

**IRIS RECOGNITION USING IMAGE PROCESSING
AND NEURAL NETWORK**

**A THESIS SUBMITTED TO THE GRADUATE
SCHOOL OF APPLIED SCIENCES OF
NEAR EAST UNIVERSITY**

BY

MUHAMMAD WAHEED

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING**

NICOSIA, 2014

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Muhammad Waheed

Signature:

Date:

ACKNOWLEDGMENTS

First of all I would like to extend our sincere and humble gratitude to Almighty Allah whose blessings and guidance has been a real source of all my achievements in my life. I feel a deep sense of obligation and greatly thank to my project supervisor Assist.Prof.Dr. Boran Şekeroğlu who is always there for me help and without their help and encouragement, it was impossible for me to achieve this task. I feel greatly indebted for his sincere guidance. I will be failing in my duties if I don't express my warmest thank to Prof. Dr. Adil Amirjanov for his guidance and encouragement, for his guidance and valuable suggestions in reading materials and other difficulties of the project. All the year he helps me in study problem.

I would like to thank my friends in Near East University Master Program (Rashid Khan, wameedh Raad Fathel, Rami R. Mustafa, Awad Jihad, Sipan Slivany). It is my bounden duty to pay tributes to my worthy all staff members of Department of Computer Engineering.

My parents, Thank you for your unconditional support with my studies I am honoured to have you as my parents. Thank you for given me a chance to prove and improve myself through all my walks of life. Please do not ever change. I love you.

My brothers and sisters, hoping that with this research I have proven to you that there is no mountain higher as long as God is on our side. Hoping that, you will walk again and be able to fulfil your dreams.

ABSTRACT

This research is aimed to design an iris recognition system. There are two main steps to verify the goal. First, applying image processing techniques on the picture of an eye for data preparation. Second, applying Neural Networks techniques for identification.

The image processing techniques display the steps for getting a very clear iris image necessary for extracting data from the acquisition of eye image -in standard lighting and focusing-. This picture contains the entire eye (iris, pupil and lashes). So, the localization of the iris is very important. Locating the Iris is done by following the darkness density of the pupil. The new picture has iris with pupil -in its narrow size- and this picture is not perfectly clear. Therefore, it should be enhanced to bring out the pattern. The enhanced picture is segmented into 100 parts, then a standard Deviation (STD) can easily be computed for every part. These values will be used in the neural network for the identification.

Neural network techniques explain two ways for comparisons, Linear Associative Memory Neural Network and Back Propagation Neural Network. The Back Propagation Neural Network succeeded in identification and getting best results because it attained to Recognition Rate equal to 90%, while the Linear Associative Memory Network attained to Recognition Rate equal to 80%.

Keywords: Back Propagation Neural Network, Linear Associative Memory Neural Network
Artificial neural network, Near infrared, Standard deviation

ÖZET

Bu araştırma, iris tanıma sisteminin dizayn edilmesini amaçlamaktadır. Hedefin doğruluğunu kanıtlamak için iki adım vardır. Birincisi, veri hazırlanması için göz görüntüsü üzerine görüntü işleme tekniklerinin uygulanmasıdır. İkincisi ise, tanımlama için Sinir Ağları tekniklerinin uygulanmasıdır.

Görüntü işleme teknikleri, göz görüntüsünün –standard ışıklandırma ve odaklanmada - alınmasından veri elde etmek için gerekli olan çok net bir iris görüntüsü almaya yönelik adımları göstermektedir. Bu görüntü, tüm gözü (iris, gözbebeği ve kirpikler) içermektedir. Böylece, irisin yerinin belirlenmesi çok önemlidir. İris yerinin belirlenmesi, gözbebeğin siyahlık yoğunluğunun izlenmesiyle yapılmaktadır. Yeni görüntünün gözbebeği ile irisi vardır-dar ebatı içerisinde-ve ayrıca bu resim tam net değildir. Dolayısıyla, şekil düzeninin ortaya çıkarılması için görüntünün büyütülmesi gerekmektedir. Büyütülmüş görüntü, 100 parçaya bölünmekte ve daha sonra her parça için bir standart sapma (SP) kolaylıkla hesaplanabilmektedir. Bu değerler, tanımlama yapmada sinir ağı içerisinde kullanılacaktır.

Sinir ağı teknikleri, karşılaştırmalar için, Lineer İlişkisel Bellek Sinir Ağı ve Geri Yayınımli Sinir Ağı olmak üzere iki usul açıklamaktadır. Geri Yayınımli Bellek Sinir Ağı, tanımlamayı ve en iyi sonuçları almayı başarmıştır, çünkü, Lineer İlişkisel Bellek Ağı %80’lik bir tanıma oranına erişirken Geri Yayınımli Sinir Ağı %90’lık bir tanıma oranına erişmektedir.

Anahtar Kelimeler: Geri Yayınımli Sinir Ağı, Lineer İlişkisel Bellek Sinir Ağı, Yapay Sinir Ağı, Yakın Kızılötesi, Standard Sapma

CONTENTS

ACKNOWLEDGEMENTS	II
ABSTRACT	IV
OZET	V
TABLE OF CONTENT	VI
LIST OF TALBES	IX
LIST OF FIGURES	X
LIST OF ABBREVIATIONS	XIII
CHAPTER 1: INTRODUCTION	1
CHAPTER 2: BIOMETRIC SYSTEM	3
2.1 Overview	3
2.2 Biometric system	3
2.3 Biometrical classification	5
2.3.1 Finger print recognition	5
2.3.2 Facial recognition	5
2.3.3 Voice recognition	6
2.3.4 Retina	6
2.3.5 Hand geometry	7
2.3.6 Signature dynamics	7
2.4 Iris recognition	8
2.4.1 Features of iris recognition	8
2.5 Summary	10
CHAPTER 3: THEORETICAL PRINCIPLE OF IMAGE PROCESSING	11
3.1 Overview	11

3.2 Image processing.....	11
3.2.1 Cameras and eyes.....	11
3.2.2 Filter applications.....	13
3.2.3 Morphological filtering	20
3.2.4 Histogram equalization.....	23
3.3 Summary.....	24
CHAPTER 4: THEORETICAL PRINCIPLES OF NEURAL NETWORK.....	25
4.1 Overview	25
4.2 Principle of ANN.....	25
4.3 Activation function.....	26
4.4 LAM Neural network	28
4.4.1 History of LAM	28
4.4.2 Principles of LAM.....	28
4.5 BPN.....	30
4.5.1 History of BPN	30
4.5.2 Architecture of standard BPN	31
4.5.3 Algorithm.....	32
4.5.4 Momentum Techniques.....	33
4.6 Wavelet	34
4.7 Summary.....	35
CHAPTER 5: SYSTEM DESIGN.....	36
5.1 Overview	36
5.2 System Design Requirement.....	36
5.3 Image processing	37
5.3.1 Image Acquisition	39
5.3.2 Iris localization.....	41

5.3.3 Image enhancement	41
5.3.4 Image segmentation	43
5.4 Neural network	43
5.4.1 LAM neural network suggested topology	44
5.4.2 BPN suggested topology	45
5.5 Summary	46
CHAPTER 6: EXPERIMENTAL RESULT AND COMPARISON	47
6.1 Overview	47
6.2 System features	47
6.3 Image processing result	48
6.4 Neural network result	55
6.5 Results	58
6.6 Comparing with recent researches	59
CHAPTER 7: CONCLUSIONS AND FUTURE WORK	62
7.1 Conclusion	62
7.2 Future work	63
REFERENCES	64
APPENDICES	68
A. Source code	68
B. Screen shots	75

LIST OF TABLES

Table 2.1: Comparison of biometric systems.....	10
Table 3.1: The products of a vertical vector and a horizontal vector	14
Table 3.2: Shows the median filter	19
Table 6.1: Comparisons between the two neural networks methods.....	57
Table 6.2: Recognition rates of the system for IRIS tested images.....	58
Table 6.3: Accuracy rate for iris segmentation.....	59
Table 6.4: The recognition performance of comparing with existing methods.....	60
Table 6.5: Recognition performance with Gabor Wavelet research	61
Table 6.6: Final parameter of back propagation memory neural network	61

LIST OF FIGURES

Figure 2.1: Examples of biometric characteristics that are commonly used.....	4
Figure 2.2: Sample of fingerprint picture.....	5
Figure 2.3: Sample of facial picture with anterior landmarks.....	5
Figure 2.4: Sample of retina picture.....	6
Figure 2.5: Sample of hand picture.....	7
Figure 2.6: Sample of signature picture.....	7
Figure 2.7: Sample of iris picture showing the patterns and the pupil.....	8
Figure 2.8: Iris features	8
Figure 3.1: The block diagram of image analysis process	11
Figure 3.2: Major structures in an electronic camera and the human eye.....	12
Figure 3.3: Sobel filter masks	13
Figure 3.4: Example of Sobel filter edge detection	13
Figure 3.5: Example of Gaussian blur filter effects	15
Figure 3.6: Example of edge detection results.....	16
Figure 3.7: Example of Unsharp filter results	17
Figure 3.8: Example of Average filter result	18
Figure 3.9: Example of Median filter result.....	19
Figure 3.10: The line shape structuring element	20
Figure 3.11: The square shape structuring element.....	20
Figure 3.12: The diamond shape structuring element.....	20
Figure 3.13: The rectangle shape structuring element	21
Figure 3.14: The disk shape structuring element	21
Figure 3.15: The octagon shape structuring element.....	21
Figure 3.16: Examples of different morphological operations results	22

Figure 3.17: Histogram equalization filter	23
Figure 3.18: A histogram transformation function	24
Figure 4.1: Artificial neuron model	25
Figure 4.2: Log-sigmoid activation function	26
Figure 4.3: Tan-sigmoid activation function	27
Figure 4.4: Linear activation function	27
Figure 4.5: A supervised hebbian network for learning pattern association	29
Figure 4.6: BPN with one hidden layer	32
Figure 5.1: On-line iris recognition system model	37
Figure 5.2: The steps adopted for eye image analysis	38
Figure 5.3: Alumina box with a stand carrier	39
Figure 5.4: Windows movie maker software options	40
Figure 5.5: Natural eye image needed to be pre-processed	40
Figure 5.6: Average filter iris image	41
Figure 5.7: Contrast enhancement texture iris image	42
Figure 5.8: Histogram equalization image	42
Figure 5.9: Structure of LAM network	44
Figure 5.10: BPN suggested design	45
Figure 6.1: The hardware requirements for iris recognition system	47
Figure 6.2: Flowchart for iris training and recognition system program	49
Figure 6.3: Trained images	50
Figure 6.4: Tested images	50
Figure 6.5: Results with successful image	50
Figure 6.6: Common types of low pass filter with their frequency responses	52
Figure 6.7: Example of histogram for green part of eye image	53
Figure 6.8: Example of histogram for final analysis iris image	53

Figure 6.9: STD curves of two iris segmentation image for same eye	54
Figure 6.10: STD curves of right and left iris for the same person	54
Figure 6.11: LAM network architecture	55
Figure 6.12: BPNN architecture	56

ABBREVIATIONS

LAM	Liner Associative Memory
2D	Two Dimensional
1D	One Dimensional
IMG	Image
DNA	Deoxyribonucleic Acid
IP	Image Processing
CCD	Charged Couple Device
PSF	Point Spread Function
LOG	Lalacian of Gaussian
LEN	Length
IS	Image Stabilizer (camera lens)
IQ	Image Quality
DEG	Degree
ANN	Artificial Neural Network
BPN	Backprobagation Neural Network
3D	Three Dimensional
NIR	Near Infrared
RGB	Red Green Blue
STD	Standard Deviation
DI	Diagnostic Imaging
FAR	False Acceptation Rate
FRR	False Rejection Rate
DICOM	Digital Imaging and Communications in Medicine
SSI	Solid State Imaging
DMG	Disk Image
TIF	Tagged Image File (file name extension)
FMRI	Functional Magnetic Resonance Imaging
ICIP	International Conference on Image Processing
CIS	Contact Image Sensor
FITS	Flexible Image Transport System
SI	Still Image
MIL	Matrix Imaging Library

INTRODUCTION

Since the last century, several biometric techniques were used for identification of humans. These techniques are Iris recognition, Face recognition, Fingerprint recognition, Voice recognition, etc. Each of these techniques has number of real life applications [1].

Iris Recognition refers to the automated method of verifying a match between two irises of human. Irises are one of many forms of biometrics used to identify individuals and verify their identity [1].

The aim of this thesis is design iris recognition system using Linear Associative Memory and Back propagation Neural Network. Iris recognition system is divided into two main stages. The first one is used to extract the features from the iris image, and the second stage is used for classification of patterns. Feature extracting is a very important step in iris recognition system. This thesis touches on two major classes of algorithms used for extraction of the features of face images. The recognition rate of the system depends on the meaningful data that are extracted from the iris image. So, important feature should be extracted from the images. If the features belong to the different classes and the distance between these classes are big then these features are important for given image. The flexibility of the class is also important. There can never be exact match between the images of the same iris even if they were from the same person.

There are different methods designed for image feature extraction. These methods involve different classification like Holistic and local feature-based techniques. In Holistic method the whole iris image is applied as input to recognition operation very similar to the well-known PCA based method. In second method local features are extracted such as location and local statistics of eyes for iris detection. Iris recognition techniques are considered which involves Linear Associative Memory and Back Propagation Neural Network.

The general purpose is the high confidence recognition of an individual's identity by a mathematical analysis of the random patterns that are scanned for the iris of an eye [12]. Reliable automatic recognition of persons has long been an attractive goal [9].

The goal is to achieved through following steps. First one is by applying image processing techniques for clarifying iris image necessary for extracting data by using iris acquisition, iris localization, iris enhancement and iris segmentation. The second step involves Neural

Network techniques for comparison and identification, which focuses on Linear Associative Memory (LAM) and Back Propagation Neural Network (BPNN).

The thesis is organized along seven chapters. Chapter One covers the introduction. Chapter two describes the different biometric systems especially iris recognition system. Chapter Three and Chapter four gives a theoretical basic for image processing and artificial neural networks. Chapter Five describes the image processing steps and neural network suggested topology methods. Chapter Six includes discussion and comparison of the results introduced. Finally, the conclusions and some suggestions for future works are given in Chapter Seven.

CHAPTER 2

BIOMETRIC AUTHENTICATION SYSTEMS

2.1 Overview

Biometrics is a methodology for recognizing and identifying persons based on individual and distinct physiological or behavioral characteristics. A good biometric has two basic characteristics: Stability and Distinctiveness. A stable biometric doesn't change over time. "Clearly, hair length would not be a good biometric identifier". A distinctive biometric is unique to an individual [1].

Traditional methods for personal identification are based on what a person possesses (a physical key, ID card, etc.) or what a person knows (a secret password, etc.). These methods have some problems. Keys may be lost, ID cards may be forged, and passwords may be forgotten.

In recent years, biometric personal identification grows as an interesting field from industrial and academic point of view [2].

So, biometric methods are security technologies which use human characteristics for personal identification. Iris recognition systems use iris textures as unique identifiers [3].

2.2 Biometric Systems

Biometric technologies can be divided into two major categories according to what they measure:

- Devices based on physiological characteristics of a person (such as the fingerprint or hand geometry).
- Systems based on behavioral characteristics of a person (such as signature dynamics).

[4]

Today, biometric recognition is a common and reliable way to authenticate the identity of a living person based on physiological or behavioral characteristics.

Physiological characteristics are relatively stable physical characteristics, such as fingerprint, iris pattern, facial feature, hand silhouette, etc. This kind of measurement is basically not being changed without significant duress. A behavioral characteristic is a reflection of an individual's psychological makeup as signature, speech pattern, or how one types at a keyboard.

The degree of intra-personal variation in a physical characteristic is smaller than a behavioral characteristic.

For examples, a signature is influenced by both controllable actions and less psychological factors, while speech pattern on the other hand is influenced by current emotional state, as for fingerprint template, is independent [5].

In general physiological biometrics systems are more accurate and have a higher cost than behavioral biometrics systems as shown in Figure 2.1 [1]

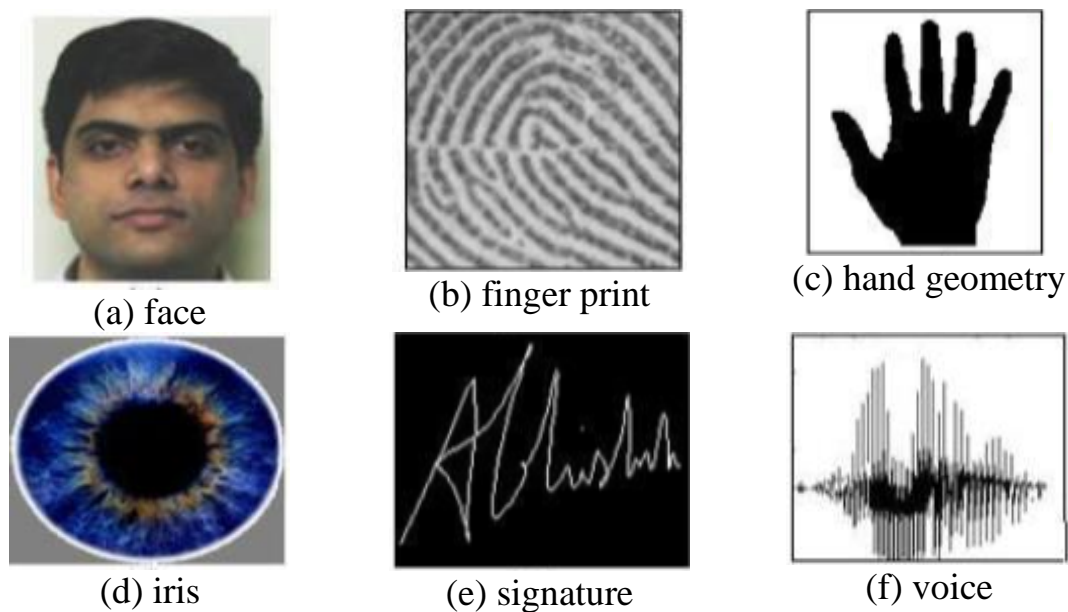


Figure 2.1: Examples of biometric characteristics that are commonly used [1]

2.3 Biometric Classification

2.3.1 Fingerprint Recognition

Fingerprint recognition technology is probably the most widely used and well-known biometric. Fingerprint recognition relies on features found in the impressions made by distinct ridges on the fingertips. There are two types of fingerprints: flat or rolled. Flat prints are an impression of only the central area of the finger pad, while rolled prints capture ridges on the sides of the finger as well as the central portion between the tip and first knuckle. Fingerprint images are scanned, enhanced and then converted into templates. These templates are saved in a database for future comparisons using optical, silicon, or ultrasound scanners. Ultrasound appears to be the most accurate one, but is rarely used; optical scanners are most commonly used See Figure 2.2 [6]



Figure 2.2: Sample of fingerprint picture [6]

2.3.2 Facial Recognition

Face recognition technology identifies individuals by analyzing certain facial features such as the upper outlines of the eye sockets or sides of the mouth. Typically, facial recognition compares a living person with a stored template, but it has also been used for comparison between static images and templates. This technology is a biometric system that can routinely be used in a secure manner, for surveillance, since a person's face is easily captured by video technology [6]. Figure 2.3 shows the set of landmarks for anterior facial recognition used in analysis:

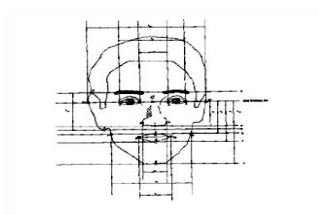


Figure 2.3: Sample of facial picture with anterior landmarks [6]

2.3.3 Voice Recognition

Voice recognition technology for identifying people is based on the differences in the voice resulting from learned speaking habits. When an individual is enrolled, the system captures samples of the person's speech as the individual says certain scripted information into a microphone or telephone multiple times. This information is known as a "pass phrase." (There are also some biometric systems available that can distinguish between people's voices without requiring a predefined phrase). The pass phrase is then converted into a digital format and distinctive characteristics (e.g., pitch, cadence, tone) are extracted to create a template for the speaker. Voice recognition templates require the most data space of all the biometric templates [6].

However, voice recognition is a biometric technique based on behavioral characteristic which can be negatively affected by the current physical condition and the emotional state. The accuracy of the Voice recognition can also be affected by the background and system noise in the input signal. This increases the false rejection rate [4].

2.3.4 Retina

Retina scan is based on the blood vessel pattern in the retina of the eye. Retina scan technology is older than the iris scan technology that also uses a part of the eye. The first retinal scanning systems was launched by EyeDentify in 1985. The main drawback of the retina scan is its intrusiveness. The method of obtaining a retina scan is personally invasive. A laser light must be directed through the cornea of the eye. Also the operation of the retina scanner is not easy [4]. See Figure. 2.4 below

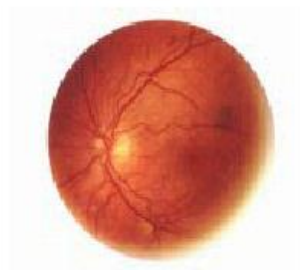


Figure 2.4: Sample of retina picture [4]

2.3.5 Hand geometry

Hand geometry is based on the fact that nearly every person's hand is shaped differently and that the shape of a person's hand does not change after a certain age. Hand geometry systems produce estimates of certain measurements of the hand such as the length and the width of fingers. Various methods are used to measure the hand. The optical method is much more common today. Optical hand geometry scanners capture the image of the hand and use the image edge detection algorithm compute the hand's characteristics. Only Two-Dimensional (2D) characteristics of the hand can be used in this case [4]. See Figure. 2.5



Figure 2.5: Sample of hand picture [4]

2.3.6 Signature dynamics

The signature dynamics recognition is based on the dynamics of making the signature, rather than a direct comparison of the signature itself afterwards. The dynamics is measured as a means of the pressure, direction, acceleration and the length of the strokes, number of strokes and their duration. Pioneers of the signature verification first developed a reliable statistical method in 1970s. This involved the extraction of ten or more writing characteristics such as the number of times the pen was lifted, the total writing time and the timing of turning points [4]. See Figure. 2.6



Figure 2.6: Sample of signature picture [4]

2.4 Iris Recognition

2.4.1 Features of Iris Recognition

Iris is a coloured circular muscle, which is beautifully pigmented giving us our eye's colour (the central aperture of the iris is the pupil). This circular muscle controls the size of the pupil so that more or less light, depending on illumination conditions, is allowed to enter the eye [1], As shown below in Figure. 2.7

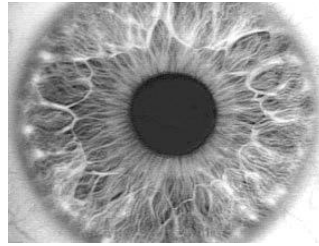


Figure 2.7: Sample of iris picture showing the patterns and the pupil [1]

It consists of lines, dots, rings, rifts, pits, crypts, freckles, striations, contraction furrows, coronas and serpentine vasculature [1,7,8].

Scientists have identified 250 features unique to each person's iris – compared with about 40 for fingerprints – and it remains constant through a person's life, unlike a voice or a face, fingerprint and hand patterns which can be changed through alteration or injury.

So, iris identification is more accurate than other high-techniques identification systems available that scans voices, faces, and fingerprints [1]. See Figure. 2.8

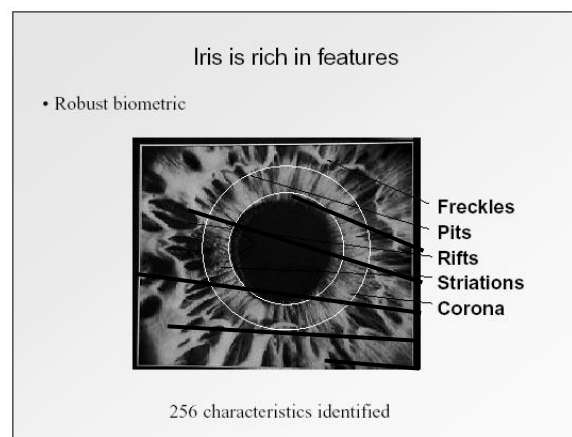


Figure 2.8: Iris features [1]

The iris begins to form in the third month of gestation and the structures creating its pattern are largely complete by the eighth month, although pigment accretion can continue into the first postnatal years [9-12]. In addition, the iris is well protected from the environment.

It is the most individually distinctive feature of the human body. No two irises are alike, not even among twins [1,13]. In fact, left and right irises of one individual are not identical. The statistical probability that two irises are never be exactly the same is estimated at 1 in 10^{72} ; Iris recognition is statistically more accurate than DNA testing [7].

For all of these reasons, iris patterns become interesting as an alternative approach to reliable visual recognition of persons when imaging can be done at distances of less than a meter, and especially when there is a need to search very large databases without incurring any false matches despite a huge number of possibilities.

Although small (11mm) and sometimes problematic to image, the iris has the great mathematical advantage that its pattern variability among different persons is enormous. Iris colour is determined mainly by the density of melanin pigment in its anterior layer and stroma, with blue irises resulting from an absence of pigment long-wavelength light penetrates while shorter wavelengths are scattered by the stroma [9].

However, the iris pattern is so highly detailed in that the identification procedure become more difficult.

Table 2.1: shows the comparison of the main biometrics [14]

Characteristic	Fingerprints	Hand Geometry	Retina	Iris	Face	Signature	Voice
Ease of Use	High	High	Low	Medium	Medium	High	High
Error incidence	Dryness, dirt, age	Hand injury, age	Glasses	Poor Lighting	Lighting, age, glasses, hair	Changing signatures	Noise, colds, weather
Accuracy	High	High	Very High	Very High	High	High	High
User acceptance	Medium	Medium	Medium	Medium	Medium	Medium	High
Required security level	High	Medium	High	Very High	Medium	Medium	Medium
Long-term stable	High	Medium	High	High	Medium	Medium	Medium

2.5 Summary

The feature of vital physiological characteristic of every human, such as fingerprint and eye, face, hand, voice and signature have achieved significant improvement in personal identification. The extracted feature of this biometrics have made a significant achievement in reducing many of the problems and weaknesses that faced with the traditional way of verifying the identity (persons) using passwords. Despite the degree of high security achieved by these techniques the accuracy recognition systems did not reach to 100% yet. For this reason the design of new techniques for feature extraction and image recognition become important computer science.

CHAPTER 3

THEORETICAL PRINCIPLES OF IMAGE PROCESSING

3.1 Overview

In this chapter, an introduction to image processing, basic principles, filtering, morphology and enhancement is presented.

3.2 Image processing

Image processing refers to the various operations performed on images that are digitally stored as an aggregate of pixels in order to obtain new image with required specifications. There are many subjects in image processing, including artistically transforming the image, finding and recognizing objects in an image and enhancing or degrading the quality of the image [18].

Images have their information encoded in the spatial domain. In other words, features in images are represented by edges, not sinusoids. This means that the spacing and number of pixels are determined by how small are the features needed to be seen, rather than by the formal constraints of the sampling theorem [19].

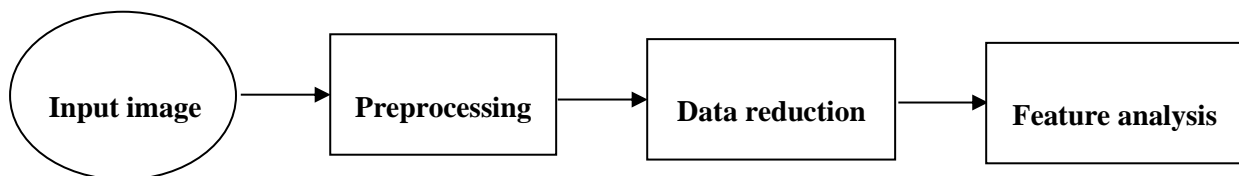


Figure 3.1: The block diagram of image analysis process [19]

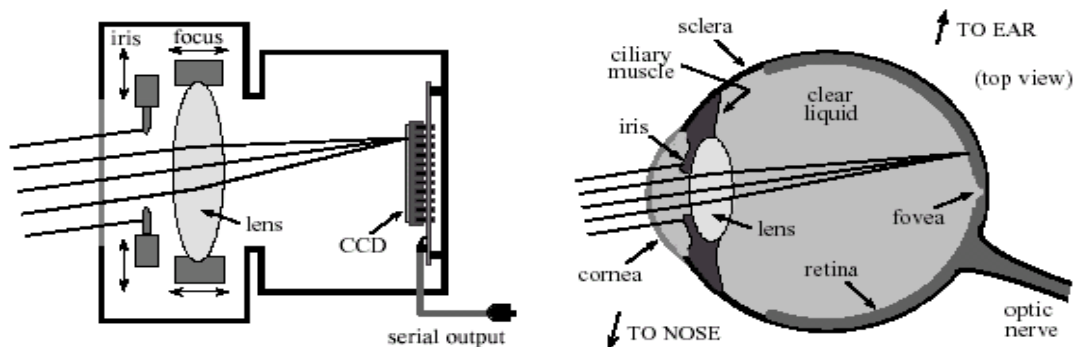
3.2.1 Cameras and Eyes

The structure and operation of the eye is very similar to an electronic camera, and it is natural to discuss them together. Both are based on two major components, a lens assembly and an imaging sensor.

The lens assembly captures a portion of the light emanating from an object and focuses it onto the imaging sensor. The imaging sensor then transforms the pattern of light into a video signal, either electronic or neural.

Figure 3.2 illustrates the major structures in Charged Coupled Device (CCD) electronic camera and the human eye, respectively.

Both are light tight enclosures with a lens mounted at one end and an image sensor at the other. The camera is filled with air, while the eye is filled with a transparent liquid. Each lens system has two adjustable parameters: focus and iris diameter. If the lens is not properly focused, each point on the object will then project to a circular region on the imaging sensor, causing the image to be blurry. In the camera, focusing is achieved by physically moving the lens towards or away from the imaging sensor. In comparison, the eye contains two lenses, a bulge on the front of the eyeball called the cornea, and an adjustable lens inside the eye. The cornea does most of the light refraction, but is fixed in shape and location. Adjustment to the focusing is accomplished by the inner lens, a flexible structure that can be deformed by the action of the ciliary muscles. As these muscles contract, the lens will flatten to bring the object into a sharp focus. In both systems, the iris is used to control how much of the lens is exposed to light, and therefore the brightness of the image projected onto the imaging sensor. The iris of the eye is formed from opaque muscle tissue that can be contracted to make the pupil (the light opening) larger. The iris in a camera is a mechanical assembly that performs the same function [19].



(a) Diagram of an electronic CCD camera. (b) Diagram of the human eye

Figure 3.2: Major structures in an electronic camera and the human eye [19]

3.2.2 Filter applications

Filtering is a technique for modifying or enhancing an image that can be used to emphasize certain features or remove other features of an image. Filtering is a neighborhood operation in which the value of any given pixel in the output image is determined by applying some algorithm to the values of the pixels in the neighborhood of the corresponding input pixel, where pixel's neighborhood is some set of pixels defined by their locations relative to that pixel [20].

Below is an overview to some common filters:

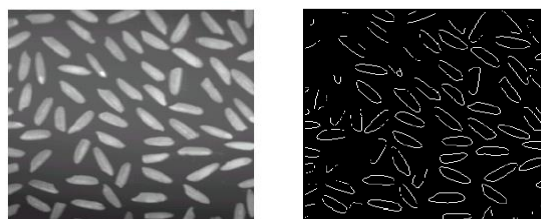
1- Sobel filter:

The Sobel filter masks are used for image edge detection; they look for edges in both the horizontal and vertical directions and then combine this piece of information into a single metric. The masks are shown below in Figure.3.3

ROW MASK			COLUMN MASK		
+1	+2	+1	-1	0	+1
0	0	0	-2	0	+2
-1	-2	-1	-1	0	+1

Figure 3.3: Sobel filter masks [19]

These masks are each convolved with the image [22]. Figure 3.4 gives on example of the Sobel filter edge detection:



(a) Original image (b) Image after Sobel filter

Figure 3.4: Example of Sobel filter edge detection [22]

The Gaussian blur filter has a smoothing effect on images. It is in fact a low pass filter. Apart from being circularly symmetric, edges and lines in various directions are treated similarly. The Gaussian blur filters have some very advantageous characteristics:

- They are separable into the product of horizontal and vertical vectors.
- Large kernels can be decomposed into the sequential application of small kernels.

So, the filter kernel of the Gaussian blur filter is decomposable in the product of a vertical vector and a horizontal vector; the possible vectors are shown in Pascal triangle Table 3.1.

When multiplied by each other, they will produce gaussian blur filters:

Table 3.1: The products of a vertical vector and a horizontal vector

Index N	Coefficients											Sum of coefficients = 2^N
0	1											1
1	1 1											2
2	1 2 1											4
3	1 3 3 1											8
4	1 4 6 4 1											16
5	1 5 10 10 5 1											32
6	1 6 15 20 15 6 1											64
7	1 7 21 35 35 21 7 1											128
8	1 8 28 56 70 56 28 8 1											256
9	1 9 36 84 126 126 84 36 9 1											512
10	1 10 45 120 210 252 210 120 45 10 1											1024
11	1 11 55 165 330 462 462 330 165 55 11 1											2048

Gaussian blur filter kernel example (order 3) [21]

$$\begin{pmatrix} 1 \\ 3 \\ 3 \\ 1 \end{pmatrix} \times \begin{pmatrix} 1 & 3 & 3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 3 & 3 & 1 \\ 3 & 6 & 6 & 3 \\ 3 & 6 & 6 & 3 \\ 1 & 3 & 3 & 1 \end{pmatrix} \quad (2.5)$$

An example of image effected by Gaussian blur filter is given in Figure. 3.5



(a) Original image

(b) Image after Gaussian filter

Figure 3.5: Example of Gaussian blur filter effects [21]

3- Laplacian of Gaussian (LOG):

The Laplacian of an image highlights regions of rapid intensity change and is therefore often used for edge detection. The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian Smoothing filter in order to reduce its sensitivity to noise.

The Laplacian $L(x,y)$ of an image with pixel intensity values $I(x,y)$ is given by:

$$\nabla^2 = L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (3.1)$$

$$\nabla^2 \approx \frac{4}{(\alpha+1)} \begin{bmatrix} \frac{\alpha}{4} & \frac{1-\alpha}{4} & \frac{\alpha}{4} \\ \frac{1-\alpha}{4} & -1 & \frac{1-\alpha}{4} \\ \frac{\alpha}{4} & \frac{1-\alpha}{4} & \frac{\alpha}{4} \end{bmatrix} \quad (3.2)$$

The 2D LoG function centered on zero and with Gaussian standard deviation σ has the form shown in equation (3.3):

$$LoG(x,y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (3.3)$$

Hence, the simple Laplacian can be seen as a limiting case of the LoG for narrow Gaussians (i.e. where: $\sigma < 0.5$ pixels) [23]. See Figure. 3.6 for LoG filter results:



Figure 3.6: Example of edge detection results [23]

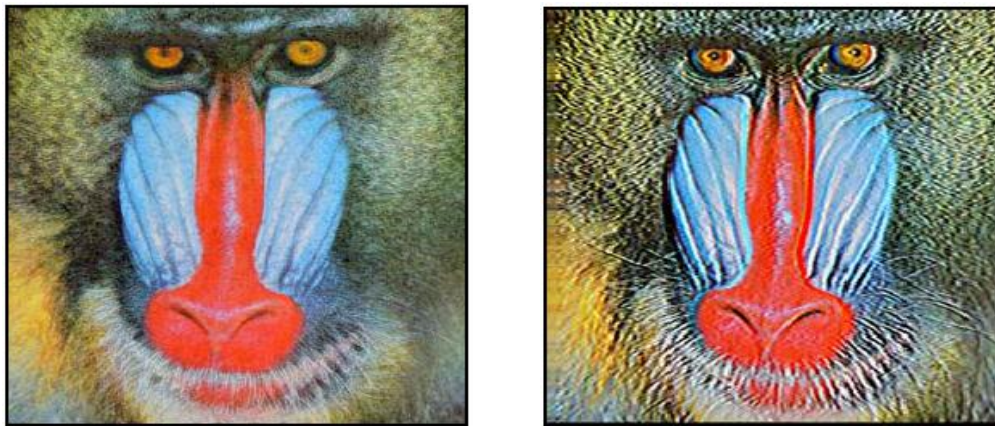
4- Unsharp filter:

The unsharp masking enhancement algorithm is a representative example of the practical image sharpening methods. Image sharpening deals with enhancing detail information in an image. The detailed information is typically contained in the high spatial frequency components of the image. The details of information include edges and, in general, correspond to image features that are spatially small. This piece of information is visually important because it delineates object and feature boundaries and is important for textures in objects [22].

An unsharp masking filter which is created from the negative of the Laplacian filter with parameter alpha is shown below in equation (3.4):

$$-\nabla^2 \approx \frac{1}{(\alpha+1)} \begin{bmatrix} -\alpha & \alpha-1 & -\alpha \\ \alpha-1 & \alpha+5 & \alpha-1 \\ -\alpha & \alpha-1 & -\alpha \end{bmatrix} \quad (3.4)$$

Alpha must be in the range of 0.0 to 1.0 [20]. See Figure. 3.7 for the image sharpened by this filter.



(a) Original image

(b) Sharpened image

Figure 3.7: Example of Unsharp filter results [20]

5- Average filter:

An averaging filter is useful for removing grain noise from a photograph. Because each pixel gets set to the average of the pixels in its neighborhood, local variations caused by grain are reduced.

That each output pixel is set to an "average" of the pixel values in the neighbourhood of the corresponding input pixel [20]. Thus, the coefficients of this filter are:

P_1	P_2	P_3
P_4	P_{AV}	P_5
P_6	P_7	P_8

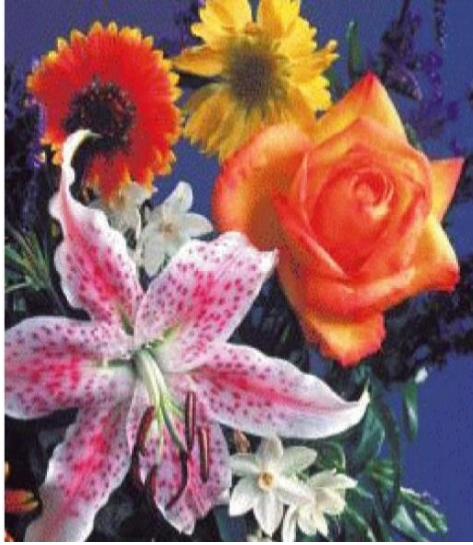
Where:

P_1, P_2, \dots, P_8 : The neighbourhood pixels.

P_{AV} : The middle pixel of mask.

So, the middle coefficient is calculated as average in the output image by equation (3.5):

$$P_{AV} = \frac{(P_1 + P_2 + \dots + P_8)}{8} \quad (3.5)$$



(a) Original image



(b) Image after average filter

Figure 3.8: Example of Average filter result [20]

6- Median filter:

Median filtering is a nonlinear operation often used in image processing to reduce “salt and pepper” noise, its goal is to simultaneously reduce noise and preserve edges. Median filtering is similar to using an averaging filter. However, with median filtering, the value of an output pixel is determined by the median of the neighbourhood pixels, rather than the mean. Median filtering is, therefore, effective to remove these outliers without reducing the sharpness of the image [20].

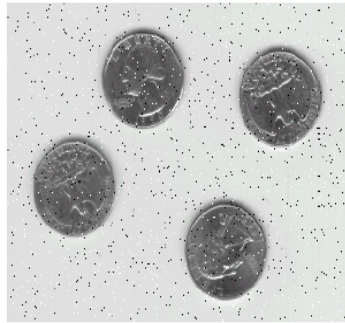
The Median filter selects the middle pixel value from the ordered set. Its filtering operation is performed on an image by applying the sliding window concept, similar to what was done with convolution [22]. These mask values will be sorted to have their orders. An example for Median filter coefficients is:

Table no 3.2: shows the median filter

20	7	8
10	9	15
2	3	1

1 2 3 7 8 9 10 15 20

It is obvious that number 8 has the middle order among other values. Then number 9 in the mask coefficient will be changed into 8. Figure 3.9 shows example for the Median filter:



(a) Original image

(b) Image after Median filter

Figure 3.9: Example of Median filter result [20]

So, Median filtering is very effective in removing sharp 'spikes' from an image. When noise is introduced in an image, lowpass filtering just 'blends' the noise, whilst median filtering succeeds in removing most of the noise [24].

3.2.3 Morphological filtering

Morphology is a technique of image processing based on structuring element shapes. The value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbours.

By choosing the size and shape of the neighbourhood, a morphological operation can be constructed that is sensitive to specific shapes in the input image. Morphological operations perform common image processing tasks, such as thinning, skeletonization and filling.

An overview to a few common Morphological shapes is shown below:

- 1- Line shape: It creates a linear structuring element, with LEN specifies the length, and DEG specifies the angle (in degrees) of the line, as measured in a counterclockwise direction from the horizontal axis. LEN is approximately the distance between the centers of the structuring element members at opposite ends of the line. Figure 3.10 shows the line shape structuring element.

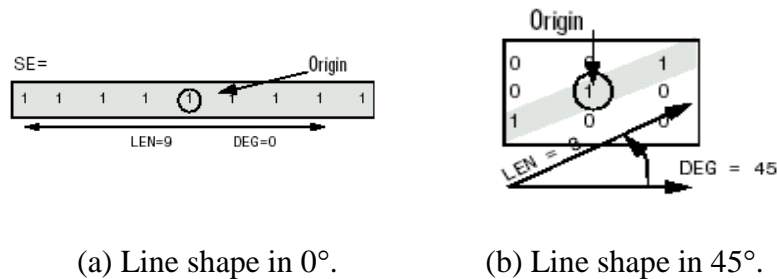


Figure 3.10: The line shape structuring element [24]

- 2- Square shape: It creates a square structuring element, with W specifying the width. The square shape created by $W \times W^T$, where W must be a nonnegative integer scalar. Figure 3.11 shows the square shape structuring element.

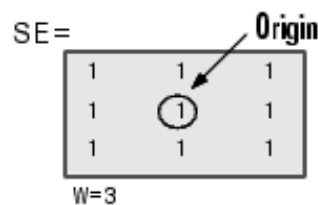


Figure 3.11: The square shape structuring element [24]

- 3- Diamond shape: It creates a diamond-shaped structuring element, with R specifying the distance from the structuring element origin to the points of the diamond. R must be a nonnegative integer scalar. Figure 3.12 shows the diamond shape structuring element.

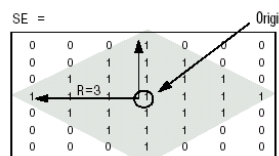


Figure 3.12: The diamond shape structuring element [24]

- 4- Rectangle shape: It creates a rectangle-shaped structuring element, with MN specifying the size. MN must be a two-element vector of nonnegative integers. The first element of MN is the number rows in the structuring element neighborhood; the second element is the number of columns. Figure 3.13 shows the rectangle shape structuring element.

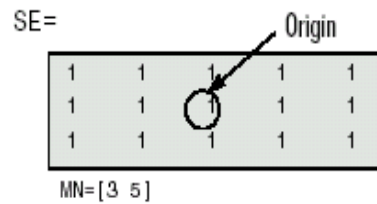


Figure 3.13: The rectangle shape structuring element [24]

- 5- Disk shape: It creates a disk-shaped structuring element, which comprises all pixels whose centers are not greater than R away from the origin. R must be a nonnegative integer. Figure 3.14 shows the Disk shape structuring element.

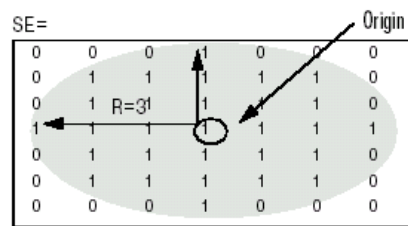


Figure 3.14: The disk shape structuring element [24]

- 6- Octagon shape: It creates an octagonal structuring element, with R specifying the distance from the structuring element origin to the sides of the octagon, as measured along the horizontal and vertical axes. R must be a nonnegative multiple of 3. Figure 3.15 shows the Octagon shape structuring element [20]

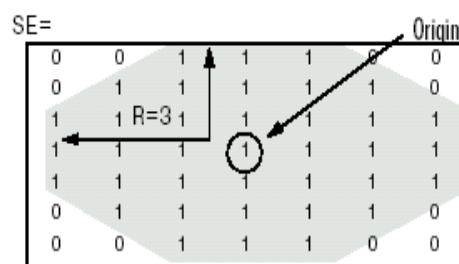
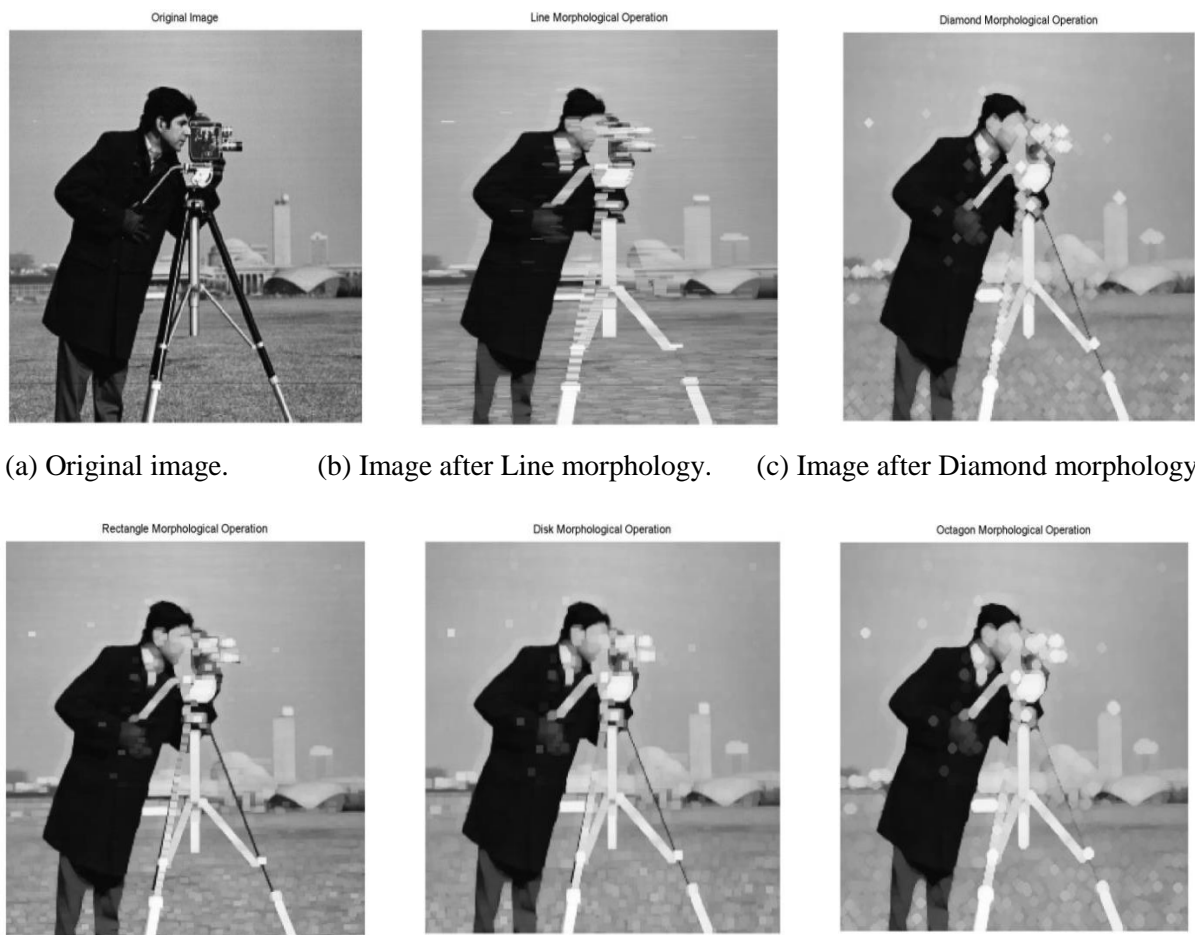


Figure 3.15: The octagon shape structuring element [24]

All morphological operations are performed by laying the structuring element on the image and sliding it across the image in a manner similar to convolution. The difference is in the operation performed. That is, the window is overlaid on the upper-left corner of the image, and the morphological operations are determined. This value is put into the output image (buffer) corresponding to the center location of the window. The window is then slid one pixel over, and the process is repeated. When the end of the row is reached, the window is slid back to the left side of the image and down one row, and the process is repeated. This process continues until the entire image has been processed.

Note that the outer rows and columns are not replaced. In practice, this is usually not a problem due to the fact that the images are much larger than the masks, and these wasted rows and columns are often filled with zeros [22]. See Figure 3.16 which gives examples for different morphological operations results.

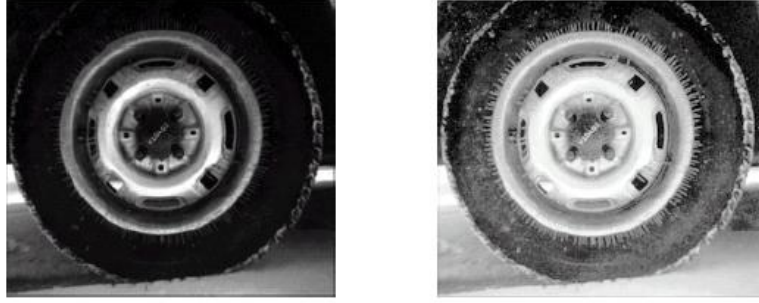


(a) Original image. (b) Image after Line morphology. (c) Image after Diamond morphology.
(d) Image after Rectangle morphology. (e) Image after Disk morphology. (f) Image after Octagon morphology.

Figure 3.16: Examples of different morphological operations results [22]

3.2.4 Histogram Equalization

Histogram equalization is an example of Histogram modeling techniques, it provides a sophisticated method for modifying the dynamic range and contrast of an image by altering that image such that its intensity histogram has a desired shape. See Figure 3.17.



(a) Original image. (b) Histogram equalization image.

Figure 3.17: Histogram equalization filter [22]

Algorithm for histogram equalization:

By considering two continuous variables $r, s \in [0,1]$ representing the level of pixels in the original and equalized images respectively. The goal of histogram equalization is to find a transform function $T(r)$ which produces a value s for every pixel value r in the original image:

$$s = T(r) \quad (3.6)$$

It is assumed that the transform function $T(r)$ is single-valued and monotonically increasing so that it is possible to define the inverse law:

$$r = T^{-1}(s) \quad (3.7)$$

An example of such transformation function is illustrated in Figure 3.18 when this formula is choosing:

$$s = T(r) = \int_0^r P_r(w) dw \quad (3.8)$$

where p_r is probability density function of original image [25]

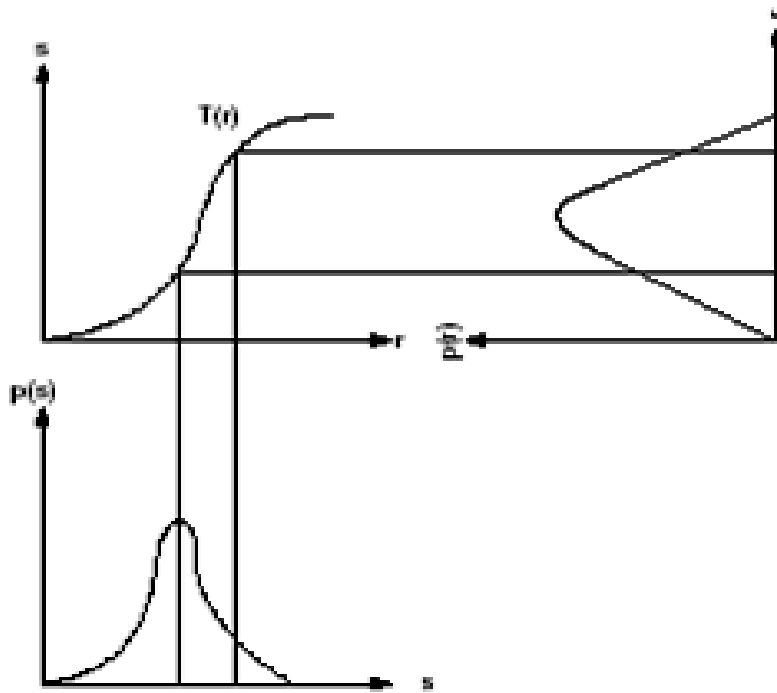


Figure 3.18: A histogram transformation function [25]

3.3 Summary

In this chapter, we identify some of the basic component of images and their some applications on the images. We apply some filtration techniques on the images to make it more precise and clear. Filter applications, morphological filtering, histogram equalisation are described in this chapter.

CHAPTER 4

THEORETICAL PRINCIPLES FOR NEURAL NETWORKS

4.1 Overview

Artificial Neural Networks (ANN) is computers whose architecture is modeled after the brain. They typically consist of many hundreds of simple processing units, which wired together in a complex communication network. Each unit or node is a simplified model of a real neuron, which fires (sends off a new signal) if it receives a sufficiently strong input signal from the other nodes to which it is connected. The strength of these connections may be varied in order to perform different tasks corresponding to different patterns of node firing activity [1].

4.2 Principles of ANN

The transmission of signals in biological neurons through synapses is a complicated chemical process in which specific transmitter substances are released from the sending side of the synapse. The effect is to raise or lower the electrical potential inside the body of the receiving cell. The neuron fires if the potential reaches a threshold.

This neuron model is widely used in ANN with some variations. Figure 4.1.

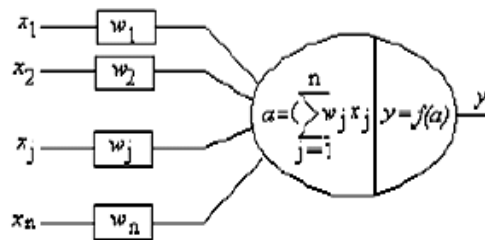


Figure 4.1: Artificial neuron model [28]

The artificial neuron presented in the above figure has n inputs, denoted as x_1, x_2, \dots, x_n . Each line connecting these inputs to the neuron is assigned a weight, denoted as w_1, w_2, \dots, w_n , respectively.

The action, which determines whether the neuron is to be fired or not, is given by the formula:

$$a = \sum_{j=1}^n w_j x_j \quad (4.1)$$

The output of the neuron is a function of its action:

$$y = f(a) \quad (4.2)$$

Originally the neuron output function $f(a)$ was a threshold function. However, linear, ramp and sigmoid functions are also widely used today. An ANN system consists of a number of artificial neurons and a huge number of interconnections among them. According to the structure of the connections [26,27].

ANN can be roughly categorized into two types in terms of their learning features: supervised learning algorithms, where networks learn to fit known inputs to known outputs, and unsupervised learning algorithms, where no desired output to a set of input is defined [14].

4.3 Activation functions

Activation function (or sometimes called a threshold function) transforms the summed up input signal, received from the summation function, into an output. The activation function can be either linear or non-linear. The type of activation function characterizes the neural network [14].

Multilayer networks often use the log-sigmoid activation function. This function is illustrated in Figure 4.2 shown below:

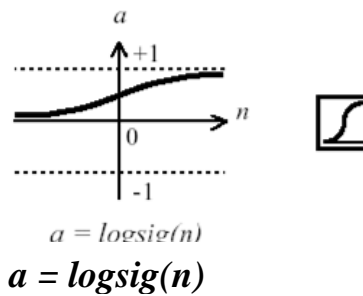


Figure 4.2: Log-sigmoid activation function [14]

This activation function generates outputs between 0 and 1 as the neuron's net input goes from negative to positive infinity.

Alternatively, multilayer networks may use the tan-sigmoid activation function. See Figure 4.3 below:

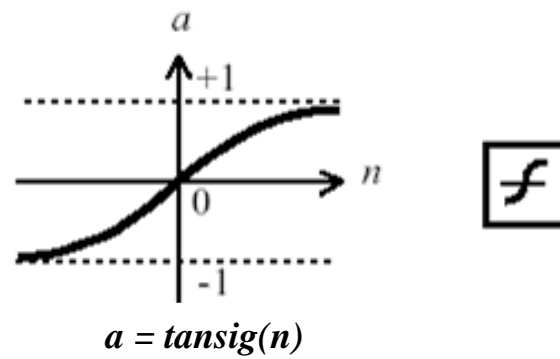


Figure 4.3: Tan-sigmoid activation function [14]

Occasionally, the linear activation function is used in BPN. This is shown in Figure 4.4:

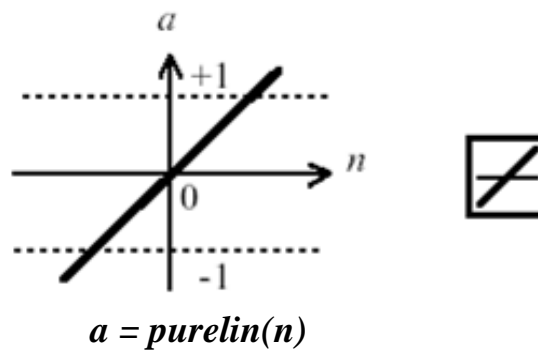


Figure 4.4: Linear activation function [14]

If the last layer of a multilayer network has sigmoid neurons, then the outputs of the network are limited to a small range. If linear activation functions are used in the output neurons, the network outputs can take on any value [28].

4.4 Linear Associative Memory (LAM) Neural Network

4.4.1 History of LAM

This network is suggested at the beginning by Tuevo Kohonen in 1972, and it was developed later in 1977 by the scientist James Anderson.

To a significant extent, learning is the process of forming associations between related patterns. Aristotle observed that human memory connects similar items (such as: ideas, sensations, etc.) which occur in close proximity or in close succession.

The patterns which associate together may be of the same type or sensory modality (e.g., a visual image may be associated with another visual image) or of different types (e.g., the fragrance of fresh-mown grass may be associated with a visual image or a feeling). Memorization of a pattern (or a group of patterns) may be considered to be associating the pattern with itself. An important characteristic of the associations formed is that an input stimulus which is similar to the stimulus for the association will invoke the associated response pattern. Our ability to recognize a person (either in person from a photo) is an example of the capability of human memory to respond to new situations. It is relatively difficult to program a traditional computer to perform this task easily [29].

4.4.2 Principles of LAM

An associative memory net may serve as a highly simplified model of human memory, and the Hebb rule is the simplest and most common method of determining the weights for an associative memory neural net [29]. The Hebb rule learning algorithm will be explained in the next chapter.

A supervised learning situation can be adapted by basing the weight adjustment for the connection on the desired output of the neuron rather than the actual output. The associations are given by a set of ordered pairs, $\{ \langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \dots, \langle x_b, y_t \rangle \}$, where x_i and y_i are the vector patterns to be associated. So, for each vector pair $\langle X_i, Y_i \rangle$ the input vector X_i is the key to get the output vector Y_i .

By supposing that the length of the x_i is n and the y_i is m , the network has two layers, an input layer of size n and output layer of size m , as shown in Figure 4.5:

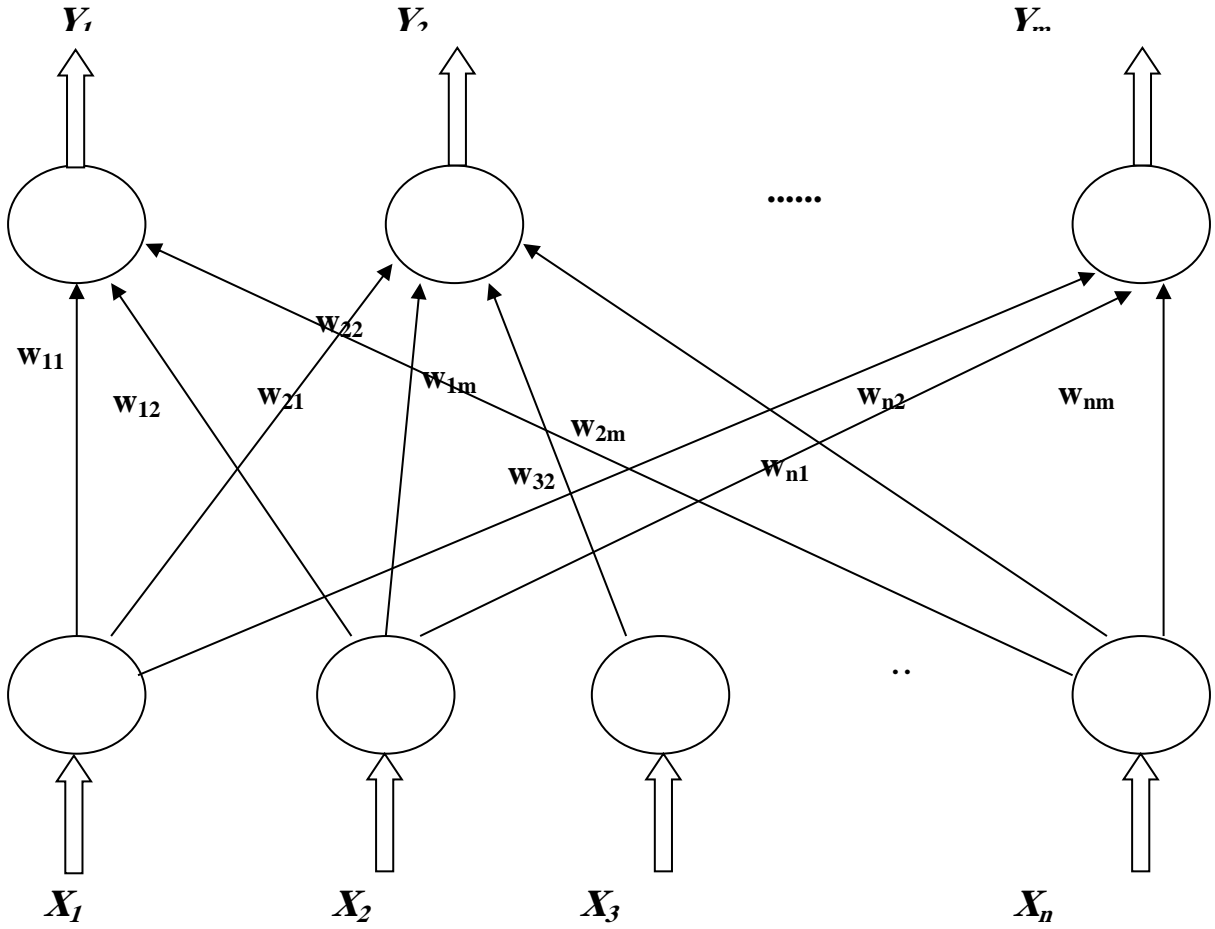


Figure 4.5: A supervised hebbian network for learning pattern association [29]

The weights are initialized as follows:

$$W = X_1 \times Y_1 + X_2 \times Y_2 + \dots + X_i \times Y_i \quad (4.3)$$

A network, which maps input vectors to output vectors using this formula for weight assignment, is called linear associator. [1]

The general equation for weights calculation is shown in the equation (4.4) below:

$$W_{ij} = \sum_{P=1}^m [X_i^P]^T Y_j^P \quad (4.4)$$

Where

p: number of vector pairs.

m: sum of all the vector pairs.

X: the input of the network.

Y: the decided output of network.

The equation vectors verify the equation (4.5):

$$W = \sum_{p=1}^m X_p^T Y_p \quad (4.5)$$

Then, the output vector Y is computed as equation (4.6):

$$Y_p = X^{(P)} W_{ij} = \sum_i x_i^{(P)} W_{ij} = \sum_j (x^{(p)} x^{(q)}) y_j^{(q)} \quad (4.6)$$

Where

(P): number of the previous vector pairs

(q): number of the current vector pairs

When the whole input vectors are normalized to one unit this will led to equation (4.7) [17]:

$$\sum_{j=1}^n (x_i^{(P)})^2 = 1 \quad \text{every} \quad p=1, 2, \dots, m \quad (4.7)$$

4.5 Back Propagation Neural Network (BPNN)

4.5.1 History of BPNN

The demonstration of the limitations of single-layer neural networks was a significant factor in the decline of interest in neural networks in the 1970s [29]. The BPN was created by generalizing the Windrow-Hoff learning rule to multiple-layer networks and nonlinear differentiable activation functions. Properly trained BPN tends to give reasonable answers when presented with inputs that they have never seen.

Typically, a new input leads to an output similar to the correct output for input vectors used in training that are similar to the new input being presented.

This generalization property makes it possible to train a network on a representative set of input/target pairs and get good results without training the network on all possible input/output pairs [28]. The training of a network by Back Propagation involves three stages: the feed forward of the input training pattern, the calculation and Back Propagation of the associated error, and the adjustment of the feed forward phase. Even if training is slow, a trained net can produce its output very rapidly. More than one hidden layer may be beneficial for some applications, but one hidden layer is sufficient for most purposes [29]

4.5.2 Architecture of standard BPNN

A multilayer neural network with one layer of hidden units (the Z units) is shown in Fig. 2.30. The output units (the Y units) and the hidden units also may have biases. The bias on a typical output unit Y_k is denoted by w_{0k} ; the bias on a typical hidden unit Z_j is denoted v_{0j} . These bias terms act like weights on connections from units whose output is always 1. Only the direction of information flow for the feed forward phase of operation is shown. During the Back Propagation phase of learning, signals are sent in the reverse direction. The algorithm in the next section is presented for one hidden layer, which is adequate for a large number of applications [29].

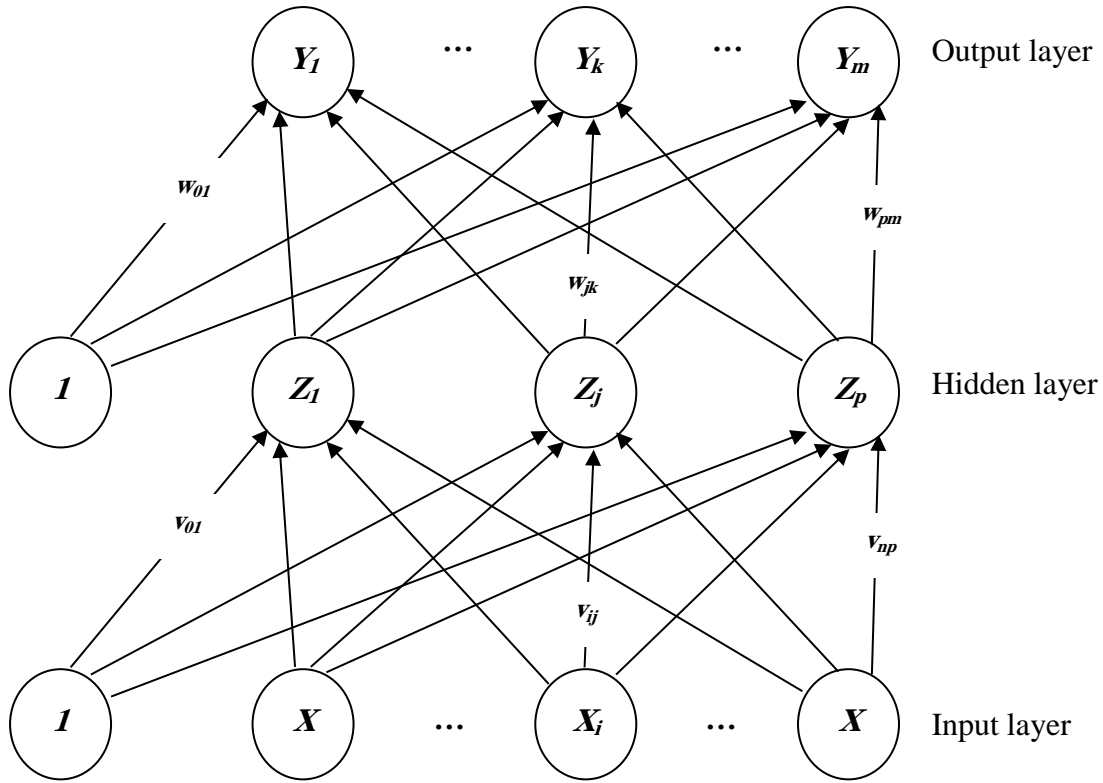


Figure 4.6: BPNN with one hidden layer [29]

4.5.3 Algorithm

As mentioned earlier, training a network by Back Propagation involves three stages: the feed forward of the input training pattern, the Back Propagation of the associated error, and the adjustment of the weights.

During feed forward, each input unit (X_i) receives an input signal and broadcast this signal to each of the hidden units $Z_1 \dots Z_p$ through the (v) weights. Each hidden unit then computes its activation and results in its signal (z_j) to each output unit through the (w) weights. Each output unit (Y_k) computes its activation (y_k) to form the response of the net for the given input pattern. During training, each output unit compares its computed activation y_k with its target value t_k to determine the associated error for that pattern with that unit. Based on this error, the factor δ_k ($k=1 \dots m$) is computed, δ_k is used to distribute the error at output unit Y_k back to all units in the previous layer (the hidden units that are connected to Y_k). It is also used (later) to update the weights between the output and the hidden layer.

In a similar manner, the factor δ_j ($j=1 \dots p$) is computed for each hidden unit Z_j . It is not necessary to propagate the error back to the input layer, but δ_j is used to update the weights between the hidden layer and the input layer. When all the δ factors have been determined, the weights for all layers are adjusted simultaneously.

The adjustment to the weight w_{jk} (from hidden unit Z_j to output unit Y_k) is based on the factor δ_k and the activation z_j of the hidden unit Z_j . The adjustment to the weight v_{ij} from input unit X_i to hidden unit Z_j is based on the factor δ_j and the activation x_i of the input unit.

An epoch is one cycle through the entire set of training vectors. Typically, many epochs are required for training a BPN. The foregoing algorithm updates the weights after each training pattern is presented. A common variation is batch updating, in which weight updates are accumulated over an entire epoch before being applied [29].

4.5.4 Momentum technique

Due to the time required to train a Neural Network, many researchers devoted their efforts for improving the speed up techniques. Various efforts range from optimizations of current algorithms to development of original algorithms.

One of the most commonly discussed extensions is momentum. BPN, which is frequently used in Neural Network training, often takes a great deal of time to converge on an acceptable solution. Momentum is a standard technique that is used to speed up convergence and maintain generalization performance [30].

In BPN with momentum, the weight change is in a direction that is combination of the current gradient and the previous gradient; that is a modification of gradient descent -whose advantages arise chiefly when some training data are very different from the majority of the data (and possibly even incorrect), is desirable to use a small learning rate to avoid a major disruption of the learning direction when a very unusual pair of training patterns is presented. However, it is also preferable to maintain training at a fairly rapid pace as long as the training data are relatively similar.

Convergence is sometimes faster if a momentum term is added to the weight update formulas. In order to use momentum, weight updates from one or more previous training patterns must be saved.

For example, in the simplest form of BPN with momentum, the new weights for training step $t+1$ are based on the weights at training steps t and $t-1$. The weight update formulas for BPN with momentum are:

$$w_{jk}(t+1) = w_{jk}(t) + \alpha \delta_k z_j + \mu [w_{jk}(t) - w_{jk}(t-1)] \quad (4.8)$$

or

$$\Delta w_{jk}(t+1) = \alpha \delta_k z_j + \mu \Delta w_{jk}(t) \quad (4.9)$$

and

$$v_{ij}(t+1) = v_{ij}(t) + \alpha \delta_j x_i + \mu [v_{ij}(t) - v_{ij}(t-1)] \quad (4.10)$$

or

$$\Delta v_{ij}(t+1) = \alpha \delta_j x_i + \mu \Delta v_{ij}(t) \quad (4.11)$$

Where the momentum parameter μ is constrained to be in the range from 0 to 1, exclusive of the end points.

When using momentum, the net is proceeding not in the direction of the gradient, but in the direction of a combination of the current gradient and the previous direction of weight correction [29].

4.6 Wavelet

Addison describes a wavelet as a mathematical function used to divide a given function or a continuous-time signal into different frequency components and study each component with a resolution that matches its scale[37]. The wavelets are scaled and are the translated copies (known as daughter wavelets”) of a finite-length or fast-decaying oscillating waveform known as the mother wavelet”).

Wavelet transforms have advantages over traditional Fourier transforms for representing functions that have discontinuities and sharp peaks and or accurately deconstructing and reconstructing finite non-periodic, and/or non-stationary signals [38] These underlying characteristics make wavelets applicable for creating the feature vector that is necessary in the iris recognition algorithm.

4.7 Summary

In this chapter, we identify some of the neural network techniques that we use for the recognition of the images from the data base. In this chapter, basically we use two algorithms from the neural network. Linear Associative Memory Neural Network (LAM), and Backpropagation Memory Neural Network (BPM).

CHAPTER 5

SYSTEM DESIGN

5.1 Overview

In this chapter the design of Iris recognition system has been performed. The chapter will also explain the image processing and Neural network. Image processing will show you how to extract the images and which technique is used for the extraction of the images. And Neural Network is used for the identification of the iris images. For iris recognition Linear Associative Memory and Back Propagation Neural Network techniques are used.

5.2 System Design Requirements

System requirements are extracted from the previous theoretical principles. They will be detailed in the next chapter for the practical work software as leading to the best results in analysis. A digital image captured by camera having an eye in Three-Dimensional (3D) matrix RGB image must be divided into an intensity image for dealing with 2D matrix image and the binary image will be used for pupil darkness density extraction. A convolution is important for each image filter type performed, where an unsharp filter will be used for producing heavy sharpness image and an average filter performed on iris image for noises extinction. Morphological dilation performed by using square structuring element shape into reaching a reference point for iris image extraction and disk structuring element shape for iris image contrast enhancement. Histogram equalization is very effective in image detail enhancement.

In identification, a LAM network and BPNN are used each one alone. A linear activation function used in LAM network and a tan-sigmoid activation functions with momentum technique are used in BPNN. The comparisons between the two methods will be illustrated in chapter Six. Figure 5.1 shows an on-line iris recognition system design model.

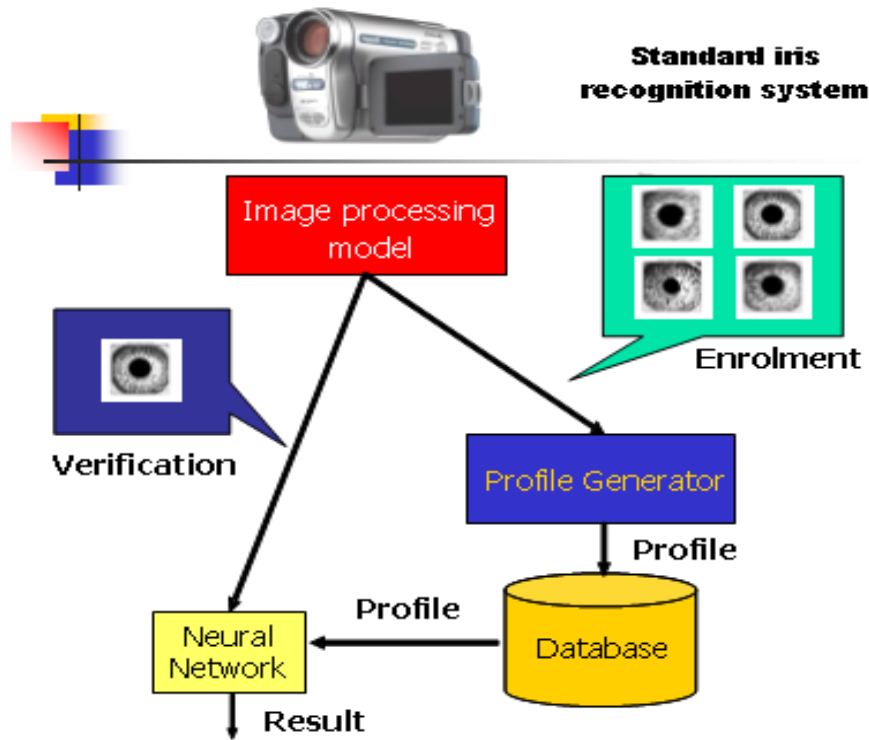


Figure 5.1: on-line iris recognition system model [30]

5.3 Image processing

The acquired image always contains not only the ‘useful’ parts (iris), but also some ‘irrelevant’ parts (e.g. eyelid, pupil etc.). Besides, under some conditions, the brightness is not uniformly distributed. For the purpose of analysis, the captured image needs to be preprocessed [2]. To capture the rich details of iris patterns, an imaging system should resolve a minimum of 50 pixels in iris radius [11]. This chapter explains the steps of image preprocessing starting from data acquisition which consists of sharpened image and converting to an intensity image, after that image localization which consists of histogram equalization image, lowpass filtered image, converting to a binary image, morphological dilation process and extracting the iris image, then image enhancement which consists of average filtered image, enhancement contrast filtered image and histogram equalization image. Finally image segmentation for data pattern extraction. Figure 5.2 shows the block diagram of this steps which is adopted in processing the eye image.

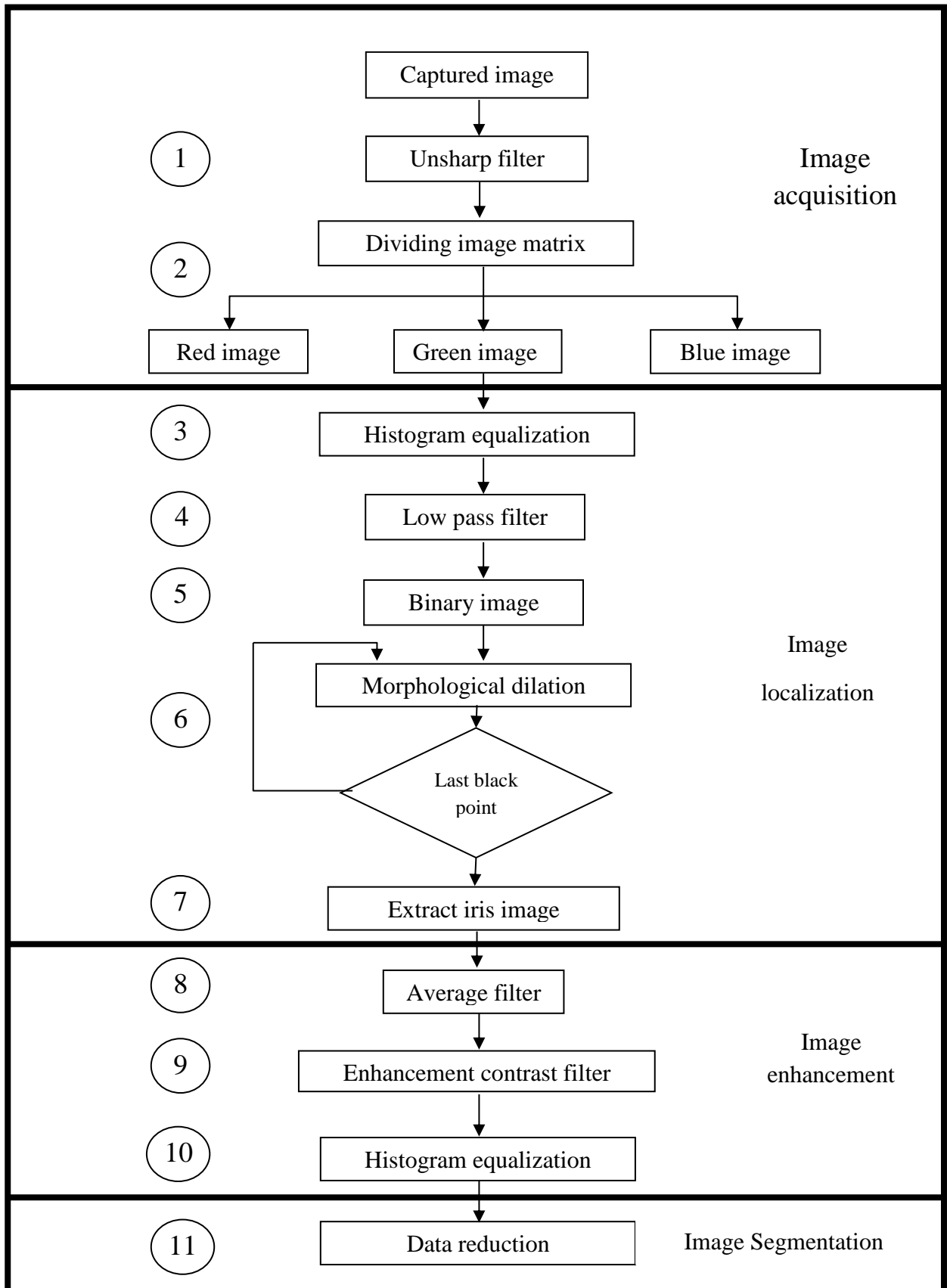


Figure 5.2: The steps adopted for eye image analysis [11]

5.3.1 Image acquisition

The essential step in iris recognition system is the proper acquisition of the eye image. Since iris is small in size and dark in colour (especially for Asian people), it is difficult to acquire good images for analysis using the standard CCD camera and ordinary lighting. Therefore, a device for image acquisition must be designed, which can deliver iris image of sufficiently high quality [2]. The process of measurement should be fast, comfortable for the person under test [12].

After many experiments on different cameras, it became distinct that the digital video camera type Sony Digital Camcorder (DCR)-TRV265E, which is explained in appendix A, with the box of alumina holding by a stand carrier and having a hole to put the eye on it, in order to curtain any external lights, can be a good way to capture the image. See Figure 5.3.

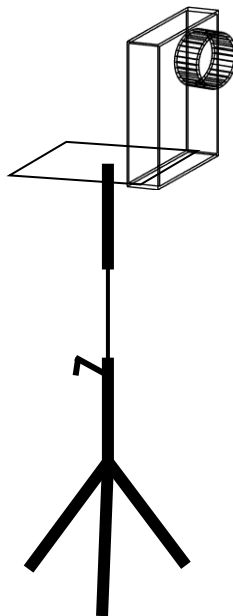


Figure 5.3: Alumina box with a stand carrier [12]

The Near-Infrared (NIR) wavelengths spreading by the camera light accessory even darkly pigmented irises reveal rich and complex features [11]. Furthermore, The size of the pupil may change due to the variation of the illumination and the hippus, and the associated elastic deformations in the iris texture may interfere with the results of pattern matching.

For the purpose of accurate texture analysis, it is necessary to compensate this deformation [2]. Therefore, by using the continuous white light from the video camera a fixed narrow pupil size can be gotten.

Captured image is of type Joint Pictures Expert Group (JPEG) with the size 240×320×3 pixels.

The original iris image has low contrast and may have non-uniform illumination caused by the position of the camera light source. These may impair the result of the texture analysis [2]. So, there is a need for preprocessing operations. The image will be captured like a photo from windows movie maker software. See Figure 5.4 for the windows movie maker software options and Figure 5.5 for captured image example needed to be preprocessed.

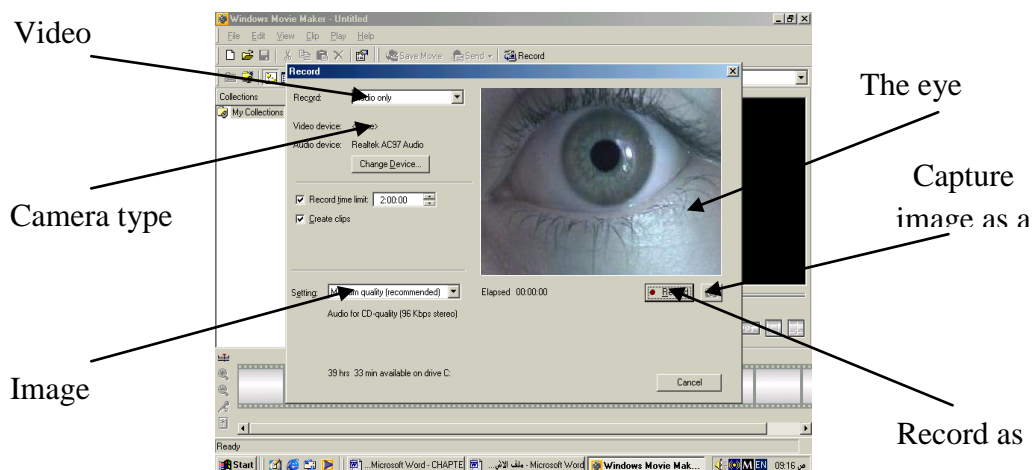


Figure 5.4: Windows movie maker software options [2]

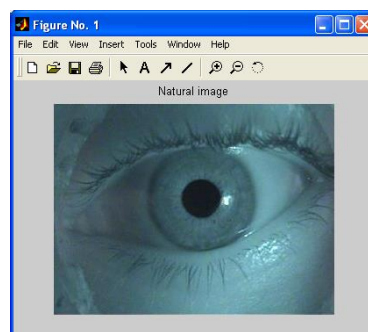


Figure 5.5: Natural eye image needed to be preprocessed [2]

5.3.2 Iris localization

Compared with the other parts of the eye, the pupil is much darker. To locate the iris in eye image, it can be followed the darkness density of the pupil. Darkness density pupil method is more efficient than other method to locate the iris [2].

Other method, like integro-differential operators that are sensible to the specular spot reflection of the non-diffused artificial light pointing to the center of the user's eye. This detection strategy consists in using firstly an edge calculation technique to approximate the position of the eye in the global image (center of the pupil), and secondly integro-differential operators to search more precisely pupil boundary, iris center and iris boundary. So, whenever this spot takes place in the pupil near from the iris/pupil frontier, the detection of the inner boundary of the iris fails [5].

It is easy to detect the inner boundary between the pupil and the iris, but the outer boundary of the iris is more difficult to detect because NIR makes low contrast between the two sides of the boundary [2].

5.3.3 Image enhancement

The previous image needs to reduce the effect of non-uniform distortion and illumination. In addition, it needs to make the texture more precise and clearer. Then the iris texture will be able to extract the data which are important for comparison.

- 1- Average filter: An averaging filter is useful for removing distortion from the iris image pattern, because each pixel gets set to the average of the pixels in their neighborhood. Local variations caused by grain are reduced when calculating the average in the output image by using equation (2.10). See Figure 5.6:

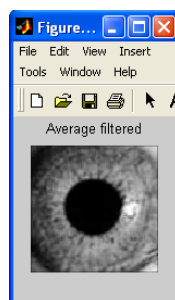


Figure 5.6: Average filter iris image [2]

2- Enhancement contrast filter: Morphological Top-hat and bottom-hat filtering can be used together to enhance contrast in an image [20]. This is done by the following steps:

- First, by creating a flat disk-shaped structuring element, with R equal to 3, as shown in Fig. 2.20.
- Second, by making Top-hat filter into iris image (by using the disk-shaped) to get the top values with there positions. Also making bottom- hat filter (again, by using the disk-shaped) to get the bottom values with there positions in the image.
- Finally, by adding the top-hat image to the original image, and then subtracting the “bottom-hat” image from the result, the texture details will be more deeper and clearer.

Then the iris image enhancement in contrast is performed. See Figure 5.7:

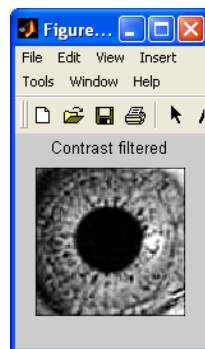


Figure 5.7: Contrast enhancement texture iris image [2]

3-Histogram equalization: It is the final step for image analysis and it is important to equalize the values of the previous texture image and the output image will contain a uniform distribution of intensities. See Figure 5.8.

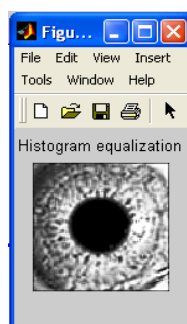


Figure 5.8: Histogram equalization image [2]

(This image is the final step of images analysis)

5.3.4 Image segmentation

To provide suitable computations to the next stage (neural network), the final image will be divided into 100 segments, each segment has 10×10 pixels, then a standard deviation (STD) can be taken for each square segment.

STD statistics is a measure of variability equal to the square root of arithmetic average of the squares of the deviations from the mean in a frequency distribution. In practice STD which calculates the average value for nested pixels is considered to be the more suitable one for data conversion [1]. It is calculated according to equation (5.1) shown below [17]:

$$STD_j = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (A_i - M_i)^2} \quad j = 1, 2, \dots, 100 \quad (5.1)$$

Where N represents the number of pixels in each section A_i represents the intensity of i-th pixel in the section, M_i represents the mean value of window's pixels, Depending on equation (5.2):

$$M_i = \frac{1}{N} \sum_{i=0}^{N-1} A_i \quad (5.2)$$

From equation 5.2 proposed system gets the 100 STD values from the last iris texture image analysis. This data will be used as an input to the neural network chosen topology.

5.4 Neural network

For the purpose of identification and comparisons, two ways are used in neural network techniques: Linear Associative Memory method, Back propagation Neural Network method.

The networks are trained for pattern classification. The training process requires a set of examples of proper network behaviour (network inputs p and target outputs t). During training, the weights and biases of the network are iteratively adjusted to minimize the network performance for feedforward [28].

5.4.1 LAM network suggested topology

The LAM network which is selected in matching process has 100 nodes in the input layer and the one node in the output layer. The suggested structure of LAM network is shown below in Figure 5.9.

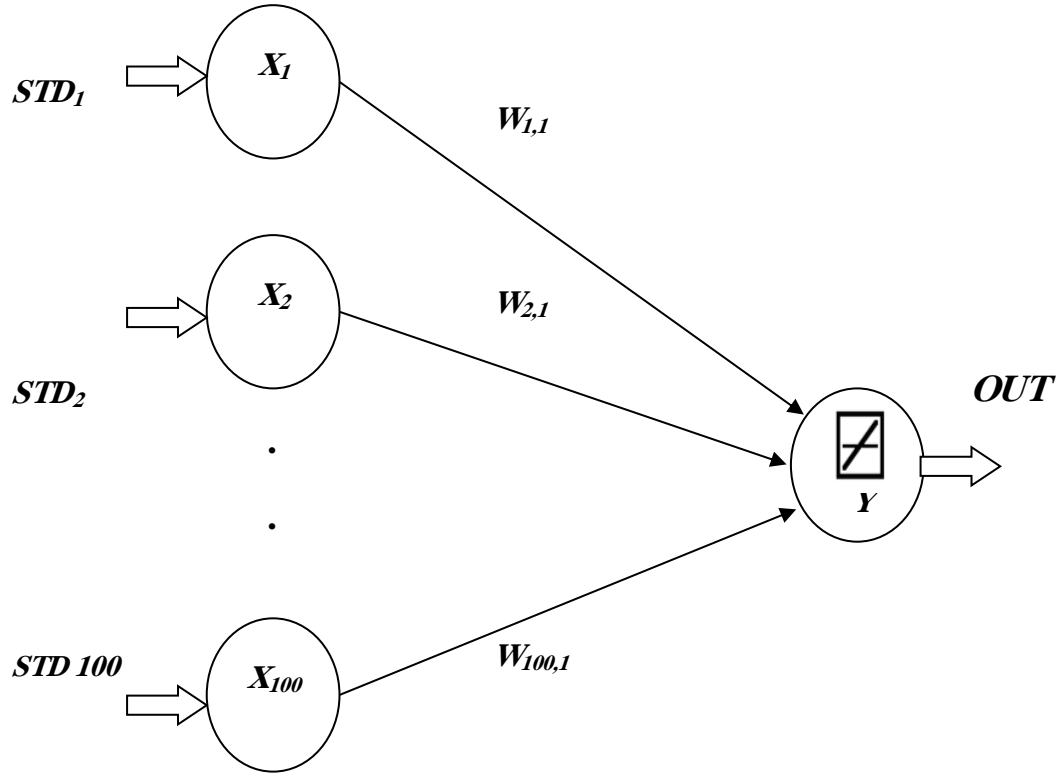


Figure 5.9: Structure of LAM network [17]

As shown in this figure, the STD values represent the input values to the network. After computing the weights by using equation (2.17), the output of the iris image is calculated by equation (5.3) shown below [17]:

$$Y = \sum_{i=1}^{100} X_i W_{i1} \quad (5.3)$$

5.4.2 BPNN Suggested Topology

The BPNN which is suggested has one hundred nodes in the input layer, three nodes in the hidden layer and one node in the output layer. This topology requires hidden units ranging from 0 to 3, the more hidden units the better results but complex computation. The activation functions used are tan-sigmoid activation functions for the whole neurons on the hidden and output layer. And momentum technology is used to speed up convergence. As mentioned previously, image segmented into 100 square matrixes and the standard deviation (STD) calculated for each segment as ($STD_1, STD_2, \dots, STD_{100}$). These values flow parallel to the input layer for training. The calculated weights (including biases) are stored in the database file and become the comparison base to detect any other iris image. Every iris image to be tested enters the network in the same way, the output values will be compared with the outputs of the original image. Then the error will be compared with 0.00001 tolerance value, and this is the least value of any tested image. Because as much as the error value is less, more accurately the image is recognized properly.

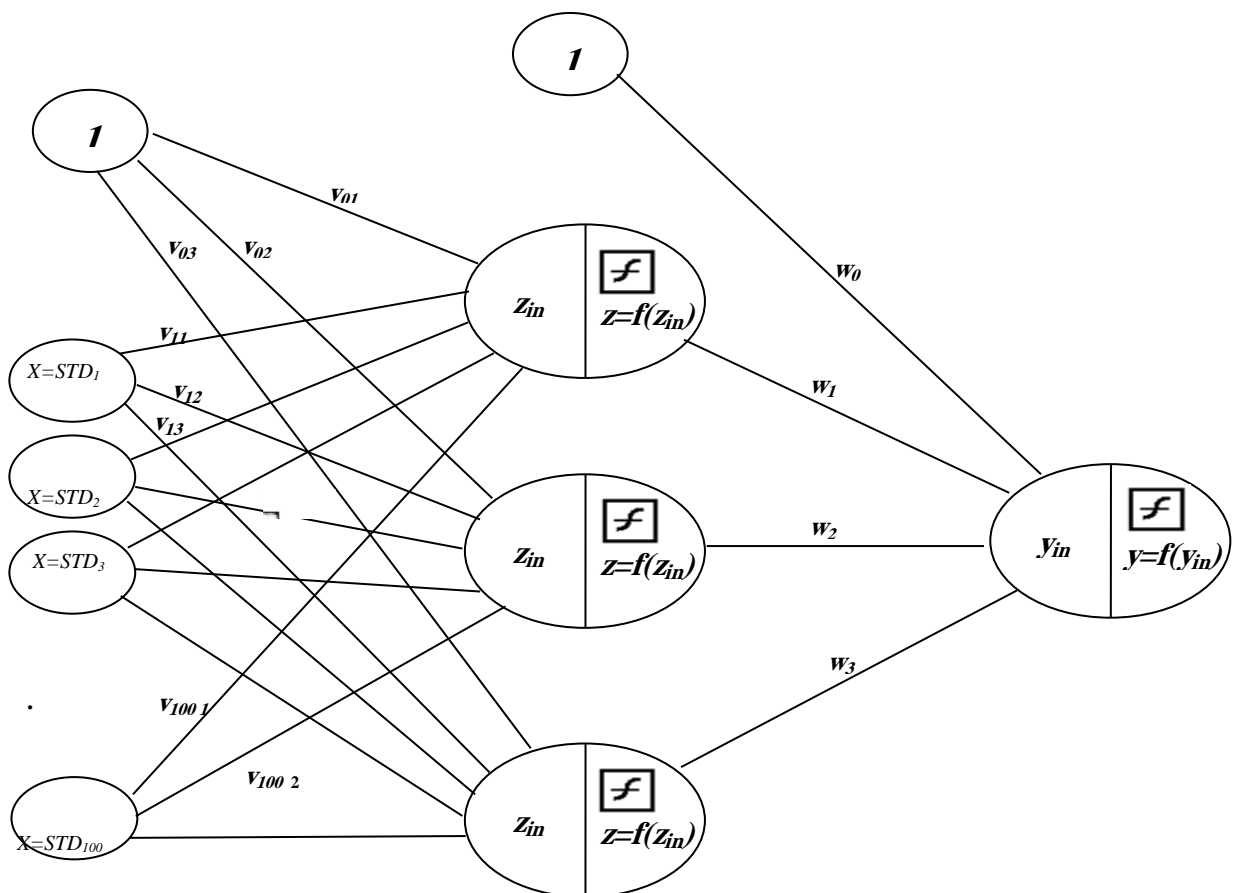


Figure 5.10: BPN suggested design [20]

5.5 Summary

In this chapter, we identify the system that we are going to use in the future. We use some of the properties from the image processing like image acquisition, image localization, image segmentation and image enhancement. And then we use the two algorithms from the neural network Linear Associative Memory Neural Network (LAM) and Backpropagation Memory Neural Network (BPM) for the recognition of images from the database.

CHAPTER 6

EXPERIMENTAL RESULTS AND COMPARISON

6.1 Overview

This chapter describes the system hardware and software with reviews to its performance in simplicity, fast and easy to use. Also, it displays the obtained results of image processing and neural networks with explaining the best way that can be used to have a good compression and less executive error.

To make sure that the system efficiency is high, two expression rates are accredited in measurement. These rates are called the False Acceptance Rate (FAR)/False Rejection Rate (FRR). The values of the rates are bound to a certain security threshold [4].

6.2 System features

The system designed for iris recognition is simple but expensive. It consists of (1) Alumina box with a stand carrier, (2) Sony video camera in high specifications (having NIR light and large video zooming), (3) Complete computer system (having: speakers, SVGA card, USB controller, Ram 2 GB, windows 7, hard disk 120 GB, processor 1.6 GHz etc.) and (4) The software program of iris recognition with MATLAB program in version 6.5. See Figure 6.1 which exposes the system hardware.



Figure 6.1: The hardware requirements for iris recognition system [4]

As mentioned earlier, the image is captured using windows movie maker software, when pressing a photo captured image icon the software program need to input a name for the image with type JPEG.

This method is more faster than taking image by the hardware bottom of camera to save it on a digital tape then translates its file to the computer.

This system provides authentication and small time recognition of an individual's identity by mathematical analysis for the iris random patterns of an eye. Figure 6.2 shows the flow chart that describes the system program performance. The program execution steps are found in appendix B.

6.3 Image processing results

Image processing is a leading part of this work to get an enhancement iris pattern image for data extraction. The results obtained from the applications of image processing techniques on the captured eye image are.

FLOW CHART FOR PROPOSED SYSTEM

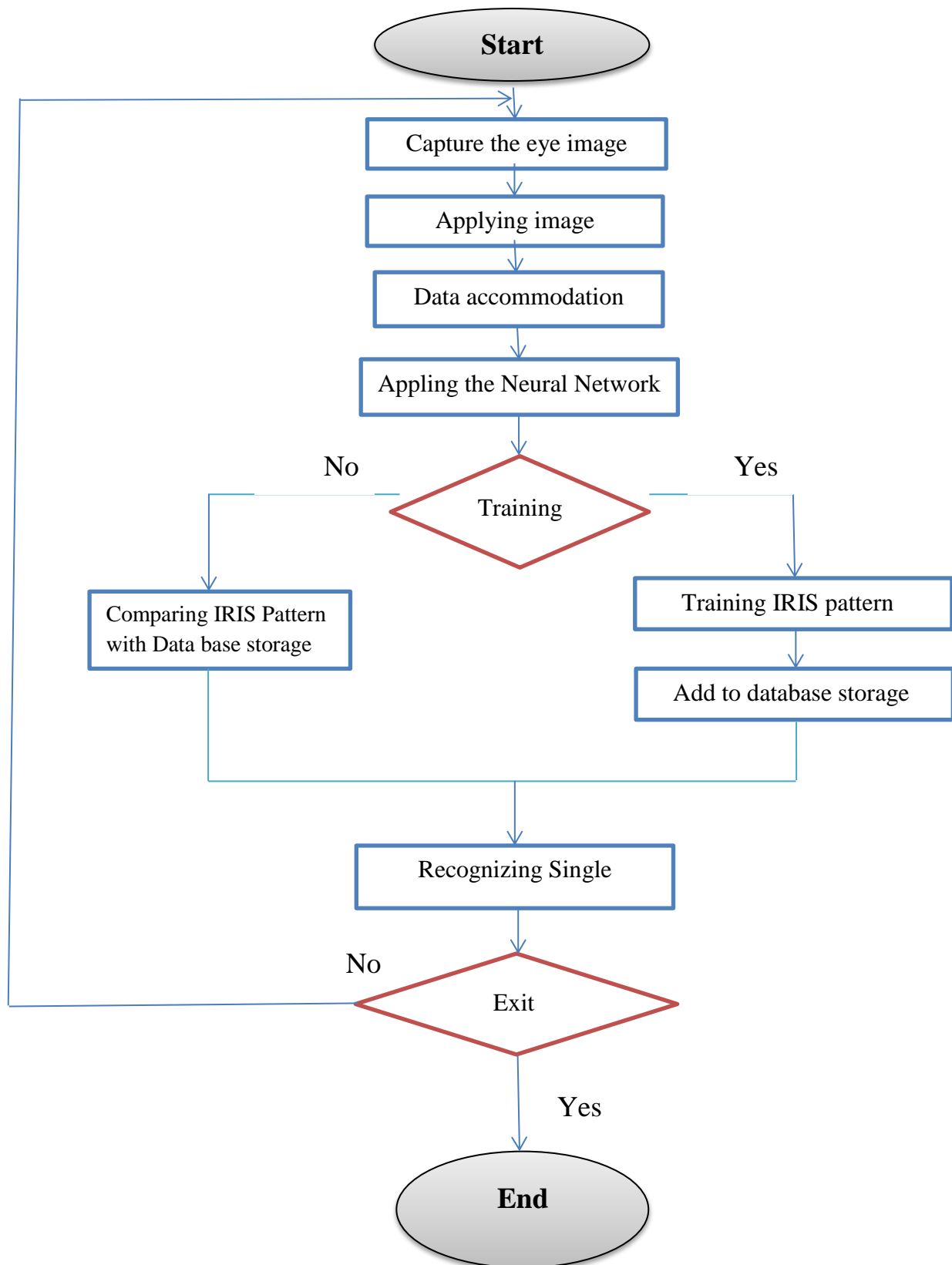


Figure 6.2: Main Flow chart for IRIS Training and recognition system program

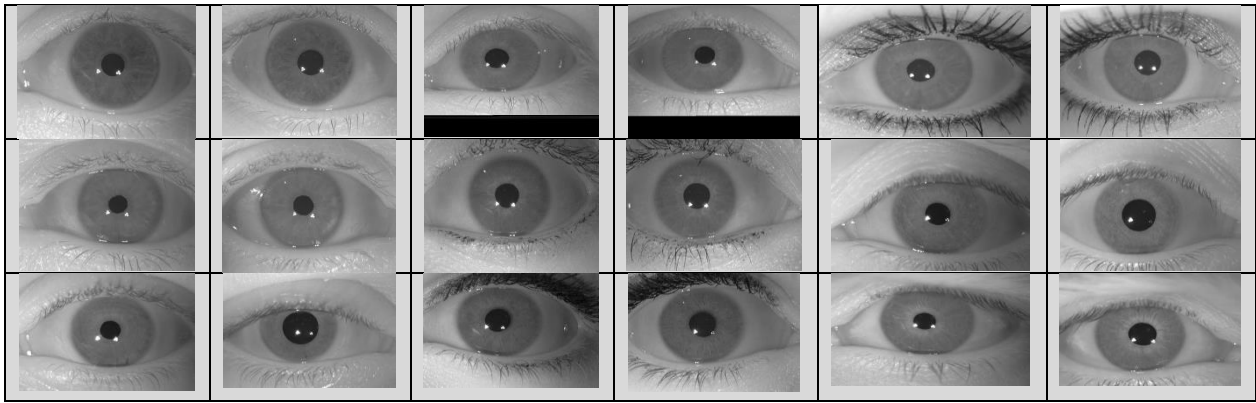


Figure 6.3: trained images

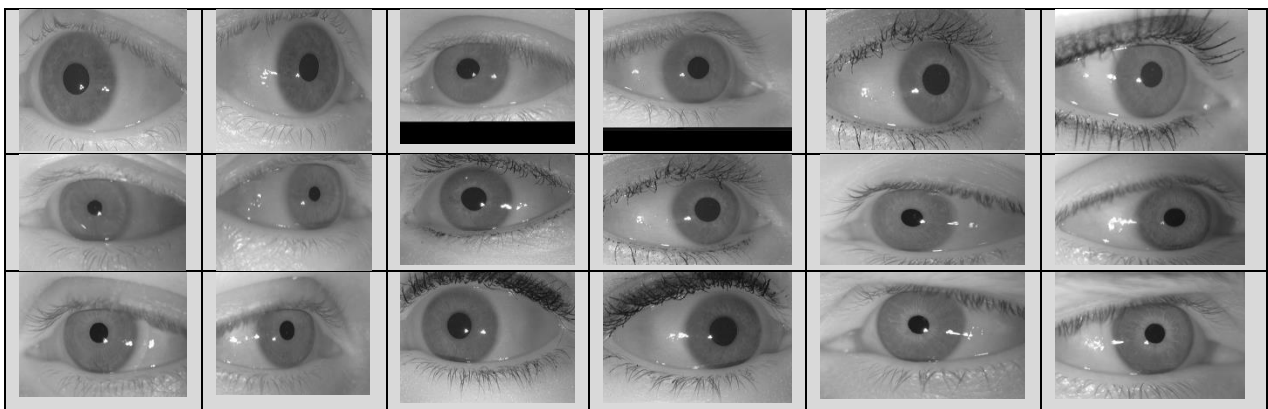


Figure 6.4: tested images

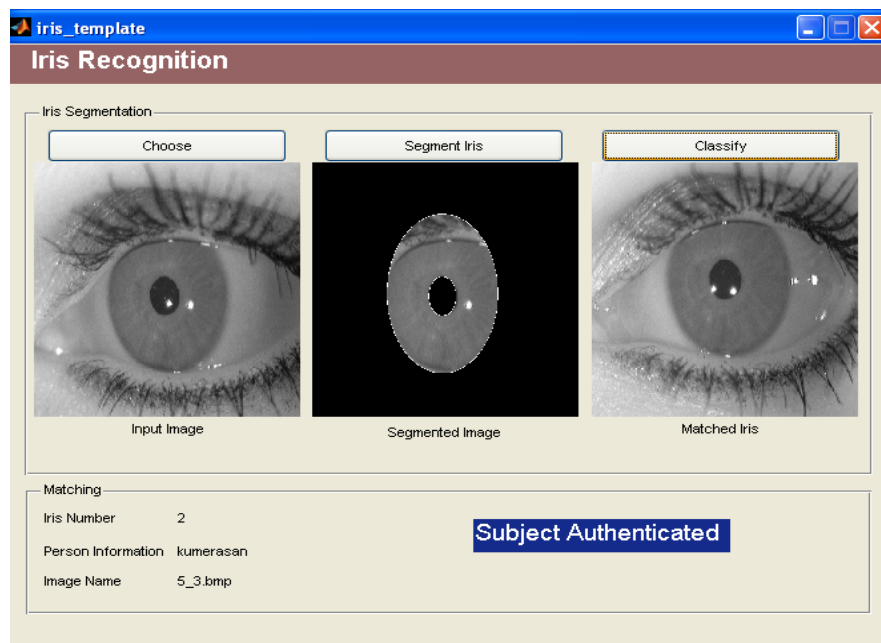


Figure 6.5: Result with successful image.

Table 6.1: information about the images in the database

Information about images in database	
Extension of image	BMP
dimension of image	640x480
Width	640 pixels
Height	480 pixels
Size	301 kb
Color of images	Gray scale

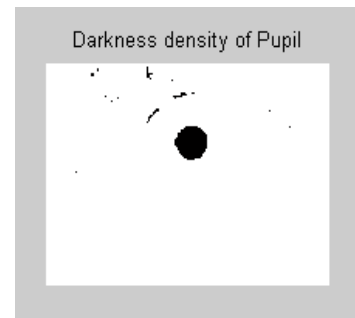
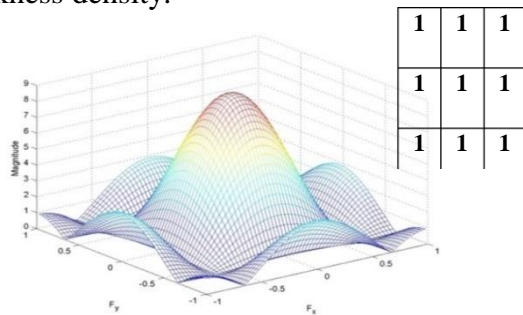
1- Image Acquisition:

The image acquisition is the first and important part of standard iris recognition systems. A box of alumina is established in front of a digital video camera and holding by a stand carrier. It is an effective way for eye image acquisition because it provides standard functions which are curtaining any external lights, aligning the human's eye against the camera lens and avoiding human's head or camera vibration and providing a suitable camera movements.

2- Iris Localization:

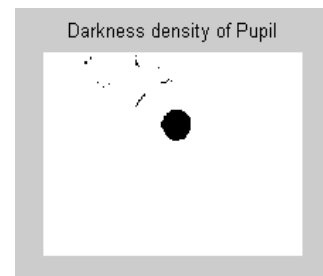
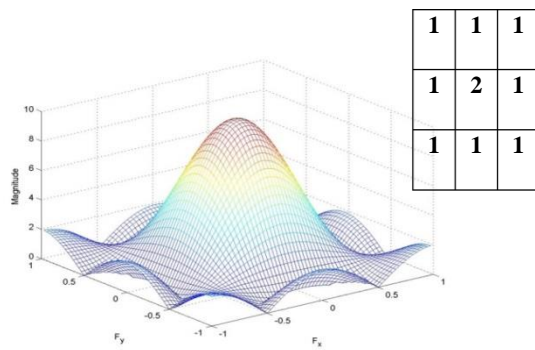
Locating the Iris can be done by following the darkness density of pupil by dilation processes. As mentioned in Chapter Three, this method is more efficient to locate the iris than the other method where the location is found by integro-differential operators [22]. To get the best way for following the darkness density of the pupil the low pass filter was used. But for choosing the effected mask of low pass filter, the coefficients in the mask may be biased. This is typically for application-specific reasons. The effected mask may be weighted the center pixel, or the diagonal pixels more heavily than the other pixels. These types of masks are often multiplied by $1/N$ to get one, where N is the sum of the mask coefficients. This is the equivalent of linearly remapping the image data after the convolution [22]. Figure 6.3 shows the common types of low pass filter masks with their frequency responses.

The figure also clarifies that the last mask gives the best result in pupil darkness density images, but the first two masks get some other black regions in the eye image with the pupil darkness density.



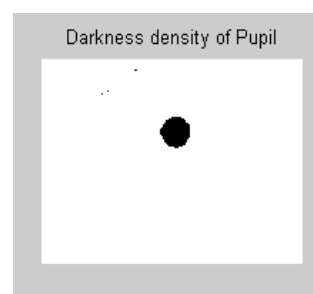
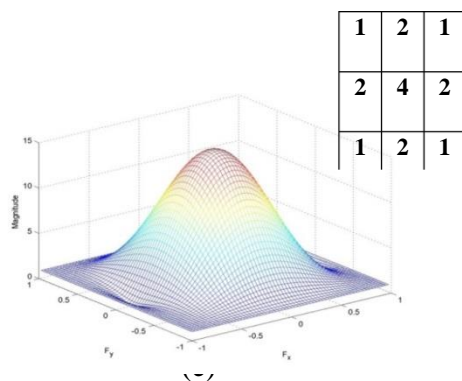
(a) The lowpass kernel mask with its frequency response

(b) The darkness density image implemented by the mask



(c) The lowpass kernel mask with its frequency response

(d) The darkness density image implemented by the mask



(e) The lowpass kernel mask with its frequency response

(f) The darkness density image implemented by the mask

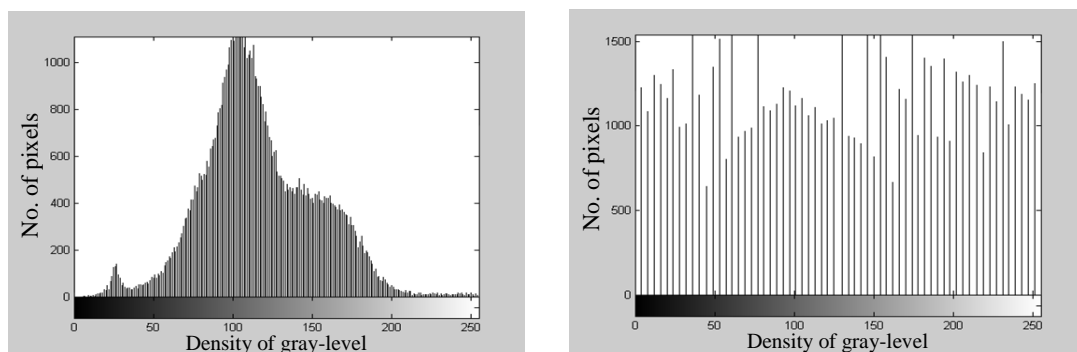
Figure 66: Common types of lowpass filter masks with their frequency responses [22]

3- Image enhancements:

Histogram equalization may have very effective method used for enhancing images. In the original image histogram (An image histogram is a chart that shows the distribution of intensities in an intensity image) the gray-levels of the image are too close to each other. And this image's density function is not uniform or near uniform. This means that the probabilities of the gray-levels are not equal. An ideal image's density function should be uniform, i.e. the probabilities of the gray-levels should be equal to each other. When the original histogram transformed to a uniform histogram (histogram equalization) the new image will be close to ideal. Histogram equalization method may be used twice before image segmentation:

- a) For the green part of sharpened image (to equalize the data of an eye image).
- b) For the final image analysis step (to equalize the data of an iris image).

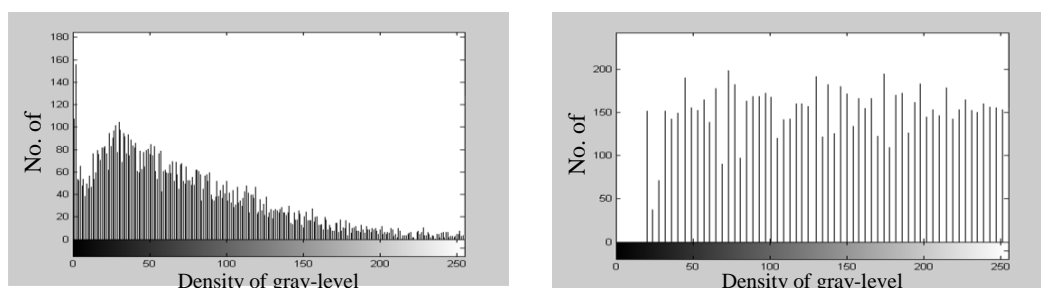
Figure 6.4 and Figure 6.5 show the histogram equalization of the same image for the green part of sharpened image and for the final image analysis respectively:



(a) before the histogram equalization

(b) After the histogram equalization

Figure 6.7: Example of histograms for green part of eye image [22]



(a) Before the histogram equalization

(b) After the histogram equalization

Figure 6.8: Example of histograms for final analysis iris image [22]

4- Image segmentation:

It is important to see differences between the iris images for different persons and a convergence between the iris images for the same person in data values. After image segmentation, taking STD values for each image segment may give suitable results because of its features, where it reduces the huge numbers of iris image data and exhibits the variations between image segments. This step prepares the iris pattern data to be used in the next stage neural network. Figure. 6.6 shows that the STD values for two images had the same iris pattern. It proves that image data is nearly for the same iris person pattern.

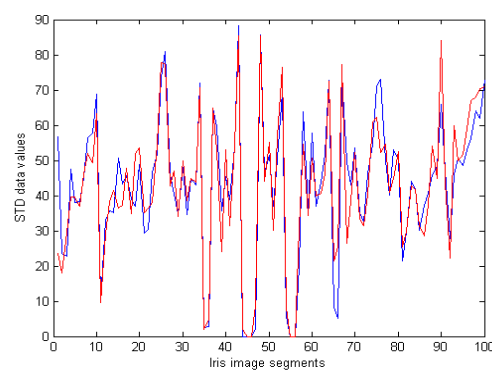


Figure 6.9: STD curves of two iris segmentation images for the same eye [27]

In the previous figure, the little differences for the values between the two images are due to the random noise from camera and the small change in illumination. This equilization for the same iris images STD values will lead to give different STD curves between different iris images. Figure 6.7 below shows the differences between the right iris and the left iris for the same person:

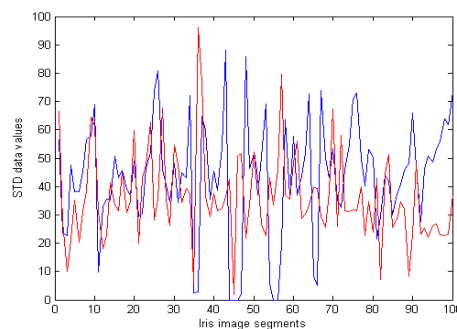


Figure 6.10: STD curves of the right iris and the left iris for the same person [23]

6.4 Neural networks results

Two types of neural network were used for the recognition as explained in Chapter Three. The number of Iris images used in training for the two types are 50 samples. The input data used for training is the STD of each iris segment:

1- LAM method:

The LAM network topology, as shown in Figure 6.8, is a single-layer consisting of 100 nodes for input and 1 node for output. It has 100 weights to be in the database file. The data input stream is parallel for each iris image. By using this network in the system, an execution time is about 45 seconds or less for iris image training due to the parallel feedfor word data stream.

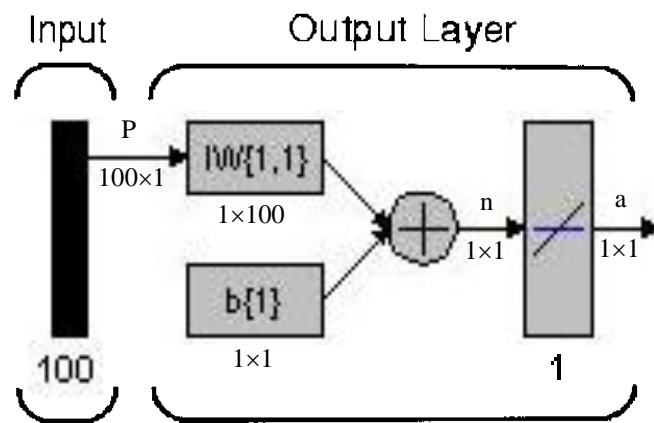


Figure 6.11: LAM network architecture [25]

This LAM network topology attained Recognition rate 80%. These rates are imputed to the normalization for image data, because the division by parameter (AA) in equation 3.5 may give too close results for two different images. Table 4.2 shows samples for weights which are extracted from different iris patterns, these weights differ from one image to another and when tested iris input to the system each STD data value will multiply to its weight in these iris trained images. Thus when tested iris image is the same trained iris image. Each STD data in fact is close equal to its weight value.

2- BPNN method:

The BPNN topology, as shown in Figure 6.9, is a multiple-layer consisting of 100 node for input, 3 nodes for hidden and 1 node for output. It has 307 weights and biases to be stored in the database file. The data input stream is parallel for each iris image.

By using this network in the system, an execution time is about 35 seconds or less for iris image training due to the feedback word in BPNN learning algorithm.

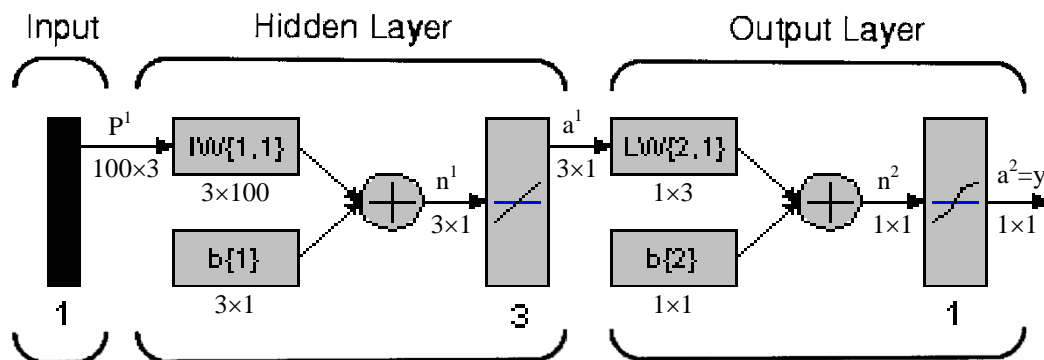


Figure 6.12: BPNN architecture [25]

This BPN topology attained recognition rate 90%. These rates are imputed to the tan-sigmoid transfer function which is used in the output layer. Table 4.3 shows samples for weights extracted from different iris patterns. These weights may be used in image testing. Then, the output of BPN checked to define the person.

3- Comparisons:

The BPNN topology used is more accurate than the LAM network topology used in recognition because of its Recognition percentages for the irises tested images, But it needs more execution time for learning. Furthermore, BPNN topology has used number of neurons more than the LAM network. Table 6.1 shows the differences between these two used methods.

Table 6.1: Difference between the two neural networks methods

Comparison subject	LAM	BPNN
Network type	Single-layer	Multi-layer
Feed forword flow	100 input / 1 output	100 input / 3 hidden / 1 output
Database stored	100 weights	307 weights
Recognition rate	80%	90%
Tolerance for comparison with error	0.0000001	0.00001
Program generalization Speed	45 seconds	35 seconds
Size of input images	640x480	640x480

In Linear Associative Memory Neural Network we use only single layer network but in Backprobagation Neural Network we use multi layer network, in LAM we have 100 inputs and one output but in BPNN we have 100 inputs layer 3 hidden layers and one output, the recognition rate of BPNN is more than the LAM for the iris tested images, the size of the images are same in the both networks. BPNN topology used is more accurate than the LAM network topology used in recognition because of its recognition percentages for the iris tested images, but it need more execution time for learning.

6.5 Results

The simulation results of the Iris recognition system are obtained using LAM and BPNN feature extraction techniques. I am going to perform the simulations in three different experiments.

In first experiment, 138 images of 23 persons are taken. In image database each person has 6 different images. The number of noisy tested images in this experiment is 46. Using LAM 40 images was recognized successfully and recognition rate was 86.9%. Using BPN 42 images was recognized successfully and recognition rate was 91.3%.

In the second experiment, 231 images are taken. In database these images are belonging to 33 persons. Each person has 7 different images. Number of noisy tested images that is used in this experiment was 99. Using LAM, 91 images was recognized successfully with recognition accuracy 91.9%. Using BPNN, 92 images was recognized successfully with recognition accuracy 92.9%.

In the third experiment, 320 images are taken. These images are belonging to 40 persons and each person has 8 different images. Number of noisy tested images that are used in this experiment was 80. Using LAM, 71 images was recognized successfully with recognition accuracy 88.7%. Using BPNN, 75 images was recognized successfully with recognition accuracy 93.8%.

Table 6.2: recognition rates of the system for IRIS tested images.

Subjects	Tested Iris	Irises in database	LAM		BPNN	
			R.R	E.R	R.R	E.R
23	46	138	86.9%	13.1%	91.3%	8.7%
33	99	231	91.9%	8.1%	92.1%	7.9%
40	80	320	88.7%	11.3%	99.5%	0.5%

6.6 Comparing with recent researches

1) Comparing with the Neural Network

Now-a-days there are large numbers of researches are on the internet for the identification of irises, and one of them is neural network. It uses the CASIA database which is one of the largest database available in the public domain. The database contains 756 images of 108 different persons. Experiments are performed in two different stages iris segmentation and iris identifications [33].

At first stage rectangular area algorithm is used for the localization of the iris. The average time for the detection of inner and outer boundary of the iris images was 0.14sec. The accuracy was 98.62%. In the meantime the proposed system used the darkness density pupil and the average time for the detection of inner and outer boundary was 7sec and the accuracy rate was 97%.

At the second stage the recognition of irises are performed using the Neural Network techniques. 50 persons images are selected for the iris database for the classification. The detected irises after normalization and enhancements are scaled by using averaging. This help to reduced the size of the neural network. The images are represented by the matrices. The matrices are the input signal for the neural network. The output of the neural network are classes of iris pattern. For each set of iris images, the two patterns are used for the training and other two are used for the testing. The recognition rate for the Neural Network is 99.25%. In the same time, the proposed system uses the same CASIA database. 50 persons images are taken and each has four different images, two of them are used for the testing and two of them are used for the identifications. The algorithm used for the proposed system shows the result of 99.50%.

Table 6.3 Accuracy rate for iris segmentation

Methodology	Accuracy rate	Average time
Daughman	57.7%	90s
Wildes	86.49%	110s
Neural network	98.62%	0.14s
Proposed	97%	7s

Table 6.4 The recognition performance of comparing with existing methods

Methodology	Accuracy
Boles	92.64%
Liman	94.9%
Neural network	99.25%
Proposed	99.50%

2) Comparing with Gabor Wavelet method

In my second comparison, the proposed system is compared with the Gabor Wavelet method. This method also took the images from the CASIA. Before there was lot of problems for the iris images, and due to lack of iris images and to promote the research, National Laboratory of Pattern Recognition (NLPR), Institute of Automation (IA), Chinese Academy of Sciences (CAS) they provide iris database freely for the iris recognition researches. Iris recognition is one of very active research topic due to its high accuracy and uniqueness. In every iris recognition method first you have to segment the iris, and in Gabor Wavelet method, the Daugman's method is used for the segmentation of the iris, and in the proposed system, the darkness of the pupil is used for the segmentation of the iris [34].

Next step is the identification of the iris using Gabor Wavelet method. 10 persons images are selected from the iris database for the classification, and every persons have six different images. The top three images are used for the testing and other three are used for the pattern recognition. They use the Daugman's methods to iris regions segmentation and use Gabor wavelet for feature extraction. At last, in the identification stage they calculate Hamming distance between a test image & a training image. The smallest distance among them is expressed, that test image belongs to this class. The recognition rate is showed in table 6.5. And in the same time proposed system uses the images of 10 different persons and every person has 4 images of eye. In proposed system the top 2 images are used for testing and the next two images are used for the training purpose

Table 6.5: Recognition performance with Gabor Wavelet research

Methodology	Accuracy
Gabor wavelet	96.5%
Proposed system	99.50%

Table 6.6: Final parameters of Back Propagation Memory Neural Network

False Acceptance rate	10%
False Rejection rate	10%
Learning rate	0.2
Momentum rate	0.9
Recognition rate	90%
Error rate	0.00001
Speed	35 sec

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this work, an iris recognition system has been designed. The system hardware consists of Digital video camera (type: Sony DCR-TRV265E) with alumina box holding by a stand carrier and complete computer system (having speakers, SVGA card, USB controller, etc.), and some of images I took online.

The software built program consists of two stages, image processing stage for getting an enhancement iris pattern image and neural network stage for recognition by utilizing the data of the iris pattern. The neural network stage consists of two phases: training phase for human iris identifying and testing phase for deciding whether the human iris exists on the database or not.

Two methods of neural network have been used in this thesis LAM method and BPNN method. The latter yielded the best results compared with the other method.

NIR light is very important to clarify the details of iris pattern, and the white light is important for constricting the pupil to its narrow size. Dealing with JPEG type of pictures proved to be successful. Getting the iris location is performed by following the darkness density of pupil. Data equalization is obtained by using histogram equalization at the beginning and at the end of image analysis. Iris image can be converted to the matrix for STD values instead of pixels intensity. STD values attained good results. ANN has fast mathematical methods for identification. Back propagation Neural Network can be used for iris pattern classification. Momentum technique with multiple activation function in Back Propagation has advantages in speed and accuracy. LAM network is a fast way for identification but it has high rate of percentage error compared with the other employed methods. The iris recognition system designed is general, easy to use, fast and compatible

with different computers. The proposed system compared with many other recent researches and this system achieved higher recognition and speed.

7.2 Future work

There are some ideas for future work and recommendations to improve this research, the following points display these ideas:

- The system may be interfaced with control system on the organization entrance door.
- Identification report can be created for each tested person by the system. This report may have: name, job, address, ... etc.
- Multi-online persons recognitions system can be designed.
- The iris picture can be captured at suitable distance by using high performance camera.
- Iris and retina patterns recognition can be designed and implemented in one combinational system.

REFERENCES

- [1] R. A. Mohammed, N. B. Ibrahim, A. S. Nori, "Iris Identification Software (IIS)", Raf. Jour. Sci., Vol. 15, No. 1Comp., Math. & Stati., Special Issue, pp. 1-10, 2004.
- [2] Y. Zhu, T. Tan, Y. Wang, "Biometric Personal Identification Based on Iris Patterns", Proc. Int. Conf. on Pattern Recognition, Vol. 2, pp. 801-804, 2000.
- [3] R. Roizenblatt, P. Schor, F. Dante, J. Roizenblatt, R. Belfort, "Iris recognition as a biometric method after cataract surgery", BioMedical Engineering OnLine, 3:2, pp. 1-7, 2004.
- [4] Z. Ríha, V. Matyáš, "Biometric Authenrication Systems", Faculty of Informatics Masaryk University (FIMU), FIMU-RS-2000-08.
- [5] C. Tisse and et. al. "Person identification technique using human iris recognition",
- [6] P. Rosenzweig, A. Kochems, A. Schwartz, "Biometric Technologies: Security, Legal, and Policy Implications", Published by The Heritage Foundation, Legal Memorandum No. 12, June 2004.
- [7] Dory L. Hudspeth, Sarah J. Spinks, "Iris Scan Technology", Faulkner Information Services, Publication Date: 0109, Publication Type: TUTORIAL, 2001.
- [8] J. Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE transactions on pattern analysis and machine intelligence. Vol. 15, No. 11, November 2003.
- [9] J. Daugman, "How Iris Recognition Works", IEEE transactions on circuits and systems for video technology, Vol. 14, No.1, January 2004.
- [10] J. Daugman, C. Downing, "Epigenetic randomness, complexity and singularity of human iris patterns", The Royal Society, Proc. R. Soc. Lond. B (2001) 268, 1737-1740.
- [11] J. Daugman, "The importance of being random: statistical principles of iris recognition", The journal of the pattern recognition society, Pattern Recognition 36 (2003), 279 – 291.
- [12] T. Burghardt, "Inside Iris Recognition", Master course in global computing and multimedia, university of Bristol, November 2002.

- [13] K. W. Nam, K. L. Yoon, J. S. Bark, W. S. Yang, "A feature extraction method for binary Iris code construction", Proceedings of the 2nd International Conference on Information Technology for Application, ICITA 2004.
- [14] H. A. Al-Sumaidae, "Off-Line Handwritten Signature Verification Using Backpropagation Neural Network", M. Sc. Thesis, Technical college, Mosul/Iraq, January 2005.
- [15] M. Dunker, "Don't blink: Iris recognition for biometric identification", SANS institute as part of GIAC practical repository, pp.1-17, 2004.
- [16] J. Daugman, "How Iris Recognition Works", p. 6 Daugman, John Homepage 7, November 2003.
- [17] R. Amer, "Design a software application for Iris Identification by Artificial Neural Network", M. Sc. Thesis, Computers and Mathematical Sciences College, University of Mosul/Iraq, June 2001.
- [18] W. Y. Chang, "Image Processing with Wreath Products", M. Sc. Thesis, Harvey Mudd college, Department of Mathematics, May 2004.
- [19] S. W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing", California Technical Publishing, ISBN 0-9660176-3-3, Ch. 23, Second Edition, 2009.
- [20] The MathWorks Inc., "Image Processing Toolbox For Use with MATLAB", Ver. 6.5, 2008, MA, USA.
- [21] F. patina, "An introduction to digital image processing", homepage yov408, pp. 1-49, July 2003.
- [22] S. E. Umbaugh, "Computer vision and image processing: A practical approach using CVIPtools", Prentice Hall PTR, 2008.
- [23] R. Fisher, S. Perkins, A. Walker and E. Wolfart, "Laplacian / Laplacian of Gaussian", 2003:
- [24] S. Maerivoet, "An introduction to image enhancement in the spatial domain", University of Antwerp, Department of mathematics and computer science, November 2000.

- [25] J. S. Lim, "Two-dimensional signal and image processing", Prentice Hall signal processing series, 2009.
- [26] N. Shindala, "Speaker Recognition using neuro-fuzzy method", M. Sc. Thesis, Computer Engineering college, university of Mosul/Iraq, 2004.
- [27] E. Gelenbe, H. U., "Lecture Notes on Neural Networks", METU, 2004.
- [28] The Math Works Inc., "Neural Network Toolbox, For Use with MATLAB", Ver. 6.5, 2008, MA, USA.
- [29] L. Fausett, "Fundamental of Neural Networks, Architectures, Algorithms and Applications", Printice Hall Int. Snc., 2004.
- [30] E. Istook, T. Martinez, "Improved backpropagation learning in neural networks with windowed momentum", International journal of neural systems, vol. 12, no. 3&4, pp. 303-318, 2002.
- [31] J. C. Russ, "Seeing the Scientific Image", Proceedings RMS, Vol. 39/2, June 2004.
- [32] P. M. Mather, "Computer processing of remotely-sensed images", Paul M. Mather, British Library, 2007.
- [33] : W. W. Boles and B. Boashash, "A human identification technique using images of the iris and wavelet transform," IEEE Trans. Signal Process., vol. 46, no. 4, pp. 1185–1188, Apr. 2008
- [34] L. Ma, T. Tan, Y. Wang, and D. Zhang, "Personal identification based on iris texture analysis," IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 12, pp. 1519–1533, Dec. 2003
- [35]: L. Yu, D. Zhang, and K. Wang, "The relative distance of key point based iris recognition," Pattern Recognit., vol. 40, no. 2, pp. 423–430, Feb. 2007.
- [36] Seongwon Cho*, Jaemin Kim School of Electronic and Electrical Engineering, Hongik University 72-1 Sangsu-dong, Mapo-gu, Seoul 121-791, Korea Department of Computer Engineering, Seokyeong University,

- [37] S. Lim, K. Lee, O. Byeon, and T. Kim, "Efficient Iris Recognition through Improvement of Feature Vector and Classifier", ETRI Journal, Vol.23, No.2, June, 2009.
- [38] J. Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence".IEEE Transl. on Pattern Analysis and Machine Intelligence, Vol.15, issue 11, 2003.

APPENDIX A

SOURCE CODE

```
function [inputImg,CurrentPage,Feature,f1] =test()
fig = figure(...
'Units','characters',...
'Color',[0.925490196078431 0.913725490196078 0.847058823529412],...
'IntegerHandle','off',...
'InvertHardcopy',get(0,'defaultfigureInvertHardcopy'),...
'MenuBar','none',...
'Name','IRIS recognition using image processing and neural network',...
'NumberTitle','off',...
'PaperPosition',get(0,'defaultfigurePaperPosition'),...
'Position',[50.8 10 132.4 39.3846153846154],...
'Resize','off',...
'UseHG2','off',...
'HandleVisibility','callback',...
'Tag','figure1',...
'UserData',[],...
'Visible','on');

lblHeading = uicontrol(...
'Parent',fig,...
'Units','characters',...
'BackgroundColor',[0.584313725490196 0.388235294117647 0.388235294117647],...
'FontSize',15,...
'FontWeight','bold',...
'ForegroundColor',[1 1 1],...
'HorizontalAlignment','left',...
'Position',[-0.2 36.7692307692308 132.4 2.53846153846154],...
'String',' Iris Recognition',...
'Style','text',...
'Tag','text1');

pnlMain = uipanel(...
'Parent',fig,...
'Units','characters',...
'Title','Iris Segmentation',...
'Tag','uipanel1',...
'UserData',[],...
'Clipping','off',...
'Position',[2.6 11.4615384615385 127.4 24]);

lblInputImage = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'CDATA',[],...
'Position',[2 1.92307692307692 38.2 1.38461538461538],...
```

```
'String','Input Image',...
'Style','text',...
'Tag','text2',...
'UserData',[]);
```

```
lblSegmentedImage = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'CDATA',[],...
'Position',[43.4 1.69230769230769 38.2 1.38461538461538],...
'String','Segmented Image',...
'Style','text',...
'Tag','text3',...
'UserData',[]);
```

```
imgSegmentedImage = axes(...
'Parent',pnlMain,...
'Units','characters',...
'Position',[42.8 3.61538461538462 40 16.4615384615385],...
'CameraPosition',[0.5 0.5 9.16025403784439],...
'CameraPositionMode',get(0,'defaultaxesCameraPositionMode'),...
'Color',get(0,'defaultaxesColor'),...
'ColorOrder',get(0,'defaultaxesColorOrder'),...
'LooseInset',[16.458 2.32692307692308 12.027 1.58653846153846],...
'XColor',get(0,'defaultaxesXColor'),...
'YColor',get(0,'defaultaxesYColor'),...
'ZColor',get(0,'defaultaxesZColor'),...
'Tag','axes3');
```

```
imgInputImage= axes(...
'Parent',pnlMain,...
'Units','characters',...
'Position',[1 3.61538461538462 40 16.4615384615385],...
'CameraPosition',[0.5 0.5 9.16025403784439],...
'CameraPositionMode',get(0,'defaultaxesCameraPositionMode'),...
'Color',get(0,'defaultaxesColor'),...
'ColorOrder',get(0,'defaultaxesColorOrder'),...
'LooseInset',[16.458 2.32692307692308 12.027 1.58653846153846],...
'XColor',get(0,'defaultaxesXColor'),...
'YColor',get(0,'defaultaxesYColor'),...
'ZColor',get(0,'defaultaxesZColor'),...
'Tag','axes1');
```

```
lblMatchedIris = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'CDATA',[],...
'Position',[85 1.92307692307692 38.2 1.38461538461538],...
'String','Matched Iris',...
'Style','text',...
```

```
'Tag','text4',...
'UserData',[]);
```

```
imgMatchedIris = axes(...
'Parent',pnlMain,...
'Units','characters',...
'Position',[84.8 3.61538461538462 40 16.4615384615385],...
'CameraPosition',[0.5 0.5 9.16025403784439],...
'CameraPositionMode',get(0,'defaultaxesCameraPositionMode'),...
'Color',get(0,'defaultaxesColor'),...
'ColorOrder',get(0,'defaultaxesColorOrder'),...
'LooseInset',[16.458 2.32692307692308 12.027 1.58653846153846],...
'XColor',get(0,'defaultaxesXColor'),...
'YColor',get(0,'defaultaxesYColor'),...
'ZColor',get(0,'defaultaxesZColor'),...
'Tag','axes2');
```

```
cmdChoose = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'Callback','iris_template_export("pushbutton6_Callback",gcbo,[],guidata(gcbo))',...
'Position',[3 20.0769230769231 36 2.15384615384615],...
'String','Choose',...
'Tag','pushbutton6',...
'Callback',@cmdChoose_callback);
```

```
cmdSegment = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'Callback','iris_template_export("pushbutton7_Callback",gcbo,[],guidata(gcbo))',...
'Position',[44.6 20.0769230769231 36 2.15384615384615],...
'String','Segment Iris',...
'Tag','pushbutton7',...
'Callback',@cmdSegment_callback);
cmdClassify = uicontrol(...
'Parent',pnlMain,...
'Units','characters',...
'Callback','iris_template_export("pushbutton8_Callback",gcbo,[],guidata(gcbo))',...
'Position',[86.2 20.0769230769231 36 2.15384615384615],...
'String','Classify',...
'Tag','pushbutton8',...
'Callback',@cmdClassify_callback);
```

```
pnlResult = uipanel(...
'Parent',fig,...
'Units','characters',...
```

```

'Title','Matching',...
'Tag','uipanel2',...
'Clipping','on',...
'Position',[2.8 2.46153846153846 127 8.53846153846154]);

```

```

lblIrisNumber = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[2.2 4.92307692307692 24.6 1.61538461538462],...
'String','Iris Number',...
'Style','text',...
'Tag','text9',...
'UserData',[]);

```

```

lblPersonInformation = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[2.2 2.76923076923077 24.6 1.61538461538462],...
'String','Person Information',...
'Style','text',...
'Tag','text10',...
'UserData',[]);

```

```

lblImageName = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[2.2 0.846153846153846 24.6 1.61538461538462],...
'String','Image Name',...
'Style','text',...
'Tag','text11',...
'UserData',[]);

```

```

lblIrisNumberValue = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[22.2 4.92307692307693 24.6 1.61538461538462],...
'String','2',...
'Style','text',...
'Tag','text12',...
'UserData',[]);

```

```

lblPersonInformationValue = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[22.2 2.76923076923077 24.6 1.61538461538462],...
'String','Input Image',...
'Style','text',...
'Tag','text13');

lblImageNameValue = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'CData',[],...
'HorizontalAlignment','left',...
'Position',[22.2 0.846153846153846 24.6 1.61538461538462],...
'String','Input Image',...
'Style','text',...
'Tag','text14');

lblResult = uicontrol(...
'Parent',pnlResult,...
'Units','characters',...
'BackgroundColor',[0.0784313725490196 0.168627450980392 0.549019607843137],...
'CData',[],...
'FontSize',14,...
'ForegroundColor',[1 1 1],...
'HorizontalAlignment','left',...
'Position',[66.8 3.84615384615385 38.6 2.15384615384615],...
'String','- ',...
'Style','text',...
'Tag','text15',...
'UserData',[]);

function cmdChoose_callback(hObj,eventdata)
    clear;
    f1= uigetfile({'*.jpg;*.bmp'},'Select the Image file');
    inputImg=imread(f1);
    subplot(imgInputImage);imagesc(inputImg);colormap(gray); axis off;
end

function cmdSegment_callback(hObj,eventdata)
    eyeimage=inputImg;
    [circleiris circlepupil imagewithnoise,fcmap_a,fcmap_b] = segmentiris(eyeimage);
    imagewithcircles = uint8(eyeimage);

    %get pixel coords for circle around iris
    [x,y] = circlecoords([circleiris(2),circleiris(1)],circleiris(3),size(eyeimage));
    ind2 = sub2ind(size(eyeimage),double(y),double(x));

    %get pixel coords for circle around pupil
    [xp,yp] = circlecoords([circlepupil(2),circlepupil(1)],circlepupil(3),size(eyeimage));

```

```

ind1 = sub2ind(size(eyeimage),double(yp),double(xp));

innerboundary=imagewithcircles;
outerboundary=imagewithcircles;
segment=imagewithcircles;
innerboundary(ind1) = 255;
outerboundary(ind2) = 255;
x_size=size(eyeimage,1);
y_size=size(eyeimage,2);

xcent=circleiris(1);
ycent=circleiris(2);
radius=circleiris(3);
xcent1=circlepupil(1);
ycent1=circlepupil(2);
xc=round(1000*rand(1));
xc1=round(1000*rand(1));
fcm_a1=fcm_a(1:52,xc:xc+600,1:3);
fcm_b1=fcm_b(1:104,xc1:xc1+600,1:3);
for i=1:x_size
    for j=1:y_size
        inside =(i-xcent).^2 + (j-ycent).^2 < radius^2 ;
        inside_p =(i-xcent1).^2 + (j-ycent1).^2 < circlepupil(3)^2 ;
        if inside==0
            segment(i,j)=0;
        end
        if inside_p==1
            segment(i,j)=0;
        end
    end
end

segment(ind1) = 255;
segment(ind2) = 255;
% Write circles overlayed
%imagewithcircles(ind2) = 255;

subplot(imgSegmentedImage);imagesc(segment);colormap(gray); axis off;
Feature=segment;
end
function cmdClassify_callback(hObj,eventdata)
    [template, mask] = createiristemplate(f1);
    if (exist('iris_database.dat')==2)
        load('iris_database.dat','-mat');
        iris_number=iris_number;
        best_matching=zeros(iris_number,1);

        myfile=load('mynnet.mat','-mat');
        for scanning=1:iris_number

```

```

temp = data{scanning,1};
mas = data{scanning,2};

hd=gethammingdistance(temp, mas, template, mask, 2);
best_matching(scanning)=hd;
end
[distanza_minima,posizione_minimo]=min(best_matching)
if (distanza_minima<2)
    mynnet=myfile.mynnet;

    test1=template(1:3)';
    result=sim(mynnet,test1)
    if(result>distanza_minima)
        [distanza_minima,posizione_minimo]=min(best_matching);
    end

    set(lblIrisNumberValue,'String',int2str(posizione_minimo));
    set(lblPersonInformationValue,'String',informations{posizione_minimo});
    set(lblImageNameValue,'String',files{posizione_minimo});
    resImage=imread(files{posizione_minimo});
    subplot(imgMatchedIris);imagesc(resImage);colormap(gray); axis off;
    set(lblResult,'String','Subject Authenticated');
else
    set(lblResult,'String','Subject Not Authenticated');
end

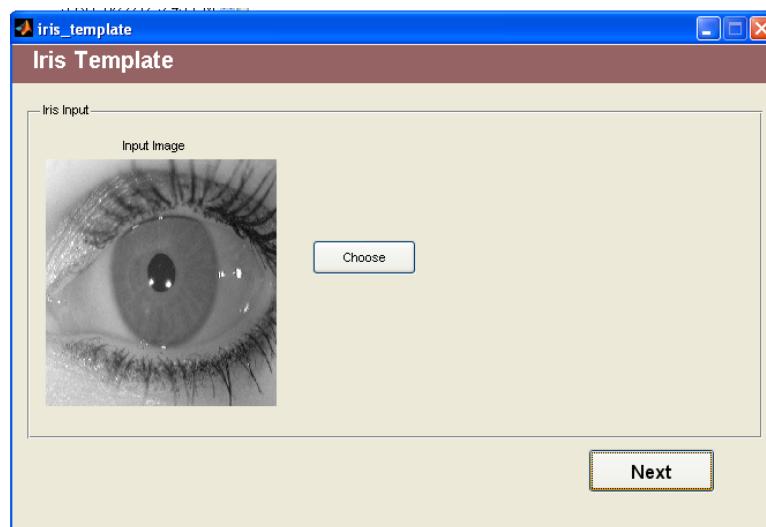
else
    hinfo = msgbox('Database is Empty');
end
end
end

```

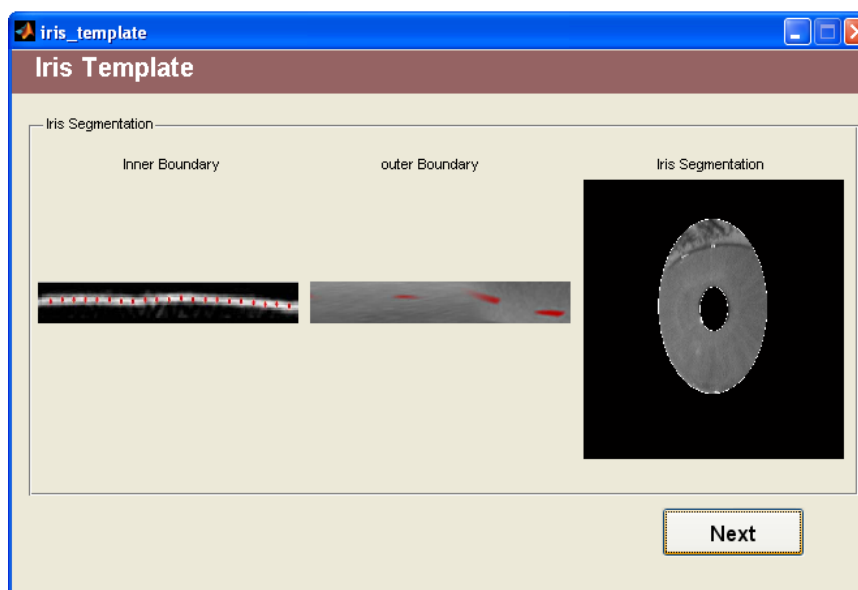
APPENDIX B

SCREEN SHOTS

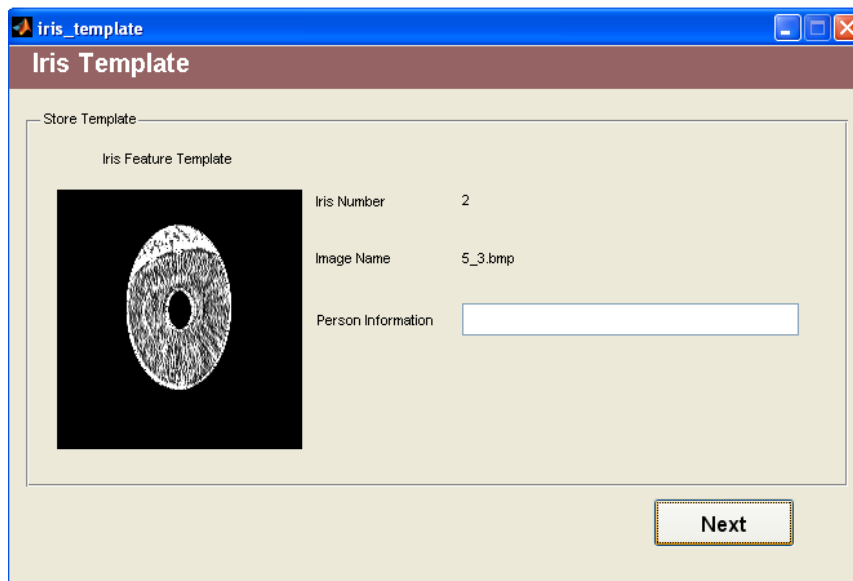
1. Input Iris Image



2. Iris Segmentation



3.Store Iris Template



4. Iris Recognition

