Journal of Applied Sciences 4 (4): 611-618, 2004
ISSN 1607-8926

Multi-matcher Based Fingerprint Identification System

Nasir Rehan and Khalid Rashid
Department of Computer Science, International Islamic University, Islamabad, Pakistan

Abstract: Fingerprint identification system being the most developed and emphasized among the various practically employed biometric systems, has yet to make a long journey to prove its effectiveness in terms of speed and accuracy. Of various options for increasing systems efficiency, one may be the combination or fusion of various contemporary fingerprint identification systems. This study reveals the possibility for introducing a fingerprint identification system based on different fingerprint matching systems, working in close collaboration. Two different fingerprint representation schemes are evaluated and a merger point for the two is suggested.

Key words: Biometrics, fingerprint, efficiency, evaluation, combination

INTRODUCTION

Fingerprint identification systems are increasingly employed into business, trading and living fields for automatic personal identification. Though various fingerprint representation schemes have been developed and tested over the years but none had been able to yield satisfactory results. For example, a state-of-the-art fingerprint identification and classification algorithm by Anil *et al*.[1] reports accuracies of 92.2% for five and 94.5% for four class classifications. These claims prove to be false when tested in the laboratory environment. Though these may be great achievements but in situations where the desired error rate is negligibly small, then the need for further refining existing technologies becomes inevitable. Among many directions for addressing accuracy issues, there is an option for combining or running in parallel the existing fingerprint identification algorithms and coming up with a final decision (i.e. differentiating between genuine and imposter). This study takes into account two different fingerprint representation schemes and highlights fusion or merging point for both the representation schemes.

Minutiae-based fingerprint identification system: Most fingerprint representation systems follow the minutiae-based approach. A minutiae-based fingerprint verification system extracts minutiae points (Fig. 1) from the fingerprint images and matching decision is based on the correspondence of minutiae (i.e. it's type and position) between the stored template and current input fingerprint image.

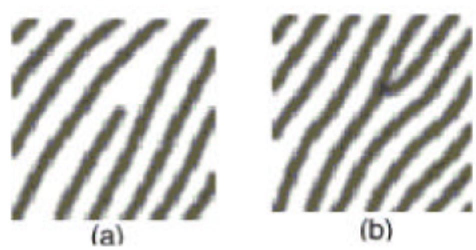

Fig. 1: Two most widely used minutia features
a) Ridge Ending b) Ridge Bifurcation

Minutiae-based fingerprint identification systems use a large number of successive processing steps. In general, following are the steps in a minutiae-based system:

- Orientation field estimation
- Segmentation
- Binarization
- Morphological operations
- Minutiae extraction
- Post-processing
- Registration of minutiae templates

A brief description of the above-mentioned processing steps is given below:

Orientation field estimation: The orientation field represents the directional flow of ridges and valleys in a fingerprint image. A number of methods have been proposed to estimate the orientation field of fingerprint image[2]. To calculate the ridge direction, the fingerprint

Corresponding Author: Nasir Rehan, Department of Computer Science, International Islamic University, Islamabad, Pakistan
Tel: 92-51-5470085 E-mail: nasirrehan@yahoo.com

image is divided into a number of non-overlapping blocks and an orientation representative of the ridges in the block is assigned to the block, based on the analysis of grayscale gradients within that block. The complete steps for orientation estimation algorithm by Hong and Anil[3].

Segmentation: The task of a fingerprint segmentation algorithm is to decide that which part of the image belongs to the foreground, originating from the contact of a fingertip with the sensor, and which part to the background. Accurate segmentation is especially important for reliable extraction of features like minutiae and singular points. Therefore, the main goal of the segmentation algorithm is to discard the background to reduce the number of false features originated due to noise. Segmentation algorithm by Asker and Sabih[4] gracefully segments the fingerprint image.

Binarization: The method of transforming grayscale pixel values to either black or white pixels corresponding to a ridge and valley, respectively is called binarization. The process can be carried out using a multitude of techniques[5]. Binarized image can be obtained using either global or adaptive thresholding. A brief description of the two is given below.

Global thresholding: Global thresholding involves the formulation of a histogram consisting of the number of pixels versus the pixel value. In this technique, the highest black and white pixel values within grayscale image are determined. The two major peaks found are assumed to be the most commonly used dark and light pixels. The middle range pixels will then be used to discriminate between black and white pixels for binarization (Fig. 2).

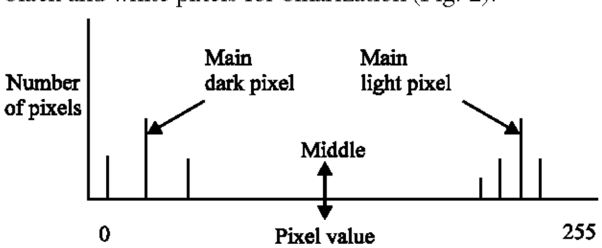

Fig. 2: Histogram for calculation of most numerous light and dark pixel values

Adaptive thresholding: The method, also known as contrast enhancement binarization, involves passing a low pass filter over the image and using the resulting grayscale pixel number to discriminate between black and white pixels. The low pass filter does not process edges that are one pixel wide in the image. Low pass filtering involves a spatial convolution process within a window (Fig. 3). These windows can be quite large. The image is convolved with this filter and the process is repeated until a new binarized image is obtained.

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Fig. 3: Convolution matrix (3×3) for low pass filtering

Morphological operation: Various morphological operations are performed on binarized image. The most important of them include thinning and spur removal. A thinned fingerprint image is one in which each ridge is one pixel in width. Thinning is achieved by successive deletion of pixels from all sides of the image. Each of the four sides are eroded away (Fig. 4) according to some set template[6].

If the image matches with template, the middle pixel is removed. Initially the two north images are processed, and then the other compass points are overlaid. Once all eight matrices have been sampled on the entire image, the process is repeated again on the newly formed image. Processing only stops when no more pixels can be deleted. Similarly spur operation is performed on thinned image to remove false bifurcations and ends caused by thinning. This removes some real end points from their original locations too. Rosenfeld and Kak explained how successive deletion works[7].

North	South	East	West
0 0 0 X 1 X X 1 1	1 1 X X 1 X 0 0 0	0 X 1 0 1 1 0 X X	X X 0 1 1 0 1 X 0
X 0 0 1 1 0 X 1 X	X 1 X 0 1 1 0 0 X	0 0X 0 1 1 X 1 X	X 1 X 1 1 0 X 0 0

Fig. 4: Eight matrices based thinning method

Minutiae extraction: Once the thinned ridge map is available, minutia location is not a tiresome job then. The technique uses a sample window (Fig. 5) to detect key minutia features (ridge endings and bifurcations).

Ridge pixels with one ridge pixel neighborhood are identified as ridge endings and those with three ridge pixel neighbors are identified as ridge bifurcations (Fig. 6 and 7). However, the entire minutia thus detected are not genuine due to image processing artifacts and the

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

The matrix checks for pixel values in each neighborhood of pixel P1 and then compares the outcome with the characteristics table

Fig. 5: Matrix for traversing through thinned image

CN	Characteristic
0	Isolated point
1	End point
2	Continuing point
3	Bifuraction poin

Fig. 6: Checking ridge characteristics

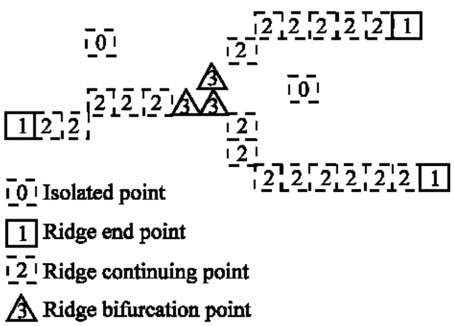

Fig. 7: Thinned ridged map with minutiae being marked

noise in the fingerprint image. The excess bifurcations and endpoints associated with noise must be removed to maintain accuracy levels[6].

Post-processing: At this stage, the genuine minutiae are gleaned from the extracted minutiae using a number of heuristics. For instance, too many minutiae in a small neighborhood may indicate noise and they could be discarded. Similarly very close ridge endings oriented anti-parallel to each other may indicate spurious minutia generated by a break in the ridge either due to poor contrast or a cut in the finger. Also two very closely located bifurcations sharing a common short ridge often suggest extraneous minutia generated by bridging of adjacent ridges as a result of dirt or image processing artifacts.

Thus all detected minutiae are passed through a process in which a minutia point is selected and all it's neighbors within a specified radius are marked and these minutiae points including the center or selected minutia point are all deleted (Fig. 8). In the end there are only 70 -90 minutiae points left for registration with the system.

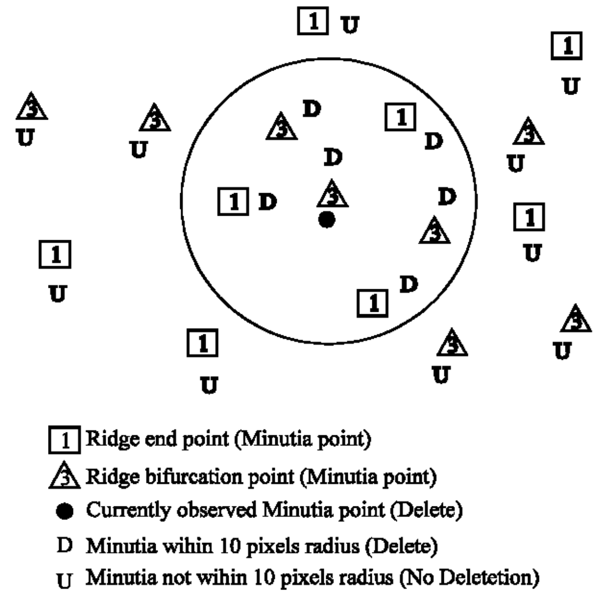

Fig. 8: Minutia deletion strategy

Minutiae registration: Feature extraction stage in minutia-based system is concluded by registration of acquired minutiae. Many features regarding a particular minutia point are recorded in the database. These features include information like total number of minutiae points detected, position of each minutia point, minutiae type, it's angle (direction) and most important of all, the minutia's position relative to other minutia points in the same fingerprint. A fingerprint can have up to 60-90 minutiae detected, and a valid match is generally accepted between two fingerprints if both prints have 8 to 17 common minutia points.

A microscopic approach, called minutia matching is commonly used for matching in minutia based fingerprint systems. Matching procedure involve each minutia's relative position to other minutia in a fingerprint (Fig. 9).

Total minutiae points	Minutia position (x, y)	Minutia angle/ direction	Minutia type (RE/RB)

Fig. 9: Most common minutiae information in database

Challenges associated with minutiae-based system: There are various factors or constraints, whose net affect adversely hinders the efficiency of minutia-based systems. Segmentation of fingerprint-textured images still remains a difficult problem in fingerprint identification systems. Without significant segmentation, the system's efficiency is questioned, as false minutiae are detected off the original fingerprint image. The most successful approach towards fingerprint segmentation includes Gabor Filters, which is the prime characteristic of Filterbank based fingerprint system[8], but this has it's own computational constraint which will affect the time within which the system generates the matching result.

The thinning algorithms haven't yet been designed to perfection. The correspondence between the original fingerprint image and the resulting thinned image has to be perfect, as it is at this stage the minutia points are determined and stored for use in matching stage. The contemporary thinning algorithms engulf many true minutia points and generate many false minutia points.

The heuristics employed for the deletion of minutia points further add to the deficiency of minutia-based systems. Many of the valid minutia points are deleted and certain false minutiae points survive for registration in the system database.

Filterbank-based fingerprint identification system: Filterbank based fingerprint identification system not only takes into account the local anomalies (i.e. minutia points) in the ridge structure but also the global pattern of ridges and furrows, inter-ridge distances, and overall pattern of ridge flow. This system is based on the fact that most textured images contain a limited range of spatial frequencies. Textured regions possessing different spatial frequency, orientation and phase can be easily discriminated by decomposing the image into several spatial frequency and orientation channels. Jain and Farrokhnia derived a global representation of texture by decomposing the input image into different frequency and orientation components using a Gabor Filterbank[9]. This representation for oriented texture of fingerprints was inspired by Daugman's work[10] on iris recognition and the success of Gabor Filterbank[8,9].

The four main steps in this filterbank based identification algorithm involve[8]:

- Reference point determination in fingerprint image and cropping,
- Tessellation of region around the reference point,
- Filtration of the tessellated area in six different directions using bank of Gabor Filters,
- Computation of Average Absolute Deviation (AAD) to yield feature vectors, also called the FingerCode, inspired by Daugman's IrisCode.

A brief description of the above-mentioned processing steps is given below:

Reference point location: The point of maximum curvature of the concave ridges in the fingerprint image is taken as the reference point. Center point location algorithm[4,8] gracefully determines the point of most curvature by determining the normals of each fingerprint ridge and then following them inwards towards the center. The algorithm has the following steps:

- Apply a pixel-wise adaptive 2-D Gaussian lowpass Wiener filter to reduce noise from the fingerprint image. The filter uses neighborhood of size 5 | x | 5 to estimate the local gradient mean and standard deviation.
- Divide the input fingerprint image into non-overlapping blocks of size 10 | x | 10.
- Determine x and y magnitudes of the gradient (G_x and G_y) at each pixel in each block, by taking the average of the two neighboring pixels.
- Apply the same 2-D Gaussian lowpass filter on the x and y gradients as above to smooth out the gradients.
- With each block, compute the slope perpendicular to the local orientation of each block using equation (1).

$$\Theta = \frac{1}{2}\tan^{-1}\left(\frac{\sum_{j=1}^{10}\sum_{j=1}^{10} 2G_x(i,j)(i,j)}{\sum_{j=1}^{10}\sum_{j=1}^{10} G_x^2(i,j) - G_y^2(i,j)}\right) + \frac{\pi}{2} \quad (1)$$

- Only looking at blocks with slope values ranging from 0 to π/2, trace a path down until a slope is encountered that is not ranging from 0 to π/2 and mark that block (Fig. 10).

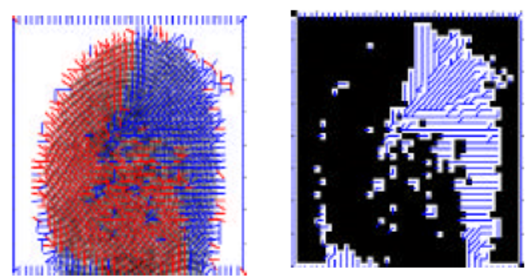

Fig. 10: Depicting inward slope determination

- The block that has the highest number of marks will compute the slope in the negative y direction and output an x and y position which will be the center point of the fingerprint (Fig. 11).

Once this point is determined, fingerprint image is then cropped to a 223 | x | 223 pixels centered on the calculated reference point.

Fig. 11: Picking up block with higher number of markings

Tessellation: The region of interest in the fingerprint image is the collection of all sectors around the center point where each sector is computed in terms of a particular radius and angle. This demarcation of the cropped area in sectors is called tessellation. The cropped fingerprint image is divided into 5 concentric bands centered on the reference point. Each band has a radius of 20 pixels except for the inner most, which has a radius of 12 pixels. So the total radius of the sectorization is 111 pixels similar to that of cropped image. Each band is divided into 12 sectors (Fig. 12).

Fig. 12: Reference point marked and area of interest tessellated

The sectorization is performed as 6 equi-angular Gabor filters[8] are used for feature extraction. These Gabor filters are aligned with the 12 wedges formed by the bands. In other words, each sector will capture information corresponding to each Gabor filter. The center band is ignored because it has very small area to be of any use. The radius of the sectorization is chosen to avoid the effects of circular convolution in applying Gabor filters. Another reason for sectorization is the normalization process. Each sector is individually normalized to a constant mean and variance to eliminate variations in gray-scale values in the fingerprint pattern, which are incurred due to scanning noise and pressure variations.

Equation (2) is used for normalizing pixel intensities in each sector of the cropped fingerprint image.

$$N_1(x,y) = \begin{cases} M_0 \dfrac{\sqrt{V_0 \times (I(x,y) - M_i^2)}}{V_i}, & \text{if } I(x,y) > M_i \\ M_0 \dfrac{\sqrt{V_0 \times (I(x,y) - M_i^2)}}{V_i}, & \text{otherwise} \end{cases} \quad (2)$$

M_0 is constant mean and V_0 is constant the variance, both having values = 100. I is sector number, M_i is the mean of sector, and V_i is the variance of sector.

Gabor filtration: The normalized image is then passed through a bank of Gabor filters. Each filter is performed by producing a 23 x 23 filter image for 6 angles (0, π/6, π/3, π/2, 2π/3 and 5π/6), and convolved with the fingerprint image. Spatial domain convolution is rather slow, so multiplication in the frequency domain is preferred. However, this involves more memory to store real and imaginary coefficients. The purpose of applying Gabor filters is to remove noise while preserving ridge valley structures and to provide information contained in a particular direction in image. A fingerprint convolved with a 0-oriented (0 degree) filter accentuates those ridges that are parallel to that angle (0 degree) and smoothes out the ridges of all other directions. Filters tuned to other angles work in similar manner. These six directional-sensitive filters capture most of the global ridge directionality information as well as the local ridge characteristics present in fingerprint (Fig. 13)[8].

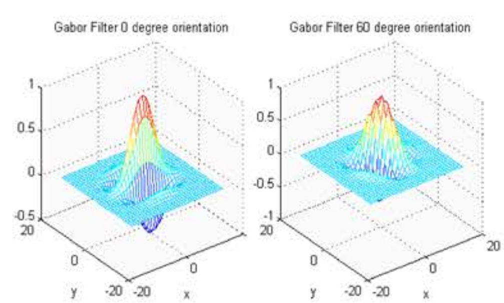

Fig. 13: 0 and 60 degree oriented Gabor filter

Equation (3) is the definition of Traditional Gabor filter[8].

$$G(x, y, f, \theta) = \exp\left\{\frac{-1}{2}\left[\frac{x'^2}{\delta_x^2} + \frac{y'^2}{\delta_y^2}\right]\right\}\cos(2\pi f x')$$

$$x = x\sin\theta + y\cos\theta, \quad (3)$$

$$y = x\cos\theta - y\cos\theta$$

The parameters, δ_x and δ_y are Gaussian constants whose values are empirically determined and set to 4.0. Very small value will make the filter ineffective in removing noise while too large a value will destroy ridge and furrow details. The filter frequency f is set equal to average inter ridge frequency (1/k), where k is the average inter-ridge distance which is determined to be on average 10 pixels in 500 dpi image.

Feature vector: After obtaining 6 filtered images, the variance of pixel values is calculated in each sector. Higher variance value in a sector means that the ridges in that sector are parallel in direction to that of applied Gabor filter. Whereas the low variance value indicates that the ridges are orthogonal to the angle of applied Gabor filter. So the filtering will smooth out the orthogonal (non-parallel) ridges. Corresponding to tessellation, a total of 360-variance values (6 angles × 60 sectors) are extracted known as feature vectors of the fingerprint.
Equation (4) is used for calculating variance values:

$$V_{i\theta} = \sqrt{\sum_{K_i} (F_{i\theta}(x, y) - P_{i\theta})^2} \quad (4)$$

Where, $F_{i\theta}$ represents the individual pixel value in the ith sector. $P_{i\theta}$ is the mean of the pixel values. K_i is the number of pixels in the ith sector.

Challenges associated with filterbank-based system: Filterbank based representation[8] is a newly emerging technique. Though it is just in its nurturing stage but still it has achieved remarkable success. Various factors that cause setback to its efficiency are:

First step in this fingerprint representation is the reference point determination (core point location) which itself, in certain cases is very much unreliable. Even though this multi-resolution reference point location algorithm is accurate but it fails to detect correct reference point in very low quality images leading to a reject or even worse, a false rejection/matching in the identification system. The reference point location algorithm also has higher error rate in consistently locating the reference point in arch type fingerprints due to the absence of singular points (maximum curvature ridge). This factor can be addressed by introducing multiple reference points (core and delta point location).

The current implementation is not rotation invariant. For this reason the rotation is handled in matching stage, which adds to the time for matching thereby hindering system's efficiency.

The current implementation of Filterbank representation extraction takes longer than a typical minutia-extraction algorithm. This is due to a large bank of Gabor filters (depending on the number of angles applied i.e. 6 or more angles) convolved with the original image. Gabor filtering is an expensive process in terms of computation. More than sixty percent of systems operational time is attributed to Gabor filtering stage. Replacement can be thought of a single rotation invariant Gabor filter that not only accentuates the ridges in all directions in a single convolution but will also reduce the size of Finger-Code (feature vectors) used for fingerprint matching. For the time being, convolution can be made significantly faster by dedicated DSP processors or performing the filtering in the frequency domain. Furthermore the presently employed matching algorithms are simple. An improved matching algorithm would improve the overall systems performance.

Proposed multi-matcher based fingerprint identification system: The architecture of the proposed system, for multiple-matcher based fingerprint identification system (Fig. 14). Fingerprint system based on multi matcher can be of many variations. The architecture highlights only one of many combinational or decision level integration point. For example, the proposed system relies first on one matcher but when that representation scheme seems to fail or degrade, the task is handed over to the other representation scheme. Another possibility could be that both matchers could be run in parallel and at the end the final decision could be reached upon by evaluating the result of the individual matcher.

The system put forth gives more importance to filterbank matcher than minutia. Because minutia matcher, being older and mature in terms of fingerprint identification technology, haven't got any definite failure or weak point. Its performance doesn't deteriorate due to a single factor, as is the case in filterbank-based system (i.e. reference point calculation algorithm though extremely efficient but at times it is also extremely unreliable factor). Also minutia-based system had been the focus point for fingerprint systems for a very long time now and it has evolved to its full strength and further developments in this particular direction have ceased to some extent or are very slow. Therefore alternate contemporary techniques should be given a chance for further nourishment.

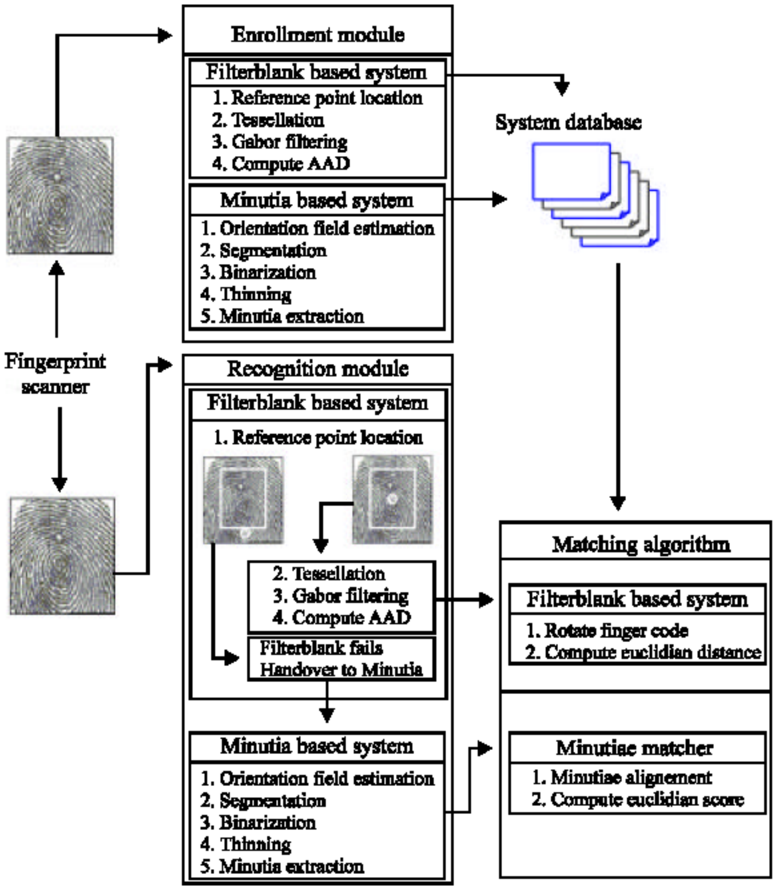

Fig. 14: Architecture of proposed multi-matcher based fingerprint identification system

This multi-matcher based system in its enrollment module extracts and stores fingerprint signature using both Filterbank and Minutia based systems accordingly. Next time when the person comes for authentication then his fingerprint signature is first extracted using Filterbank technique. Here we have made a check for reference point that if it is marked at a point (near the edge of fingerprint image), corresponding to which the system is unable to select the area for tessellation, then the system doesn't entertain the task of authentication through Filterbank technique. The authentication task is handed over to minutia system whose outcome will definitely be not as deteriorating as would have been the case, if Filterbank technique alone had been employed.

There may be certain constraints associated with these multi-matcher based systems like efficiency in terms of speed, decision level fusion when both the systems comes up with their own authentication outcomes, separate database management, etc. But these issues aren't hard to overcome and will definitely be addressed once an upper bound for the performance of multi-matcher based fingerprint identification system is established.

CONCLUSION

Multi-matcher based fingerprint identification systems though have been talked about a lot but haven't yet received any practical consideration until now. If appropriate efforts been utilized in this direction it may prove to be the cure for deficiencies that are present in the present day fingerprint identification systems.

Presented in this study is just one of many possible variations of multi-matcher based fingerprint systems. Further investigations will also focus on the precedence of fingerprint matchers for decision making in a multi-matcher based fingerprint identification system relative to their individual performance results obtained when operated individually.

REFERENCES

1. Jain, A.K. , L. Hong, S. Pankanti and R. Bolle, 1997. An identity-authentication system using fingerprints. Proc. IEEE., 85: 1365 1388.
2. Kawagoe, M. and A. Tojo, 1984. Fingerprint pattern classification. Pattern Recognition, 17: 295-303.

3. Hong, L. and A.K. Jain, 1999. Classification of fingerprint images. 11th Scandinavian Conference on Image Analysis; Kangerlussuaq, Greenland.
4. Asker, M.B. and S.H. Gerez. Segmentation of Fingerprint Images, Department of Electrical Engineering, University of Twente. Laboratory of Signals and Systems.
5. Klette, R. and P. Zamperoni, 1995. Hand Book of Image Processing Operators. John Wiley and Sons, New York.
6. Anthony, L., 1999. Fingerprint Recognition. Department of Electrical Engineering, University of Queensland, St. Lucia, 4067.
7. Kasaei, K., M. Deriche and B. Boashash, 1997. Fingerprint feature enhancement using block-direction on reconstructed images. International Conference on Information, Communications and Signal Processing, pp: 721-725.
8. Anil, J., S. Prabhakar and L. Hong. Fingercode: A Filterbank for Fingerprint Representation and Matching, CPS Technical Report Library, MSU.
9. Anil, J. and F. Farrokhnia, 1991. Unsupervised texture segmentation using gabor filters. Pattern Recognition, 24: 1167-1186.
10. Daugman, J., 1993. High confidence recognition of persons by a test of statistical independence. IEEE Trans. Pattern Anal. Machine Intell., 15: 1148-1161.