J Inf Process Syst, Vol.12, No.1, pp.83~99, March 2016
http://dx.doi.org/10.3745/JIPS.02.0020

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Robust Fingerprint Matching System Using Orientation Features

Ravinder Kumar*, Pravin Chandra**, and Madasu Hanmandlu***

Abstract
The latest research on the image-based fingerprint matching approaches indicates that they are less complex than the minutiae-based approaches when it comes to dealing with low quality images. Most of the approaches in the literature are not robust to fingerprint rotation and translation. In this paper, we develop a robust fingerprint matching system by extracting the circular region of interest (ROI) of a radius of 50 pixels centered at the core point. Maximizing their orientation correlation aligns two fingerprints that are to be matched. The modified Euclidean distance computed between the extracted orientation features of the sample and query images is used for matching. Extensive experiments were conducted over four benchmark fingerprint datasets of FVC2002 and two other proprietary databases of RFVC 2002 and the AITDB. The experimental results show the superiority of our proposed method over the well-known image-based approaches in the literature.

Keywords
Circular ROI, Core Point Detection, Image-Based Fingerprint Matching, Orientation Features

1. Introduction

In the past, thumb impressions were the single mode of identification of individuals in commercial, financial, and civilian domains. With the advent of fingerprint sensors, all fingers are used for the improved identification of individuals. A fingerprint-matching algorithm compares the ridge and valley patterns of two fingerprints and returns a matching score between 0 and 1, where 0 indicates a non-match and 1 indicates a perfect match. The score between 0 and 1 represents a partial match where the binary decision (match/non-match) is based on the threshold value varying from application to application. For example, in time monitoring and attendance marking applications, a higher threshold value may be used; whereas, in security/forensic applications, a low threshold value is chosen [1].

One of the approaches used for fingerprint matching is based on the comparison of minutiae pairs extracted from fingerprint images [1-4]. Minutiae points extracted from both the input and the query image are matched using a point-matching algorithm. This method is suitable for one-to-many matching of fingerprints, but is not compatible with all fingerprint sensor technologies, higher sensor

Manuscript received August 31, 2013; first revision November 14, 2014; accepted January 14, 2015; onlinefirst May 21, 2015.
Corresponding Author: Ravinder Kumar (ravinder_y@yahoo.com)
* Dept. of Computer Science and Engineering, Ansal Institute of Technology, Gurgaon 122001, India (ravinder_y@yahoo.com)
** University School of Information and Communication Technology, GGS Indraprastha University, Dwarka Delhi 110078, India (chandra.pravin@gmail.com)
*** Dept. of Electrical Engineering, Indian Institute of Technology, Haus-Khas Delhi 110016, India (mhmandlu@gmail.com)

resolutions, and large sensors. Minutiae-based methods may not lead to successful matching if the two fingerprint images do not have the same number of minutiae points and if they do not possess the portions of the fingerprints with significant information. A security system where the number of minutiae points is a limiting factor may not work. The false minutiae points resulting from a low quality of fingerprint ridge details also pose a problem during matching.

Recent approaches for fingerprint recognition consider the overall fingerprint characteristics rather than minutiae points alone, and they utilize more discriminatory information [5-16]. They employ image-based algorithms to deal with more data points in the form of curvature, density, and ridge thickness, which make them less dependent on the size of the fingerprint sensor. The perfect matching of various fingerprint patterns between the input and the query images is obtained by aligning them in the same direction. A threshold value describing the smallest allowable deviation is used for matching. These methods work well on fingerprints containing a fewer number of minutiae points and they also do not suffer from low quality, processing speed, and variable length feature vectors.

In this paper, we present an image based fingerprint matching system consisting of the following steps: preprocessing, core point detection, extraction of circular region of interest (ROI), alignment of extracted ROI by correlation maximization, extraction of orientation features, and similarity evaluation using the modified Euclidian distance. The highlights of this paper are as listed below:

- The proposed system is robust to rotation and translation alignment in which the best alignment of two fingerprints is achieved by maximizing the correlation between their extracted orientation fields.
- The proposed approach utilizes the orientation field (OF) features for evaluating the similarity between test and trainee fingerprint images This approach requires very few preprocessing steps as compared to other approaches in the literature, which require very complex and computationally expensive steps for preprocessing and the extraction of features.
- Circular ROI makes the proposed approach computationally efficient because only the region around the reference point is considered. This region has the most discriminatory information than that of the boundary regions.
- The modified Euclidian distance makes the proposed approach more robust to the matching, which allows an efficient selection of the threshold value [17].

The paper is organized as follows: Section 2 provides the details of the related works on the image-based fingerprint matching. Section 3 presents the systematic methodology of the proposed method. Section 4 gives an experimental design for database generation and matching. Section 5 discusses the experimental results, and conclusions are given in Section 6.

2. Related Works

Recently, image-based methods have gained popularity amongst researchers because of their relevance to automatic fingerprint matching. The prime objective of all these methods is to extract the discriminatory information from low quality fingerprint images where the performance of the minutiae-based methods is at stake. Most of the image-based methods bank on the local texture feature analysis using filter banks [1]. The Gabor filter-based approaches are proven to be effective in capturing

the local ridge features in both the spatial and the frequency domains [7,12]. A compact and fixed length new texture descriptor called fingercode, which was devised by Jain et al. [7], utilizes both the global and local discriminatory information of fingerprints. This descriptor extracts the texture features by applying Gabor filters in the region around the fingerprint reference point. Sha et al. [12] have proposed an improved fingercode for a filter bank based fingerprint matching system, which combines the directional features with average absolute deviation features. Another Gabor filter-based discriminator with low computational complexity called local binary pattern (LBP), which was proposed by Nanni and Lumini [9,10], is based on a multi-resolution analysis of the local fingerprint patterns. LBP provides the gray scale local texture descriptor. The fingerprints that are to be matched are first aligned using their minutiae and then these are divided into sub-windows. Each sub-window is convolved with 16 Gabor filters and the invariant LBP histograms are computed from the convolved image. The weighted Euclidean distance matcher is used for the evaluation of similarity between test and trainee fingerprint image. However, the performance is degraded on low quality images where all the minutiae points are not extracted precisely and reliably.

Gabor filter-based descriptors are not robust to rotation. To make them robust to rotation, each fingerprint has to be represented with 10 associated templates and the template with the minimum matching score is considered to be the rotated version of the input fingerprint image. Therefore, these methods require significantly larger storage space and higher processing time, and they also underperform due to a lack of effective alignment methods to make them robust to rotation and translation.

Tico et al. [13,18] proposed a method for fingerprint recognition based on local texture features extracted from the wavelet transform of a discrete image. The ROI of 64×64 pixels is cropped around the core point (detected manually). It is then divided into four non-overlapping blocks that are each of 32×32 in size and a set of wavelets up to level 4 are applied to each block. At each level the standard deviation of the wavelet coefficients is computed over 48 blocks to get a feature vector length of 48 for each fingerprint. The similarity is calculated using the Euclidian distance between the feature vectors of the test and the trained fingerprint images.

Amornraksa and Tachaphetpiboon [5] have also proposed a local texture analysis method based on the discrete cosine transform (DCT) for fingerprint matching. This method involves steps that are almost similar to the ones used by Tico's method, except that the feature vector length is six per block (or a total length of 24 for four sub-images) instead of 48. However, the two features in [5,13,18] are not robust to rotation.

Jin et al. [8], using the integrated wavelets and Fourier-Mellin transform (WFMT), developed a transform-based descriptor. This method is not vulnerable to rotation, transformation, and shape distortion. Multiple WFMTs are used to form a reference WFMT feature to reduce the variability of the fingerprint images. But this makes the matching process time-consuming.

Hybrid descriptors combined with Gabor filters are also suggested in the literature. Ross et al. [11] describe a hybrid approach that combines a set of minutiae with a ridge feature map extracted through a set of Gabor filters. The energy of the filtered image, which is comprised of the ridge feature map and a set of minutiae pairs, is used for evaluating the similarity between test and trainee fingerprint images.

Benhammadi et al. [6] have also proposed a hybrid descriptor by combining the orientations of the minutiae to construct the minutiae texture map. Considering the orientation features enhances the performance of hybrid descriptors, but the accuracy is degraded on low quality images.

In this paper, we present a robust fingerprint matching system that employs the orientation features for evaluating the similarity between test and trainee fingerprint images.

These features possess the most dominant characteristics of a fingerprint image and are extracted reliably and precisely even from low quality images. Our method applies the image enhancement to increase the clarity between ridges and valleys in conjunction with the automatic core point detection. Robustness to rotation and translation of the system is achieved by extracting a circular ROI with a radius of 50 pixels centered at the core point of the fingerprint images. To simplify the complex process of feature extraction as discussed above, we have used the orientation of the extracted ROI as the feature vector for matching. For the similarity measure, we have used the modified Euclidean distance method. A sequence of extensive experiments conducted on different databases proves the robustness and effectiveness of the proposed method.

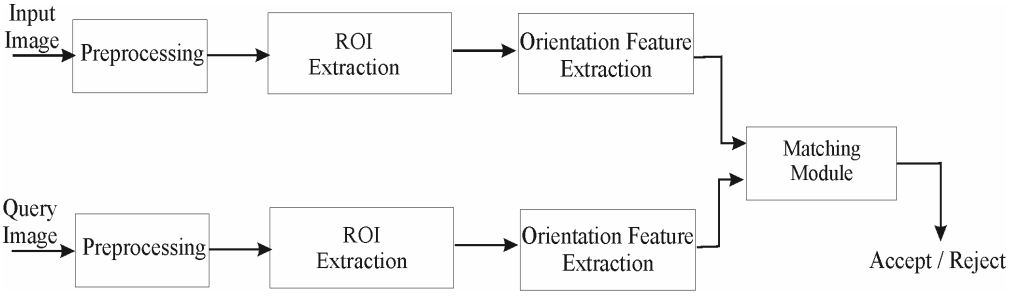

Fig. 1. Block diagram of proposed fingerprint matching system. ROI= region of interest.

3. The Proposed Algorithm

A block diagram shown in Fig. 1 describes the detailed methodology of the proposed fingerprint matching system. Both the input and the query fingerprint images are first preprocessed and then a circular ROI with a radius of 50 that is centered at the core point is extracted. The orientation features from each ROI are extracted and matched using the proposed modified Euclidean distance, which is further compared with the threshold to make the matching decision.

3.1 Preprocessing

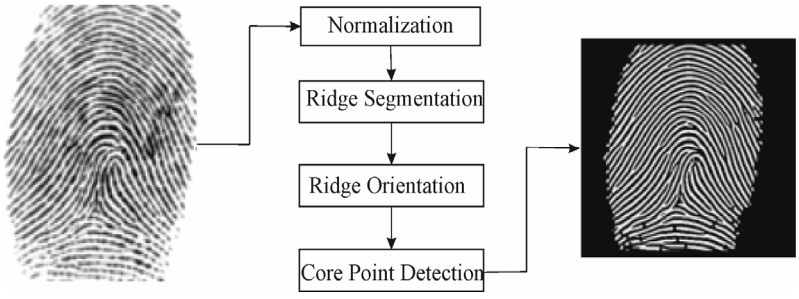

Fig. 2. Illustration of the preprocessing step.

Our intent is to improve the clarity between the ridge and furrow structure in the fingerprint images, thus facilitating the feature extraction process. The steps for the pre-processing stage are given in Fig. 2. The output of this stage is a processed image, which is suitable for further processing.

The following steps are performed at the preprocessing stage:

i. *Normalization:* an input fingerprint image is normalized so that it has the pre-specified mean and variance.

ii. *Ridge Segmentation:* the normalized fingerprint image is segmented to separate the background from the foreground.

iii. *Ridge Orientation:* the orientation value at each pixel of the segmented fingerprint image is estimated.

iv. *Core Point Detection:* the Poincare index method is used to detect a singular point (core point).

3.1.1 Normalization

The main objective of the normalization process is to reduce the variation in the gray level values along the ridges and valleys [16]. This operation does not change the clarity of the ridges and valleys and also does not influence the ridge breaks, intra-ridge holes, and parallel touching ridges.

Let $I(i, j)$, M_I, and V_I denote the gray level, and the estimated mean and variance of the image I, respectively. The normalized gray level $N(i,j)$ at the pixel (i,j) is defined as:

$$N(i,j) = \begin{cases} M_o + \sqrt{\frac{V_o \times [I(i,j) - M_I]^2}{V_I}}, & if\ I(i,j) > M_I \\ M_o - \sqrt{\frac{V_o \times [I(i,j) - M_I]^2}{V_I}}, & otherwise \end{cases} \quad (1)$$

where M_0 and V_0 are the desired mean and variance, respectively.

3.1.2 Ridge segmentation

Another important step in the fingerprint image preprocessing is segmentation. It is the process of separating the foreground (fingerprint area) from the background of an image. Segmentation is necessary to avoid the inclusion of false features, due to the presence of noise in the feature extraction process.

Thresholding is used in many applications for image segmentation due to it being the least complex and being easy to implement. In fingerprint images, the separation of the background and foreground is affected by the striped and oriented pattern and not by the average intensity [1]. For effective segmentation, local thresholding (block intensity) based methods are more effective as compared to global thresholding [1].

Mehtre et al. [19] have proposed a segmentation method for fingerprints and other images whose histogram is bimodal based on its directional image. This segmentation method, which is called the directional method, has some limitations in the case of images with a perfectly uniform distribution of gray regions, as pointed out in [20]. A composite method using variance and directional criterion overcomes the difficulties associated with the region of uniform gray values. Ratha et al. [21] used the

average masking on the orientation field of the window for image smoothing and then applied a waveform projection-based algorithm for the ridge segmentation.

We used a mean and variance based segmentation method for extracting the ridges from the background, as discussed in [21]. The algorithm below gives the segmentation method steps.

Algorithm for Segmentation

1. Divide the input image I into non-overlapping blocks of size w × w.
2. Compute the mean value of each block using:

$$\mathrm{M}(I) = \frac{1}{w^2} \sum_{i=-\frac{w}{2}}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}}^{\frac{w}{2}} I(i,j) \tag{2}$$

3. Compute the standard deviation $std(I)$ using the mean value from Step 3, as follows:

$$std(I) = \sqrt{\frac{1}{w^2} \sum_{i=-\frac{w}{2}}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}}^{\frac{w}{2}} \big(I(i,j) - M(I)\big)^2} \tag{3}$$

4. Empirically select a threshold value. If $std_I > threshold$, the block is considered to be a ridge region, otherwise, the block belongs to the background.

3.1.3 Ridge orientation

Computing the ridge orientation of the image is a common step in all algorithms dealing with reference point detection and fingerprint matching. Ridge orientation computation algorithms are classified into two categories: gradient-based methods [7,22-24] and filter-based methods [25,26]. Filter-based methods are less accurate and are also computationally more expensive than the gradient-based methods. This is due to the use of limited numbers of filters. However, these methods are less prone to noise.

In this paper, we have used the gradient-based approach for the computation of ridge orientation. The ridge orientation is denoted as $\Theta(i,j)$ (in radian), which is an angle between the ridges and horizontal axis representing the local ridge orientation at pixel(i,j). For the orientation computation, the algorithm described by Hong et al. [25] is used with a slight modification. The Gaussian operator in Eq. (4) is used to compute G_x and G_y, which are the gradient magnitudes in the x and y directions, respectively.

$$h_g(x,y) = e^{-(x^2+y^2)/(2\sigma^2)} \tag{4}$$

Because of the noise-corrupted ridge valley structures and low gray level contrast in the image, the computed local ridge orientation may not have the correct value. As the local ridge orientation varies very slowly in the neighborhood, except at the singular point, a low-pass filter is employed to adjust the erroneous local ridge orientation. To perform low-pass filtering, the orientation image needs to be

converted into a continuous vector field as follows:

$$\varphi_x = cos\ (2\Theta(x,y)) \tag{5}$$

$$\varphi_y = sin\big(2\Theta(x,y)\big) \tag{6}$$

where φ_x and φ_y are the x and y components of the vector field, respectively. The Gaussian low-pass filter is applied to the vector field using:

$$\varphi_x'(x,y) = \sum_{u=-\frac{l}{2}}^{\frac{l}{2}} \sum_{v=-\frac{l}{2}}^{\frac{l}{2}} w(u,v)\,\varphi_x(x-uw, y-vw), \tag{7}$$

$$\varphi_y'(x,y) = \sum_{u=-\frac{l}{2}}^{\frac{l}{2}} \sum_{v=-\frac{l}{2}}^{\frac{l}{2}} w(u,v)\,\varphi_y(x-uw, y-vw), \tag{8}$$

where w is a two-dimensional low-pass filter of size $l \times l$ for an unit integral. Finally, the local ridge orientation at each pixel (i,j) is obtained using Eq. (9).

$$\Theta(i,j) = \frac{1}{2} tan^{-1}\left(\frac{\varphi_x'(i,j)}{\varphi_y'(i,j)}\right) \tag{9}$$

3.1.4 Core point detection

To align the fingerprint images, detecting an accurate reference point is essential. The Poincare index method is widely used for singular point detection [1,27-30].

Let (i,j) be the position of an element θ_{ij} in the orientation image enclosed by a digital curve (with N points). Then, the Poincare index $P(i,j)$ is computed from:

$$P(i,j) = \sum_{k=0}^{N} \Delta(k) \tag{10}$$

where,

$$\Delta(k) = \begin{cases} \delta(k), & if|\delta(k)| < 90^{\circ} \\ \delta(k) + 180, & if|\delta(k)| \le -90^{\circ} \\ \delta(k) - 180, & otherwise \end{cases}$$

where $\delta(k) = \theta\big(i_{(k+1)\ mod\ N}, j_{(k+1)\ mod\ N}\big) - \theta(i_k, j_k)$ is the ordered difference between the two neighboring elements of θ_{ij} in the orientation image. The sequential ordering is done in a clockwise direction from 0 to $(N-1)$, and the size of the closed curve is chosen as 3 (i.e., N = 8, and $(k+1) mod\ 8$ signifies that $d_8 = d_0$). It can be easily shown that on the closed curves the Poincare index assumes only one of the discrete values: 0°, $\pm 180^{\circ}$, 360° in the case of fingerprint singularities. Thus, the Poincare index is set according to:

$$P(i,j) = \begin{cases} 0^{\circ}, & if\ pixel\ (i,j)\ does\ not\ belong\ to\ singular\ region \\ 360^{\circ}, & if\ pixel\ (i,j)\ belong\ to\ a\ Whorl\ type\ singular\ region \\ 180^{\circ}, & if\ pixel\ (i,j)\ belong\ to\ a\ Core\ type\ singular\ region \\ -180^{\circ}, & if\ pixel\ (i,j)\ belong\ to\ a\ Delta\ type\ singular\ region \end{cases}$$

In this work, we are only interested in the core points. As such, we have considered a region of interest by processing a 2×2 mask for detecting the core point with the value $180°$. Fig. 3 shows the detected core points using the Poincare index method for the left and the right loop images.

(a) (b)

Fig. 3. Detected core points shown by '+' for (a) left loop and (b) right loop.

3.2 ROI Extraction

For computing a feature vector we used the pre-defined area (ROI) around the detected reference point, rather than the entire fingerprint. We selected the circular ROI rather than a rectangular ROI, which introduces nonlinear distortion on rotation. We obtained a circular ROI by circularly cropping the fingerprint image $I(i,j)$ with a radius of 50 pixels around the reference point. This is denoted by $C(i, j)$, which is obtained by using:

$$C(i,j) = I(i,j), \forall\ i,j, such\ that\ \sqrt{(i-x)^2 + (j-y)^2} \leq 50 \tag{11}$$

where x and y are the coordinates of the reference point. The main advantage of circular cropping is that the area around the core point is the same for every rotation, which is not the case with a rectangular/square ROI mapped area around the core point.

4. The Experiment

We used a MATLAB 7.12.0 (R2011a) running on iCore5 setup processor installed with 4 GB of RAM using Windows 7 operating system for conducting all experiments reported in this paper. The databases used for the experiments are described in the rest of this section.

4.1 Databases

The proposed method was evaluated on fingerprint images taken from the public database FVC2002

set_A [31]. In this database, the fingerprint impressions are acquired by using the modern capacitive and optical sensors, as shown in Table 1.

Table 1. Summary of four FVC2002 databases

	Sensor type	Image size (pixel)	Set A (w×d)	Resolution (dpi)
DB1	Optical sensor	388×374 (142 K)	100×8	500
DB2	Optical sensor	296×560 (162 K)	100×8	569
DB3	Capacitive sensor	300×300 (88 K)	100×8	500
DB4	SFinGe v2.51	288×384 (108 K)	100×8	About 500

FVC2002 contains four distinct subsets: DB1, DB2, DB3, and DB4. Each dataset consists of fingerprint impressions for 100 subjects with eight impressions per subject at various rotations in the range of $-30°$ to $+30°$ (determined empirically) and translations. The resolution of the images in FVC2002 is 500 dpi (dots per inch) and in DB2 they are 569 dpi.

The second database was generated by selecting the first impression for each subject from all four datasets of FVC2002 by rotating them 10 times by an angle selected randomly between $-30°$ to $30°$. This generated 10 test images for each impression. This database, RFVC2002, was used to demonstrate the robustness of the proposed method for rotation and translation.

The third database was collected from 60 individuals using SecuGen Hamster IV FIPS 201/PIV, which is an FBI compliant optical fingerprint scanner. For each finger, 10 images were taken at different rotations, which resulted in a total of 600 images. All the images were scanned at a resolution of 500 dpi as recommended by the FBI and named 'AITDB'. Here, the database capturing was done in a controlled environment to have a resemblance to a real scenario. The empirically observed rotation was in the range of $[-35°$ to $40°]$.

4.2 Alignment of Images

The performance of any fingerprint matching algorithm relies on the perfect alignment of images. We maximized the correlation between the orientation values of pixels around the reference point of the image. The correlation coefficient ρ between the sample orientation image X and the query orientation image Y was computed from:

$$\rho = \frac{cov(X,Y)}{\sigma_x \sigma_y} \tag{12}$$

where the covariance is given by $cov(X,Y) = \sigma_{xy} = E[(X-\bar{X})(Y-\bar{Y})]$ and σ_x, σ_y are the standard deviations of X and Y. There is a perfect alignment of X and Y for $\rho = 1$, and they are unaligned for $\rho = 0$. If the value of ρ is between 0 and 1 they are partially aligned. Table 2 shows the computed correlation coefficient between the sample fingerprint image (FVC2002DB1_A /1_1.tif) and three query images FVC2002DB1_A /1_1.tif, FVC2002DB1_A /1_7.tif, and FVC2002DB1_A /1_8.tif).

As can be observed in the table, the best alignment is achieved at the maximum value of the correlation coefficient. The query image will have perfect alignment with the sample image when $\rho = 1$. Perfect alignment is observed for query images 2 and 3 at angles $-6°$ and $-7°$, respectively. The

negative value of the angle indicates anti-clockwise rotation, whereas, a positive value indicates clockwise rotation.

Table 2. Correlation coefficients between sample and query images at different values of orientation

Angle of rotation (in degree)	ρ		
	Query image 1	Query image 2	Query image 3
−30°	0.459	0.554	0.539
−20°	0.665	0.827	0.816
−10°	0.879	0.999	0.998
0°	1.000	0.970	0.968
10°	0.803	0.828	0.728
20°	0.616	0.692	0.572
30°	0.476	0.580	0.487

4.3 Extraction of Orientation Features

The orientation of the ridges is an intrinsic characteristic of the fingerprint images. To compute the orientation feature vector, the cropped image is divided into non-overlapping blocks that are 5×5 in size [32]. The column variance feature vector of the sample block is computed from:

$$\sigma_k = \sum_{i=1}^{n} (\Theta(i,k) - \mu_k)^2 \ \ for\ k = 1, \dots, m \tag{13}$$

And that of the query block is computed from:

$$\sigma'_k = \sum_{j=1}^{n} (\Theta'(i,j) - \mu'_k)^2 \quad for\ k = 1, \dots, m \tag{14}$$

where Θ and Θ' are the orientation fields of the cropped sample and the query images, respectively. μ_k and μ'_k are the mean value of the k^{th} column of the orientation images Θ and Θ', respectively, and m and n are the number of columns and rows of the cropped image blocks, respectively. Similarly, the row variance feature vectors of sample and query blocks are computed using:

$$\sigma_l = \sum_{j=1}^{m} (\Theta(i,j) - \mu_l)^2 \quad for\ l = 1, \dots, n \tag{15}$$

$$\sigma'_l = \sum_{j=1}^{m} (\Theta'(i,j) - \mu'_l)^2 \quad \text{for } l = 1, \dots, n \tag{16}$$

where μ_l and μ'_l are the mean values of the l^{th} row of the orientation images Θ and Θ', respectively. The next step is to compute the distance between the two feature vectors using:

$$d = \sqrt{\frac{1}{2}\sum_{k=1}^{n}(\sigma_k' - \sigma_k)^2 + \sum_{l=1}^{m}(\sigma_l' - \sigma_l)^2} \tag{17}$$

Finally, the computed distance was compared with the user specific threshold to make a decision of acceptance or rejection. The choice of threshold value depends upon the type of application.

5. Experimental Results and Discussion

The experiments were conducted to evaluate the performance of the proposed method on the database FVC2002 set_A (which contains four subsets: DB1, DB2, DB3, and DB4), RFVC2002, and AITDB. The proposed method operates in a verification mode where the false acceptance rate (FAR), the false rejection rate (FRR) and genuine acceptance rate (GAR) are evaluated as:

$$FAR = \frac{Number\ of\ accepted\ imposter\ claims}{Total\ number\ of\ imposter\ claims} \tag{18}$$

$$FRR = \frac{Number\ of\ rejected\ genuine\ claims}{Total\ number\ of\ genuine\ claims} \tag{19}$$

$$GAR = 100 - FRR \tag{20}$$

A series of experiments was conducted on the datasets of FVC2002, RFVC2002, and AITDB to compute the GAR and FAR at various user specific thresholds. The genuine matches and imposter matches were found as listed below.

- For genuine matches, each impression (say g) in the database was matched against the remaining impressions (say h) of the same finger to compute GAR. Accordingly, the FRR was computed as $100 -$ GAR for each subset. To avoid computation duplications, only matching either g with h or h with g was done. The total number of genuine tests for each dataset of FVC2002 (eight impressions per finger were matched against the remaining seven impressions for all 100 fingers for 100 subjects) was: $\big((8 \times 7)/2\big) \times 100 = 2800$.
  For RFVC2002 (10 impressions per finger were matched against the remaining nine impressions for all 100 fingers) was: $\big((10 \times 9)/2\big) \times 100 = 4500$.
  For AITDB (10 impressions per finger were matched against the remaining nine impressions of same finger for all 60 fingers) was: $\big((10 \times 9)/2\big) \times 60 = 2700$.
- For imposter matches, the first impression of each finger in the dataset as matched against the first impression of the remaining fingers in the dataset to compute FAR. The total number of imposter tests for each dataset of FVC2002 and RFVC2002 (100 fingers were matched against the remaining 99 fingers) was: $\big((100 \times 99)/2\big) = 4950$.

For AITDB (60 fingers were matched against the remaining 59 fingers) it was: $\left((60 \times 59)/2\right) \times 60 = 1770$.

We excluded the fingerprint image with the reference point that was detected very close to the boundary and also excluded the one with the reference point that could not be detected by the algorithm. A further evaluation of the proposed method was done by using an equal error rate (EER) at which FAR is equal to FRR [21]. The smaller the value of EER, the better the performance of the system. Table 3 shows the results in terms of GAR and FAR at various thresholds values.

Table 3. Summary of experimental results of the proposed method for various threshold values (unit, %)

Threshold value (%)	Performance parameter	FVC2002 Database				RFVC2002	AITDB
		DB1_A	DB2_A	DB3_A	DB4_A	DB_R	DB1
1.5	GAR	3	100.0000	99.5000	99.8000	99.9800	100.0000
	FAR	10.6700	12.6000	14.8000	10.0800	9.8000	7.8000
1.3	GAR	99.5500	99.5500	99.1000	99.5000	99.6000	99.7000
	FAR	6.0700	5.3500	10.6500	7.1500	3.9200	2.3400
1.1	GAR	98.1900	98.9200	98.1500	98.2000	98.9000	98.8000
	FAR	2.4900	1.2000	7.4000	4.0300	2.0500	0.7000
1.0	GAR	97.4500	97.9900	97.5900	97.9900	98.1900	98.5700
	FAR	0.7600	0.0300	5.4700	3.5200	0.8800	0.5400
0.9	GAR	97.0300	97.5300	97.1500	97.3500	97.8900	98.1300
	FAR	0.4900	0.1400	4.1400	2.1100	0.5800	0.3100
0.8	GAR	96.7800	96.9700	96.3700	96.6700	97.2700	97.8700
	FAR	0.3200	0.1000	2.5160	1.0000	0.4100	0.1700
0.7	GAR	96.1000	96.2200	95.3500	95.9000	96.7000	97.2700
	FAR	0.2000	0.0800	1.7700	0.6800	0.3100	0.0700
0.6	GAR	95.3400	95.5300	94.4800	95.3400	95.9400	95.9400
	FAR	0.1600	0.0700	1.3300	0.5700	0.2300	0.0230
0.5	GAR	94.9000	94.6800	94.0200	94.5000	95.2500	95.6500
	FAR	0.1300	0.0400	0.8800	0.4000	0.0900	0.0156
0.4	GAR	93.2300	93.5100	93.2300	93.9000	94.6300	94.9700
	FAR	0.1100	0.0200	0.5900	0.3000	0.0200	0.0078
0.3	GAR	92.6500	92.9700	92.5400	92.9500	93.9500	93.7500
	FAR	0.0900	0.0010	0.1480	0.2000	0.0100	0.0039
0.2	GAR	91.5600	92.2600	91.3600	91.9600	92.9600	92.9600
	FAR	0.0500	0.0000	0.0880	0.1008	0.0900	0.0008
0.1	GAR	90.9800	91.9300	90.4300	90.8300	91.8300	91.8300
	FAR	0.0000	0.0000	0.0296	0.0000	0.0010	0.0000
0.0	GAR	90.2100	90.9100	88.9100	89.9100	90.9100	90.9100
	FAR	0.0000	0.0000	0.0148	0.0000	0.0000	0.0000

GAR=genuine acceptance rate, FAR=false acceptance rate.

The average EERs on four datasets namely DB1, DB2, DB3, and DB4 of FVC2002, RFVC2002, and AITDB are 3.43%, 2.65%, and 32.34%, respectively, with an overall average of 3.12%. Figs. 4 and 5 show the ROCs of the proposed method on different databases.

It is shown in Fig. 4 that the average EER of the proposed method is better on the FVC2002 DB3 and AITDB than on other datasets and Fig. 5 shows that the best results are from AITDB and the RFVC 2002DB.

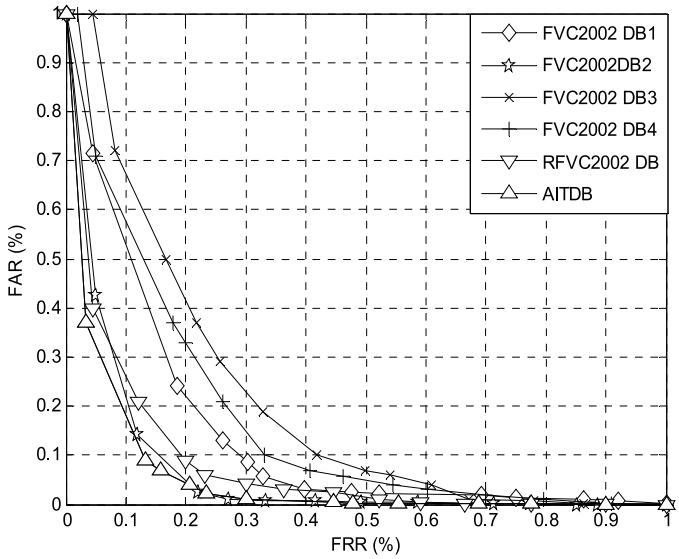

Fig. 4. ROC (FRR vs. FAR) of the proposed method on three databases FVC2002, RFVC2002, and AITDB.

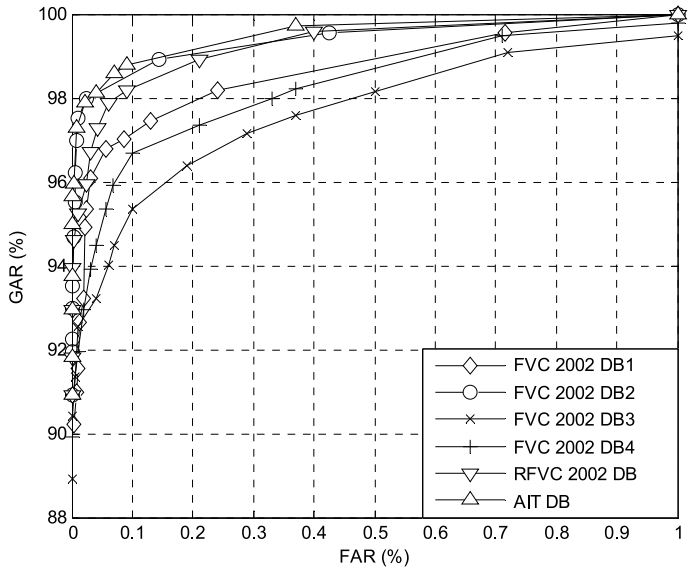

Fig. 5. ROC (FAR vs. GAR) of the proposed method on three databases FVC2002, RFVC2002, and AITDB.

5.1 Comparison with Other Image Based Matchers

A comparison of the proposed method is made here with other image based approaches, which include the following methods:

- GABOR is an image-based matcher that uses the Gabor features to describe the fingerprint image [11].
- LBP, local binary pattern based descriptor [9].
- DCT, discrete cosine transform based fingerprint matcher [5].
- WFMT, Fourier-Mellin transform based matcher [8].

We used the same parameters for experiments, as were used in the above methods in the literature. These methods were tested on FVC2002 for the comparative performance analysis in terms of EER (%). For evaluating the similarity between test and trainee fingerprint images, the Euclidian distance was employed. The results of these comparisons are given in Table 4 and the ROC curve is shown in Fig. 6.

Table 4. Equal error rate (%) comparison of proposed method with existing methods

Method	DB1	DB2	DB3	DB4	Average
GABOR [11]	**1.87**	3.98	4.64	6.21	4.17
LBP [9]	7.00	6.2	9.9	7.5	7.65
DCT [5]	2.96	5.42	6.79	7.53	5.68
WFMT [8]	2.43	4.41	5.18	6.62	4.66
Proposed	3.12	**2.89**	**4.34**	**3.4**	**3.43**

LBP=Local Binary Pattern, DCT=discrete cosine transform, WFMT=wavelets and Fourier-Mellin transform.

It is clear, as seen in Table 4 (figures in bold indicate the best results) and from ROC curve in Fig. 6, that the proposed method outperforms other image based matchers on DB2, DB3, and DB4, whereas, the Gabor-filter-based method performs better on DB1. Moreover, the average EER of the proposed method is better than that of the other methods.

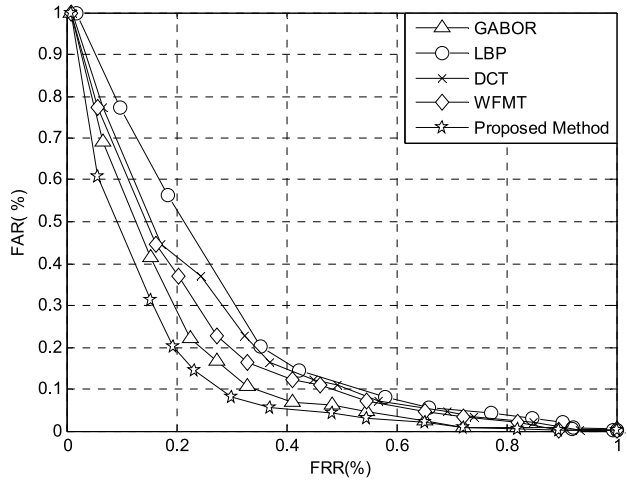

Fig. 6. ROCs for the comparison of the proposed method with the existing methods. LBP=Local Binary Pattern, DCT=discrete cosine transform, WFMT=wavelets and Fourier-Mellin transform.

5.2 Discussion

The superiority of the proposed method has been established by comparing its performance with that of the existing image based methods in the literature. This includes the score level fusion based hybrid matching system that utilizes the minutiae and ridge feature map that are extracted via Gabor filters [11]. The results of our proposed method were also compared with those of an invariant LBP based hybrid matcher [9], DCT [5], and WFMT [8] based methods. Our proposed method outperformed these methods on FVC2002 DB2, BD3, and DB4 and achieved an average EER of 3.43%. However, the Gabor-filter-based method outperforms on DB1 with an EER of 1.87% versus the proposed method, which has an ERR of 3.12%. This is because of the better quality of images in DB1.

We demonstrated the robustness of our proposed method by extracting the circular region of the interest around the extracted reference point. The circular ROI has a radial symmetry that ensures that the same region under angular displacement aligns with the ROI. We demonstrated the translational invariance by using the ROI around the reference point. The limitation of the proposed system is that it cannot handle images whose reference points are very close to the boundaries and images that do not have any landmark. To address the first issue, we assumed that matching should be performed using the proposed system in a controlled environment, where the query image should be scanned again if no reference point is detected or if the reference point is close to the boundary. To address the second issue, we used the complex filtering-based reference point detection method, which will return the ridge point with the highest curvature as the landmark point for the purpose of ROI extraction. The proposed system gives EERs of 2.65% and 2.43% on RFVC2002 and AITDB, respectively, and shows robustness to rotation and translation.

6. Conclusions

We have developed a simple and robust fingerprint matching system by exploiting the discriminatory information in the form of orientation features that are present in fingerprints. The use of circular ROIs does not change the shape and size of the ROI under rotation. The proposed method gives promising results and outperforms the well-known image based methods on DB2, DB3, and DB4 datasets of the FVC2002 set_A. To prove the effectiveness of the proposed method towards robustness, it was tested on the datasets that were generated by rotating the images from the FVC2002 database and a real AITDB.

The contributions of this paper are as follows: the exploitation of orientation features and the extraction of a circular ROI, which makes the proposed system computationally faster by avoiding the whole image and the improved matching.

Further work is required to design a mechanism for fingerprint matching with a partial circular ROI and to also reduce the size of the feature vector by tessellating the ROI into sub-windows.

References

[1] D. Maltoni, D. Maio, D., A. K. Jain, & S. Prabhakar, *Handbook of Fingerprint Recognition,* 2nd ed. London: Springer, 2009.

[2] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: a new representation and matching technique for fingerprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2128-2141, 2010.

[3] T. Y. Jea and V. Govindaraju, "A minutia-based partial fingerprint recognition system," *Pattern Recognition*, vol. 38, no. 10, pp. 1672-1684, 2005.

[4] N. K. Ratha, K. Karu, S. Chen, and A. K. Jain, "A real-time matching system for large fingerprint databases," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 799-813, 1996.

[5] T. Amornraksa and S. Tachaphetpiboon, "Fingerprint recognition using DCT features," *Electronics Letters*, vol. 42, no. 9, pp. 522-523, 2006.

[6] F. Benhammadi, M. N. Amirouche, H. Hentous, K. B. Beghdad, and M. Aissani, "Fingerprint matching from minutiae texture maps," *Pattern Recognition*, vol. 40, no. 1, pp. 189-197, 2007.

[7] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 846-859, 2000.

[8] A. T. B. Jin, D. N. C. Ling, and O. T. Song, "An efficient fingerprint verification system using integrated wavelet and Fourier–Mellin invariant transform," *Image and Vision Computing*, vol. 22, no. 6, pp. 503-513, 2004.

[9] L. Nanni and A. Lumini, "Local binary patterns for a hybrid fingerprint matcher," *Pattern Recognition*, vol. 41, no. 11, pp. 3461-3466, 2008.

[10] L. Nanni and A. Lumini, "Descriptors for image-based fingerprint matchers," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12414-12422, 2009.

[11] A. Ross, A. K. Jain, and J. Reisman, "A hybrid fingerprint matcher," *Pattern Recognition*, vol. 36, no. 7, pp. 1661-1673, 2003.

[12] L. Sha, F. Zhao, X. Tang, "Improved fingercode for filterbank-based fingerprint matching," in *Proceedings of International Conference on Image Processing (ICIP 2003)*, Barcelona, Spain, 2003, pp. 895-898.

[13] M. Tico, E. Immonen, P. Ramo, P. Kuosmanen, and J. Saarinen, "Fingerprint recognition using wavelet features," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2001)*, Sydney, Australia, 2001, pp. 21-24.

[14] J. C. Yang, S. Yoon, and D. S. Park, "Applying learning vector quantization neural network for fingerprint matching," in *AI 2006: Advances in Artificial Intelligence.* Heidelberg: Springer, 2006, pp. 500-509.

[15] J. C. Yang and D. S. Park, "A fingerprint verification algorithm using tessellated invariant moment features," *Neurocomputing*, vol. 71, no. 10, pp. 1939-1946, 2008.

[16] J. C. Yang and D. S. Park, "Fingerprint verification based on invariant moment features and nonlinear BPNN," *International Journal of Control Automation and System*, vol. 6, no. 6, pp. 800-808, 2008.

[17] R. Kumar, P. Chandra, and M. Hanmandlu, "Fingerprint matching based on orientation feature," *Advanced Materials Research*, vol. 403-408, pp. 888-894, 2011.

[18] M. Tico, P. Kuosmanen, and J. Saarinen, "Wavelet domain features for fingerprint recognition," *Electronics Letters*, vol. 37, no. 1, pp. 21-22, 2001.

[19] B. M. Mehtre, N. N. Murthy, S. Kapoor, and B. Chatterjee, "Segmentation of fingerprint images using the directional image," *Pattern Recognition*, vol. 20, no. 4, pp. 429-435, 1987.

[20] B. M. Mehtre and B. Chatterjee, "Segmentation of fingerprint images: a composite method," *Pattern Recognition*, vol. 22, no. 4, pp. 381-385, 1989.

[21] N. K. Ratha, S. Chen, and A. K. Jain, "Adaptive flow orientation-based feature extraction in fingerprint images," *Pattern Recognition*, vol. 28, no. 11, pp. 1657-1672, 1995.

[22] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 905-919, 2002.

[23] A. K. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 302-314, 1997.
[24] S. Wang and Y. Wang, "Fingerprint enhancement in the singular point area," *IEEE Signal Processing Letters*, vol. 11, no. 1, pp. 16-19, 2004.
[25] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777-789, 1998.
[26] A. R. Rao, *A Taxonomy for Texture Description and Identification*. New York: Springer, 1990.
[27] S. Jirachaweng, Z. Hou, W. Y. Yau, and V. Areekul, "Residual orientation modeling for fingerprint enhancement and singular point detection," *Pattern Recognition*, vol. 44, no. 2, pp. 431-442, 2011.
[28] T. Joshi, S. Dey, and D. Samanta, "A two-stage algorithm for core point detection in fingerprint images," in *Proceedings of TENCON 2009-2009 IEEE Region 10 Conference*, Singapore, 2009, pp. 1-6.
[29] K. Karu and A. K. Jain, "Fingerprint classification," *Pattern Recognition*, vol. 29, no. 3, pp. 389-404, 1996.
[30] M. Kawagoe and A. Tojo, "Fingerprint pattern classification," *Pattern Recognition*, vol. 17, no. 3, pp. 295-303, 1984.
[31] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2002: second fingerprint verification competition," in *Proceedings of the 16th International Conference on Pattern Recognition*, Quebec City, Canada, 2002, pp. 811-814.
[32] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 1009-1014, 2003.

RAVINDER KUMAR http://orcid.org/0000-0003-2117-5734

He received the M.Tech. degree in Computer Science & Engineering in 1998 from GJ University of Science and Technology, Hisar, India. Currently, he is Assistant Professor with Ansal Institute of Technology, Gurgaon and submitted Ph.D. to USICT GGSIPU Delhi, India. His research interest is in the image processing and biometrics.

Pravin Chandra http://orcid.org/0000-0002-6555-3832

He received Ph.D. in IT from GGSIP University, Delhi in 2004. Since 1999, he has been with the University School of ICT, GGSIP University, Delhi. Currently, he is professor at University School of ICT, GGSIP University and Controller of Examination of the University. His is interest is in the mathematical properties of neural networks and the relation of these properties to observed physical properties.

Madasu Hanmandlu http://orcid.org/0000-0002-2818-5479

He received the B.E. degree in Electrical Engineering from Osmania University, Hyderabad, India, in 1973, the M.Tech. degree in power systems from R.E.C. Warangal, Jawaharlal Nehru Technological University (JNTU), India, in 1976, and the Ph.D. degree in control systems from Indian Institute of Technology, Delhi, India, in 1981. He is currently professor in Electrical Engineering Department of IIT Delhi. He has guided 30 Ph.D. and 150 M.Tech. students. He has handled several sponsored projects. He is presently an Associate Editor of both Pattern Recognition Journal and IEEE Transactions on Fuzzy Systems and a reviewer to other journals such as Pattern Recognition Letters, IEEE Transactions on Image Processing and Systems, Man and Cybernetics. He is a senior member of IEEE.