International Journal of Image and Graphics
Vol. 14, No. 3 (2014) 1450010 (27 pages)
© World Scientific Publishing Company
DOI: 10.1142/S0219467814500107

GLCM-Based Multiclass Iris Recognition Using FKNN and KNN

S. B. Kulkarni*
Graphic Era University, Dehradun, Uttarakhand
Department of Computer Science and Engineering
SDM College of Engineering and Technology
Dharwad 580002, India
sbkulkarni_in@yahoo.com

Raghavendrarao B. Kulkarni
Department of Computer Science and Engineering
Shiradi Sai Engineering College, Bangalore 562106
eliterbk@gmail.com

U. P. Kulkarni
Department of Computer Science and Engineering
SDM College of Engineering and Technology
Dharwad 580002, India
upkulkarnir@yahoo.com

Ravindra S. Hegadi
Solapur University, Solapur 413255, India
rshegadi@gmail.com

Received 31 December 2012
Accepted 22 June 2014
Published 21 August 2014

Iris recognition is one of the important authentication mechanism used extensively in biometric applications. The majority of the applications use single class iris recognition with normalized iris image. The proposed technique uses multi class iris recognition with region of interest (ROI) iris image on supervised learning. In this paper, the term ROI is referred as Un-normalized iris. The iris features are extracted using gray level co-occurrence matrix (GLCM) and a multiclass training vector is created. Further, iris image is classified based on fuzzy K-nearest neighbor (FKNN) and KNN classification. Test samples features are matched with the stored repository by various matching techniques such as max fuzzy vote, Euclidean distance, cosine and cityblock. The experiment

*Corresponding author.

is carried on standard database CASIA-IrisV3-Interval and result shows that multiclass approach with ROI segmented iris has better recognition accuracy using FKNN and KNN.

Keywords: GLCM; FKNN; KNN; multiclass; ROI.

1. Introduction

Iris recognition is important in recent days to identify a person uniquely. As the computation begins with the object, pervasive computing needs an object to be identified before further processing. Hence, object recognition is an essential part of the pervasive computing. Further, these objects are put into computation to identify them uniquely. For such reason, machine learning techniques can be adopted in pervasive and ubiquitous computing. Wherever secure system is necessary, an authentication mechanism is applied which may range from home, office to border security.

Typically any recognition system consists of: (a) image acquisition, (b) preprocessing, (c) classification and (d) matching.

(a) *Image acquisition*: This is the first step in the iris recognition system. The object that must be recognized is to be captured and processed to get its features.

(b) *Preprocessing*: In the preprocessing, the captured image is to be segmented for extracting required features. The features can be extracted using any feature extraction technique suitable to the application. Before extracting, the required feature image needs to be analyzed for presence of noise which will reduce the accuracy. Hence, presence of unwanted noise must be removed. Then to compare any two images of same class or different class, the size of the image should be same size or the number of features must be constant. There are various feature extraction techniques that are available but the present work focuses on extracting the iris features which are rotationally invariant. Since iris is not a fixed object, it moves as eye moves. Hence, movement of eye leads to the deformation of the iris. It is difficult to recognize the deformed iris in multi direction, hence, gray level co-occurrence matrix (GLCM) features that possesses rotational invariant features are used to overcome the drawback of deformed iris feature extraction. The features extracted by applying GLCM feature extraction technique are energy, contrast, correlation, homogeneity, autocorrelation, dissimilarity, inertia and entropy.[1–4] Similarly, in this paper, 11 features of iris are extracted using GLCM feature extraction technique. Noise removal involves removal of eyelashes. Normalization is the process of transformation of the object to a fixed dimension.

(c) *Classification*: The object captured is to be classified; hence a class ID is assigned to multiple objects belonging to each class. The left and right irises of a person do not match to each other, hence both of them belongs to different classes. In this proposed work, fuzzy K-nearest neighbor (FKNN) and KNN classification techniques are used to classify iris image.

(d) *Matching*: The test object is matched with the stored feature vector to test for similarity. The matching techniques: maximum fuzzy vote, euclidean distance, cosine and cityblock algorithms are analyzed.

In this paper, KNN and FKNN classification methods are applied to know true acceptance (TA) and false rejection (FR) and K-fold approach is applied to analyze the classification accuracy with (5-1), (5-2), (5-3), (5-4). In (5-1) combination, 5 indicates the number of training samples and 1 indicates the number of test samples. Likewise, CASIA Iris V3 interval database is tested. It is difficult to recognize the iris, if minimum numbers of training samples are used, so proposed an algorithm to achieve more accuracy in minimum number of training samples using region of interest (ROI) based iris features. The classification of images involves two phases: one is training phase and the other is testing phase. In training phase, the properties of image features are isolated and a vector of training class is created. In the subsequent testing phase, features extracted are matched with the training features.

The remaining sections of this paper are organized as follows: Section 2 describes related work. Section 3 deals the feature extraction technique. Section 4 deals with the method proposed. Section 5 gives experimental data and result analysis.

2. Related Work

Survey on iris biometric was carried by Bowyer *et al.* they reported other researchers' methods and existing method used by other researches.[5] In that, one such technique is that, the iris boundaries are detected using integro-differential operator and then divided the iris into four portions (top, bottom, left, right). Further, top and bottom portions are discarded due to occlusion. They use the three-quarters of the iris region closest to the pupil. Iris features are extracted using circular symmetric feature (CSF) based on Gabor filter. An algorithm to detect the pupil that finds large connected component of pixels with a constraint that the pixel intensity values are below a certain given threshold was suggested by Floam and Safir.[5] They suggested that a description of individual's iris can be stored which can then be used for verification task. Daugman's integro-differential operator method and unwrapping makes normalization assumption such as the iris stretches linearly when pupil dilates and contracts. Wyatt opposed the assumption of Daugmann's and said that "it does not perfectly match the actual deformation of Iris".[5] Savitri *et al.* applied the Gabor wavelet, local binary pattern (LBP) and histogram of oriented gradient (HOG) techniques to extract features on specific portion of the iris to show that half portion of the iris is enough for iris recognition instead of entire image.[6]

A technique employed by Masek and Kovesi is to remove the occlusion due to eyelid and eye lashes.[7] The proposed technique here is to eliminate the noise present in the iris region by masking. Masek has applied normalization technique in the preprocessing. Masek extended the original Hamming distance (HD) method, which was used to match the two iris patterns. Features extracted are of binary bits, therefore bit-wise comparison was necessary and HD was preferred to match the iris patterns. HD algorithm is used in such a way that, only significant bits were considered while matching between two iris templates. In HD, only those bits in the iris pattern that correspond to zero bits in noise masks of both iris patterns were used in the calculation. The HD was calculated only for those bits which are generated from the true iris region. Since the algorithm uses the insignificant bits present in the mask to compute HD, it degrades the overall performance.

A technique for angle compensation in non-ideal Iris recognition using bi-orthogonal wavelet approach was proposed by Stephine *et al.*[8] They applied Daugman's integro-differential operator and performed segmentation at multiple resolution. Further they used modified version of Daugman's by replacing the integral along the elliptical contour with an integral along the arc-shaped boundary. In this method, they replaced 2D images of the known 3D calibration object by a set of 2D iris images from different subjects with a known angle of 0°, 15°, 30°. These images are used to model in different angles mentioned. A testing image is then transformed using the selected model plane to a frontal view image. In this connection, a plane very close to the model is used. In another method, angular distortions are estimated. In this method, the images must be trained for each angle. Here, for each of the angle (0°, 15° and 30°) an average model planes were calculated from the 50 training images. However, the deformation is considered in only one degree i.e. horizontal direction but the multiple deformations and skewing of eye are not yet addressed.

As integro-differential operator is computationally expensive, Yuan and Shi used Circular Hough transform, along with Daugman's rubber sheet model for normalizing the Iris.[9–11] The computation cost is reduced. They further proposed Iris feature extraction using 2D phase congruency. Phase congruency is related to phase information rather than amplitude information within the image. Phase congruency was first defined in terms of Fourier series expansion of a signal at a particular location. Phase component is the corresponding component of the amplitude component at that point of normalized iris image. "2D phase congruency provides invariance to variance in image illumination and some other condition". After phase congruency, the normalized iris is down sampled by 4×4 windows and concatenated every low value to vector and completes the vector with 1024 bits.

The best bits in an iris code work was carried out by Hollingsworth *et al.*[12] They used the unwrapped iris image with 20×240 pixels. In this they applied 30% and 40% of iris code making it to consider only the best bits by removing non-iris portions such that they considered only 5 to 12 rows instead of entire 20 rows of iris code.

A post-classifier method was proposed by Rahulkar and Holambe, in which a normalized iris-image with a size of 64×360 is being used.[13] In this upper half-iris portion is used to extract most discriminating iris features.

The above techniques uses normalized iris images for iris recognition and the features extracted are non-rotational of iris image, hence in this work rotational free features of iris is considered by using GLCM feature extraction techniques. They applied feature reduction by considering less size or by considering less radial distance from center of pupil to iris. As stated above, in pre-processing, computation time for normalization of iris image is more, hence ROI-based iris is considered for this work.

3. Feature Extraction Technique

Feature extraction is the essential part in pattern recognition wherein the goal of pattern recognition is to reduce the number of features such that the features are distinct to classify the images; images can be classified by shape, color and texture, etc.[14] Azizi and Pourreza proposed that, the GLCM is proved to be a good technique as it provides reasonable accuracy and is invariant to iris rotation.[15]

In a statistical texture analysis, texture features were computed on the basis of statistical distribution of pixel intensity at a given position relative to others in a matrix of pixel representing image. Depending on the number of pixels or dots in each combination, there are first-order statistics, second-order statistics or higher-order statistics. Feature extraction based on GLCM is the second-order statistics that can be used to analyze image as a texture.[16] GLCM is also called gray tone spatial dependency matrix. It is a tabulation of the frequencies or how often a combination of pixel brightness values in an image occurs. Figure 1 represents the formation of the GLCM of the gray-level (four levels i.e. 0, 1, 2, 3) image at the distance $d = 1$ and the direction of 0°.

Figure 1(a) is an example of image matrix of pixels intensity representing image with four levels of gray. Here intensity levels, level 0 and 1 are marked with a thin box. The thin box represents pixel-intensity 0 along with 1 in horizontal direction with distance 1 as level 1 is 0's neighbor (in the horizontal direction or the direction of 0°). There are two occurrences of such pixels. Therefore, the GLCM matrix generated (Fig. 1(b)) value 2 in row 0, column 1. Similarly, row-0 and column-0 of GLCM matrix gives a value 2, because there are two occurrences in which pixels with value 0 has pixels 0 as its neighbor (horizontal direction). As a result, the pixels matrix representing in Fig. 1(a) is transformed into GLCM matrix as shown in Fig. 1(b). Figure 1(c) depicts the normalized co-occurrence matrix. It is calculated

0	$\boxed{0}$	$\boxed{1}$	1	1
0	$\boxed{0}$	$\boxed{1}$	1	1
0	2	2	2	2
3	3	1	2	2
2	2	3	3	3

(a)

	0	1	2	3
0	2	$\boxed{2}$	1	0
1	0	4	1	0
2	0	0	5	1
3	0	1	0	3

(b)

0.1	0.1	0.05	0
0	0.2	0.05	0
0	0	0.25	0.05
0	0.05	0	0.15

(c)

Fig. 1. GLCM Feature extraction: (a) image matrix with four-level gray image, (b) GLCM matrix with distance = 1 and direction = 0° and (c) normalized co-occurrence matrix.

based on dividing the GLCM matrix (Fig. 1(b)) by sum of pair of GLCM matrix. The sum of GLCM matrix in this example is $\sum P(i,j) = 20$.

Following equations from "Eqs. (1) to (7)" are used to extract the GLCM features of iris image.

$$\text{Energy: } f1 = \sum_i \sum_j P(i,j)^2, \tag{1}$$

$$\text{Contrast: } f2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i,j) \middle| \, |i-j| = n \right\}, \tag{2}$$

$$\text{Correlation: } f3 = \sum_i \sum_j P(i,j) \frac{(i-\mu_x)(i-\mu_y)}{\sigma_x \sigma_y}, \tag{3}$$

where μ_x, μ_y and σ_x, σ_y are the means and standard deviation of p_x, p_y

$$\mu_x = \sum_i \sum_j i \cdot P(i,j) \quad \mu_y = \sum_i \sum_j j \cdot P(i,j),$$

$$\sigma_x = \sqrt{\sum_i \sum_j (i-\mu)^2 P(i,j)} \quad \text{and} \quad \sigma_y = \sqrt{\sum_i \sum_j (j-\mu)^2 P(i,j)}$$

for the symmetrical GLCM, $\mu_x = \mu_y$ and $\sigma_x = \sigma_y$

$$\text{Homogeneity: } f4 = \sum_i \sum_j \frac{1}{1+(i-j)^2} P(i,j), \tag{4}$$

$$\text{Autocorrelation: } f5 = \sum_i \sum_j (ij) P(i,j), \tag{5}$$

$$\text{Dissimilarity: } f6 = \sum_i \sum_j |i-j| \cdot P(i,j), \tag{6}$$

$$\text{Inertia: } f7 = \sum_i \sum_j (i-j)^2 P(i,j), \tag{7}$$

where $\mu_x = \mu_y$ and $\sigma_x = \sigma_y$ are mean and standard deviation along x and y axis.

$$\begin{aligned} \text{Energy} &= 0.10^2 + 0.10^2 + 0.05^2 + 0.00^2 + 0.00^2 + 0.20^2 + 0.05^2 + 0.00^2 \\ &\quad + 0.00^2 + 0.00^2 + 0.25^2 + 0.05^2 + 0.00^2 + 0.05^2 + 0.00^2 + 0.15^2 \\ &= 0.01 + 0.01 + 0.0025 + 0 + 0 + 0.04 + 0.0025 + 0 + 0 \\ &\quad + 0 + 0.0625 + 0.0025 + 0 + 0.0025 + 0 + 0.0225 \\ &= 0.155. \end{aligned}$$

$$\begin{aligned}
\text{Contrast} &= (0-0)^2 * 0.10 + (0-1)^2 * 0.10 + (0-2)^2 * 0.05 + (0-3)^2 * 0.00 \\
&\quad + (1-0)^2 * 0.00 + (1-1)^2 * 0.20 + (1-2)^2 * 0.05 + (1-3)^2 * 0.00 \\
&\quad + (2-0)^2 * 0.00 + (2-1)^2 * 0.00 + (2-2)^2 * 0.25 + (2-3)^2 * 0.05 \\
&\quad + (3-0)^2 * 0.00 + (3-1)^2 * 0.05 + (3-2)^2 * 0.00 + (3-3)^2 * 0.15 \\
&= 0 + 0.1 + 0.2 + 0 + 0 + 0 + 0.05 + 0 + 0 + 0 + 0 + 0.05 + 0 + 0.2 \\
&\quad + 0 + 0 \\
&= 0.6. \\
\text{Entropy} &= 0.10 * \ln(0.10) + 0.10 * \ln(0.10) + 0.05 * \ln(0.05) + 0.00 \\
&\quad + 0.00 + 0.20 * \ln(0.20) + 0.05 * \ln(0.05) + 0.00 \\
&\quad + 0.00 + 0.00 + 0.25 * \ln(0.25) + 0.05 * \ln(0.05) \\
&\quad + 0.00 + 0.05 * \ln(0.05) + 0.00 + 0.15 * \ln(0.15) \\
&= 0.23 + 0.23 + 0.149 + 0.322 + 0.149 + 0.346 \\
&\quad + 0.149 + 0.149 + 0.284 \\
&= 2.008. \\
\text{Mean } (\mu) &= 0 * 0.10 + 0 * 0.10 + 0 * 0.05 + 0 * 0.00 + 1 * 0.00 + 1 * 0.20 \\
&\quad + 1 * 0.05 + 1 * 0.00 + 2 * 0.00 + 2 * 0.00 + 2 * 0.25 + 2 * 0.05 \\
&\quad + 3 * 0.00 + 3 * 0.05 + 3 * 0.00 + 3 * 0.15 \\
&= 0.20 + 0.05 + 0.50 + 0.1 + 0.15 + 0.45 \\
&= 1.45. \\
\text{Variance} &= (0-1.45)^2 * 0.10 + (0-1.45)^2 * 0.10 + (0-1.45)^2 * 0.05 \\
&\quad + (0-1.45)^2 * 0.00 + (1-1.45)^2 * 0.00 + (1-1.45)^2 * 0.20 \\
&\quad + (1-1.45)^2 * 0.05 + (1-1.45)^2 * 0.00 + (2-1.45)^2 * 0.00 \\
&\quad + (2-1.45)^2 * 0.00 + (2-1.45)^2 * 0.25 + (2-1.45)^2 * 0.05 \\
&\quad + (3-1.45)^2 * 0.00 + (3-1.45)^2 * 0.05 + (3-1.45)^2 * 0.00 \\
&\quad + (3-1.45)^2 * 0.15 \\
&= 0.210 + 0.210 + 0.105 + 0.0405 + 0.010 + 0.0756 \\
&\quad + 0.0151 + 0.120 + 0.360 \\
&= \sqrt{1.146} = 1.071.
\end{aligned}$$

$$\begin{aligned}
\text{Correlation} &= (0-1.45)*(0-1.45)*0.10+(0-1.45)*(1-1.45)*0.10\\
&\quad+(0-1.45)*(2-1.45)*0.05+(0-1.45)*(3-1.45)*0.00\\
&\quad+(1-1.45)*(0-1.45)*0.00+(1-1.45)*(1-1.45)*0.20\\
&\quad+(1-1.45)*(2-1.45)*0.05+(1-1.45)*(3-1.45)*0.00\\
&\quad+(2-1.45)*(0-1.45)*0.00+(2-1.45)*(1-1.45)*0.00\\
&\quad+(2-1.45)*(2-1.45)*0.25+(2-1.45)*(3-1.45)*0.05\\
&\quad+(3-1.45)*(0-1.45)*0.00+(3-1.45)*(1-1.45)*0.05\\
&\quad+(3-1.45)*(2-1.45)*0.00+(3-1.45)*(3-1.45)*0.15\\
&= 0.210+0.0652+0.398+0.405-0.0123+0.0756\\
&\quad+0.0426-0.0348+0.360\\
&= 1.51.\\
\text{Homogeneity} &= 1/(1+(0-0)^2)*0.10+1/(1+(0-1)^2)*0.10\\
&\quad+1/(1+(0-2)^2)*0.05+1/(1+(0-3)^2)*0.00\\
&\quad+1/(1+(1-0)^2)*0.00+1/(1+(1-1)^2)*0.20\\
&\quad+1/(1+(1-2)^2)*0.05+1/(1+(1-3)^2)*0.00\\
&\quad+1/(1+(2-0)^2)*0.00+1/(1+(2-2)^2)*0.00\\
&\quad+1/(1+(2-3)^2)*0.25+1/(1+(2-4)^2)*0.05\\
&\quad+1/(1+(3-0)^2)*0.00+1/(1+(3-1)^2)*0.05\\
&\quad+1/(1+(3-2)^2)*0.00+1/(1+(3-3)^2)*0.15\\
&= 0.10+0.05+0.01+0.20+0.025+0.125+0.01+0.15\\
&= 0.67.
\end{aligned}$$

Similarly, other features are calculated. In this way, feature vectors are generated with the texture property. Following features are considered in the work proposed using MATLAB generated GLCM features:

F1 Autocorrelation:	(autoc)
F2 Contrast:	(contr)
F3 Correlation:	(corrm)
F4 Correlation:	(corrp)
F5 Cluster Prominence:	(cprom)
F6 Cluster Shade:	(cshad)
F7 Dissimilarity:	(dissi)

F8 Energy:	(energ)
F9 Entropy:	(entro)
F10 Homogeneity:	(homom)
F11 Homogeneity:	(homop)

Feng Gui and Wu Ye-qing used GLCM combined with DCT to extract the iris features.[17] They have used a fusion of GLCM and DCT features to improve the recognition.

Azizi and Pourreza used normalization method which was referred to in Libor Masek, Peter Kovesi and Contourlet transform (CT) which allows for different directions at each scale.[7,15,18] They used 21 GLCM features with which the classification accuracy is 94.2 and they used subband along with 56 GLCM features with which classification accuracy is 96.3% using KNN classifier.

Junying Zeng *et al.* proposed the technique for Iris recognition system-based on biomimetic pattern recognition on the CASIA database as a new approach of Biomimetic pattern and their result is better by using biometric pattern.[19]

Grigorescu *et al.* compared the technique of texture features based on Gabor filter.[20] Hence, gabor energy can be applied to extract iris features and Tsai *et al.* applied the gabor filter to extract iris features.[21]

4. Method Proposed

The method proposed includes the existing method of segmentation, normalization and the proposed ROI-based iris. In this work, normalized iris image and ROI-based iris images are tested for multiclass recognition to achieve better accuracy in accepting multiple classes. Figure 2 shows the stages of multiclass iris recognition. Initially GLCM features are extracted on normalized iris image then GLCM feature vector is created. To match the test images of normalized irises FKNN and KNN are applied with different distance methods. It is found that for few cases, more number of normalized training images was required so to overcome this drawback un-normalized (ROI-based) iris images were passed for testing resulting into better matching performance by lesser number of training samples. This reduces the number of training samples.

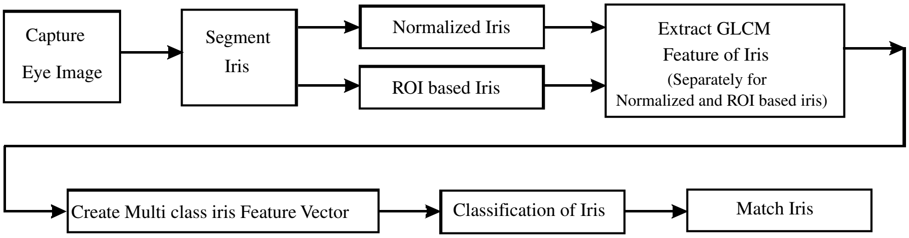


Fig. 2. Multiclass recognition.

The following steps are carried out in the proposed work.

4.1. *Segmentation*

Canny edge detection and boundary between sclera/iris, pupil/iris is detected for iris localization, along with Circular Hough transform which is used to detect the edges in the iris that differentiates the inner and outer part of the iris.[7] The center coordinates x_c and y_c, and the radius r, which define circle as given in "Eq. (8)". Further roipoly and inverse roipoly are applied to separate the iris region from rest of eye image. Figure 3 shows the visual example of normalized and un-normalized iris image after the segmentation.

4.1.1. *Hough transform*

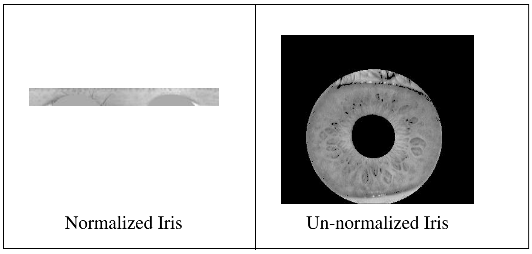
The Hough transform is a standard computer vision algorithm that can be used to determine the parameters of simple geometric objects, such as lines and circles, present in an image. Circular Hough transform can be employed to deduce the radius and center coordinates of the iris and pupil regions. Initially, an edge map is generated by calculating the first derivatives of intensity values in an eye image and threshold is applied to get the result. From the edge map, votes are cast in Hough space for the parameters of circles passing through each edge point.[7] These parameters are the center coordinates x_c and y_c, and the radius r, which are used to define any circle according to the equation.

$$x_c^2 + y_c^2 = r^2. \tag{8}$$

A maximum point in the Hough space will correspond to the radius and center coordinates of the circle defined by the edge points. Wildes *et al.* and Kong and Zhang also make use of the parabolic Hough transform to detect the eyelids, approximating the upper and lower eyelids with parabolic arcs,[7] which is represented in Eq. (9)

$$(-(x - h_j)\sin\theta + (y - k_j)\cos\theta)^2 = a_j(x - h_i)\sin\theta + (y - k_j)\cos\theta, \tag{9}$$

where a_j controls the curvature (h_j, k_j), is the peak of the parabola and θ is the angle of rotation relative to the x-axis.



Normalized Iris

Un-normalized Iris

Fig. 3. Example of normalized and un-normalized iris image.

4.2. *Normalization of iris*

Circular region of the iris data is converted to rectangular plane using polar to rectangular conversion. Rubber sheet algorithm is used to convert to fixed dimension as shown in Fig. 4.

4.3. *Un-normalized (ROI-based) iris*

The need for generation of ROI-based iris is to eliminate the normalization process as shown in Fig. 5. The specific generated pattern is called tyre model (ROI-based iris image) because it looks like a vehicle tyre with inner (pupil) segmented circle forming the rims of the tyre and iris forming the tyre.[22]

4.3.1. *Masking the exterior part of the outer segmented circle*

To mask the exterior part of outer circle, x and y co-ordinates of the exterior circle and radius r are considered from Sec. 4.1 of previous step of segmentation. This becomes an input to ROI (inverse Roipoly) to mask outer iris ring (blackening or cropping) as shown in Fig. 6.

4.3.2. *Masking the interior part of inner segmented circle*

To mask the interior circle, the x and y co-ordinates of the circle are taken along with the radius r of the circle from Sec. 4.1 of previous step of segmentation. Then roipoly is applied to mask inner circle, as shown in Fig. 7.

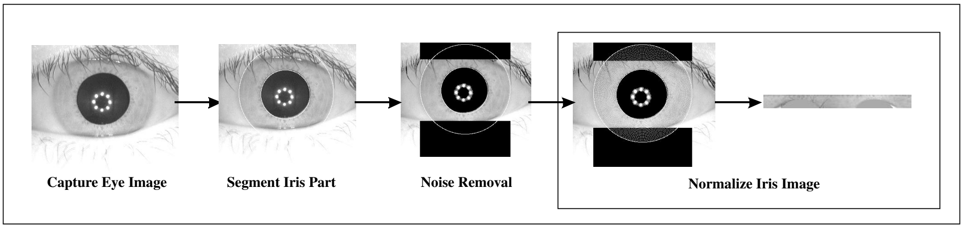


Fig. 4. Normalization of iris.

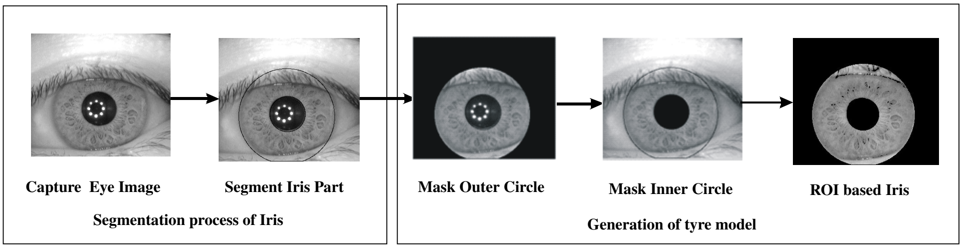


Fig. 5. Tyre model generation.

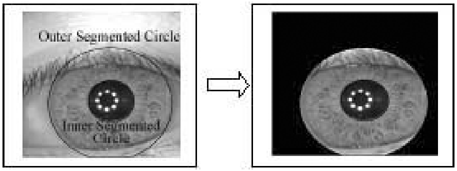


Fig. 6. Masking the exterior part of the outer segmented region.

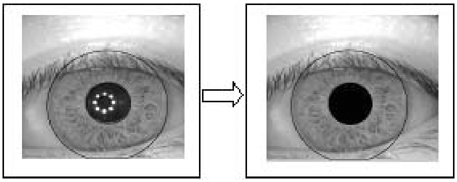


Fig. 7. Masking the interior part of inner segmented region.

4.3.3. *Cropping the iris region*

Output of Sec. 4.3.1 is fed to Sec. 4.3.2 to generate the tyre model. In the tyre model the Human iris part looks like a tyre by masking the rest of the iris part as shown in Fig. 8.

4.4. *Noise removal*

The presence of eyelash and eyelids introduce noise in the iris data. In order to remove these noises, noise pixels are identified using Algorithm 1 and removed by replacing it with zero for further computation. Noise removal is shown in Fig. 9.

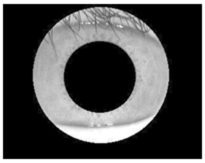


Fig. 8. ROI-based iris image.

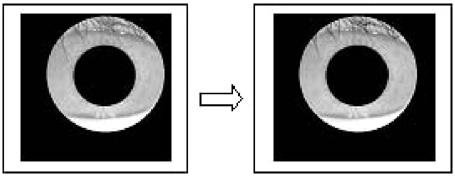


Fig. 9. Masking the non-iris part of the ROI.

Algorithm 1 Noise removal from tyre model (ROI-based iris).
Input: ROI Image
Output: ROI Image with noise free.

```
1. Read ROI Image.
2. Compute Size of an Image.
3. Store size in 'a' and 'b'.
4. For each pixel value of iris image,
   a. Replace 0 if pixel value is less than 100.
5. Write noise free image.
```

Description of the algorithm is as follows:

Algorithm 1 is proposed to eliminate the presence of noise in the iris image. It is observed from Fig. 9 that, iris gray level values are ranging more than 100 (pixel value) hence, the pixel value less than 100 is the value of the eye lashes so these are replaced with zero to make it completely black. Such information is not considered in feature extraction.

4.5. *Design and algorithms*

Multiclass iris recognition design is shown in Fig. 10. The overall design is described in following steps:

1. Capture multiple eye images of multiple classes.
2. Segment the iris image.
3. Normalize the iris image.
4. Extract GLCM features of normalized iris image.
5. Create feature vector of all the classes.
6. Assign the class ID to multiple classes of objects.
7. Store the multiple classes with their respective class in a database.
8. Create feature vector for test or sample image (single or multiple objects, multiple classes with multiple objects).
9. Apply classification algorithm.
10. Match the feature vectors with the stored repository.
11. If all classes are not matched, then replace normalized iris by un-normalized (ROI-based) iris.
12. Repeat steps 8–10 for matching by changing different distance methods.

Block 1: In this, the image captured is segmented, normalized and the GLCM features are extracted, initially GLCM feature vector is created and stored this acts as training vector for all the classes. Similarly testing vector is created. FKNN and KNN classification is applied on these feature vector and matching is performed on different distance metrics maximum fuzzy vote, Euclidean distance, cosine and cityblock.

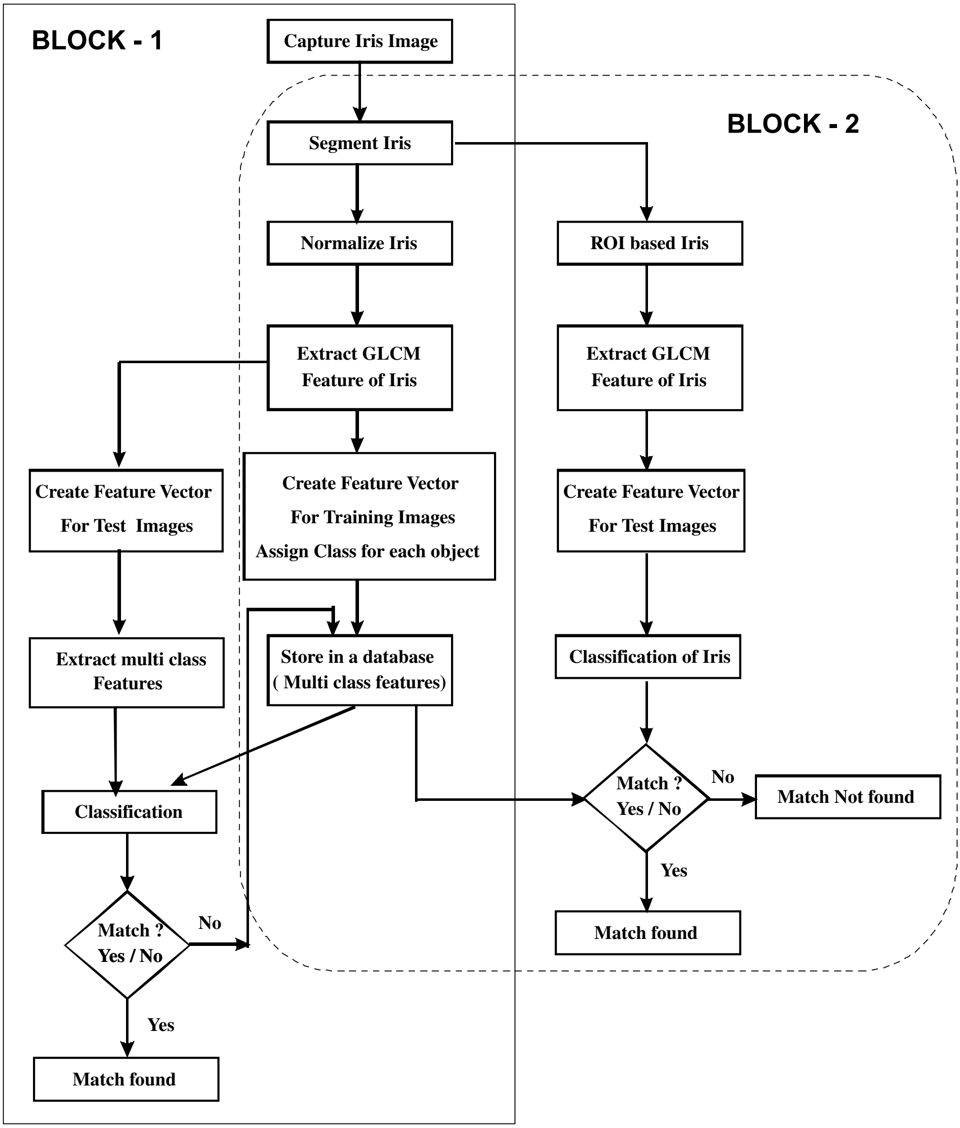

BLOCK - 1
Capture Iris Image
Segment Iris
Normalize Iris
Extract GLCM Feature of Iris
Create Feature Vector For Test Images
Create Feature Vector For Training Images Assign Class for each object
Extract multi class Features
Store in a database (Multi class features)
Classification
Match ? Yes / No
No
Yes
Match found
BLOCK - 2
ROI based Iris
Extract GLCM Feature of Iris
Create Feature Vector For Test Images
Classification of Iris
Match ? Yes / No
No
Match Not found
Yes
Match found

Fig. 10. Overall design of multiclass.

In match yes/no of block 1, if yes then match is found. Otherwise it is not matched. As the data is supervised, hence match should happen, the reason for not matching is that the iris image chosen is a deformed iris. Hence the features need to be trained again. So this multi-deformed iris image features are appended to training vector as shown in block 1. This indicates that more training samples are required to accept the multi-deformed iris image.

Block 2: In continuation with Block 1 explanation, if match not found for the deformed iris, instead of training with more samples, the un-normalized (ROI-based) iris image GLCM features are extracted and matched with the same training vector the result found that it accepted. This indicates that normalization method needs more training sample as compared to un-normalized iris image match. This is the novel approach where training samples are reduced by using un-normalized iris images for matching.

Algorithm 2. Training multiple classes.
Input: Normalized Iris Image feature.
Output: Features vectors with their class ID.

1. Read Multiple image of same class.
2. Extract GLCM features (11 features).
3. Assign class to each object of same class with same class ID.
4. Repeat steps 1–3 for multiple classes of iris images until all classes.
5. Store the feature vector matrix along with respective class ID matrix in a database.

Algorithm 3. Generate feature vector. *Input:* GLCM extracted features.
Output: Feature vector

1. Read GLCM feature of an image.
2. Extract 11 features.
3. Generate a row of 11 features.
4. Feature vector row is created.

The description of the algorithms is as follows:

Algorithm 2 is used to generate the feature vector on supervised data. Initially normalized iris image is taken as an input and the GLCM features (11 features, such as autoc, contr, corm, corrp, cprom, cshad, dissi, energy, entro, homom, homop) are extracted for each image and a class is assigned to each normalized iris image.[1–4] This step is repeated for the entire iris database. Finally it generates the feature vector of entire database along with class ID.

Algorithm 3 is used to generate the feature vector of unsupervised data. Here the feature vector is created without any class.

Further, these two feature vector are taken as input to FKNN and KNN for testing the features that are close to each other by various distance measure metrics.

Feature vectors are created by placing each object in row and column specifies the features along with respective class assignment like indexing as proposed by Somnath Dey and Debasis Samanta, is as shown in Fig. 11.[23] A class is assigned to each object and a group of similar objects have the same class ID to represent a class. Similarly multi class feature vectors are created and stored in a database. Multiclass feature and the respective class assignment for 1 to n class or m to n class with m_1 to n_1 object are created in the proposed algorithm. Here m and n are beginning class and ending class, respectively. Objects concern to class can be accessed with m_1 to n_1, as m_1 is beginning object and n_1 is ending object. Hence any combination of classes and any number of combinations of objects with any number of features can be created. Extracted features are same to both training and test samples. That is number of columns of training samples is equal to the test samples.

Algorithm 4. Generate multiclass feature vector.
Input: Beginning class, ending class, beginning object, and ending object.
Output: Multiclass Feature vector with class ID.

1. Read beginning class.
2. Read ending class.
3. Read beginning object.
4. Read ending object.
5. Go to step 1 of Algorithm 2.
6. Append row vector.
7. Append same class ID for each object until all object of same class.
8. Increment Class number till all class assignment completes.
9. Repeat steps 5–8 until all objects of all classes.
10. Multiclass feature vector along with class ID is created and stored in database.

Description of the Algorithm 4 is as follows:

Algorithm 4 is an extended version of Algorithm 2 where Algorithm 3 uses multiple objects of same class hence each object feature is extracted and stored with the same class name as shown in Fig. 11.

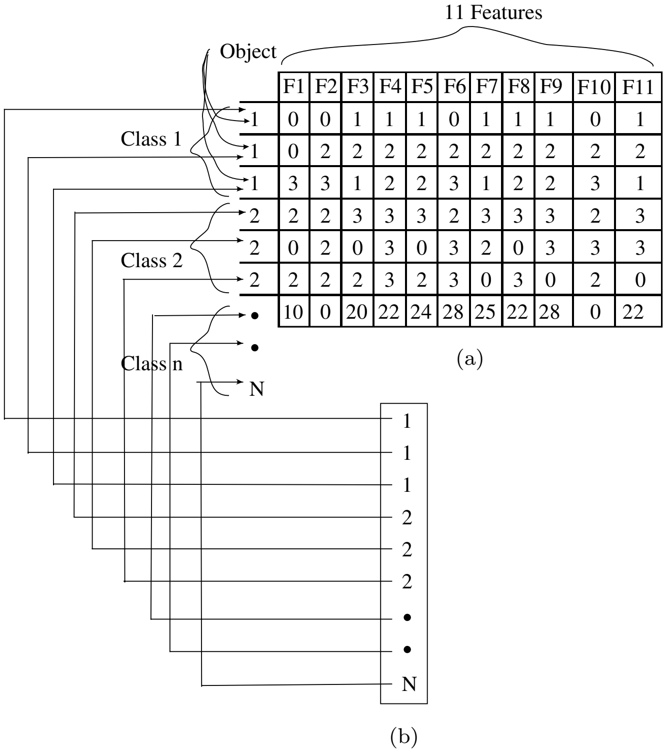
Figure 11 depicts how a feature vector and the classes are assigned to the samples. In Fig. 11(a) represents the feature vector for $1 \ldots N$ classes and (b) is the class number assignment to each sample that belongs to a particular class. Algorithm 4 describes how the multiple classes are assigned.

4.6. *Classification*

The feature vectors shown in Fig. 11 have to be classified, therefore FKNN, KNN classifiers are applied on feature vector of iris images.[1–4]

Classification includes:

- Training method:
  - Save the training examples.



	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
1	0	0	1	1	1	0	1	1	1	0	1
1	0	2	2	2	2	2	2	2	2	2	2
1	3	3	1	2	2	3	1	2	2	3	1
2	2	2	3	3	3	2	3	3	3	2	3
2	0	2	0	3	0	3	2	0	3	3	3
2	2	2	2	3	2	3	0	3	0	2	0
•	10	0	20	22	24	28	25	22	28	0	22
•											
N											

(a)

1
1
1
2
2
2
•
•
N

(b)

Fig. 11. Multiclass feature vector with class ID. (a) Multiclass feature vector with objects along row and features along column, (b) Multiple classes of group of each class mapping to multiple classes of group of objects.

- At prediction time (matching):
  - — Find the k training examples $(x_1, y_1), \ldots (x_k, y_k)$ that are closest to the test example x.
  - — Predict the most frequent class among those (LABELS) y_i's.

4.7. *Matching*

The binary pattern of the two iris are matched with hamming distance and its improvement in hamming distance is defined in Refs. 7 and 18. Here the classification again is matched with the Fuzzy criteria and distance criteria. In FKNN, class is predicted with the maximum fuzzy vote "Eq. (10)" in which the assigned members are influenced by the inverse of the distance from the nearest neighbors and their class memberships. The inverse distance serves to weight a vector's membership more if it is closer and less if it is farther from the vector under consideration. In KNN with different distance methods like Euclidean distance "Eq. (11)", cosine

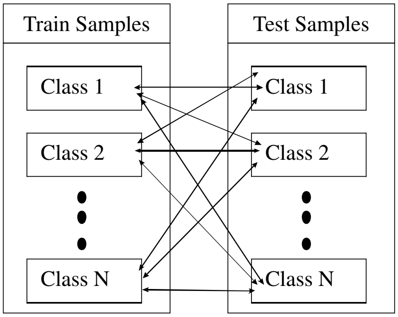


Fig. 12. Comparison of multiple classes.

"Eq. (12)" and city block "Eq. (13)" are applied for the classification.

$$\text{Maximum fuzzy vote:}\quad u_i(x) = \frac{\sum_{j=1}^{k} u_{ij}(1/\|x - x_j\|^{-1}/(m-1))}{\sum_{j=1}^{k}(1/\|x - x_j\|^{-1}/(m-1))}. \tag{10}$$

$$\text{Euclidean distance:}\quad d_{ij} = \sqrt{\sum_{k=1}^{n}(x_{ik} - x_{jk})^2}. \tag{11}$$

$$\text{Cosine distance:}\quad \cos(d_1, d_2) = (d_1, d_2)/\|d_1\|\|d_2\|. \tag{12}$$

$$\text{City block distance:}\quad d_{ij} = \sum_{k=1}^{n}|x_{ik} - x_{jk}|. \tag{13}$$

Figure 12 depicts the comparison of features on multiple classes where the train sample classes are from 1 to N and the test sample classes are from 1 to N are compared as shown in the figure. In this, if the class 1 contains 5 training samples then maximum of four test samples are compared for the same class. In this case K-fold is (5-4) is applied for $1, \ldots, N$ classes. As the samples in the multiclass testing requires more number of classes and each class with fixed number of samples but CASIA iris V3interval can provide maximum number of classes with nine samples for each class. Hence, it is chosen that the experiment can be conducted with maximum of 5-4 combination in multi class with this environment.

5. Experimental Data and Result Analysis

The proposed work is carried out on the cassia database of CASIA-IrisV3-Interval dataset which has total 249 subjects, 395 classes and number of images 2639.[25] Most of the images were captured in two sessions, with at least one month interval. For our experiment, we have considered a total of 20 classes as all classes do not have enough subjects for training. Tables 1–4 shows the selection of K-fold for highest accuracy. Highlighted marks in Tables 1–4 shows the highest accuracy i.e. 90% with

Table 1. Ten classes with K-fold 5-4.

Class with K-fold (5-4)			FKNN			KNN								
			Max. fuzzy vote			Eucl. distance			Cosine			City block		
Class no.	Total training samples	Total test samples	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy
1	5	4												
2	10	8	6	2	66.67	6	2	66.67	6	2	66.67	5	3	40.00
3	15	12	9	3	66.67	9	3	66.67	7	5	28.57	8	4	50.00
4	20	16	13	3	76.92	13	3	76.92	10	6	40.00	12	4	66.67
5	**25**	**20**	**17**	**3**	**82.35**	**17**	**3**	**82.35**	**14**	**6**	**57.14**	**16**	**4**	**75.00**
6	30	24	20	4	80.00	20	4	80.00	17	7	58.82	19	5	73.68
7	35	28	21	7	66.67	21	7	66.67	16	12	25.00	19	9	52.63
8	40	32	24	8	66.67	24	8	66.67	16	16	0.00	23	9	60.87
9	45	36	25	9	64.00	25	9	64.00	19	17	10.53	24	12	50.00
10	50	40	28	12	57.14	28	12	57.14	23	17	26.09	27	13	51.85

Table 2. Ten classes with K-fold 5-3.

Class with K-fold (5-3)			FKNN			KNN								
			Max. fuzzy vote			Eucl. distance			Cosine			City block		
Class no.	Total training samples	Total test samples	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy
1	5	3												
2	10	6	4	2	50.00	4	2	50.00	3	3	0.00	3	3	0.00
3	15	9	7	2	71.43	7	2	71.43	6	3	50.00	6	3	50.00
4	20	12	10	2	80.00	10	2	80.00	8	4	50.00	9	3	66.67
5	**25**	**15**	**13**	**2**	**84.62**	**13**	**2**	**84.62**	**11**	**4**	**63.64**	**12**	**3**	**75.00**
6	30	18	15	3	80.00	15	3	80.00	13	5	61.54	14	4	71.43
7	35	21	16	5	68.75	16	5	68.75	12	9	25.00	15	6	60.00
8	40	24	18	6	66.67	18	6	66.67	12	12	0.00	18	6	66.67
9	45	27	19	8	57.89	19	8	57.89	15	12	20.00	18	9	50.00
10	50	30	21	9	57.14	21	9	57.14	17	13	23.53	21	9	57.14

Table 3. Ten classes with K-fold 5-2.

Class with K-fold (5-2)			FKNN			KNN								
			Max. fuzzy vote			Eucl. distance			Cosine			City block		
Class no.	Total training samples	Total test samples	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy
1	5	2												
2	10	4	3	1	66.67	3	1	66.67	3	1	66.67	2	2	0.00
3	15	6	5	1	80.00	5	1	80.00	5	1	80.00	4	2	50.00
4	20	8	7	1	85.71	7	1	85.71	7	1	85.71	6	2	66.67
5	**25**	**10**	9	1	88.89	9	1	88.89	**9**	**1**	**88.89**	8	2	75.00
6	**30**	**12**	**11**	**1**	**90.91**	**11**	**1**	**90.91**	10	2	80.00	**10**	**2**	**80.00**
7	35	14	11	3	72.73	11	3	72.73	10	4	60.00	10	4	60.00
8	40	16	12	4	66.67	12	4	66.67	10	6	40.00	11	5	54.55
9	45	18	14	4	71.43	14	4	71.43	11	7	36.36	11	7	36.36
10	50	20	13	7	46.15	13	7	46.15	13	7	46.15	13	7	46.15

Table 4. Ten classes with K-fold 5-1.

Class with K-fold (5-1)			FKNN			KNN								
			Max. fuzzy vote			Eucl. distance			Cosine			City block		
Class no.	Total training samples	Total test samples	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy	TA	FR	Accuracy
1	5	1												
2	10	2	1	1	0.00	1	1	0.00	1	1	0.00	1	1	0.00
3	15	3	2	1	50.00	2	1	50.00	2	1	50.00	2	1	50.00
4	20	4	3	1	66.67	3	1	66.67	3	1	66.67	3	1	66.67
5	25	5	4	1	75.00	4	1	75.00	4	1	75.00	4	1	75.00
6	**30**	**6**	**5**	**1**	**80.00**	**5**	**1**	**80.00**	**5**	**1**	**80.00**	**5**	**1**	**80.00**
7	35	7	5	2	60.00	5	2	60.00	5	2	60.00	5	2	60.00
8	40	8	6	2	66.67	6	2	66.67	5	3	40.00	6	2	66.67
9	45	9	6	3	50.00	6	3	50.00	5	4	20.00	6	3	50.00
10	50	10	7	3	57.14	7	3	57.14	6	4	33.33	7	3	57.14

FKNN and KNN with maximum fuzzy vote[24] and Euclidean distance, respectively. The observation shows that, as the numbers of classes are increased, the accuracy is decreased and also as the number of class decreases, accuracy is decreased. Highest accuracy is observed in training 5 to 6 classes. But Table 5 show that, as number of subjects of a class were increased for training, the accuracy of the proposed method found to be 100%. In order to fix the minimum training classes with minimum subjects, optimum combination observed is between 1 and 5 classes with 6 subjects for training gives 100% accuracy. In our experiment, a test is conducted for both normalized image sets and ROI-based iris image are considered. Shifted or rotated eye in the normalized form found no match with the feature vector. Whereas the un-normalized (ROI-based) iris image found match with the feature vector. This is reported as 9th object problem (Shifted or rotated iris image) in Table 5. Matching with the features extracted from normalized image requires more training sample. To overcome this problem, in this paper, a technique ROI-based iris image with GLCM feature is proposed. The result shows 100% accuracy with K-fold 5-3. Hence, K-fold of 5-3 i.e. 5 classes with 6 subjects and any number of classes within 5 class of test samples with 3 or less than three objects (test sample) shows 100% accuracy in recognizing multiple classes of multi deformed iris and it can be extended to any number of classes with the said combination. In our experiment, we have considered K-fold training and test data comparisons. The combination is 10 to 9 classes i.e. 10 classes for training image and 9 classes for test image with 5 training objects to each class and 4 test objects to each class are compared with FKNN, and KNN classifier with different matching methods as shown in Table 1. Total training objects are 50 (10 classes × 5 objects) with 11 features (total features are 550) and 40 (10 classes × 4 objects) objects with 11 feature set (total features is 440). The total comparisons are 550 with 440 feature set further iterated till single object of class, as shown in Tables 1–4. TA means true acceptance and FR means false rejection.

The comparison between existing method of template creation[7] time with normalized iris and tyremodel (ROI-basediris) un-normalizediris is stated in Ref. 22. There is an improvement in the performance of average time by 55% in preprocessing step.[22] Tables 6 and 7 show the comparison between the proposed method with the other recent popular methods for classification. It shows that, only 11 features are considered to achieve the better performance. In this Table 7, as compared the ROI-based (un-normalized) with normalized iris, the feature difference of F1, F5 and F6 is more hence these features can be reduced. The proposed method shows improvement in classification using KNN and FKNN for multiclass recognition under supervised data. Table 8 show the comparison of GLCM features with normalized iris data and ROI-based (un-normalized) iris data. The drastic variation is in feature F1 (autocorrelation), F5 (cluster prominence) and F6 cluster shade. But other features do not have a large difference. This shows that lesser features are enough to classify and recognize in multiclass.

Table 5. Combination of K-fold with 5-1, 5-2 and 5-3.

Training		Testing		FKNN	KNN			
Class no.	Subjects	Class no.	Objects	Max. fuzzy vote	Eucl. distance	Cosine	City block	Observations
1–5	1–6	1–5	7th	√	√	√	√	All methods accepted
1–5	1–6	1–5	8th	√	√	X	√	Failure of cosine
1–5	1–6	1–5	9th	X	X	X	X	9th object problem
1–5	1–6	1–5	7th and 8th object	√	√	X	X	Failure of cosine and cityblock
1–5	1–6	1–5	8th and 9th object	X	X	X	X	9th object problem
1–5	1–6	1–5	7th, 8th and 9th object	X	X	X	X	9th object problem
1–5	1–6	2–5	7th, 8th and 9th object	√	√	X	X	9th object problem of 1st class
1–5	1–7	1–5	8th	√	√	√	√	All methods accepted
1–5	1–7	1–5	9th	X	X	X	X	9th object problem of 1st class
1–5	1–8	1–5	9th	√	√	√	√	Needs more training subjects (Hence replaced with ROI-based iris (no need of normalization) and further tested with smaller set of data in next step)
1–5	1–5	1–5	6th to 9th object	X	X	X	X	All methods failed
1–5	1–5	1–5	7th to 9th object	X	X	X	X	All methods failed
1–5	1–6	1–5	7th to 9th object	√	√	X	X	100% Classification by FKNN and KNN (Minimum training subjects required is twice the testing objects with 1–5 classes)

Table 6. KNN classifier with Haralicks's 21 feature of 3 matrices of iris normalized image with contourlet transform.

The number of classes	The correct of percentage classification (%) (KNN classifier)
20	96.6
40	88.3
60	90.8
80	89.3
100	88.5

Table 7. Comparison between proposed method and some well-known method.

Method	The correct of percentage classification (%)	The feature vector length (bit)
Daugman (1993)	100	2048
Wildes (1996)	94.18	>2048
Ma (2002)	95.02	1600
Ma (2003)	92.22	1320
Jafar Ali (2003)	92.16	87
Azizi and Pourreza (2009) (GLCM)	94.2	21
Azizi and Pourreza (2009) (GLCM with subband)	96.3	56
Our method GLCM with KNN (Min distance) and FKNN multiclass recognition	100	11 (features)

Table 8. Comparison of features of normalized iris image and ROI-based iris image.

Features	Normalized values	Un-normalized (ROI-based) iris
F1 Autocorrelation (autoc)	38.76987	10.73354
F2 Contrast (contr)	0.093096	0.284416
F3 Correlation (corrm)	0.828418	0.971673
F4 Correlation (corrp)	0.828418	0.971673
F5 Cluster Prominence (cprom)	4.409801	982.5504
F6 Cluster Shade (cshad)	0.953771	95.31092
F7 Dissimilarity (dissi)	0.089749	0.115405
F8 Energy (energ)	0.513769	0.498229
F9 Entropy (entro)	1.063968	1.280394
F10 Homogeneity (homom)	0.955683	0.958223
F11 Homogeneity (homop)	0.95546	0.956071

The experiment is conducted on HP Compaq $n \times 6320$ with 2 GB Ram, Intel Core 2 Duo processor with 1667 MHz. Tables 1–4 show the optimum selection for highest accuracy, based on this again, the experiment was conducted to get 100% accuracy in multiclass recognition. In Table 5, observation column shows the result analysis for various combination of K-fold.

6. Conclusion

Experimental results show that recognition accuracy is varying from 80% to 100%. It is observed that K-fold 5-3 is having better accuracy. Elimination of normalization leads to reduction in the training sample by 25% and an improvement of 45% in the feature extraction time by using un-normalized iris method. Multiclass iris recognition is achieved i.e. the proposed algorithm recognizes multiple classes in a single instance. As there is no transformation (shifting) of iris data hence, the approach (ROI with GLCM) satisfies iris recognition with any degree of freedom (multi deformed iris). The number of minimum training samples required are at least double the test samples is observed in Table 5. Recognition accuracy decreases as the number of classes are less than 5 and more than 6 this is because iris features are very close to each other. Recent existing method shows 96.3% accuracy with GLCM feature and subbands combination feature, having 56 features with normalized process. Whereas the proposed approach varies from 80% to 100% and also uses less number of training samples to achieve better accuracy in multi-deformed iris recognition. The proposed method works better for small set of data but it can be further improved by clustering approach, wherein each group of class made as one cluster so that multiple cluster approach can be adopted for large set of data.

Acknowledgements

The author wishes to acknowledge CASIA for providing the iris database. The shared CASIA Iris Database is available on the web.[25] The work is partially supported by the Research Grant from AICTE, Govt. of India, Reference no.: 8023/RID/RPS-114(PVT)/2011–12 Dated December, 24-2011.

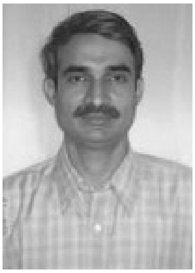
References

1. R. M. Haralick, K. Shanmugam and I. H. Dinstein, “Textural features for image classification,” *IEEE Transactions on Systems, Man and Cybernetics* **SMC-3**, 610–621 (1973).
2. L. K. Soh and C. Tsatsoulis, “Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices,” *IEEE Transactions on Geoscience and Remote Sensing* **37**, 780–795 (1999).
3. D. A. Clausi, “An analysis of co-occurrence texture statistics as a function of grey level quantization,” *Canadian Journal of Remote Sensing* **28**, 45–62 (2002).
4. http://murphylab.web.cmu.edu/publications/boland/boland_node26.html.
5. K. W. Bowyer, K. Hollingsworth and P. J. Flynn, “Image understanding for iris biometrics: A survey,” *Comput. Vision and Image Understanding* **110**, 281–307 (2008).
6. G. Savithri and A. Murugan, “Performance analysis on half iris feature extraction using GW, LBP and HOG,” *Int. J. Comput. Applications* **22**, 27–32 (2011).
7. L. Masek and P. Kovesi, “Matlab source code for a biometric identification system based on iris patterns,” *The School of Computer Science and Software Engineering*, Vol. 2, No. 4 (The University of Western Australia, 2003).

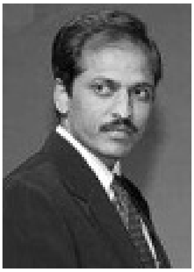
8. S. A. C. Schuckers *et al.*, "On techniques for angle compensation in nonideal iris recognition," *IEEE Trans. Syst. Man Cybern. B, Cybern.* **37**, 1176–1190 (2007).
9. X. Yuan and P. Shi, "Iris feature extraction using 2D phase congruency," in *3rd ICITA*, Vol. 2 (July 2005), pp. 437–441.
10. S. Pie and J. Horng, "Circular arc detection based on Hough transform," *Pattern Recognition Lett.* **16**, 615–625 (2005).
11. N. Alioua, A. Amine, M. Rziza and D. Aboutajdine, "Eye state analysis using iris detection based on circular Hough transform," *ICMCS* (April 2011), pp. 1–5.
12. K. P. Hollingsworth, K. W. Bowyer and P. J. Flynn, "The best bits in an iris code," *IEEE Trans. Pattern Anal. and Mach. Intell.* **31**, 964–973 (2009).
13. A. D. Rahulkar and R. S. Holambe, "Half-iris feature extraction and recognition using a new class of biorthogonal triplet half-band filter bank and flexible K-out-of-n: A postclassifier," *IEEE Trans. Inf. Forens. Security* **7**, 230–240 (2012).
14. R. Powlikar *et al.*, *Pattern Recognition* (Wiley Encyclopedia of Biomedical Engineering, New York, NY: Wiley, 2006), pp. 1–22.
15. A. Azizi and H. Pourreza, "A novel method using contourlet to extract features for iris recognition system," *Emerging Intelligent Computing Technology and Applications* **5754**, 544–554 (2009).
16. A. K. Mohanty, S. Beberta and S. K. Lenka "Classifying benign and malignant mass using GLCM and GLRLM based texture features from mammogram," *International Journal of Engineering Research and Applications* **1**, 687–693 (2011).
17. G. Feng and Y. Q. Wu, "An iris recognition algorithm based on DCT and GLCM," in *Proc. SPIE (S0277-786X)* (2007), p. 70001H-1.
18. S. B. Kulkarni, R. S. Hegadi and U. P. Kulkarni, "Improvement to libor masek algorithm of template matching method for iris recognition," in *Proc. of the International Conference & Workshop on Emerging Trends in Technology* (Mumbai, India, ACM, 2011), pp. 1270–1274.
19. Z. Zeng, Y. Zhai, J. Gan and Y. Xu, "An effective iris recognition system based on biomimetic pattern recognition," *ICISE* (2009), pp. 3583–3586.
20. S. E. Grigorescu, N. Petkov and P. Kruizinga, "Comparison of texture features based on Gabor filters," *IEEE Transactions on Image Processing* **11**, 1160–1167 (2002).
21. C. C. Tsai, J. S. Taur and C. W. Tao, "Iris recognition using gabor filters optimized by the particle swarm technique," *SMC* (October 2008), pp. 921–926.
22. S. B. Kulkarni, R. S. Hegadi and U. P. Kulkarni, *ROI Based Iris Segmentation and Block Reduction Based Pixel Match for Improved Biometric Applications*, Vol. 361 (Springer-Verlag, Berlin, Heidelberg, 2013), pp. 548–557.
23. S. Dey and D. Samanta, "Iris data indexing method using Gabor energy features," *IEEE Transaction on Information Forensics and Security*, Vol. 7 (2012), pp. 1192–1203.
24. J. M. Keller, M. R. Gray and J. A. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE Transaction on Systems Man and Cybernetics*, Vol.-SMC-15 (1985), pp. 580–585.
25. Chinese Academy of Sciences, Institute of Automation (CASIA) Iris Database, http://www.cbsr.ia.ac.cn.

Dr. S. B. Kulkarni is an Assistant Professor, SDM College of Engineering and Technology, Dharwad, Karnataka, India. He obtained his Bachelor of Engineering from Basaveshwar Engineering College, Bagalkot. He received his Masters degree in Computer Science and Engineering from BVBCET, Hubli. He has received his Doctoral degree from Graphic Era University, Dehradun, Uttarakhand, India. He has published many papers at International Journal and conferences. He is a member of ISTE. He has guided many U.G. and P.G. students. His research interests are digital image processing, pattern recognition, medical image analysis, data and image mining.

Raghavendrarao B. Kulkarni is an Assistant Professor, Shiradi Sai Engineering College, Bangalore Karnataka, India. He obtained his Diploma in Computer Science and Engineering from KHK Institute, Dharwar. He obtained his Bachelor of Engineering from KLE, Belgaum. He received his Masters degree in Computer Science and Engineering from Basaveshwar Engineering College Bagalkot. He is pursuing his Ph.D. from JNTU, Hyderabad, India. He has published many papers at International Journal and conferences. He is a member of ISTE. He has guided many U.G. and P.G. students.



Dr. U. P. Kulkarni is a Professor, SDM College of Engineering and Technology, Dharwad, India. Dr. Umakant Kulkarni obtained his BE degree from Karnataka University, Dharwad in the year 1989, ME degree from PSG College of Technology, Coimbatore in the year 1991 and Ph.D. from Shivaji University, Kolhapur in the year 2007. He has published many papers at International Journal and IEEE Conferences in the areas of Pervasive and Ubiquitous Computing, Distributed Data Mining, Agents Technology and Autonomic Computing. He is Member of IETE and ISTE. He served as Head of Department and Chief Nodal Officer- TEQIP a World Bank funded project. He has guided many students at PG level and five research scholars are pursuing their Ph.D.s. He can be reached at upkulkarni@yahoo.com.



Dr. Ravindra S. Hegadi has completed his graduation in B.Sc. in 1991, M.C.A. in 1995, M.Phil. in 2004 and Ph.D. in 2007 from Gulbarga University, Gulbarga. He has 16 years of teaching and 12 years of research experience. Presently he is serving as Associate Professor at Solapur University, Solapur, India. He has published 27 research articles in reputed International Journals and 43 research articles in National and International Conference proceedings. He has also authored two books. He is a member of many international research and academic bodies including IEEE. He has visited Australia, Hong Kong and USA to present his research articles in International Conferences. He is a member of editorial and reviewer board of different journal. He has executed UGC sponsored Major research project and organized workshops. His research interests are digital image processing, pattern recognition, medical image analysis, data and image mining.