates. The process, therefore, was tested on 842 and 838 images for VA and ED based image maps respectively.

7.3.3. Results

Table 7-4: Hybrid approach outcome of Fuzzy C Means/Projection based segmentation

Technique	Method	Rate/Percentage (VA) Out of 842		Rate/Percentage (ED) Out of 838	
Hybrid Approach	Direct extraction from edge maps	785	93.23%	779	92.95%
	Masking	803	94.24%	790	94.27%

7.4. Recognition Test

The final step in recognizing a license plate is identifying the single characters extracted from the segmentation stage. An approach based on Principal Component Analysis is presented earlier in chapter 6.

7.4.1. Criteria of success

The standard of success is simply a high recognition rate of the isolated characters.

7.4.2. Test Data

The test data originates from the 852 test images extracted and isolated during the previous tests. The actual test data is the single characters extracted from the license plates. The characters are identified using principal components of the blocks extracted from the two dimensional image matrices.

7.4.3. Results

The system was trained and tested with PCA on a large set of input data. (852 input images). First the system was trained on the dataset available and based on the training data; a database of classifiers for different classes of characters was created. The system was tested on the new dataset to get the overall recognition rate. In the case of the proposed system, there are 27 set of classes for the 27 characters used on the license plate. From the dataset, 70% was used to train the PCA and 30% was used for testing. The overall recognition using PCA is shown in Table 7-5.

Table 7-5: Success rate for individual characters using PCA

Character	Recognized English form	Recognition Rate
Alif	**A**	**95%**
Baa	**B**	**92%**
Haa	**H**	**98%**
Daal	**D**	**73%**
Raa	**R**	**71%**
Seen	**S**	**67%**
Suad	**Sa**	**68%**
Tua	**T**	**97%**
Aien	**Ae**	**77%**
Quaf	**Q**	**88%**
Kaaf	**K**	**95%**
Laam	**L**	**93%**
Meem	**M**	**100%**
Noon	**N**	**94%**
Waow	**W**	**81%**
Yaa	**Y**	**100%**
One	**1**	**100%**

Two	**2**	**85%**
Three	**3**	**85%**
Four	**4**	**100%**
Five	**5**	**100%**
Six	**6**	**85%**
Seven	**7**	**95%**
Eight	**8**	**95%**
Nine	**9**	**100%**
Zero	**0**	**100%**
	Average	**97.46%**

7.5. System Test

In the previous sections, the components were examined separately in terms of performance. It is also necessary to test the combination of the components. In real life, an error made in the initial stages of the system ripples through the system, thereby, affecting the overall performance.

The system is designed and developed in Matlab 7.1 for the Identification of Saudi Arabian license plates. The input image to the system is a 24-bit image of size 900×600. The test images were taken under various illumination conditions at angular and distance measurements shown in Fig 7-1. The experiments were performed for the following cases:

- Images taken under normal illumination conditions
- Images taken under various skews
- Distorted Plates
- Occluded Plates

The whole system is achieving a high recognition rate. Table 7-6 gives an overall performance evaluating results for the license plate extraction, segmentation and character recognition. A failure is generally encountered in the cases of extremely poor quality of image. A few of such cases are presented at the end of the chapter. It is shown that the system correctly isolated 803 plates in the case of VA based edge detection and 790 plates using ED based technique. The individual testing in the recognition phase over the characters was carried out over characters extracted from 30% of the license plates. The overall system successfully identified 791/803 plates in the case of VA based technique giving an overall accuracy of 92.82% and 742/790 plates in the case of ED based implementation giving an accuracy of 87.09% over the total 852 plates. The whole system design was implemented using Intel Pentium® 1.7 MHz Dual Processor machine under Matlab 7.1 IDE.

The experimental results show that the shortcoming of the proposed system was mainly due to the following factors:

1. Poor image quality due to unstable imaging stance of the photographer.
2. Unrealistic tilt (Fig 7-7) in the image.
3. Extremely low light.

4. Extreme depression in the image angle.
5. High intensity light incident over the image.

There was a slight error in the segmentation was of the system mainly due to the presence of illegal label between the character blocks. Since the standard Saudi Arabian license plate format follows a single font, the recognition phase doesn't have the complexities that are usually present in the classification of unconstrained recognition.

Table 7-6: Results for extraction, segmentation recognition phase (VA)

	Extraction	Segmentation	Recognition	Overall
Success	842/852	803/842	791/803	791/852
Percent Recognition	98.82%	95.36%	98.5%	92.84%

Table 7-7: Results of extraction, segmentation and recognition phase (ED)

	Extraction	Segmentation	Recognition	Overall
Success	838/852	790/838	742/790	742/852
Percent Recognition	98.35%	94.27%	93.92%	87.09%

7.6. Summary

The chapter covered the test on the individual phases of an LPR system with some of the comparisons with the work previously done in this area. For the extraction phase the proposed method showed a better result then the extraction done by Hough Transform as well as Vertical Edge matching. An improvement in accuracy and a drastic improvement in the application processing time were recorded when comparing all the techniques with the proposed one. For segmentation, the system proved error proneness to tilt of high degrees and distance. An overall system was also presented attaining good results.

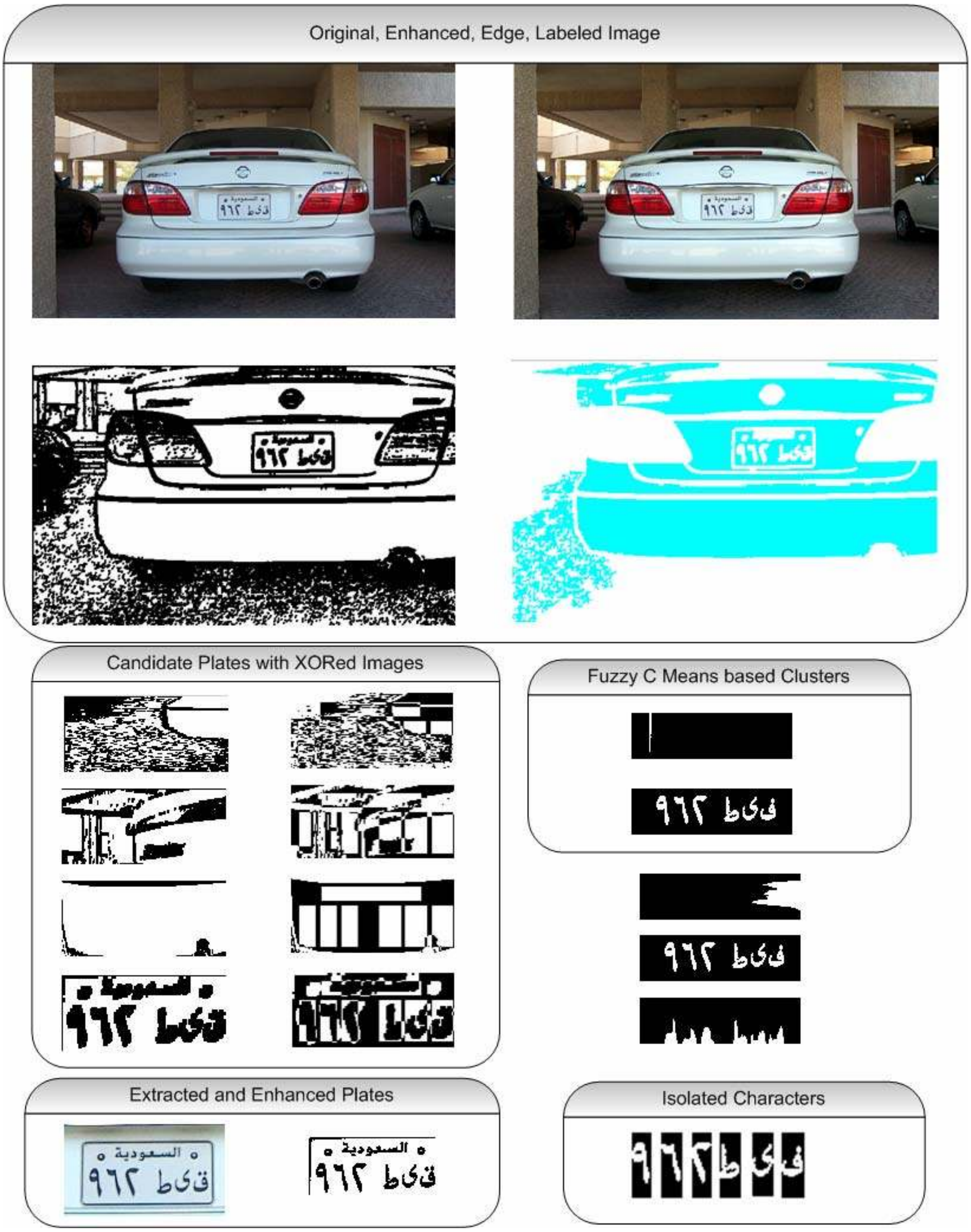

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-2: Image taken under normal illumination conditions with similar plate and background Hue and Intensity (VA implementation)

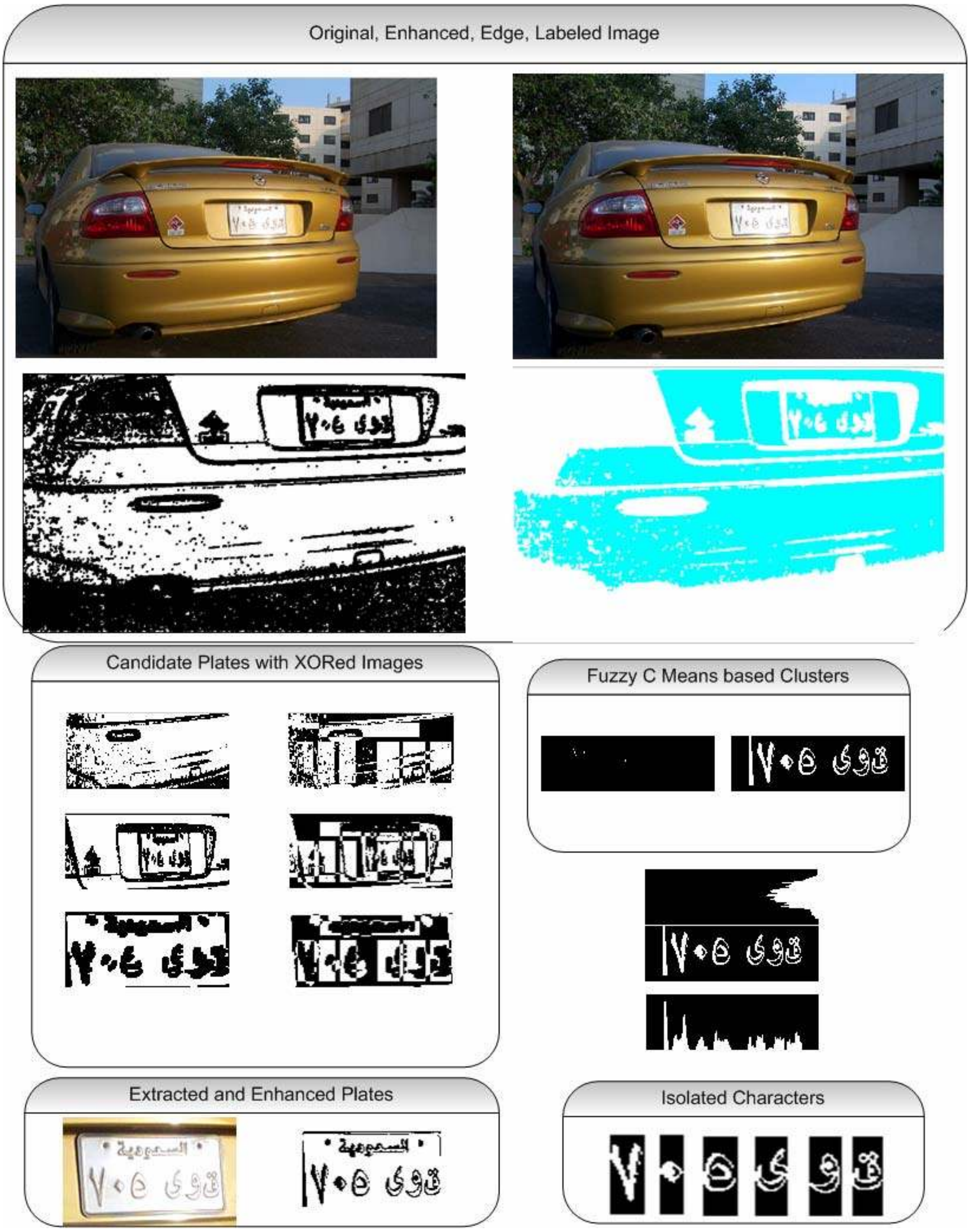

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-3: Image taken under the effect of glare with significant intensity variations at the plate surface area (VA implementation)

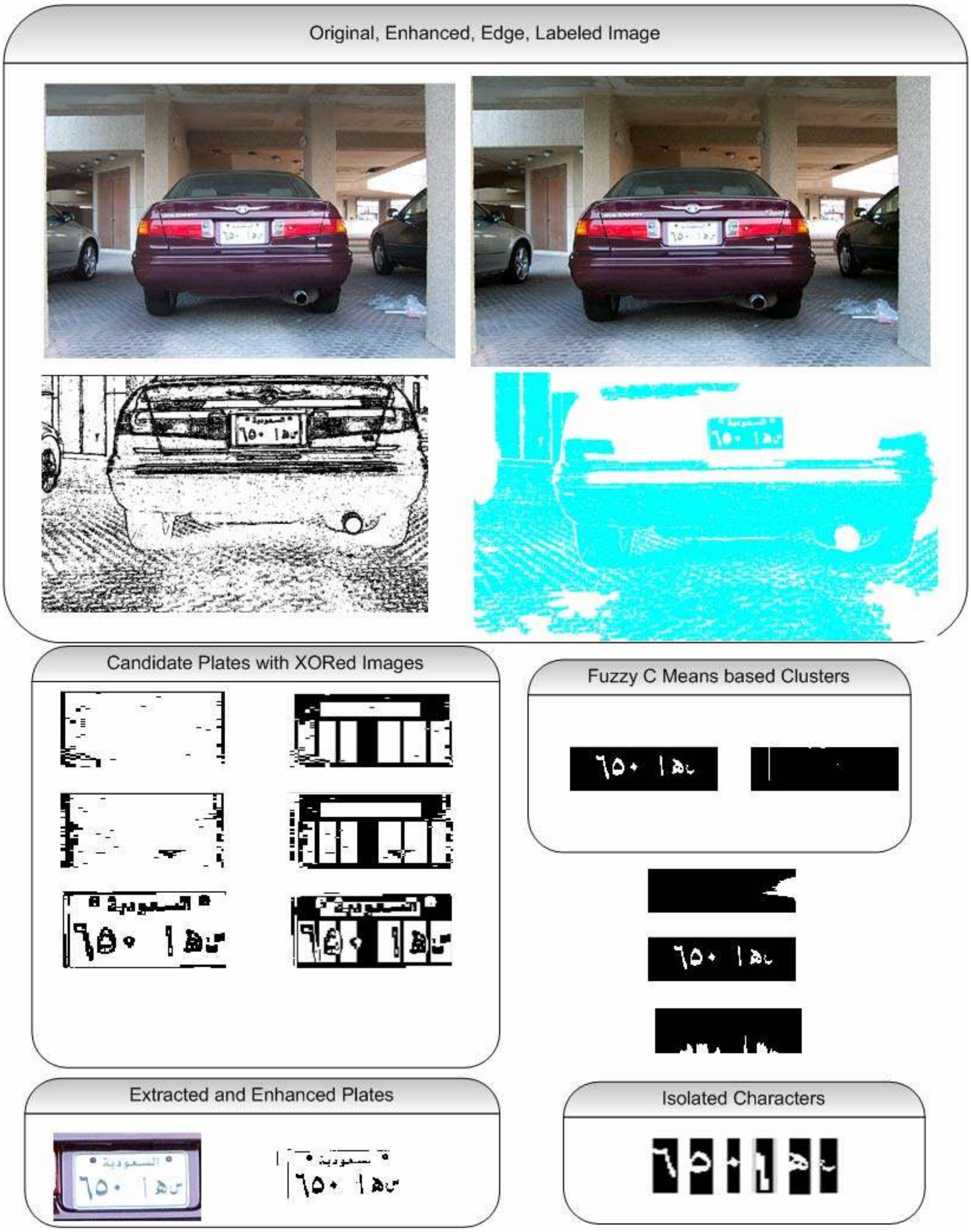

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-4: Image of a car with dim license plate and characters low in color profile (ED implementation)

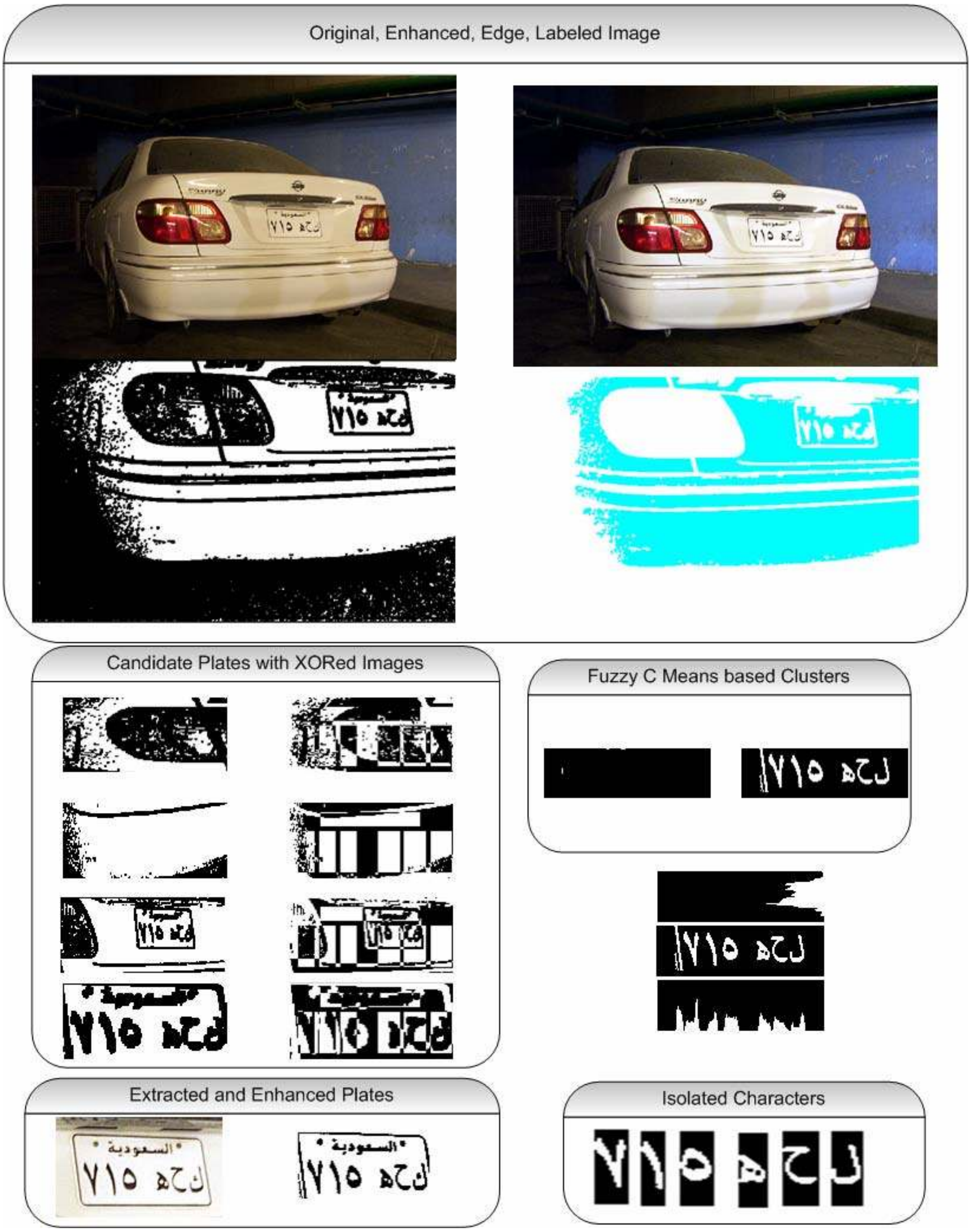

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-5: Image taken under low light. The plate's background color is similar to the foreground (VA implementation)

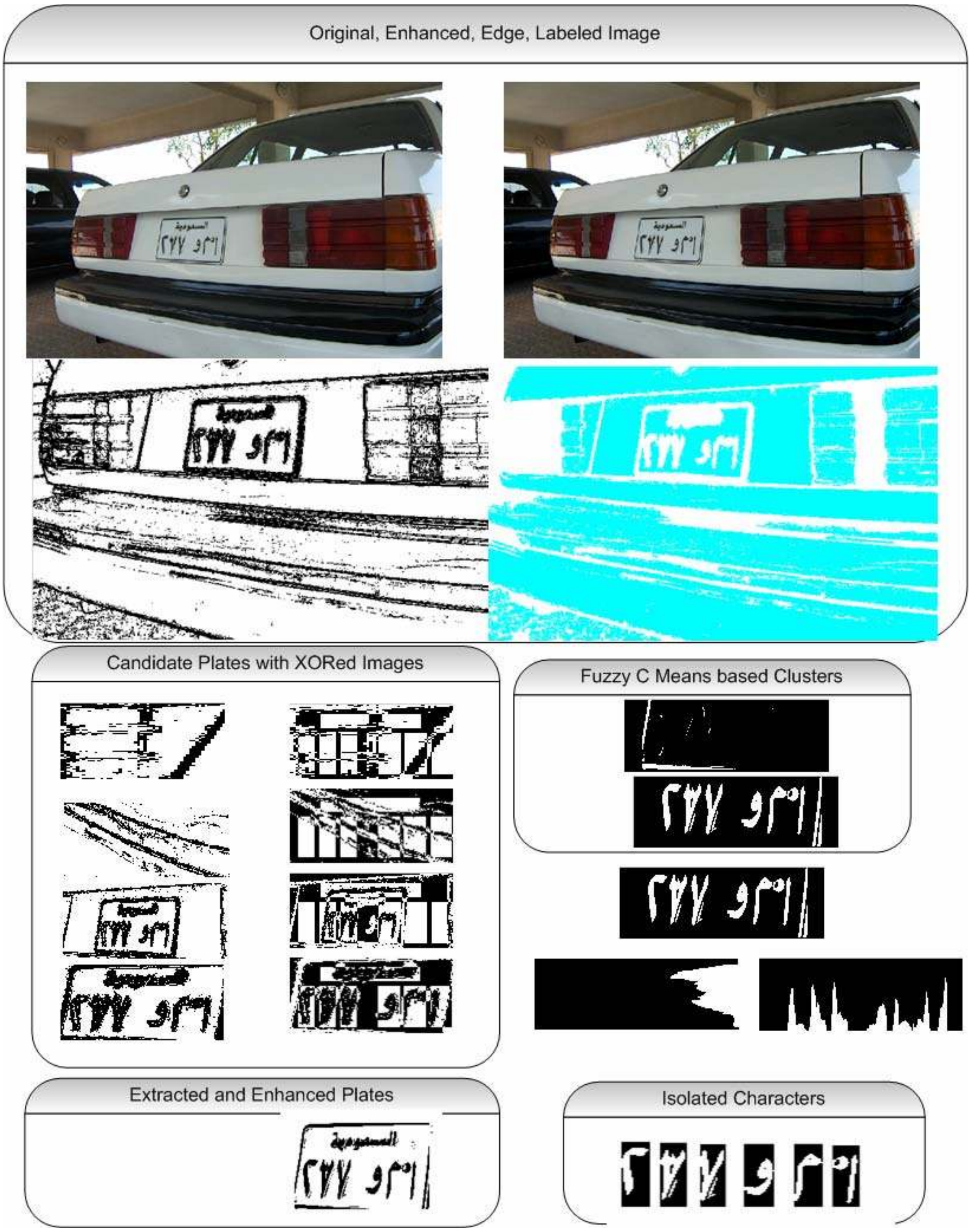

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-6: Plate with an extreme tilt (ED implementation)

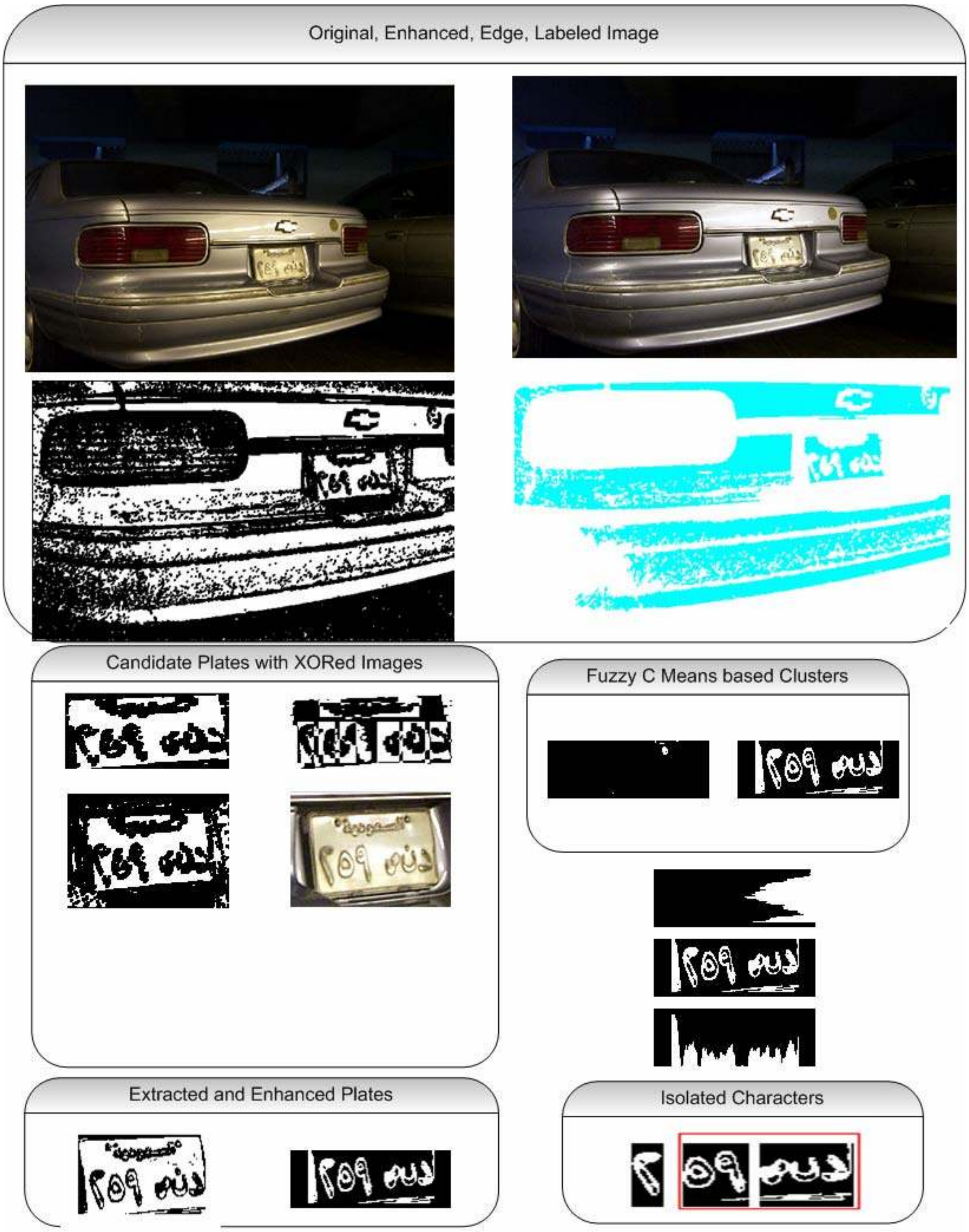

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-7: Failure case: Image taken under extreme low light with artificial illumination directly incident over a bent plate (VA implementation)

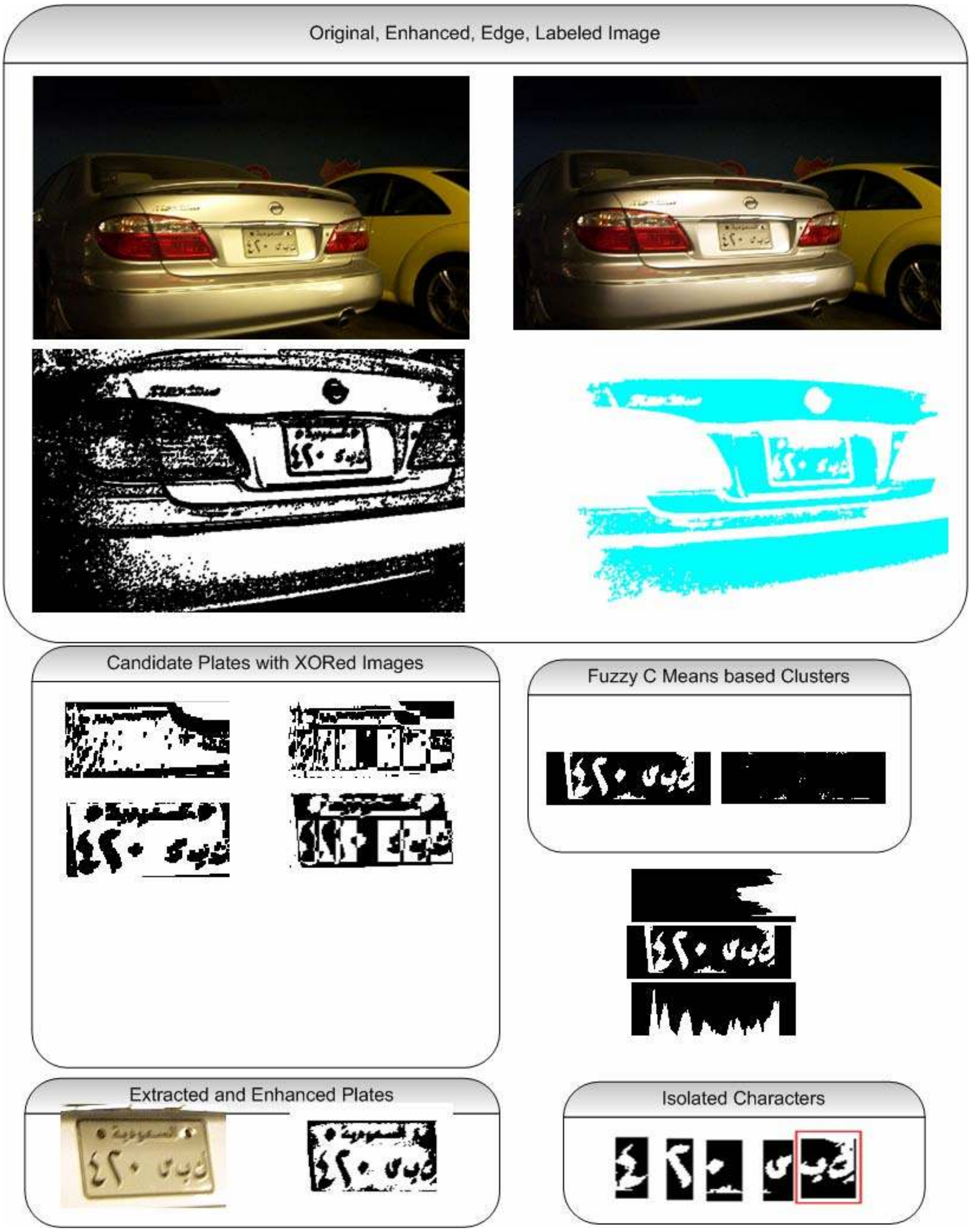

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-8: Failure Case: Image taken at low light with the plate's color saturation similar to the background color saturation. (VA implementation)

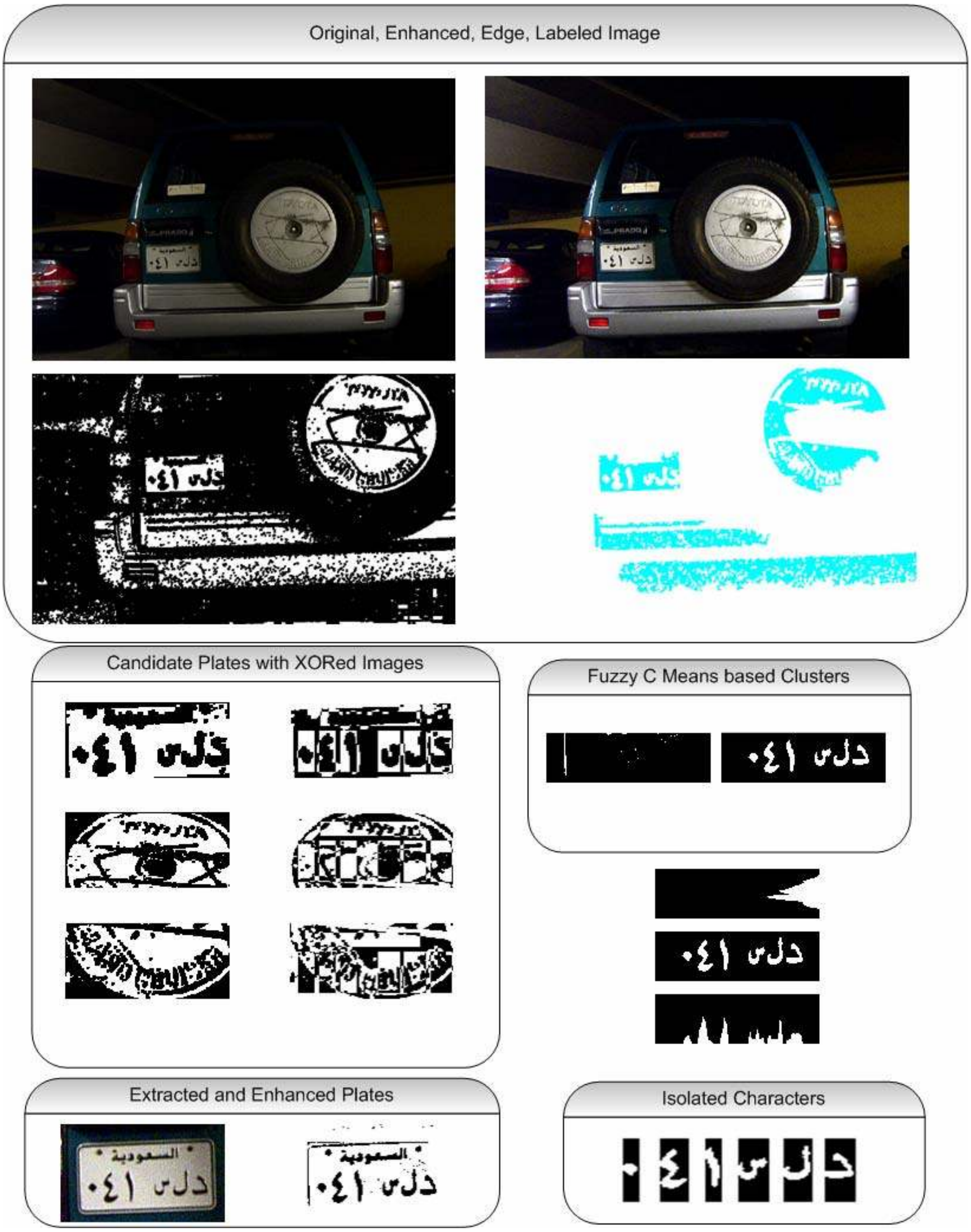

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-9: Image taken at low light and plate under shadow effect(VA implementation)

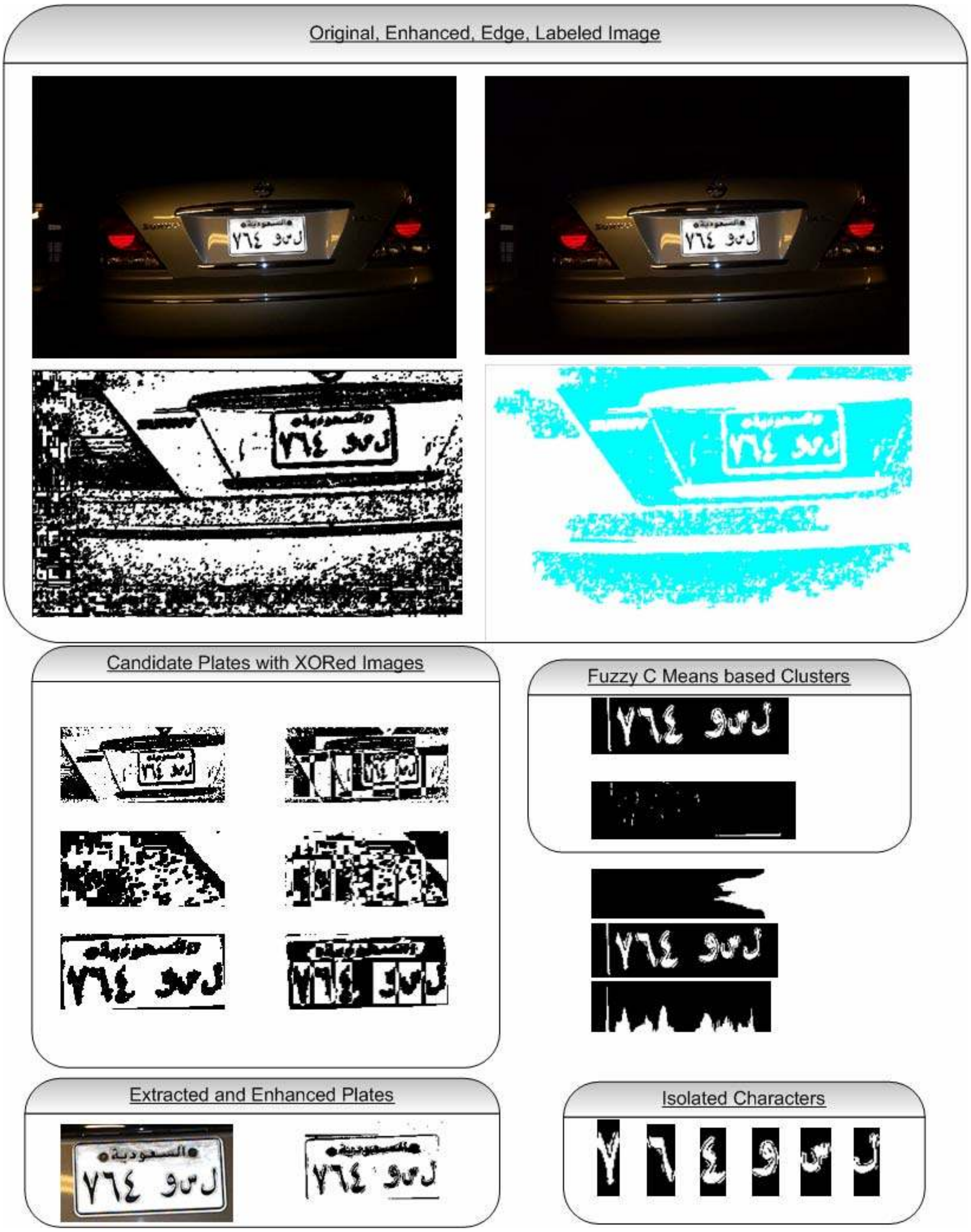

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-10: Plate under direct illumination of camera flash (VA implementation)

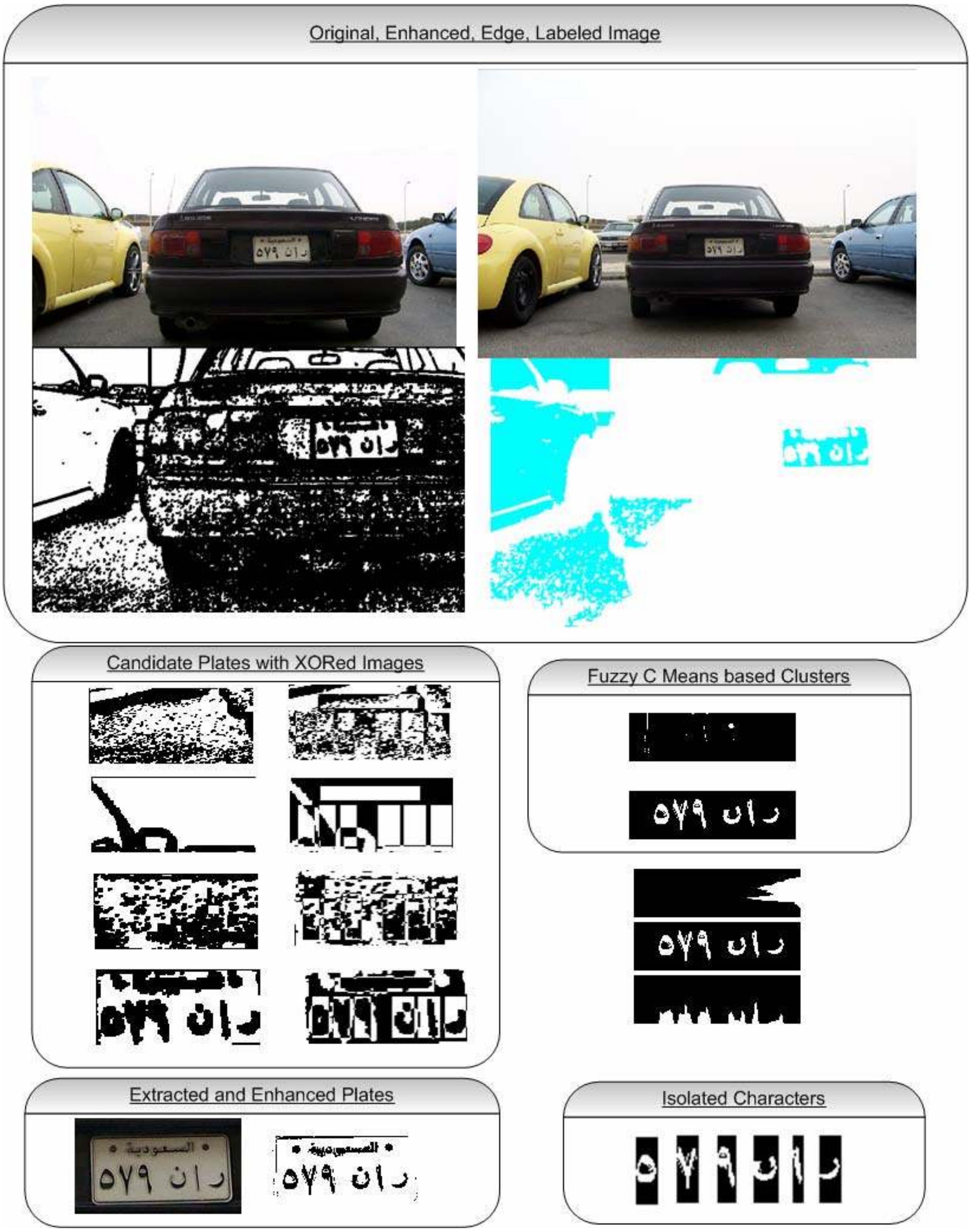

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-11: Image taken at sunset under receding light (VA implementation)

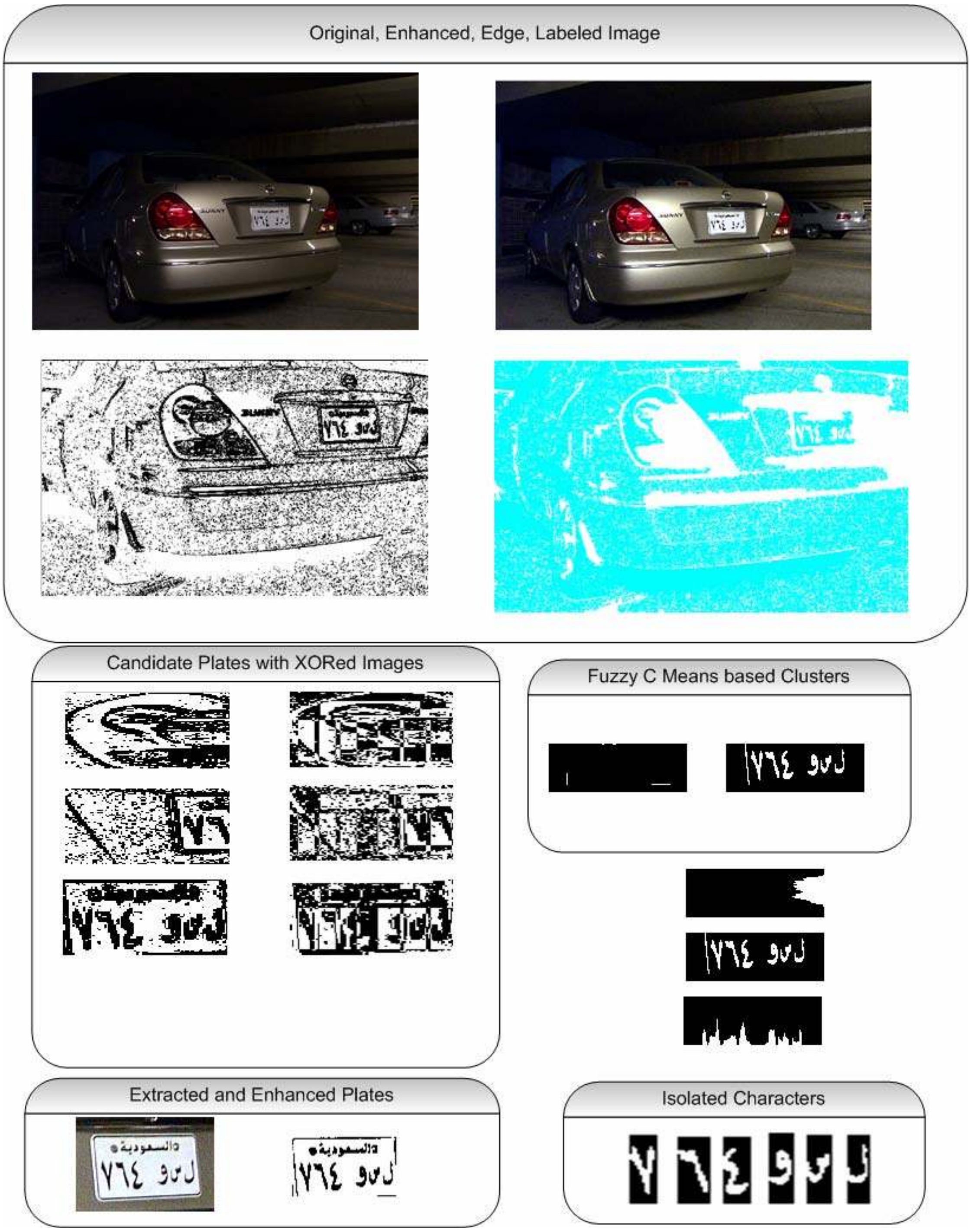

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-12: A tilted image under low light (ED implementation)

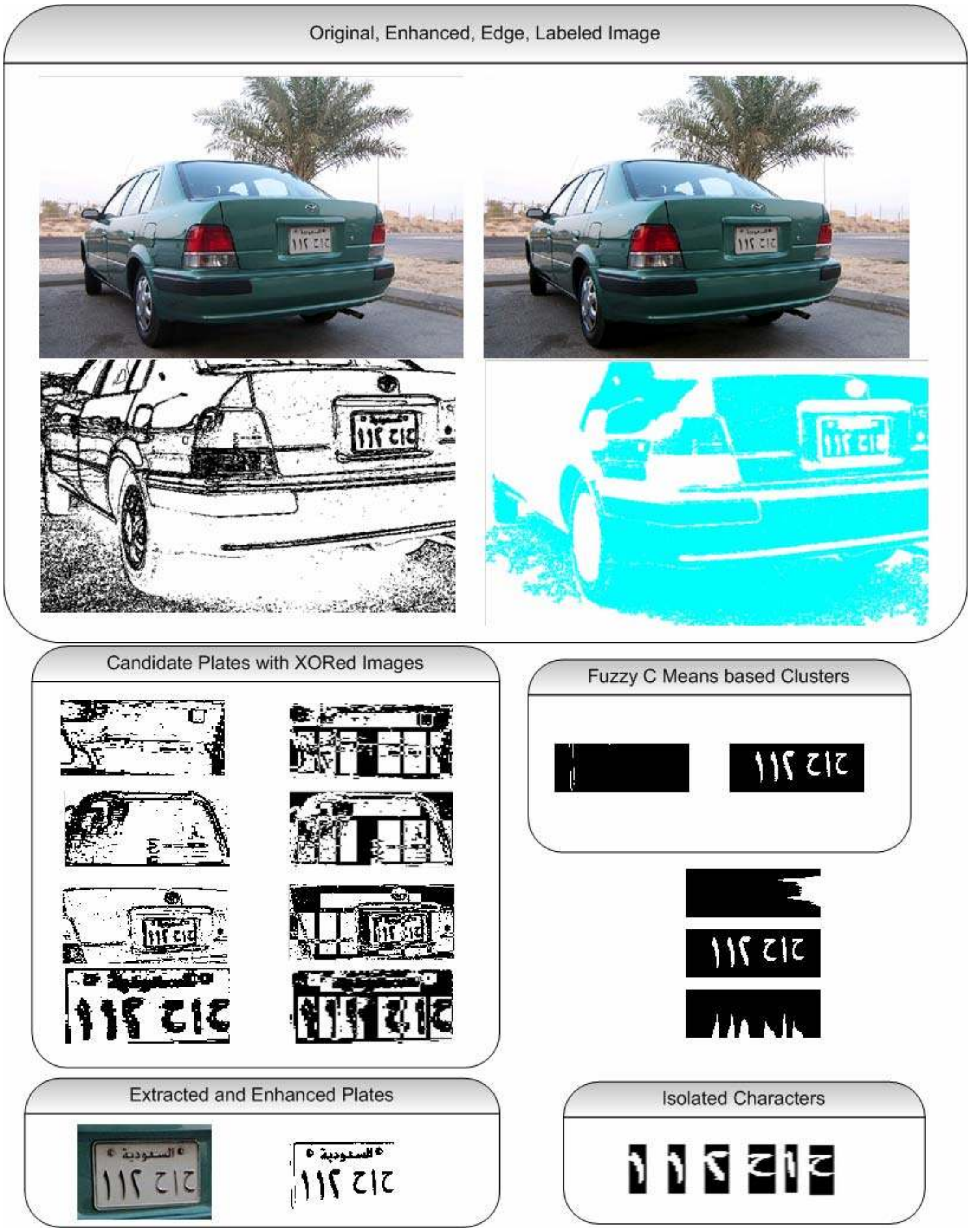

Original, Enhanced, Edge, Labeled Image
Candidate Plates with XORed Images
Fuzzy C Means based Clusters
Extracted and Enhanced Plates
Isolated Characters

Figure 7-13: A tilted image at low light (ED implementation)

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1. Conclusion

The purpose of this thesis has been to investigate the scope of automatic license plate recognition under minimal restrictions. The main objective was aimed at contributing towards the research in the fields of machine vision, pattern analysis and image processing. The system developed investigates the possibility of automating the whole process of license plate recognition for a wide range of environments. Given an input image, the system extract extracts the license plate, isolates the characters, and finally identify the characters. For each task, a set of methods were proposed, designed and developed. The input image was first passed through an image enhancement and intensity adjustment algorithm to obtain better detail of snaps taken in low light. For the extraction, a connected component labeling technique was used to obtain a number of regions. The technique was performed on the color edge maps based on Vector Angle and Euclidean Distance based techniques. The edge profile of these maps was enhanced using a fuzzy edginess calculating procedure. The final edge maps thus obtained were processed to obtain 8-connected regions that were finalized a

The number of regions were shortened using a Fuzzy technique based compactness mapping and comparing algorithm. The final candidate was decided on the minimal mismatch criteria based on Hamming Distance using XORed image matrices.

The method based on projection profiles were not very successful but a hybrid approach based on clustering regions of interest first improved the performance of projection based segmentation drastically.

For the recognition process, a technique based on Principal Component Analysis was used. The technique was able to recognize even the characters that were structurally incomplete or broken especially the character ح that normally losses information at its thinner part in distant or noisy images.

In order for the system to be robust and reliable, it should be able to combine the three separate stages and to recognize the license plates in high percentages, in order to keep the manual work as minimal as possible. This implies that the success rate of the system should be 100%.

8.2. Future Work

The system explores the core possibilities available in image enhancement and restructuring.

Steps can be taken to further enhance the image areas with highest probability of license plate presence.

At the moment the system uses two core techniques for extracting edges namely Vector Angle and Euclidean Distance. Both the techniques have their own significances discussed earlier. The fact is widely accepted that ED and VA metrics take into account the intensity and chromaticity information. To exploit their particular features, specific combination operators can be worked upon. These operators can be further enhanced to HSV domain, thereby, utilizing the flexible human perception of Hue, Saturation and Intensity domains.

The issue of compactness threshold for candidate selection at the moment is hard-coded on the basis of thresholds set by experience. The system can be trained using some pattern recognition technique over a large set of license plates taken under various conditions. This can make the number of candidates much shorter in environments much complex and detailed.

The area of segmentation uses a straight forward fuzzy clustering technique to remove noise and unrelated areas. There are no weight assignments to enhance the presence of either one of the feature vectors used. The features currently used are Centroid, Vertical and horizontal height). The system, simply, gives equal weight to all the entries. Further work can be done to define membership functions enhancing the presence of either of the features.

The approach is currently developed used sequential programming techniques. Since Matlab 7.1 has no thread support. Most of the areas in image processing like image reading, type conversions, mask based operations, noise removal, etc can be handled using thread based developments. An implementation done in a concurrent environment will enhance the running time of the application significantly.

REFERENCES

[1] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto, Robust license-plate recognition method for passing vehicles under outside environment. IEEE Transactions on Vehicular Technology, Volume: 49 Issue: 6, pp: 2309-2319, 2000.

[2] M. Sarfraz, and M. J. Ahmed, S. A. Ghazi, "Saudi Arabian License Plate Recognition System", IEEE International Conference on Geometric Modeling and Graphics, Proceedings of the 2003.

[3] B., Allen, and Hamilton, "Intelligent Transport Systems Field Operational Test", prepared for US Dept. of Transportation, Federal Highway Administration, Washington D.C., US.

[4] N., H., C., Yung, K., H., Au and A., H., S., Lai, "An approach to Korean license plate recognition based on vertical edge matching", IEEE International Conference on Systems, Man and Cybernetics, 2000, pp 2975-2980.

[5] N. H. C. Yung, K. H. Au, and A. H. S. Lai, "Recognition of vehicle registration mark on moving vehicles in an outdoor environment," IEEE International conference on Intelligent Transportation Systems, 1999, pp. 418–422.

[6] S. Draghici, "A neural network based artificial vision system for license plate recognition system ", International Journal for Neural Systems, Vol. 8, pages 112-113, February 1997.

[7] P. Davies, N. Emmott, and N. Ayland, “License plate recognition technology for toll violation enforcement,” Inst. Elect. Eng. Colloquium Image Analysis for Transport Applications, pp. 7/1–7/5, 1990.

[8] K. Yamaguchi, Y. Nagaya, K. Ueda, H. Nemoto, and M. Nakagawa,“A method for iidentifying specific vehicles using template matching,”, IEEE Int. Conf. Intelligent Transportation Systems, 1999, pp., 8-13

[9] L. Salgado, J. M. Menendez, E. Rendon, and N. Garcia, Automatic car plate detection and recognition through intelligent vision engineering, Proceedings of IEEE 33rd Annual International Carnahan Conference on Security Technology, pp. 71-76, 1999.

[10] D. G. Bailey, D. Irecki, B. K. Lim, and L. Yang, Test bed for number plate recognition applications, Proceedings of the First IEEE International Workshop on Electronic Design, Test and Applications (DELTA’02), IEEE Computer Society, 2002.

[11] T. Sirithinaphong and K. Chamnongthai, “The recognition of car license plate for automatic parking system,” in Proc. 5th Int. Symp. Signal Processing and its Applications, 1998, pp. 455–457.

[12] N. H. C. Yung, K. H. Au, and A. H. S. Lai, “Recognition of vehicle registration mark on moving vehicles in an outdoor environment,” in Proc .IEEE Int. Conf. Intelligent Transportation Systems, 1999, pp. 418–422.

[13] S. Draghici, “A neural network based artificial vision system for license plate recognition,” Int. J. Neural Systems, vol. 8, pp. 113–126, 1997.

[14] P. Davies, N. Emmott, and N. Ayland, “License plate recognition technology for toll violation enforcement”, Institute of Electrical Eng. Colloquium Image Analysis for Transport Applications, pp. 7/1–7/5, 1990.

[15] K. Yamaguchi, Y. Nagaya, K. Ueda, H. Nemoto, and M. Nakagawa, “A method for identifying specific vehicles using template matching,” in Proc. IEEE Int. Conf. Intelligent Transportation Systems, 1999, pp. 8–13.

[16] J. R. Cowell, “Syntactic pattern recognizer for vehicle identification numbers”, Image and Vision Comput., vol. 13, no. 1, pp. 13–19, 1995.

[17] R. A. Lotufo, A. D. Morgan, and A. S. Johnson, “Automatic number plate recognition,” Inst. Elect. Eng. Colloquium on Image Analysis for Transport Applications, pp. 6/1–6/6, 1990.

[18] D. Yan, M. Hongqing, L. Jilin, and L. Langang, “

[19] A high performance license plate recognition system based on the web technique”, Proceedings of the IEEE Intelligent Transport Systems, pp. 325 - 329, 2001.

[20] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, S. Yamamoto, “License plate recognition method for inclined plates outdoors”, Proceedings International Conference on Information Intelligence and Systems, pp. 304-312, 1999.

[21] P. Cornelli, P. Ferragina, M. N. Granieri, and F. Stabile, Optical recognition of motor vehicle license plates, IEEE Transactions on Vehicular Technology, vol. 44, No. 4, pp: 790-799, 1995.

[22] D. S. Kim and S. I. Chien, "Automatic car license plate extraction using modified generalized symmetry transform and image warping," in Proc. IEEE Int. Symp. Industrial Electronics, vol. 3, 2001, pp. 2022–2027.

[23] J. A. G. Nijhuis, M. H. T. Brugge, K. A. Helmholt, J. P. W. Pluim, L. Spaanenburg, R. S. Venema, and M. A. Westenberg, "Car license plate recognition with neural networks and fuzzy logic," in Proc. IEEE Int.Conf. Neural Networks, vol. 5, 1995, pp. 2232–2236.

[24] M. Yu and Y. D. Kim, "An approach to Korean license plate recognition based on vertical edge matching," in Proc. IEEE Int. Conf. Systems, Man, and Cybernetics, 2000, pp. 2975–2980.

[25] H. Hontani and T. Koga, "Character extraction method without prior knowledge on size and position information," in Proc. IEEE Int. Conf. Vehicle Electronics, 2001, pp. 67–72.

[26] X. F. Hermida, F. M. Rodriguez, J. L. F. Lijo, F. P. Sande, and M. P. Iglesias, "A system for the automatic and real time recognition of VLP's (Vehicle License Plate)," in Proc. Lecture Notes in Computer Science, 1997, vol. 1311, pp. 552–558.

[27] J. C. H. Poon, M. Ghadiali, G. M. T. Mao, and L. M. Sheung, "A robust vision system for vehicle license plate recognition using grey-scale morphology," in Proc. IEEE Int. Symp. Industrial Electronics, vol. 1, 1995, pp. 394–399.

[28] Y. S. Soh, B. T. Chun, and H. S. Yoon, "Design of real time vehicle identification system," in Proc. IEEE Int. Conf. System, Man, and Cybernetics: Humans, Information, and Technology, vol. 3, 1994, pp. 2147–2152.

[29] K. K. Kim, K. I. Kim, J.B. Kim, and H. J. Kim, Learning based approach for license plate recognition, Proceedings of IEEE Processing Society Workshop on Neural Networks for Signal Processing, vol. 2, pp: 614-623, 2000

[30] E. R. Lee, P. K. Kim, and H. J. Kim, Automatic recognition of a car license plate using color image processing, IEEE International Conference on Image Processing 1994, vol. 2, pp.301-305, 1994.

[31] S. H. Park, K. I. Kim, K. Jung, and H. J. Kim, Locating car license plates using neural network, IEE Electronics Letters, vol.35, no. 17, pp. 1475-1477, 1999.

[32] D. U. Cho, and Y. H. Cho, "Implementation of pre-processing independent of environment and recognition and template matching", The Journal of The Korean Institute of Communication Sciences, Vol. 23, no. 1, pp. 94-100, 1998.

[33] G. M. Kim, "The automatic recognition of the plate of vehicle using the correlation coefficient and Hough transform", Journal of Control, Automation and System Engineering, vol. 3, no.5, pp. 511-519, 1997.

[34] H. A. Hegt, R. J. De la Haye, and N. A. Khan, "A high performance license plate recognition system," in Proc. IEEE Int. Conf. System, Man, and Cybernetics, vol. 5, 1998, pp. 4357–4362.

[35] Pappas, T., N., "An Adaptive Clustering Algorithm for Image Segmentation", IEEE Trans, Signal Processing, Vol. 40, Issue: 4, April 1992, pp 901 – 914.

[36] Hsieh, J., W., Yu, H., S., Chen, Y., S., "Morphology Based License Plate Detection from Complex Scenes", Pattern Recognition 2002 Proceedings, Vol. 3, 11-15 Aug. 2002, pp 176-179.

[37] Cristian, M., Rosa, A., “Gray-Scale Image Enhancement as an Automatic Process Driven by Evolution”, IEEE Trans, Systems, Man and Cybernetics-Part B: Cybernetics, Vol. 34, No. 2, April 2004.

[38] Gonzalez, R., C., Wood, R., E., “Digital Image Processing", Addison Wesley, 1993.

[39] J. Cowell, and F, Hussain, A fast recognition system for isolated arabic characters, Proceedings Sixth International Conference on Information and Visualisation, IEEE Computer Society, London, England, pp. 650-654, July 2002

[40] J. Cowell, and F. Hussain, “Extracting features from arabic characters, Proceedings of the IASTED International Conference on COMPUTER GRAPHICS AND IMAGING”, Honolulu, Hawaii, USA, pp. 201-206, August 2001.

[41] H. Hasen, A. W. Kristensen, M. P. Kohler, A. W. Mikkelsen, J. M. Pedersen and M. Trangeled, Automatic recognition of license plates, Institute for Electronic System, Aalborg University, May 2002.

[42] W. Mei; Y. Li; M. Wang; and Z. Huang, “Research on number-plate recognition based on neural networks”, Proceedings of the 2001 IEE Processing Society Workshop on Neural Networks for Signal Processing XI, . pp. 529-538, 2001.

[43] B. C. Zidouri, S. Chinveeraphan, and M. Sato, Recognition of machine printed arabic characters and numerals based on minimum covering run, IEICE Trans. Inf. & syst., vol. E78-D, pp. 1649-1655, 1995

[44] A., S., Johnson, B. M. Bird, “Number plate matching for automatic vehicle identification, IEE Colloquium on Electronic Image and Image Processing in Security and Forensic”, 1990.

[45] M. M. M. Fahmy, Automatic number plate recognition: Neural Network Approach, Proceedings of VNIS’94 Vehicle Navigation and Information System Conference, 1994.

[46] H. J. Choi, A study on the extraction and recognition of a car number plate by image processing, Journal of the Korean Institute of Telematics and Electronics, vol. 24, pp. 309-315, 1987.

[47] S. K. Kim, D. W. Kim, and H. J. Kim, A recognition of vehicle license plate using a genetic algorithm based segmentation, Proceedings of 3rd IEEE International Conference on Image Processing, vol. 2, pp. 661-664, 1996.

[48] Coetzee, C., Botha, C., Weber, D.,” PC based number plate recognition system” ,IEEE International Symposium on Industrial Electronics, Volume 2, 7-10 July 1998, Pages 605-610, vol.2

[49] S. Draghici, “A neural network based artificial vision system for license plate recognition system ”, International Journal for Neural Systems, Vol. 8, pages 112-113, February 1997.

[50] M. A. Arregui, J. A. Mitxelena, “Automatic Recognition of Number Plates”, selected paper from the IVth Spanish Symposium, vol. 1, pages 239-258, 1992

[51] T. R. Cimmins, W. M. Brown, “Image Algebra and Automatic Shape Recognition”, IEEE Transaction Aerospace and Electronic Systems, vol AES-21 No. 1, January 1985.

[52] Bailey, D.G., Irecki, D., Lim, B.K., Yang, L.," Test bed for number plate recognition applications", The First IEEE International Workshop on Electronic Design, 29-31 Jan. 2002, Page(s): 501 -503.

[53] http://www.dpl-surveillance-equipment.com

[54] http://www.singaporegateway.com

[55] http://www.perceptics.com/license-plate-reader.html

[56] http://www.intertraffic.com

[57] A. Koschan, "A Comparative Study On Color Edge Detection," in Proc. 2nd Asian Conference on Computer Vision ACCV'95, Singapore, Vol. III, pp. 574-578, December 1995.

[58] Dony, R., D., Wesolkowski, S., "Edge Detection on Color Images using RGB Vector Angles", Proceedings of, IEEE Canadian Conf on Electrical & Computer Engineering, Vol. 2, 9-12 May, 1999, pp 687-692.

[59] Wesolkowski, S., Jernigan, E., "Color Edge Detection using jointly Euclidean and Vector Angle", Vision Interface, 99, 19-21 May, 1999 Watson, G., S., "Statistics on Spheres", Wiley Interscience, 1983.

[60] P. Trahanias and A.N. Venetsanopoulos, "Color edge detection using vector statistics", IEEE Trans. on Image Processing, vol. 2, pp. 259-264, April 1993.

[61] Y. Yang, Colour edge detection and segmentation using vector analysis, Master's thesis, Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, 1995.

[62] R., C., Gonzalez and R., E., Woods, Digital Image Processing, Addision Wesley, 1993.

[63] H., R., Tizhoosh, Fuzzy Image Processing, Springer, 1997. “http://pami.uwaterloo.ca/tizhoosh/wfip.htm”

[64] Bezdec, J.C., Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.

[65] R., O., Duda, P., E., Hart, D., G., Stork, “Pattern Classification”, John Wiley & Sons, Inc.

[66] http://www.singaporegateway.com/optasia/config.htm

[67] http://www.avsupply.com/details/reg-license.html

[68] http://www.htsol.com/Products/SeeCar.html

[69] http://www.perceptics.com

[70] http://www.pearpoint.com/products.htm

[71] J.S.Weszka and A.Rosenfeld, Histogram modification for threshold selection, IEEE Trans. SMC-9 (1979).

[72] D. Gabor, "Theory of Communication," J. Institute of Electrical Engineers, vol. 93, no. 26, Nov. 1946, pp. 429-457.

[73] D. Daugman, "High-Confidence Visual Recognition of Persons by a Test of Statistical Independence," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 15, no. 11, Nov. 1993, pp. 1148-1160.

[74] Source: AIM Japan, Automatic Identification Seminar, Sept.14, 2001

[75] H. Wechsler et al., Face Recognition: From Theory to Application, Springer-Verlag, Berlin, 1998.

[76] B. Schneier, "Cryptographic Design Vulnerabilities," Computer, vol. 31, no. 9, Sept. 1998, pp. 29-33.

VITAE

- Syed Adnan Yusuf.
- Born in Karachi, Pakistan on October 06, 1977.
- Received Bachelor of Engineering degree in Computer Systems Engineering from NED University of Engineering and Technology, Karachi, Pakistan in 2000.
- Worked as a Software developer/Web Engineer in Breezesoft (PVT) Ltd.
- Software QA Engineer in a CDC of Pakistan (Ltd).
- Joined KFUPM in January 2002.
- **Publications:**
- ***"Color Edge Enhancement based Fuzzy Segmentation of License Plates"***, The Proceedings of IEEE International Conference on Information Visualization (IV'2005)-UK, IEEE Computer Society Press, USA, Accepted.(Held: 6-7th July, 2005)
- ***"A System for Sign Language Recognition using Fuzzy Object Tracking"***, The Proceedings of IEEE International Conference on Information Visualization (IV'2005)-UK, IEEE Computer Society Press, USA, Accepted.(Held: 6-7th July, 2005)

- "***Segmentation of License Plates using Fuzzy Clustering",*** 11th World Congress of ***International*** Fuzzy Systems Association (IFSA 2005), July 28-31, 2005, Beijing, China. Accepted.
- **"Identification of License Plates under extreme outdoor conditions"**, International Journal of Information Technology and Web Engineering (IJITAW), http://www.idea-group.com/journals/details.asp?id=4768, Idea Group, Inc.