

HYBRID ROBUST IRIS RECOGNITION APPROACH USING IRIS IMAGE PRE-PROCESSING, TWO-DIMENSIONAL GABOR FEATURES AND MULTI-LAYER PERCEPTRON NEURAL NETWORK/PSO

Seminar Report

*Submitted in partial fulfillment of the requirements for
the award of degree of*

BACHELOR OF TECHNOLOGY

In

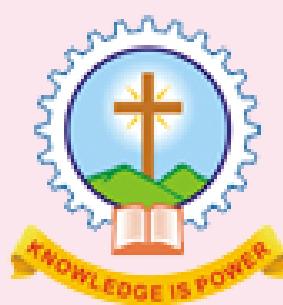
COMPUTER SCIENCE AND ENGINEERING

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted By

CHRISTO JOBY



Department of Computer Science & Engineering
Mar Athanasius College Of Engineering Kothamangalam

HYBRID ROBUST IRIS RECOGNITION APPROACH USING IRIS IMAGE PRE-PROCESSING, TWO-DIMENSIONAL GABOR FEATURES AND MULTI-LAYER PERCEPTRON NEURAL NETWORK/PSO

Seminar Report

*Submitted in partial fulfillment of the requirements for
the award of degree of*

BACHELOR OF TECHNOLOGY

In

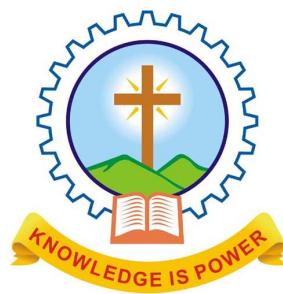
COMPUTER SCIENCE AND ENGINEERING

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

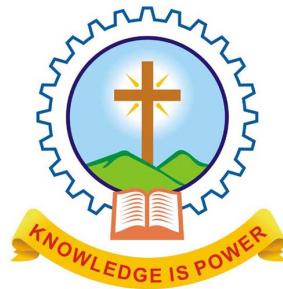
Submitted By

CHRISTO JOBY



Department of Computer Science & Engineering
Mar Athanasius College Of Engineering Kothamangalam

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING
KOTHAMANGALAM**



CERTIFICATE

*This is to certify that the report entitled **Hybrid robust iris recognition approach using iris image pre-processing, two-dimensional gabor features and multi-layer perceptron neural network/PSO** submitted by Mr. CHRISTO JOBY, Reg. No. MAC15CS022 towards partial fulfilment of the requirement for the award of Degree of Bachelor of Technology in Computer science and Engineering Engineering from APJ Abdul Kalam Technological University for December 2018 is a bonafied record of the work carried out by him under our supervision and guidance.*

.....

Prof. Joby George
Faculty Guide

.....

Prof. Neethu Subash
Faculty Guide

.....

Dr.Surekha Mariam Varghese
Head of the Department

Date:

Dept. Seal

ACKNOWLEDGEMENT

First and foremost, I sincerely thank the God Almighty for his grace for the successful and timely completion of the seminar.

I express my sincere gratitude and thanks to Dr. Solly George, Principal and Dr. Surekha Mariam Varghese, Head Of the Department for providing the necessary facilities and their encouragement and support.

I owe special thanks to the staff-in-charge Prof. Joby George, Prof. Neethu Subash and Prof. Joby Anu Mathew for their corrections, suggestions and sincere efforts to co-ordinate the seminar under a tight schedule.

I express my sincere thanks to staff members in the Department of Computer Science and Engineering who have taken sincere efforts in helping me to conduct this seminar.

Finally, I would like to acknowledge the heartfelt efforts, comments, criticisms, co-operation and tremendous support given to me by my dear friends during the preparation of the seminar and also during the presentation without whose support this work would have been all the more difficult to accomplish.

ABSTRACT

Biometric technology has brought new recognition systems to identify individuals both physically (iris, veins, face, fingerprints, and palm prints) and behaviorally (gait, voice, and signature). The traits are applicable for detecting crime in security systems because they offer attractive features such as stability and uniqueness. Many methods are already present which feature shortcomings such as computational complexity, long run times, and high memory consumption remain. This paper shows an ideal human iris recognition approach based on a multi-layer perceptron neural network and practical swarm optimization (PSO) algorithm to train the network in order to increase generalization performance. A combination of these algorithm was used as a classifier. A pre-processing step is performed to the iris images to improve the results and two-dimensional gabor kernel feature extraction is applied. The data is normalized, trained, and tested using the proposed method. A practical swarm optimization algorithm is applied to train the Neural Network for data classification.

Contents

Acknowledgement	i
Abstract	ii
List of Figures	iv
List of Abbreviations	v
1 Introduction	1
2 Related works	6
2.1 Iris image pre-processing	6
2.2 2D Gabor kernel method	11
2.3 Pattern matching	13
3 The proposed method	18
4 Conclusion	23
References	24

List of Figures

Figure No.	Name of Figures	Page No.
1.1	Example of an iris pattern, imaged monochromatically at a distance of about 35 cm.	1
1.2	Implementations in Security systems	2
1.3	Example of an iris security system in an airport	5
2.1	Detection of circular boundaries of pupil and iris.	7
2.2	Circular Hough Transform of four points on a cirlce	7
2.3	Circular Hough Transform of four points on 3 cirlces	8
2.4	Iris normalisation (a) Original image, (b) Noise removal, (c) Polar coordinate, (d) Polar coordinate noise	8
2.5	Daugman's rubber-sheet model	9
2.6	Image segmentations	10
2.7	Iris Anatomy	11
2.8	Gabor Filter Bank, (a) Gabor kernel filter bank, (b) Result of 2D gabor kernel, (c) Enlarged view of (b)	12
2.9	Each circular node represents an artificial neuron and an arrow represents a connection from the output of one artificial neuron to the input of another.	14
2.10	A particle swarm searching for the global minimum of a function	16
3.1	Steps for Recognition that were followed.	18
3.2	Block diagram of iris recognition based on the MLPNN PSO method	20
3.3	Basic Multi-Layer Perceptron Neural Network	21
3.4	EER	22

LIST OF ABBREVIATION

ANN	Artificial NN
PSO	Particle Swarm Optimisation
TSP	Travelling Salesman Problem
ICM	Intersecting Cortical Model
MLPNN	Multi-Layer Perceptron Network
SOM	Self-Organising Map
RBF	Radial Basis Function
GA	Genetic Algorithm
HVS	Human Visual System
ROI	Region Of Interest
UCI	University of California at Irvine
MSE	Mean Square Error
MMSE	Minimum Mean Square Error
CMC	Cumulative Match Scores Characteristics
FMNN	Functional Modular NN
PCA	Principal Component Analysis
DWT	Discrete Wavelet Transform
BPNN	Back Propagation NN

PNN	Probabilistic NN
WPNN	Wavelet PNN
ACFO	Adaptive Central Force Optimisation
MDANN	Multi-Dimensional Artificial NN
GAR	Genuine Accepted Ratio
EER	Equal Error Rate
FAR	False Accept Rate
FRR	False Reject Rate
ROC	Receiver Operating Characteristic

Introduction

Reliable automatic recognition of persons has long been an attractive goal. As in all pattern recognition problems, the key issue is the relation between inter-class and intra-class variability: objects can be reliably classified only if the variability among different instances of a given class is less than the variability between different classes. For example, in face recognition, difficulties arise from the fact that the face is a changeable social organ displaying a variety of expressions, as well as being an active three-dimensional (3-D) object whose image varies with viewing angle, pose, illumination, accoutrements, and age. It has been shown that, for mug shot images taken at least one year apart, even the best current algorithms can have error rates of 43% – 50%. Fig 1.1 shows the outline overlay shows results of the iris and pupil localization and eyelid detection steps. The bit stream in the top left is the result of demodulation with complex-valued two-dimensional (2-D) Gabor wavelets to encode the phase sequence of the iris pattern.. Against this intra-class (same face) variability, inter-class variability is limited because different faces possess the same basic set of features, in the same canonical geometry.

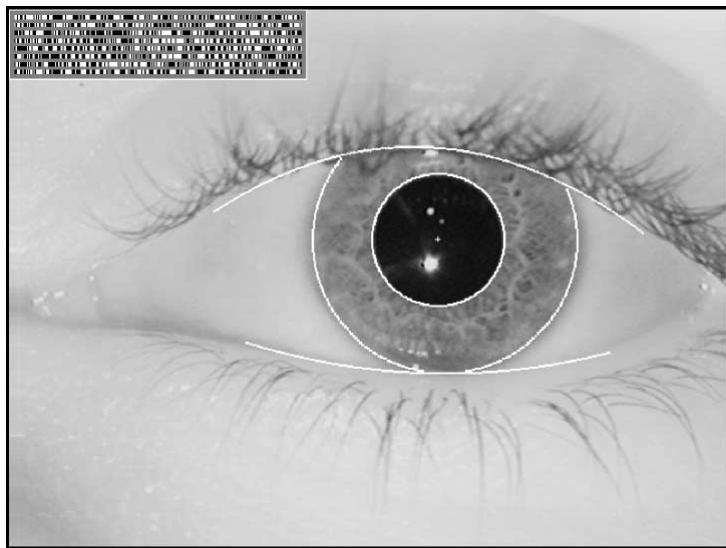


Fig. 1.1: Example of an iris pattern, imaged monochromatically at a distance of about 35 cm.

Machine intelligence algorithms such as machine learning, evolutionary computing, and neural networks (NNs) have made progress in soft computing, computer vision, and pattern recognition. The recent increase in the number of terrorist attacks and crime and the use of e-commerce means that heightened security measures are required [1]. The use of iris recognition can be seen in Security Systems. Biometric technology has brought new recognition

systems to identify individuals both physically (iris, veins, face, fingerprints, and palm prints) and behaviourally (gait, voice, and signature) [2]. All of these methods are useful approaches, but iris recognition is best. Intelligent systems with fewer coefficient errors and high reliability are required for iris recognition [3].

The diaphragm of the human iris is a circular area set between the cornea and lens of the eye [4]. The diameter of the iris is about 12 mm and the size of the pupil varies from 10% to the more common 80% of the iris [5]. The stability of iris texture throughout a lifetime makes it a proper object for discrimination. It is also an internal organ which can be clearly observed with the naked eye and has a unique physiological structure. Biometric systems can be used for mobile phones, medical testing, computer systems security, credit cards, secure access to buildings, social services, and secure electronic banking [4]. One of the most significant biometric recognition systems is iris recognition, which can identify a person using the features of his/her iris [6].



Fig. 1.2: Implementations in Security systems

It is known that iris texture is an effective means of identifying an individual [6]. Numerous methods have been developed for this such as that by Tallapragada and Rajan [7], who proposed a combination of the grey-level co-occurrence matrix (GLCM) and Haar wavelet features to enhance iris recognition.

1.0.1 Gray level co-occurrence matrix

A co-occurrence matrix or co-occurrence distribution is a matrix that is defined over an image to be the distribution of co-occurring pixel values (grayscale values, or colors) at a given offset. The offset, $(\Delta x, \Delta y)$, is a position operator that can be applied to any pixel in the image (ignoring edge effects): for instance, $(1, 2)$ could indicate "one down, two right". An image with p different pixel values will produce a pp co-occurrence matrix, for the given offset. The $(i, j)^{th}$ value of the co-occurrence matrix gives the number of times in the image that the i^{th} and j^{th} pixel values occur in the relation given by the offset. For an image with p different pixel values, the pp co-occurrence matrix C is defined over an nm image I , parameterized by an offset $(\Delta x, \Delta y)$, as:

$$C_{\Delta x, \Delta y}(i, j) = \sum_{x=1}^n \sum_{y=1}^m \left\{ \begin{array}{ll} 1, & \text{if } I(x, y) = i \text{ and } I(x + \Delta x, y + \Delta y) = j \\ 0, & \text{otherwise} \end{array} \right\} \quad (\text{Equ: 1.1})$$

1.0.2 Previous generation of recognitions

The authors used features of frequency and spatial domain, but the accuracy rate was low. To address this issue and obtain a reasonable accuracy rate, the present study proposes the two-dimensional (2D) gabor kernel to extract features from the human iris to develop a novel model based on MLPNNPSO. The method is effective, low cost, and produces precise results in optimisation. There are no previous examples of applications of a combined MLPNN and PSO algorithm for iris recognition.

Daugman [8] developed 2D gabor filtration to encrypt iris patterns. Abiyev and Altunkaya [9] presented a novel iris recognition system based on NNs for identifying individuals biometrically and also proposed a fast algorithm to localise iris boundaries. Cho and Kim [10] used 2D wavelet transform based on Haar wavelets for feature extraction to recognise the human iris and applied Learning Vector Quantization (LVQ) NN for classification purposes. They applied the dimensional winner-selection approach. An LVQ system is represented by prototypes $W = (w(i), \dots, w(n))$ which are defined in the feature space of observed data. In winner-take-all training algorithms one determines, for each data point, the prototype which is closest to the input according to a given distance measure. The position of this so-called winner prototype is then adapted, i.e. the winner is moved closer if it correctly classifies the data point or moved away if it classifies the data point incorrectly.

LVQ is the supervised counterpart of vector quantization systems. LVQ can be understood as a special case of an artificial neural network, more precisely, it applies a winner-take-all Hebbian learning-based approach. It is a precursor to self-organizing maps (SOM) and related to neural gas, and to the k-nearest neighbor algorithm (k-NN). LVQ was invented by Teuvo Kohonen.

Srivastava et al. developed a human recognition system based on the fusion of evolu-

tionary fuzzy clustering with the Minkowski distance and then used functional modular NNs (FMNNs) for classification. Ye et al. proposed iris imaging in real-time pre-estimation based on a back propagation NN (BPNN) using multiple independent BPNNs to extract the overall and contour features and localise the iris image. They used various training weights to calculate the pre-estimation output. Ma et al. introduced a verification method based on gabor filtration and wavelet moments for iris feature extraction. They also employed principal component analysis (PCA) and BPNN to reduce dimensionality and for classification, respectively. Dhage presented a method based on discrete wavelet transform (DWT) feature extraction and binary PSO for feature selection and radon transforms to recognise iris patterns.

Nedjah et al. developed hardware architecture for implementing ANNs based on an MLP to decrease the processing time. Kennedy and Eberhart presented a primary binary version of the PSO. The travelling salesman problem (TSP) and permutation flow shop sequencing problem are different types of discrete PSO algorithms. The PSO algorithm is a swarmbased evolutionary approach presented by Eberhart and Kennedy that is an effective optimisation device.

To complete the proposed novel human iris recognition system, the current study extended the ANN-based MLP approach and then developed a PSO algorithm. The proposed method and related algorithms were then compared. The following differences were identified for the proposed approach as opposed to the previous research discovered in the literature review:

- A new hybrid recognition approach using MLPNN and PSO is proposed that will increase the confirmed accuracy of iris recognition.
- The 2D gabor kernel is used on normalised images to extract iris features.

The current paper proposes a method for the iris recognition domain, but the method can also be used for facial recognition application systems as developed by Ahonen et al. The proposed method and algorithm can be performed in any domain of iris recognition, as for example, in airports for anti-terrorism control, in border control for iris identification as a passport, shown in Fig. 1.3 for secure access to bank accounts in automated teller Machines and to log into a computer.

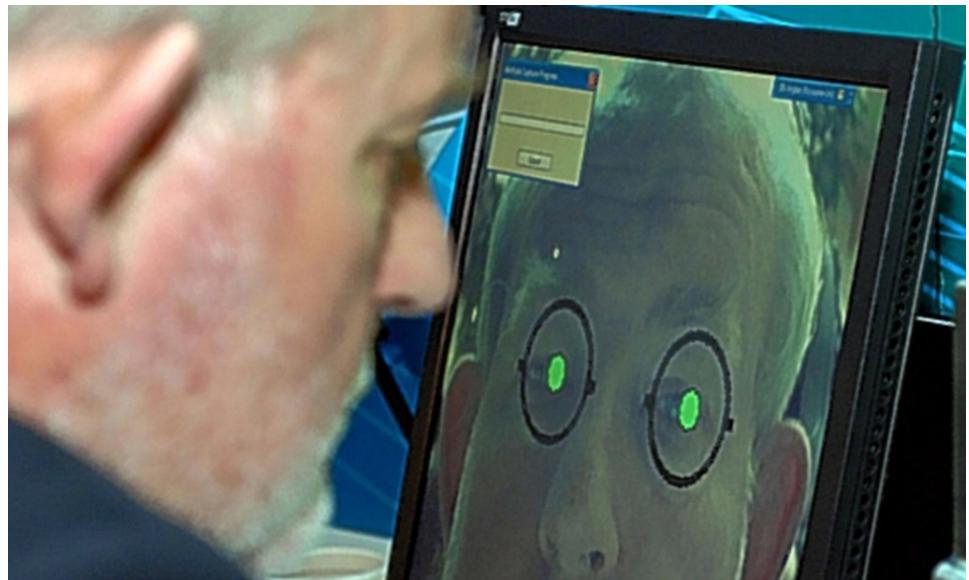


Fig. 1.3: Example of an iris security system in an airport

Related works

2.1 Iris image pre-processing

There are more people on Earth than ever before, owning more things, and swapping more information every single day. Security has never been more important but ironically, thanks to the computing power at everyone's disposal never easier to crack. Traditionally, security relies on things that are difficult to do quickly: locks are physically difficult to bust open without the correct metal keys, while information secured by encryption (computerized scrambling) is hard to access without the right mathematical keys. But this kind of security has a basic flaw: with the right key, even the wrong person can quickly gain access. Good performance for iris recognition requires high-quality images of the iris. The pre-processing steps comprise two phases: iris localisation and normalisation. These were performed on the iris images to obtain the region of interest (ROI) of the iris and minimise the noise of the images.

In the first phase, the iris localisation must be carried out without error. This step detects borders internally and externally then separates the sclera and pupil. A number of methods have been developed for this purpose. The present paper uses the circular Hough transform algorithm. In the second phase, the rubber-sheet model of Daugman was used for normalisation to convert the image from the space of the Cartesian coordinates to polar coordinates.

2.1.1 Iris localisation

The grey-scale format must apply in the captured image. The holes (regions in which dim pixels surround light pixels) that exist in a grey-level image must be identified. A canny edge detection operator must be used to make an edge map on a grey-level image. The circular Hough transform utilised by Wildes and Ahmadi was employed on specific areas on the edge image to assess the definite inner and outer iris circle parameters.

Canny edge detection is used to create an edge map[2]. The boundary of the iris is located by using canny edge detection technique. These parameters are the centre coordinates x and y, the radius r, which are able to define any circle according to the equation,

$$x^2 + y^2 = r^2 \quad (\text{Equ: 2.1})$$

2.1.2 Circular hough transform

The Hough transform is a feature extraction technique used in image analysis.

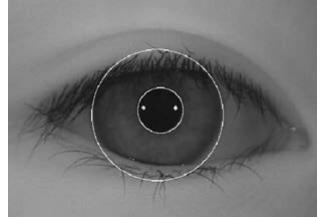


Fig. 2.1: Detection of circular boundaries of pupil and iris.

The Circle Hough Transform (CHT) is a feature extraction technique for detecting circles (Fig 2.1). It is a specialization of Hough Transform. The purpose of the technique is to find circles in imperfect image inputs. The circle candidates are produced by voting in the Hough parameter space and then select the local maxima in a so-called accumulator matrix. In a two-dimensional space, a circle can be described by:

$$(x - a)^2 + (y - b)^2 = r^2 \quad (\text{Eqn: 2.2})$$

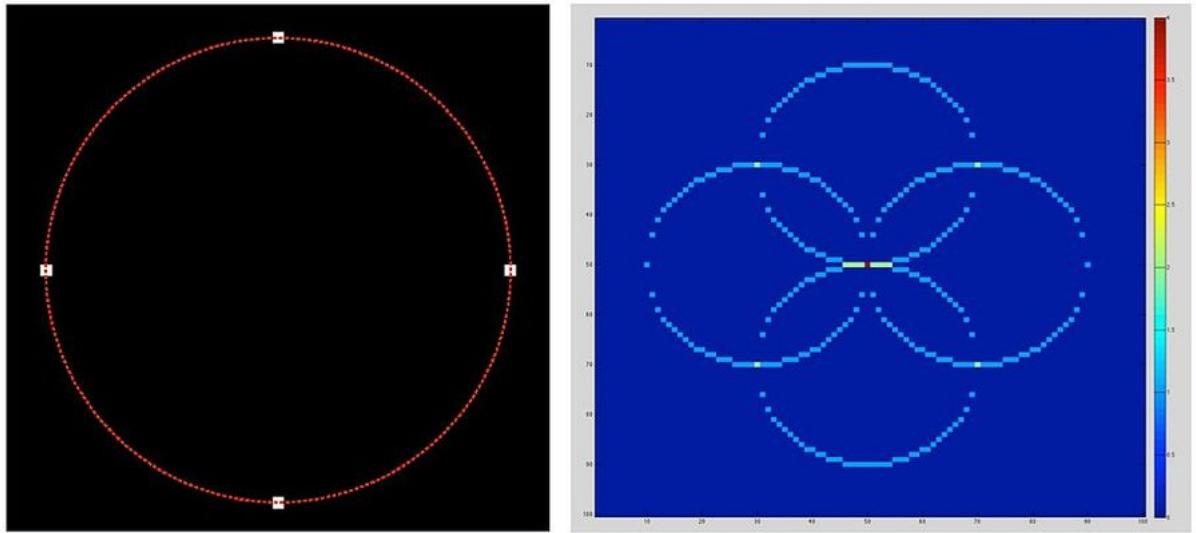


Fig. 2.2: Circular Hough Transform of four points on a circle

Consider 4 points on a circle in the original image (Fig 2.2 - left). The circle Hough transform is shown in the right. Note that the radius is assumed to be known. For each (x, y) of the four points (white points) in the original image, it can define a circle in the Hough parameter space centered at (x, y) with radius r . An accumulator matrix is used for tracking the intersection point. In the parameter space, the voting number of points through which the circle passing would be increased by one. Then the local maxima point (the red point in the center in the right figure) can be found. The position (a, b) of the maxima would be the center of the original circle.

For finding multiple circles using the same radius

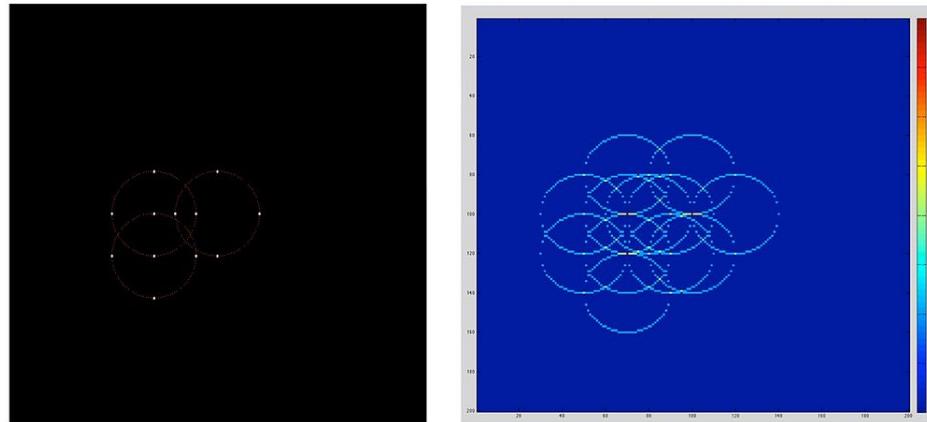


Fig. 2.3: Circular Hough Transform of four points on 3 cirlces

Note that, in the accumulator matrix (Fig. 2.3), there would be at least 3 local maxima points.

2.1.3 Image normalisation

Remapping to provide altered measurements of the iris image from Cartesian to polar coordinates is called normalisation. In the current paper, the ring of the iris was reformed to a rectangular square of composition having a stabilised size (64 512) (Fig. 2.4). Daugman's rubber-sheet model was employed for iris normalisation as shown in Fig. 2.5.

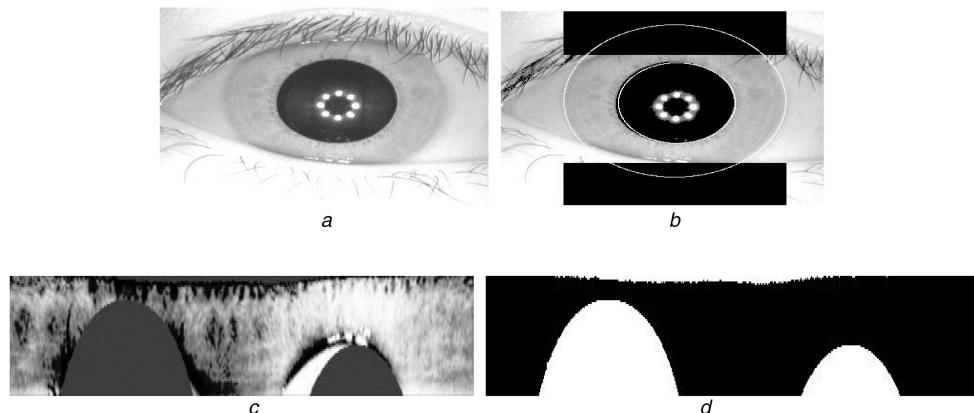


Fig. 2.4: Iris normalisation (a) Original image, (b) Noise removal, (c) Polar coordinate, (d) Polar coordinate noise

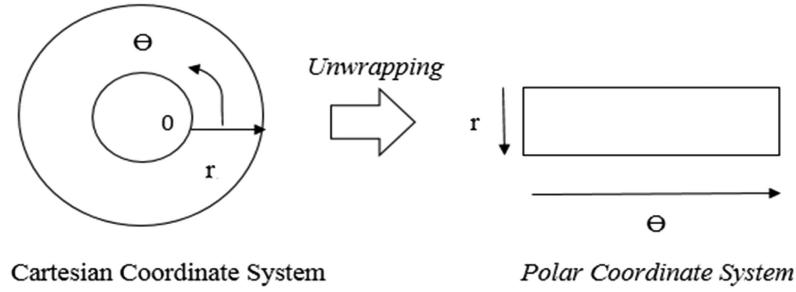


Fig. 2.5: Daugman's rubber-sheet model

Remapping to provide altered measurements of the iris image from Cartesian to polar coordinates is called normalisation. In the current paper, the ring of the iris was reformed to a rectangular square of composition having a stabilised size (64 512) (Fig. 2.11). Daugman's rubber-sheet model was employed for iris normalisation as shown in Fig. 2.2. The rubber-sheet model is defined in (1)(3), where the images of iris area are $I(x, y)$, the main Cartesian coordinates are (x, y) , the polar coordinates are (r, θ) , and x_l, y_l, x_p , and y_p are the iris and pupil boundaries coordinate in the direction

$$I(x(r, \theta), y(r, \theta))I(r, \theta) \quad (\text{Eq: 2.3})$$

$$x(r, \theta) = (1r)x_p(\theta) + r x_l(\theta) \quad (\text{Eq: 2.4})$$

$$y(r, \theta) = (1r)y_p(\theta) + r y_l(\theta) \quad (\text{Eq: 2.5})$$

where the iris coordinates are (x_i, y_i) and (x_p, y_p) and the pupil together with the x and y directions in the Cartesian coordinate system.

It accounts for variations in pupil size due to changes in external illumination that might influence iris size.

It ensures that the irises of different individuals are mapped onto a common image domain in spite of the variations in pupil size across subjects.

It enables iris registration during the matching stage through a simple translation operation that can account for in-plane eye and head rotations.

This hybrid network architecture is fundamentally different from that of the traditional system paradigm. It is established based on distributed vehicular cloudlets and units, and highlights the performance of local information and energy processing with the collaboration of nearby vehicular resources.

2.1.4 Image segmentation

The main, and most important, step in iris recognition is image segmentation. Several image quality measures were used to assess and compare the dilation ratio and occlusion acquired from this step. Identification of the pupil and limbic boundaries and eyelash and eyelid

removal were done at this time (Fig. 2.6).

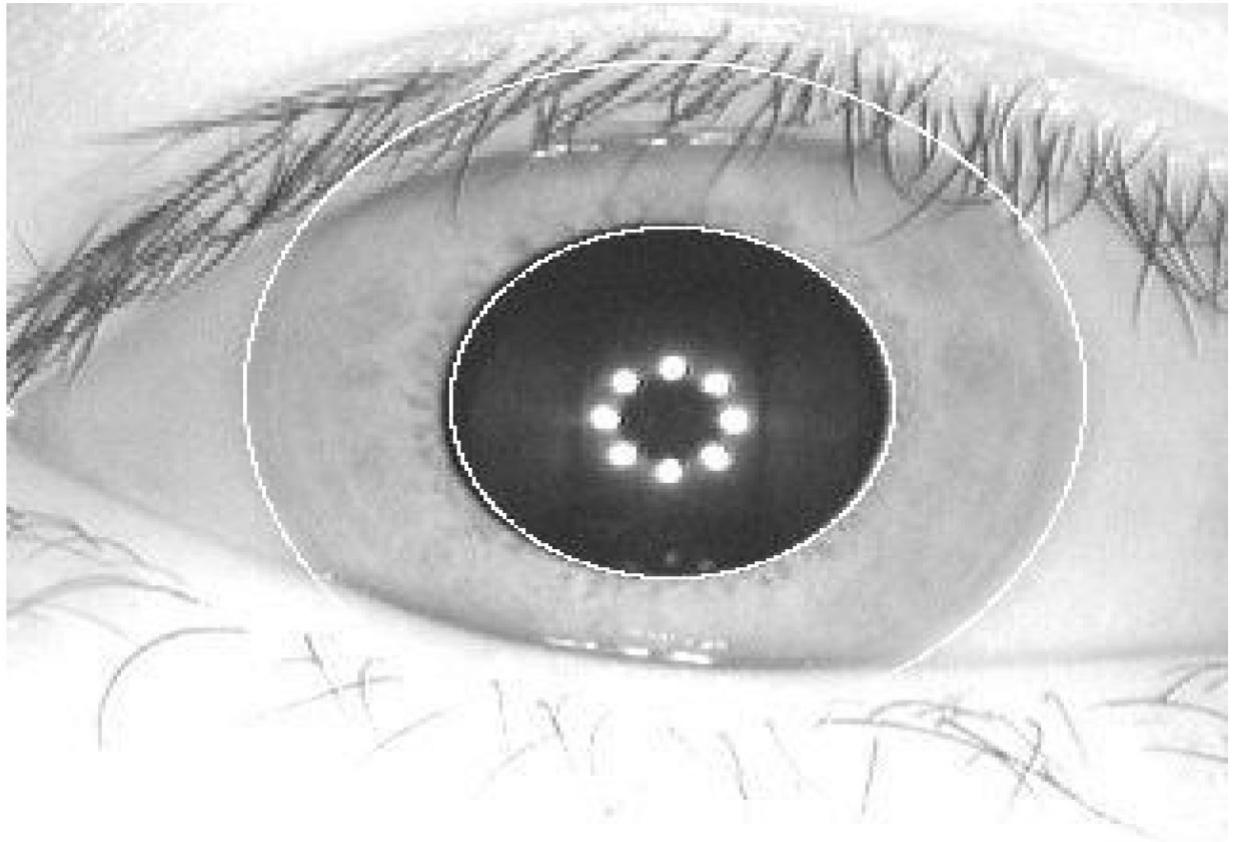


Fig. 2.6: Image segmentations

2.1.5 Iris anatomy

The iris is the colored ring of tissue around the pupil through which light enters the interior of the eye. Two muscles, the dilator and the sphincter muscles, control the size of the iris to adjust the amount of light entering the pupil. The sclera, a white region of connective tissue and blood vessels, surrounds the iris. A clear covering called the cornea covers the iris and the pupil. The pupil region generally appears darker than the iris. However, the pupil may have specular highlights, and cataracts can lighten the pupil. The iris typically has a rich pattern of furrows, ridges, and pigment spots. The surface of the iris is composed of two regions, the central pupillary zone and the outer ciliary zone.

The collarette is the border between these two regions. The minute details of the iris texture are believed to be determined randomly during the fetal development of the eye. They are also believed to be different between persons, twins and between the left and right eye of the same person. The color of the iris can change as the amount of pigment in the iris increases during childhood. An example of iris image is shown in below in Fig 2.7.:

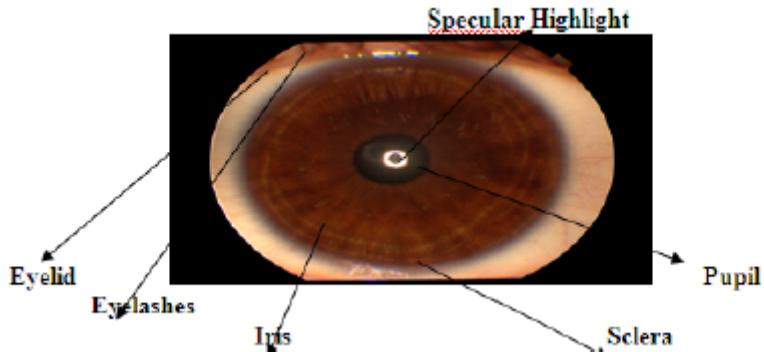


Fig. 2.7: Iris Anatomy

2.2 2D Gabor kernel method

Multichannel filtration changes the components of the spatial frequency of the various textures to extract the features of a texture. Decomposing the input image into different images using their textural information is a specific characteristic of this filter. Afterwards, the data is used to extract the features. The gabor filter bank was used to explain the channels in the spatial domain and frequency domain simultaneously. This prominent feature of gabor filtering has the advantage of joint localisation. The Fig. 2.8 shows Gabor Filter Bank. The gabor filter parameters are chosen precisely to determine the suitability of the human visual system.

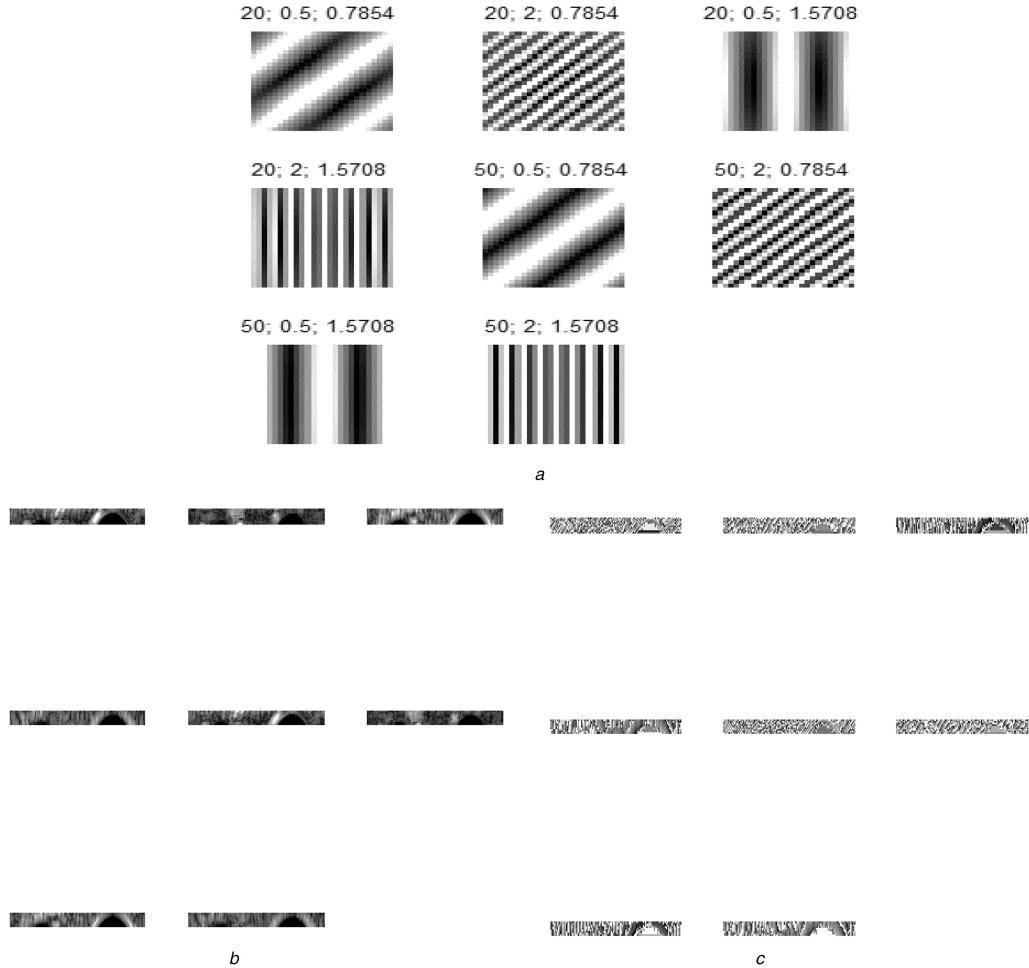


Fig. 2.8: Gabor Filter Bank, (a) Gabor kernel filter bank, (b) Result of 2D gabor kernel, (c) Enlarged view of (b)

The superfluous features of gabor reduce the dimensions of the required features. The gabor kernel approach was used to solve this issue. Each trained sample was twisted with regard to all gabor kernels. The distance within and between the classes was calculated using these twisted outcomes and the classes were used to choose the gabor kernel. To enhance algorithm performance and the power of discrimination, a method of learning the extracted features is required. Some approaches to this are machine learning, ANN, and evolutionary computing methods.

In the spatial domain, the 2D gabor filter is defined as follows:

$$g\lambda\theta\psi\sigma\gamma(x, y) = \exp\left(-\frac{x'^2 + \gamma^2y'^2}{2\sigma^2}\right)\cos(2\pi\frac{x'}{\lambda} + \psi) \quad (\text{Equ: 2.6})$$

where

$$x' = x\cos(\theta) + y\sin(\theta) \quad (\text{Equ: 2.7})$$

$$y' = y\cos(\theta) - x\sin(\theta) \quad (\text{Equ: 2.8})$$

in which σ is the sinusoidal function wavelength in $f = (1/\lambda)$, θ is the gabor filter orientation, $\psi = 90^\circ$ is the true value of the gabor filter phase offset, $\psi = 0$ is the fanciful value of the gabor filter, σ is the bandwidth, and γ is the aspect ratio.

The standard deviation sigma of the Gaussian factor determines the (linear) size of the receptive field. Its ellipticity and herewith the ellipticity of the receptive field ellipse is determined by the parameter gamma, called the spatial aspect ratio. It has been found to vary in a limited range of $0.23 < \gamma < 0.92$. Sigma cannot be controlled directly in the applet. Its value is determined by the choice of the parameters λ and b .

The parameter λ is the wavelength and $1/\lambda$ the spatial frequency of the cosine factor in Eq. (2.6). The ratio σ/λ determines the spatial frequency bandwidth of simple cells and thus the number of parallel excitatory and inhibitory stripe zones which can be observed in their receptive fields. The half-response spatial frequency bandwidth b (in octaves) and the ratio σ/λ are related as follows:

$$\delta = \log_2 \frac{\frac{\sigma}{\lambda}\pi + \sqrt{\frac{\ln 2}{2}}}{\frac{\sigma}{\lambda}\pi - \sqrt{\frac{\ln 2}{2}}} \quad (\text{Equ: 2.9})$$

$$\frac{\sigma}{\lambda} = \frac{1}{\pi} \sqrt{\frac{\ln 2}{2}} \cdot \frac{2^b + 1}{2^b - 1} \quad (\text{Equ: 2.10})$$

Neurophysiological research has shown that the half-response spatial-frequency bandwidths of simple cells vary in the range of 0.5 to 2.5 octaves in the cat (weighted mean 1.32 octaves) and 0.4 to 2.6 octaves in the macaque monkey (median 1.4 octaves). While there is a considerable spread, the bulk of cells have bandwidths in the range 1.0-1.8 octaves. Some researchers propose that this spread is due to the gradual sharpening of the orientation and spatial frequency bandwidth at consecutive stages of the visual system and that the input to higher processing stages is provided by the more narrowly tuned simple cells with half-response spatial frequency bandwidth of approximately one octave. Since λ and σ are not independent when the bandwidth is fixed, only one of them, lambda, is considered as a free parameter which is used in the applet.

2.3 Pattern matching

2.3.1 ANN and MLP

ANNs are smart universal computational algorithms comprising a number of neurons used to solve specific problems. These systems are programmed for special applications by learning the classification of data and pattern recognition. This involves adjustments for connecting the synapses between neurons to form a hierarchical structure. ANNs work as parallel processors that are connected such as a graph to provide fast decision making. Each neuron of this network is called a node. The high degree of correlation among features with no prior knowledge of the

dataset can be used to form composite hypotheses of this network. For example in Fig 2.8, where we see different layers of Artificial Neural Network.

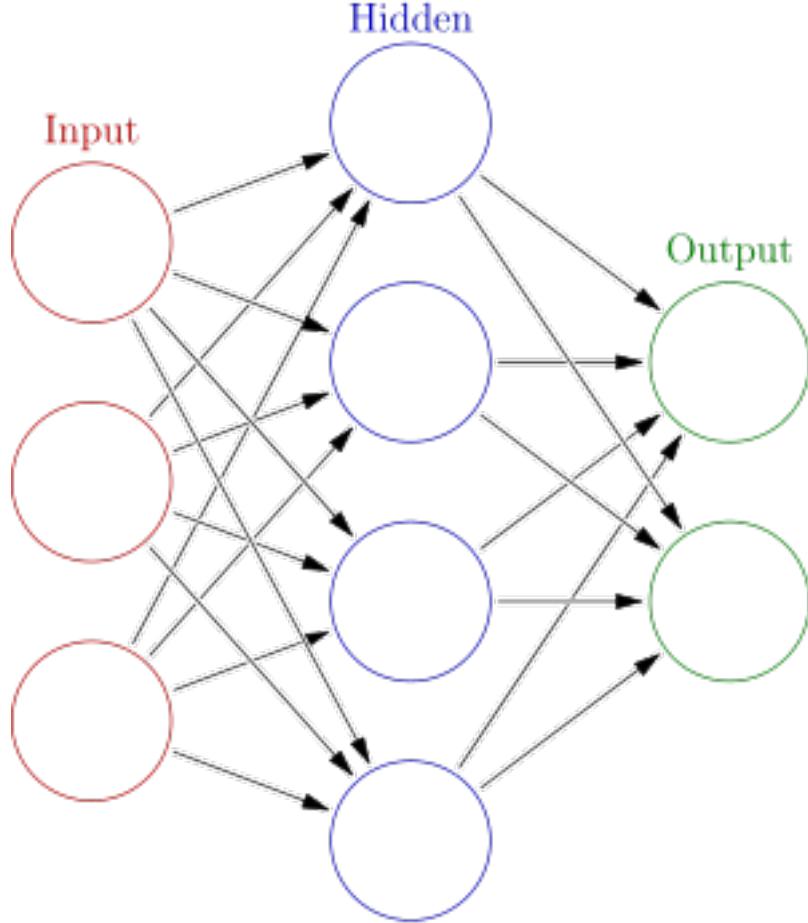


Fig. 2.9: Each circular node represents an artificial neuron and an arrow represents a connection from the output of one artificial neuron to the input of another.

Classification of non-separable problems requires more layers. A single-layer perceptron can separate linear problems. In a network with one layer, only the output layer counts as a node. In a network with more layers, the hidden and output layers count as nodes and the input layers do not; thus, there is no need for calculation.

To learn the MLPNN, network weights (w) should be defined to minimise the function of the output cost in which out is the final layer output and delineates the output from error function E . The learning MLPNN can be written as:

$$E_{SSE} = \frac{1}{2} \sum_p \sum_j (\text{targ}_j^p - \text{out}_j^{(N)p})^2 \quad (\text{Equ: 2.11})$$

where E_{SSE} is the squared error, targ is the target output, p is the training pattern, j is a neuron, out_j is the actual output of output neuron j , and N is the number of data points. Equation (2.8) updates the weights using gradient descent sets as

$$\Delta w = -\eta \frac{\partial E(w)}{\partial w} \quad (\text{Equ: 2.12})$$

where η is the learning rate. The training of the MLPNN algorithm includes the steps as shown in Algorithm 2.

Algorithm 1 Steps for training MLPNN algorithm

- 1: Selection of training patterns for the learning network.
 - 2: Defining N input, N-1 hidden layers, and N output as fully connected layers with their prior layers by weight of connection
 - 3: Random production of initial weights
 - 4: Choosing error function $E(w)$ and η correctly.
 - 5: Updating the weight of all weights (w) and all training patterns (p) $\Delta w = -\eta \frac{\partial E(w)}{\partial w}$
 - 6: Repetition of step 5 until the error function of the network becomes sufficiently small
-

2.3.2 Particle swarm optimisation

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. The population based global minimum of function can be seen in Fig. 2.9. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles.

Algorithm 2 Steps of PSO Algorithm

- 1: Define the preliminary particle.
 - 2: Compute the fitness function for all particles.
 - 3: Continue with step 4 if step 3 $\not\in$ personal best; otherwise go to step 5.
 - 4: Allot the values of the present fitness to new personal jump to step 6.
 - 5: Save prior personal best. Go to step 6.
 - 6: Allot the personal best of the best particle for the global best.
 - 7: Compute the velocity of all particles.
 - 8: Use the velocity values to update the data for all particles.
 - 9: If the target attained, continue to step 10; otherwise repeat steps 2-9.
-

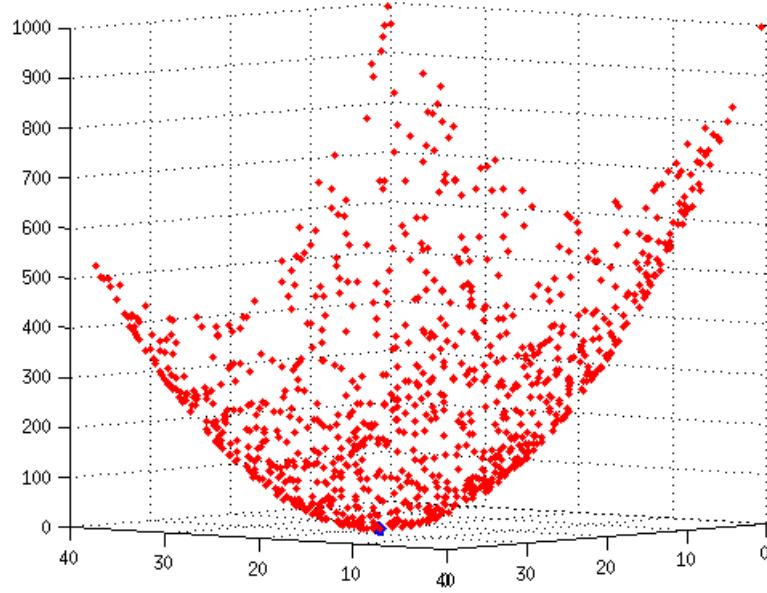


Fig. 2.10: A particle swarm searching for the global minimum of a function

Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called *pbest*. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbors of the particle. This location is called *lbest*. When a particle takes all the population as its topological neighbors, the best value is a global best and is called *gbest*.

The particle swarm optimization concept consists of, at each time step, changing the velocity of (accelerating) each particle toward its *pbest* and *lbest* locations (local version of PSO). Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward *pbest* and *lbest* locations. Another reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications. Particle swarm optimization has been used for approaches that can be used across a wide range of applications, as well as for specific applications focused on a specific requirement.

Based on a population of particles that is initialised randomly. In this algorithm, a collection of particles that can move are used to seek through the search space at high speed. PSO benefits from one end to the other from manifold distinct optimisation algorithms that are easy to execute. PSO algorithms have been successfully presented in several papers and applications.

Objective function (f) is used to assess the positions of the particles and the following equation obtains the personal best position:

$$y_i = \begin{cases} y_i, & \text{if } f(x) \geq f(y_i) \\ x_i, & \text{if } f(x_i) < f(y_i) \end{cases} \quad (\text{Equ: 2.13})$$

The global best position is defined as [42]

$$y^* \in \{y_0, y_1, y_2, \dots, y_s\} = \min\{f(y_0), f(y_1), \dots, f(y_s)\} \quad (\text{Equ: 2.14})$$

The particle speed and random location are updated at each iteration by integrating the current solution, personal best solution, and global best solution of each particle using (2.11) and (2.12) as

$$V_i = w \times V_i + c_1 r_1 (P_i - X_i) + c_2 r_2 (P_g - X_i) \quad (\text{Equ: 2.15})$$

$$X_i = X_i + V_i \quad (\text{Equ: 2.16})$$

where w is the inertia constant, c_1 and c_2 are the velocity coefficients (2), r_1 and r_2 are random numbers, and V_i falls between $[V_{max}, V_{max}]$ in which V_{max} denotes the maximum velocity. These steps are repeated until the velocity approaches zero. To assess the optimality criteria, a fitness function is used. The PSO algorithm comprises the steps shown in Chapter 2 (Algorithm 3).

The proposed method

In this section, the feature extraction, as an image pre-processing step to find the ROI, and the MLPNN PSO algorithm, as an iris recognition step, are described.

The Steps which were involved in iris recognition were (Fig.3.1)



Fig. 3.1: Steps for Recognition that were followed.

3.0.1 Feature extraction

For feature extraction, the gabor filter extracted textures and visual features. This method is defined as

$$\Psi_{\mu,v}(z) = \frac{\|k_{\mu,v}\|}{\sigma^2} e^{-\left(\left(\|k_{\mu,v}\|^2 \|z\|^2\right)/2\sigma^2\right)} \cdot \left(e^{ik_{\mu,v}^- z} - e^{-(\sigma^2/2)}\right) \quad (\text{Equ: 3.1})$$

where μ and v are the stand and scale for the direction of the wavelet, respectively, $\Psi_{\mu,v}$ is the gabor filter function, $k_{\mu,v}$ is the vector of the wavelet, σ is the filter bandwidth of the wavelet, $z = (x, y)$ are the coordinates of the pixel, and i is the complex operator.

We used 20 and 50 scales at orientations 0.7854 and 1.5708 as shown in Table 2.1. These parameters were obtained by trial and error. The vector of the wavelet is defined in (3.3), in which k_{max} is the maximum frequency

$$k_{\mu,v} = k_v^{i\phi\mu} \quad (\text{Equ: 3.2})$$

$$k_v = k_{max}/f^v \quad (\text{Equ: 3.3})$$

Parameter Name	Values and Method
matrix size	26
scales	[26, 50]
orientations	[0.7854, 1.5708]
frequencies	[0.5000 2]
centre points	[13 13]
create method	cross-product

Table 3.1: Parameters of gabor kernel feature extraction

The completion step for extraction of features using the gabor filter was done in the μ direction at scale using convolution operations. The pixel value of the iris image is $I(z)$ as defined in (3.4)

$$G_{\mu v}(z) = \psi_{\mu,v}(z) * I(z) \quad (\text{Equ: 3.4})$$

3.0.2 Multi-Layer Perceptron Neural Network Particle Swarm Optimization algorithm

The MLPNN PSO algorithm has been proposed to identify and detect the iris region. Although NN performance depends on accurate weight selection, exact structure also plays an important role in NN performance. In fact, the NN structure is formed by a number of neurons that are connected accurately. Learning NN with a large dataset requires an appropriate structure for iris recognition systems. The optimisation of an evolutionary structure for NNs has been shown to be a strongly effective strategy for selecting the structure in addition to the weights. The evolutionary algorithm can be added to the prior NN structure in the form of an approximated optimised pattern.

The proposed MLPNN architecture consists of 100 neurons in the hidden layer. In the input layer, there are 280 inputs, each corresponding to a pre-processed iris image. Finally, there is one neuron in the output layer. The basic structure of the MLPNN was altered to prepare it to apply the PSO algorithm as an optimisation tool to specify the MLPNN optimal weights. In the training section of the MLPNN, the MLPNN training function was modified using the PSO algorithm. An evolutionary method of PSO was written into the function. Instead of using the optimisation-based feature

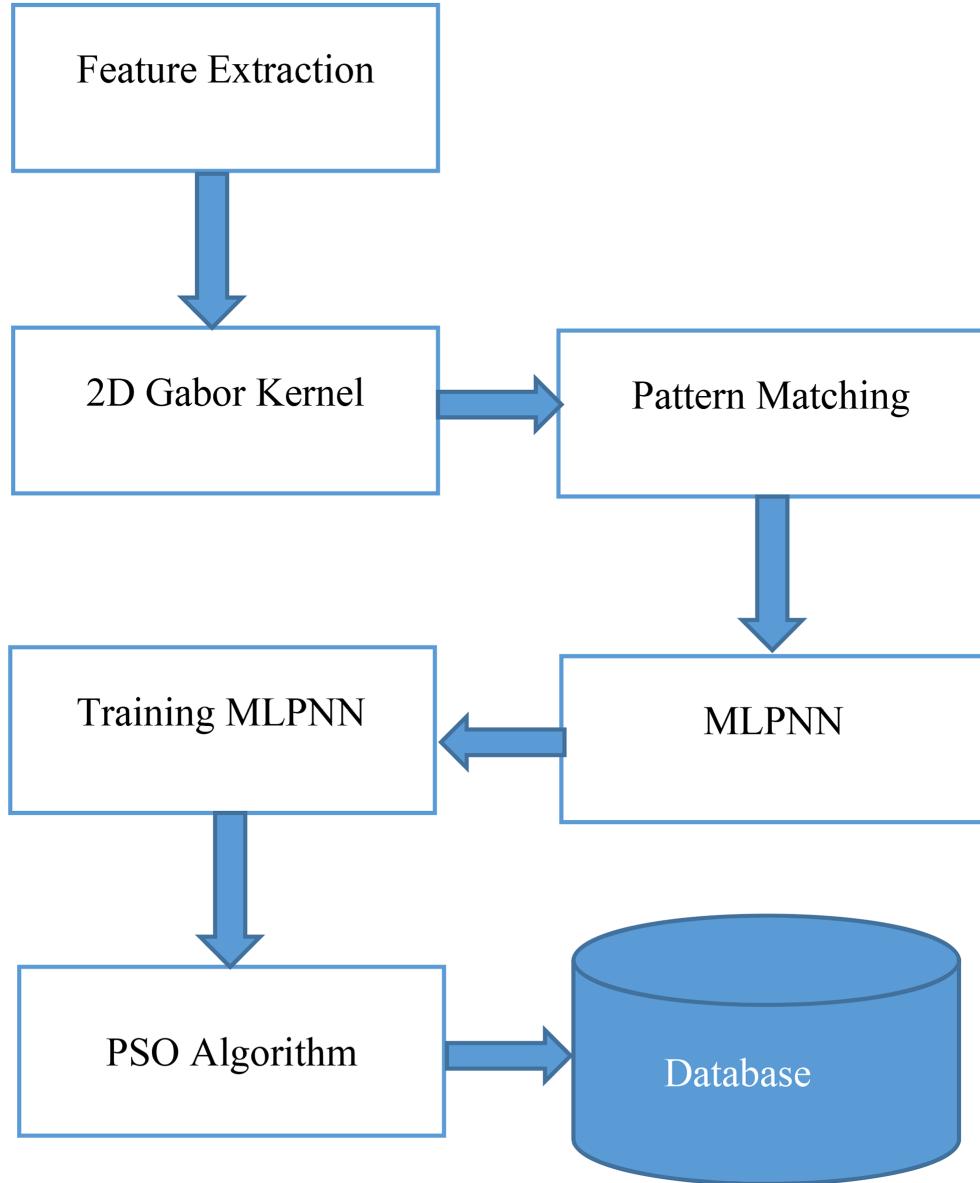


Fig. 3.2: Block diagram of iris recognition based on the MLPNN PSO method

gradient in the MATLAB NN toolbox, the evolutionary methodbased PSO was used. The arguments for the function stand without change because it is not desirable to add or remove additional items to the function. The structure of the MLPNN PSO algorithm is shown in Fig. 3.2.

3.0.3 Performance evaluation

In the proposed method, PSO is located in the inner and outer loops. The first loop is the inner loop, which is tasked to find the value of the training error related to the proposed structure, and the outer loop is tasked with optimising the NN structure (Fig. 3.3). The global best is considered for those particles which provide the minimum mean square error (MMSE). The corresponding personal best is considered for all particles in the outer layer. Subsequently,

the particles update their own structures in each iteration using their global and personal bests.

The weights of the MLPNN are important and must be specified. These are input weight (IW), layer weight (LW), and bias (b). In addition to the inputs entered into each neuron, a constant bias value was added to create the output. This MLPNN parameter must also be trained; hence, the underlying issue is accessible to these parameters. In the current paper, IW equaled 100 486 and was the number of neurons used in the first layer for the first element in which 486 was the number of inputs in the network. LW equaled 1 100 and denoted 100 neurons to 1 output layer. The bias weight was 100 1. The passive parameters of this network were calculated for IW, LW, and b to be 101, 100, and 100, respectively. This totaled 301 passive parameters that must be specified; thus, from the perspective of optimisation, 301 dimension problems were involved.

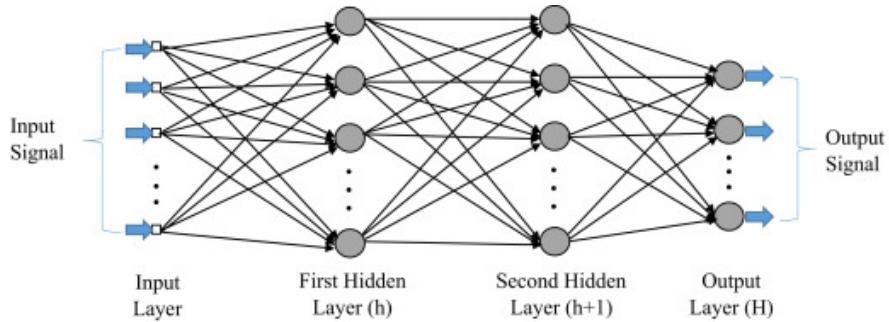


Fig. 3.3: Basic Multi-Layer Perceptron Neural Network

Apart from the type of algorithm used, the cost function of the algorithm had to be prepared. The xx vector was defined for the population, then each row of the population vector was read. This vector converted to an understandable network by changing from vector mode to matrix mode. The matrices replaced the different network sections and the network output was the same structure with the weights replaced. When the network returned, to obtain output, the new network was trained with the new data. A cost function was defined to minimise the error between the real network and the NN. This meant that the cost function encouraged the networks to be close to real output and vice versa.

The network was designed to consider 486 layers for the input layer and 100 layers for the hidden layer. Tansig was used for the activation function, Levenberg Marquardt for training the network and MSE for the network performance. The Levenberg Marquardt function obtained better results and was fast, but required a lot of memory.

After preparing the cost function for training the NN, the optimised PSO algorithm was designed and its parameters which were important for optimising the training process were defined. The convergence rate was influenced tremendously by one of these parameters. In this paper, in order to make a valid comparison, the parameters utilised to evaluate the effect of convergence rate were extremely similar. The number of maximum fails was set to 15, which had no influence on MLPNN performance. The maximum epoch number, which is the

Number of Iterations	Particle Size	Cognition Coefficient	Social Coefficient
100	200	2	2
100	500	0.5	.5
100	200	2	4
100	1000	0.5	4.5
300	500	2	2
2000	200	2	2

Table 3.2: Parameters of PSO Algorithm

stopping condition of the validation step, was set to 2000 and the global minimum was set to 0.001. For time management, no limitation was set on training (it was set to infinity). The network topology for PSO was followed by the global best. Using this topology, all particles were located in the neighbourhood of other particles and were attracted to the centre of the search space at the same time.

The parameters used in PSO were the particle size (200, 500, 1000), cognitive acceleration coefficient (c_1 : 0.5, 2), and social acceleration coefficient (c_2 : 2, 4, 4.5). The weight of inertia linearly was changed per grouping from 0.3 to 0.8. In this paper, the determination of the parameters was dependent on the results of comparison. The rate of optimality was based on the collection of parameters which were efficient for MLPNNPSO convergence. Table 5 shows the parameters used to test the proposed method. Fig. 3.4 shows the iris verification performance evaluated using the equal error rate (EER).

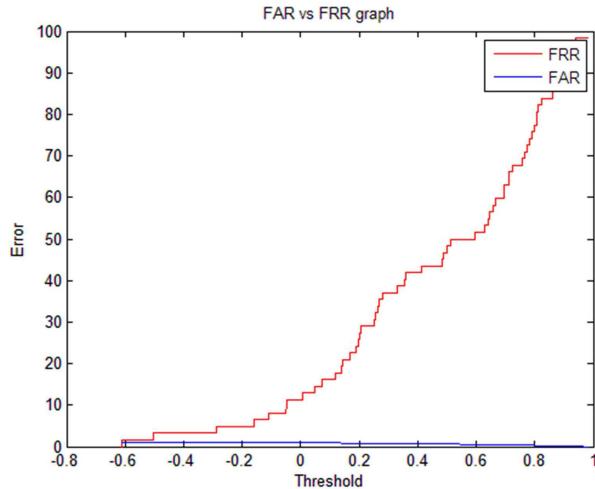


Fig. 3.4: EER

Conclusion

The present paper proposed a novel iris recognition method-based MLPNN and PSO to classify iris images. The 2D gabor kernel algorithm was used for feature extraction. The results of testing on the CASIA-iris V3 database and UCI machine learning repository databases indicate that the hybrid MLPNNPSO algorithm is an effective, appropriate, stable, robust, and competitive recognition method for human iris recognition. The experimental outcomes were better for the hybrid algorithm than when the MLPNN and PSO algorithms were used separately.

Future study will investigate the combination of fuzzy systems and MLPNN PSO methods. To increase the efficiency of the proposed algorithm and achieve better success, a proper classifier must be found for the proposed approach. A comparison of the proposed method with different classification methods is also planned using an identical database.

References

- [1] Tisse, C.L., Martin, L., Torres, L., et al.: Person identification technique using human iris recognition. Proc. Vision Interface, 2002, pp. 294299
- [2] Delac, K., Grgic, M.: A survey of biometric recognition methods. 46th IEEE Int. Proc. Symp. Electronics and Marine, 2004, pp. 184193
- [3] Yan, S., Xu, D., Zhang, B.: Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell., 2007, 29, (1), pp. 4051
- [4] Ahmadi, N., Akbarizadeh, G.: A review of iris recognition based on biometric technologies, Transylvanian Rev., 2016, 24, (4), pp. 151163
- [5] Alheeti, K.M.A.: Biometric iris recognition based on hybrid technique, Int. J. Soft Comput., 2011, 2, (4), p. 1
- [6] Tallapragada, V.S., Rajan, E.G.: Morphology based non-ideal iris recognition using decision tree classifier. IEEE Int. Conf. Pervasive Computing (ICPC), 2015, pp. 14
- [7] Daugman, J.G.: U.S. Patent No. 5,291,560, U.S. Patent and Trademark Office, Washington, DC, 1994
- [8] Abiyev, R.H., Altunkaya, K.: Neural network based biometric personal identification with fast iris segmentation, Int. J. Control Autom. Syst., 2009, 7, (1), pp. 1723
- [9] Cho, S., Kim, J.: Iris recognition using LVQ neural network. Int. Symp. Neural Networks, 2006, pp. 2633
- [10] Srivastava, V., Tripathi, B.K., Pathak, V.K.: Biometric recognition by hybridization of evolutionary fuzzy clustering with functional neural networks, J. Ambient Int. Humanized Comput., 2014, 5, (4), pp. 525537