# Ridge Distance Averaging Based Matching for Interoperability of Fingerprint Sensors

M. Selvakumar[#1], D. Nedumaran[*2]

*Dept. of Central Instrumentation and Service Laboratory[#*]*
*University of Madras*
*Chennai-600025, Tamil Nadu, India*

*Abstract*— **Fingerprint recognition techniques on minutiae-based matching algorithms have been well studied and widely accepted by many fingerprint biometric researchers. Fingerprint images taken from different sensors like swipe, thermal, optical, capacitive, etc., have different sensor characteristics like change in resolution, image size, aspect ratio, and distortion. A fingerprint image taken from a capacitive swipe sensor and the same image from an optical press sensor does not match while using minutiae-based matching techniques due to sensor characteristics called sensor interoperability. Many algorithms were developed for sensor interoperability problems, but still the problem is not completely addressed. In this paper, the interoperability of sensor matching is addressed using a simple and effective ridge distance averaging method. This proposed method was tested in the images of the fingerprint database FVC 2004, 2002 and 2000 as well as locally scanned images using various types of sensors with the same finger. The performance of the proposed method in reducing the interoperability of sensor problem was studied through a comparative study with the established minutiae matching methods. The experimental results reveal that the proposed method improves the sensor interoperability by matching the fingerprint images taken for the same figure using various types of sensors whereas the minutiae based matching method failed to match those images effectively.**

*Keywords*——**Bio-metrics, Ridge distance averaging, sensor interoperability, fingerprint minutiae matching, capacitive sensor and optical sensor.**

## I. INTRODUCTION

Biometric fingerprint recognition is one of the best measurement techniques for unique human identification [1, 2] than the other biometrics. Many algorithms for fingerprint recognition were proposed by several researchers and they were implemented for different applications [3, 4]. Various fingerprint sensors with different provisions (inbuilt algorithms) e.g., sensing technology, variation in image size, resolution, contrast quality, internal image enhancement, etc., are now readily available for the fingerprint sensor designers [5]. Numerous fingerprint capturing devices [6-8] were built with internal inbuilt algorithms, whereas some problems arose because of change in sensors or sensor interoperability problem [9-11]. Fingerprint images are captured by different sensing phenomena which show remarkable changes in image properties because of the differences in the characteristics of the sensing equipment. Changes in capturing devices or sensors like Optical (FTIR sensor or the electro-optical sensors), Solid-state sensors (capacitive sensor or the

thermal sensors), radio frequency, ultrasound sensors etc., lead to the following consequences:

- Change in lighting leads to high, low, or medium contrast gray level image capturing,
- Change in size and resolutions of fingerprint images captured due to distinct sensor characteristics,
- Change on contact pressure of the fingertip with the capturing device,
- Change in dry weather contaminants (oil and moisture) resulting in faded fingerprint capturing,
- Distortion between the fingerprint ridges due to movement artifact while capturing,
- Change in fingerprint images due to cuts or scars on the fingers while capturing.



Fig. 1(A) Image Captured By Capacitive (Swipe) Sensor Shown In (C) and (B) Image Captured By Optical (Press or Thumb) Sensor Shown In (D)

In this paper, the problems in matching the fingerprints due to sensor interoperability were studied and the solution for the same in the form of algorithm was developed. In this study, only two capturing devices are used, one is a capacitive (swipe) sensor and another is an optical (press) sensor, since such kind of swipe and press sensors images are not available in the databases employed. Capacitive swipe sensors capture the image with closer ridge distance between two parallel ridges. Similarly, optical press sensor captures the image with evenly spaced ridge distance

between two parallel ridges as shown in Fig. 1. In Fig 1, the images were taken from two different sensors (capacitive swipe and optical press) for the same finger. Fig. 1(a) shows the fingerprint image taken from a capacitive swipe sensor (Eikon) shown in Fig. 1(c). Similarly, Fig. 1(b) shows the fingerprint image taken from optical press sensor (Futronics) shown in Fig. 1(d). Detailed explanations about the sensors are given in Section III.

The fingerprint images from the capacitive swipe and optical press sensors do not match if the popular and effective minutiae-based matching method is used, due to the distance dissimilarity between the two neighbouring ridges. In this paper, we have used FVC databases [12-14], FVC 2000, FVC 2002 and FVC 2004 databases captured using various types of sensors and locally captured public images using optical press and capacitive swipe sensors for inspecting the proposed algorithm.

### A. The Sensor Interoperability Issues

A fingerprint image captured by one sensor should be matched with same fingerprint images captured by another sensor, ensuring the interoperability (or compatibility) between two different sensors. Therefore, the interoperability between the captured fingerprint images is an important issue, but only some researchers have addressed this sensor interoperability problem.

To solve the sensor interoperability problem, the common resolution and the relative resolution methods were applied to image-level and minutia-level to perform the minutiae level compensation [9]. Cao et al. adopted an ant-colony optimization algorithm for the compensation of the resolution difference of fingerprint images captured from different modes of acquisition [10]. Sensor interoperability problem was analysed based on the feature usability in segmentation [15-17]. Ross and Nadgir [18] proposed a nonlinear calibration scheme based on the Thin-Plate Spline (TPS) model, which demonstrated a pair of fingerprint sensors to calibrate the evidence of a few image pairs, acquired using the two different sensors that generated an average deformation model. To address the interoperability related problems, sensors with inbuilt quality check for enhancement process [19-20] and with inbuilt fingerprint ridge continuity check for decreasing the number of attempts in capturing the fingerprint images [21] were also attempted. Before processing, an algorithm for fingerprint image recognition identified the sensor from the captured fingerprint image based on the inherent noise characteristics of the respective sensors [22, 23] and multiple enrolment impressions were captured for the same finger using the same sensor in order to increase the coverage area, restore the missing minutiae and eliminate the spurious minutiae. Most of these techniques discussed so far attempted the interoperability of sensor and related matching techniques to certain extent, since matching is cumbersome when the same finger is captured with different sensors. Therefore, sensor interoperability problem is still a challenging task and this research work

attempts this problem by developing a novel matching algorithm and testing its applicability through performance study.

This paper is organized as follows: Section II describes the proposed fingerprint matching method. Section III describes the experimental results which include performance comparisons between images, FVC database and sensor database. A summary and scope for future work is presented in Section IV.

## II. PROPOSED FINGERPRINT MATCHING METHOD

Matching the minutiae is one of the best techniques to use due to its less template size, speed of verification, identification process, etc. Verifinger [24] is one of the popular biometric algorithms used for fingerprint matching process, which is based on minutiae matching technique that maintain equal distance between the neighbouring minutiae points of the matching images and are indicated in red lines as shown in Fig. 2. The Verifinger algorithm works under the Delaunay triangulation (DT) method in computational geometry process [25]. DT algorithm is used to form a DT net of minutia sets. The time complexity of the algorithm is, $O(N \log N)$, here $N$ is the number of minutiae.

Using DT nets, the template image and the input image are taken as minutiae edges. Minutia can be described as a vector $(x_t, y_t, \theta_t)^T$ in template image or as $(x_i, y_i, \theta_i)^T$ in input image. Meanwhile, edge can be described as a vector $(\theta_{th}, \theta_{tb}, \gamma_t, l_t)^T$ in template DT net or $(\theta_{th}, \theta_{tb}, \gamma_i, l_i)^T$ in input DT net. In vectors, $x, y$ denote x, y coordinates of minutia, $\theta, \gamma$ denote minutia angle and orientation of edges, respectively and $l$ denotes length of edge. In $(\theta_{ih}, \theta_{ib}, \gamma_t, l_t)^T$, $\theta_{th}$ corresponds to the vertex with smaller x coordinates and $\theta_{tb}$ corresponds to the vertex with larger x coordinates. The same vector can be described as $(\theta_{ih}, \theta_{ib}, \gamma_i, l_i)^T$.

Here, $T$ can be used as $T_1$, $T_2$ and $T_3$ that represent the thresholds for the length of edge $l$, minutia angle $\theta$ and orientation of edge $\gamma$, respectively. If template DT net has $M$ edges and input DT net has $N$ edges, the total number of edge comparison will be $MXN$. If no similar edge is found, the local matching end with rejection, which means the two fingerprints are not generated from the same finger.
Triangle matching is estimated using $T$ and $l$. On the left edge of the triangle is used to detect the next edges on right side. The edges on right side and the triangles are used to find the next edges on left side. If they are internal edges, then the triangle search two sides of them. Therefore, every pair of similar edges can produce one or two pairs of triangles.

From the DT nets, matching score is calculated using the relationship $MS = Max / Sqrt(M \times N)$, where M and N are number of minutiae points in the template.

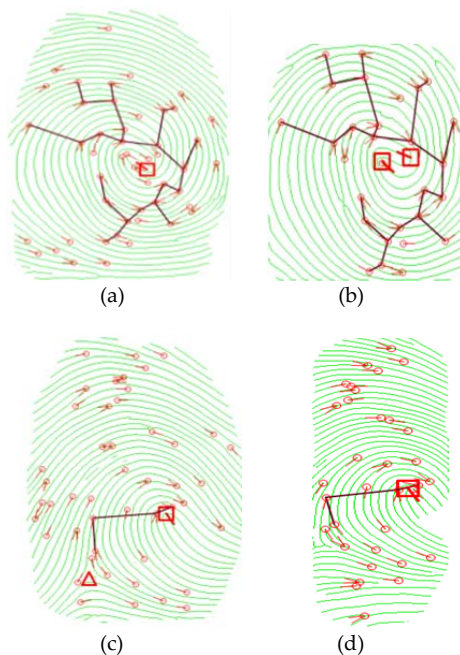

(a)                    (b)

(c)                    (d)

Fig. 2 (a) and (b) are the images captured from two different optical sensors. The red line connects the minutiae points using DT nets method, which indicate that the distance between the minutiae points are found to be same for both images. Fig. 2(c) and (d) are the images taken from optical and capacitive sensors, respectively. Here, the distance between the minutiae points are not equally spaced (triangular edges match missing), then the algorithm reads as "templates do not match".

Here the algorithms performances were found to be good in the case of same sensor but shows error in interoperability of sensors. Fig. 2(a) and (b) are the images captured using two different optical sensors that give templates having minutiae on the equally spaced ridge patterns for both the images. Similarly, Fig. 2(c) and (d) are $320 \times 480$ in size with 512 dpi resolutions and $144 \times 383$ in size with 500 dpi resolutions, respectively. These images captured using optical press and capacitive swipe sensors, in which the ridge distance is not maintained equally. If the minutiae-based algorithm is applied to these images (Fig. 2(c) and (d)), it gives templates that are not matched or matching score of zero due to the difference in the ridge distance of the two images. On the other hand, the neighbouring minutiae points in one image may be either closer or larger than the corresponding minutiae points in the other fingerprint image. This is the problem prevailing with the minutiae-based fingerprint matching algorithm. In this work, the aforementioned matching problems are attempted by developing our proposed algorithm.

The traditional minutiae based matching procedure comprises of the regular pre-processing steps viz., segmentation, enhancement, binarization, thinning and matching as shown in Fig. 3(a). In the matching step, the minutiae points are identified and the template is generated irrespective of the size of the image, scanner type, ridge distance etc. The proposed fingerprint matching procedure

keeps all the pre-processing steps as such except an additional step between the thinning and matching steps as shown in Fig. 3(b). At this step, we estimated the average ridge distance, which is used to increase or decrease the distance between the two neighbouring ridges of the other image in order to identify the correct minutiae points for determining the matching template or high matching score.
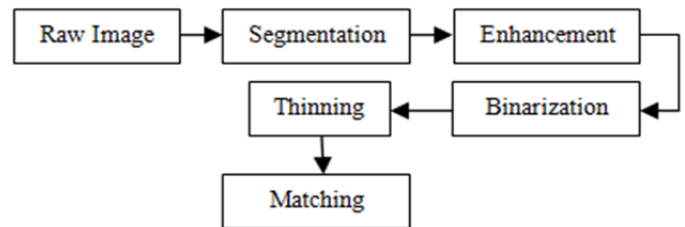


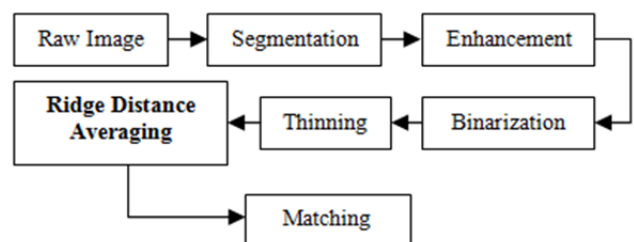Fig. 3 (a) traditional minutiae based fingerprint matching procedure.



Fig. 3 (b) proposed fingerprint matching procedure

### A. The Ridge Distance Averaging Method

Ridge distance averaging method has a simple and effective procedure as follows.

- Before the thinning process, the thickness of ridges and valleys varies from top to bottom of the fingerprint image.
- After the thinning process, the ridge thickness becomes same (of one pixel width) and the distance between two neighbouring ridges (valley pixel region) varies from top to bottom of the fingerprint image.
- Since the ridges are one pixel width thickness, it is easier to estimate the (u, v) matrix using statistical analysis for finding the average distance between the neighbouring ridges. In this work, a $32 \times 32$ matrix is employed to accommodate the maximum value pixel region in a block, since smaller matrices like $3 \times 3$ or $5 \times 5$ matrices taken more time to complete the entire process.
- Once the ridge pixel averaging process is estimated, then the Algorithm for ridge distance averaging method is applied to increase or decrease the number of pixels between two neighbouring ridges based on the average ridge distance value.
- The Algorithm automatically increases or decreases the number of pixels between the ridges to identify the matching minutiae point in order to reach a high matching score.

### B. Statistical Analysis to Find the Average Ridge Distance

The ridge distance is calculated between the two matching images using statistical analysis [26, 27]. Before

applying the statistical analysis, we have to consider the following to avoid inaccuracy in the estimation of the average ridge distance viz., ridge lines should be clear; the pixel contrast is more significant; the ridges at the minutiae points (ridge ending, ridge bifurcation, bending ridges) must be clear, that is, the pre-processing stage will be effective to perform the statistical analysis. Fig. 4 shows the thinned image ridge distance identification by the statistical method. The closer the ridge patterns to a given periodic behaviour, the greater the coefficient of the harmonic with that period. As spectral analysis enables the computation of these harmonic coefficients, it may also give effective estimates of the average ridge period.
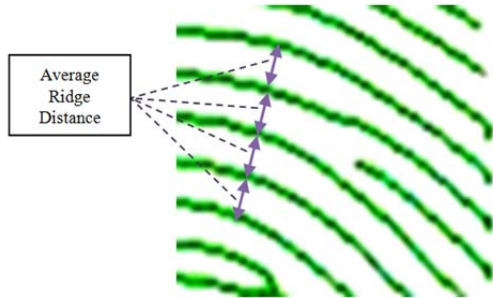


Fig. 4 definition of finding the ridge distance by a statistical analysis method in a thinned image.

If $g(x, y)$ is the gray-scale value of the pixels with coordinates $x, y \in \{0, K, N-1\}$ in an $M \times N$ image, the DFT of $g(x, y)$ is defined as follows,

$$G_{(u,v)} = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} g(x, y) e^{-2\pi j / N((x,y)(u,v))} \qquad (1)$$

where, $e^{-2\pi j / N((x,y)(u,v))} = \left( \left( \cos\left( -\frac{2\pi}{N}(x, y)(u, v) \right) \right) + j\sin\left( -\frac{2\pi}{N}(x, y)(u, v) \right) \right)^{1/2}$

where $j$ is an imaginary unit of $u, v \in \{0, A, N-1\}$, and $(x, y)(u, v) = xu + yv$ is the vector dot product. $G_{(u,v)}$ is obviously complex and $|G_{(u,v)}|$ denote the magnitude of $G_{(u,v)}$. In this study the theoretical report of this vector is defined as,

$$|G_{(u,v)}| = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \left( \left( g(x, y)\cos\left( -\frac{2\pi}{N}(x, y)(u, v) \right) \right)^2 + \left( g(x, y)\sin\left( -\frac{2\pi}{N}(x, y)(u, v) \right) \right)^2 \right)^{1/2} \qquad (2)$$

$|G_{(u,v)}|$ is the coefficients which represents the periodic characteristics of point u, v. The dominant period of signals

in an area can be determined by analysing the distribution of values of $|G_{(u,v)}|$. The whole procedure of ridge distance estimation with the spectral analysis method relies on a radial distribution function $Q(r)$ defined as follows,

$$Q(r) = \frac{1}{\kappa C_r} \sum_{(u,v) \in C_r} |G_{(u,v)}| \qquad (3)$$

Here $|G_{(u,v)}|$ represents the expression for the ridge distance resulting from a single harmonic accepted that gives $Q(r)$ the average contribution of the harmonics with ridge distance $N/r$ to the construction of the overall image, and the value of r corresponding to the maximum value of $Q(r)$ that is the incident times of dominant signal in this area. In detail, the computation of $Q(r)$ only involves the modulus of the harmonic coefficients, and the set of coefficients taken into consideration for each r is obviously invariant through any rotation of the image. Whereas $0 \leq r \leq \sqrt{2(n-1)}, C_r$ , here $C_r$ represents the set of coordinate values u, v that satisfy $\sqrt{u^2 + v^2} \, \square \, r$ and $\kappa C_r$ is the number of elements of $C_r$. The following algorithm is taken for ridge distance averaging estimation by the spectral analysis method in fingerprint images.

### C. Algorithm for Ridge Distance Averaging
Algorithm for Ridge distance averaging between two images.
**Input:** Binary thinned image X of size $u \times v \times Q(r)$ and r is kernel radius of the image
**Input:** Binary thinned image Y of size $u \times v \times k$ and r is kernel radius of the image
**Output:** Image Z of size $U \times V \times Q(r)$ with change in pixel distribution $(W \times H)$ .

> *for* $i = u$ do
>     *for* $j = v$ do
>         *for* $Q(r) > k$ do
>             Add $Y_{i+r+1}, Y_{j+r}, Y_{P+r}$ to width and
>                   height $(W \times H)$
>         *for* $Q(r) \leq k$ do
>            Remove $Y_{i-r-1}, Y_{j+r}, Y_{P+r}$ from width and
>                   height $(W \times H)$
>           $H \longleftarrow H + Y_{i-r-1} + Y_{i+r+1}$
>           $W \longleftarrow W + Y_{i-r-1} + Y_{i+r+1}$
>           $Z \longleftarrow H_{(i+j)} + W_{(i+j)}$
>         *end for*
>       *end for*
>     *end for*
> *end for*

Here $X$ is the ridge distance averaged input image, $Y$ is the ridge distance without averaged input image and $Z$ is output image with ridge distance averaged of input image $Y$. Here ' $Q(r)$ ' is the known average ridge distance (calculated using Eq. 3) with a block size $u \times v \times Q(r)$ of the input image $X$, where '$k$' is the unknown ridge distance with a block size of $u \times v \times k$ in the input image $Y$. Algorithm for ridge distance averaging method detects the value of ' $Q(r)$ ' and compares with the image $Y$, if the value of $Q(r) > k$, then it removes the pixels between the two neighbouring ridges up to it reaches the value of 'P'. At the same image $Y$, if the value of $Q(r) \le k$, then it adds the pixels between the two neighbouring ridges up to it reaches the value of ' $Q(r)$ ' in the block size of $u \times v \times k$. After this process, the height ($H$) and width ($W$) of the image is automatically adjusted to make an output image $Z$ of size $U \times V \times Q(r)$ for perfect matching. With the help of ridge distance averaging method, four sets of fingerprint images were taken from two different sensors and processed and are shown in Fig. 6.Further, FVC database and sensor public database images are tested with the proposed algorithm and the ROC curves are plotted to find the average equal error rate (EER) value for estimating the performance of the algorithm.
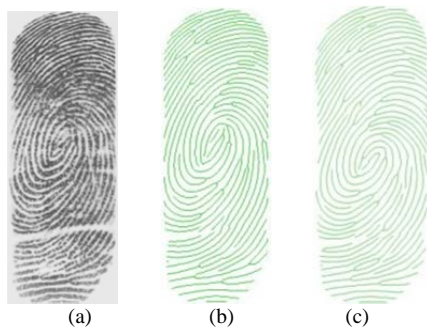


Fig. 5 (a) original image of capacitive swipe sensor, (b) without ridge averaging method (traditional procedure) and (c) proposed ridge averaging method (using a constant value $Q(r) = 3$)

Fig. 5(a) shows the raw fingerprint image taken by the capacitive swipe sensor. Fig. 5(b) is the thinned image without ridge distance averaging method (traditional procedure) and Fig. 5(c) shows the thinned image with ridge distance averaging method by applying a constant value of $Q(r) = 3$. Here, ridge distance averaging algorithm makes the automatic adjustments (adding or removing) of pixels in-between the two neighbouring ridges that yield prominent changes in the entire fingerprint image without affecting the details (minutiae points) and retaining the originality (location of the minutiae) of the fingerprint images.

## III. EXPERIMENTAL RESULTS

Several fingerprint images are taken from the public databases employ Optical and Swipe fingerprint sensors. Here four sets of fingerprint images taken from the optical press and capacitive swipe sensors are tested without ridge distance averaging method (traditional method) as well as ridge distance averaging method (proposed procedure) along with the extracted time. Two databases are used: (a) FVC for performance comparison measurements and (b) public sensor database for experimental testing (with images) and also for performance comparison measurements.

### D. Fingerprint Database Used

FVC fingerprint database with its sensor characteristics are given in Table 1(a). These are used to perform the calculation of EER values for both the Verifinger [24] and the proposed fingerprint matching algorithm. In FVC database, all the images are scanned using different sensors with different sizes and resolutions. In this paper, only the FVC database is used for performance comparison study.

Table 1(a) FVC fingerprint sensor characteristics of EER measurement for the proposed algorithm

| Database | Capturing devices | Size | Res. |
|---|---|---|---|
| FVC2000 DB1 | tOptical Sensor by KeyTronic | 300×300 | 500 dpi |
| FVC2000 DB2 | Capacitive Sensor by STM | 256×364 | 500 dpi |
| FVC2000 DB3 | Optical Sensor by Identicator | 448×478 | 500 dpi |
| FVC2002 DB1 | Optical Sensor by Identix | 388×374 | 500 dpi |
| FVC2002 DB2 | Optical Sensor by Biometrika | 296×560 | 569 dpi |
| FVC2002 DB3 | Capacitive Sensor by Precise | 300×300 | 500 dpi |
| FVC2004 DB1 | Optical Sensor by Cross match | 640×480 | 500 dpi |
| FVC2004 DB2 | Optical Sensor by Digital | 328×364 | 500 dpi |
| FVC2004 DB3 | Thermal sweeping by Atmel | 300×480 | 512 dpi |

Local sensors used in this used and their characteristics are given in Table 1(b). These are used to perform the experimental testing between the Verifinger and the proposed fingerprint matching methods. Two types of sensors (Eikon and Futronics) with different sizes and resolutions are used and more than 800 public fingerprint images were scanned from each sensor to create a sensor database for performance comparison.
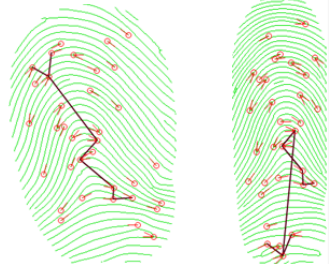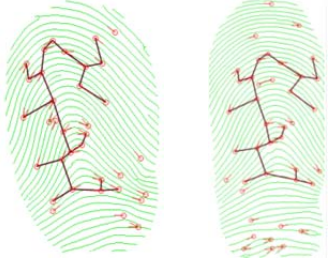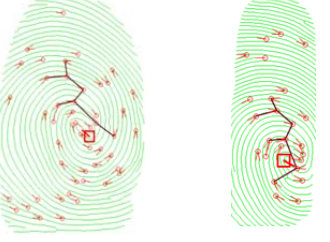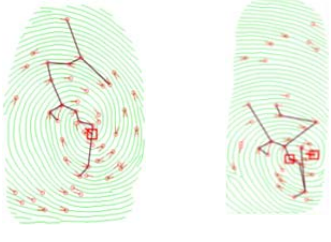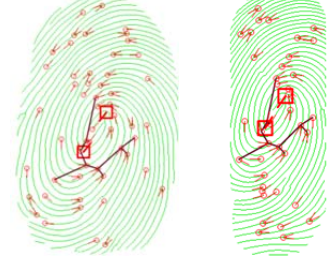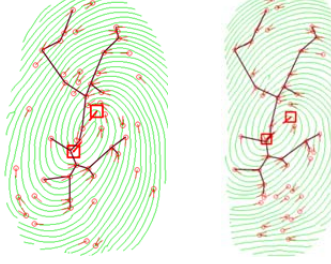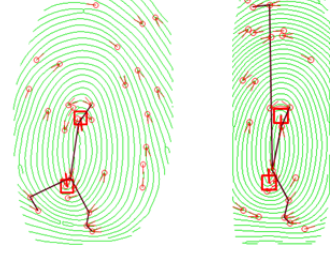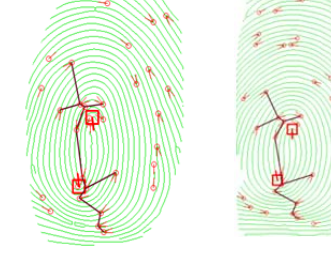
| Original (press) image with  pixels of width and height 320x480 | Original (swipe) image with 144x384 pixels of width and height. | Fingerprint matching without ridge averaging pixels. Extracted in 315ms. Templates do not match. Score: 0 | Matching with ridge distance averaging method with 241x528 pixels. Extracted in 388ms, Templates match. Score: 315 |
| Original (press) image with 320x480 pixels of width and height | Original (swipe) image with 144x384 pixels of width and height. | Fingerprint matching without ridge averaging pixels. Extracted in 288ms. Templates do not match. Score: 0 | Matching with ridge distance averaging method with 241x528 pixels. Extracted in 359ms, Templates match. Score: 83 |
| Original (press) image with 320x480 pixels of width and height | Original (swipe) image with 144x384 pixels of width and height. | Fingerprint matching without ridge averaging pixels. Extracted in 300ms. Templates do not match. Score: 0 | Matching with ridge distance averaging method with 233x526 pixels. Extracted in 332ms, Templates match. Score: 105 |
| Original (press) image with 320x480 pixels of width and height | Original (swipe) image with 44x384 pixels of width and height. | Fingerprint matching without ridge averaging pixels. Extracted in 350ms. Templates do not match. Score: 0 | Matching with ridge distance averaging method with 240x536 pixels. Extracted in 391ms Templates match. Score: 104 |
| (a) | | (b) | (c) |

Fig. 6(a) original grey-scale fingerprint image with different sensors having change in image width and height, (b) fingerprint matching without ridge distance averaging and the result gives non-matched fingerprint images, (c) fingerprint matching with ridge distance averaging and the result obtained as template matched with high matching scores.

Table 1(b) two fingerprint sensors characteristics for developing the sensor public database manually

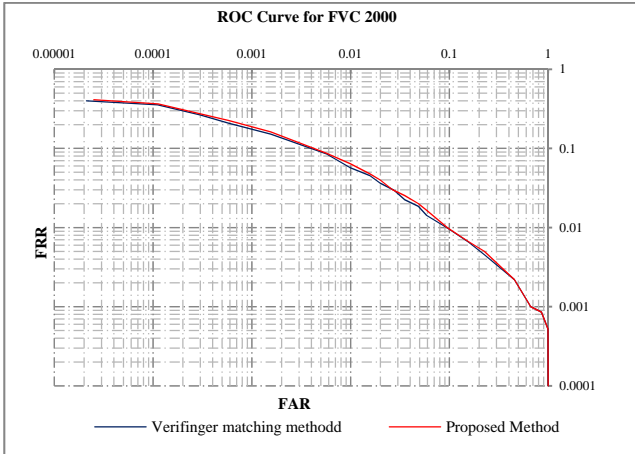| Devices | Capturing device | Size | Res. |
|---|---|---|---|
| Eikon scanner | Capacitive swipe scanner | $144 \times 384$ | 512 dpi |
| Futronics scanner | Optical press scanner | $320 \times 480$ | 500 dpi |


Fig. 7 (a) ROC curves of two algorithms on FVC2000 database
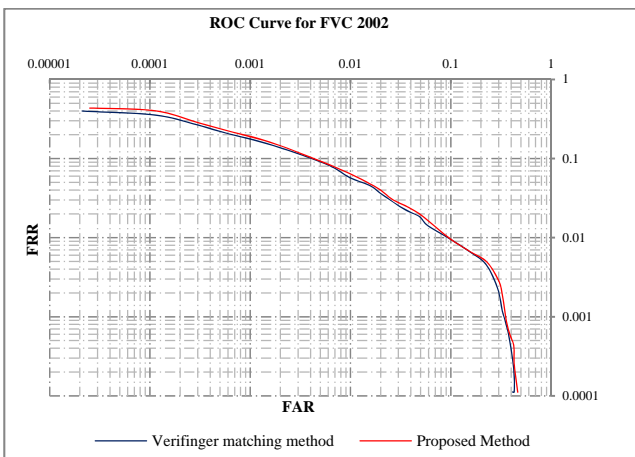

Fig. 7 (b) ROC curves of two algorithms on FVC2002 database
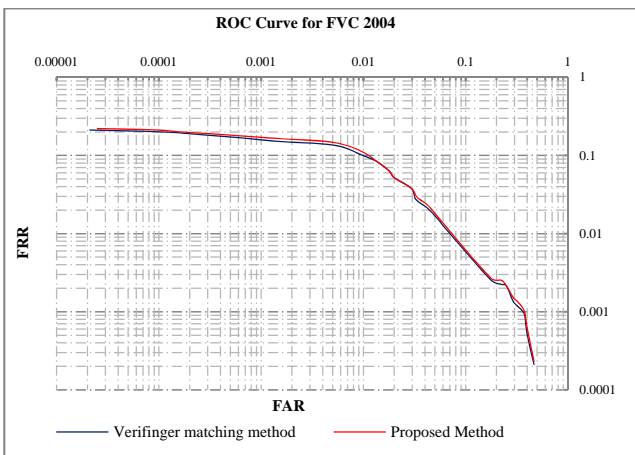

Fig. 7 (c) ROC curves of two algorithms on FVC2004 database
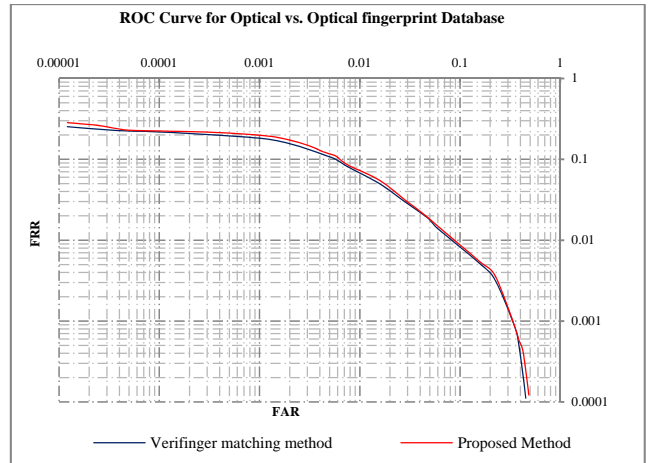

Fig. 8 (a) ROC curves on optical vs. Optical fingerprint database
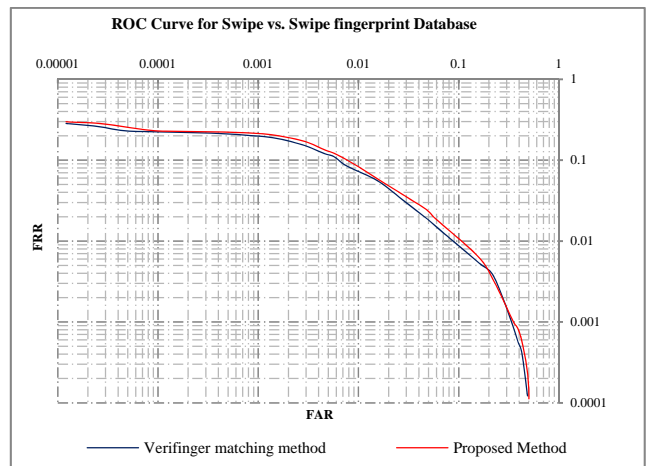

8 (b) ROC curves on swipe vs. Swipe fingerprint database
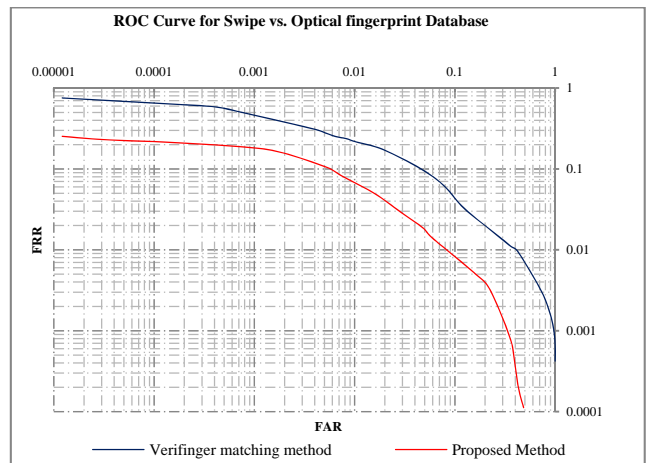

Fig. 8 (c) ROC curves on swipe vs. Optical fingerprint database

### E. Performance Comparison of Images

Fig. 6(a) shows the original gray scale images taken with two different sensors, optical (press) and capacitive (swipe) sensors reported in Table 1(b). Fig. 6(b) shows the fingerprint matching without ridge distance averaging method and the result as "templates do not match" using the Verifinger minutiae matching algorithm and Fig. 6(c) shows the fingerprint matching with ridge distance

averaging method (proposed procedure) with an auto change in pixel distribution with respect to the matching image and the result as "template matches" along with the extraction time and the matching score.

### F. Performance Comparison on FVC2004, 2002, and 2000 Databases

The receiver operating characteristic (ROC) curves of Verifinger and the proposed algorithm for FVC2004, 2002 and 2000 databases are plotted in Figs. 7(a), (b), and (c) and their corresponding EER values are reported in Table 2.

Table 2: equal error rate (EER) for verifinger vs. proposed method with FVC databases

| Matching Method | EER for FVC 2000 | EER for FVC 2002 | EER for FVC 2004 | Average EER |
|---|---|---|---|---|
| Verifinger | 2.856% | 2.744% | 2.998% | 2.866% |
| Proposed | 2.828% | 2.933% | 3.128% | 2.963% |

### G. Performance Comparison of Optical vs. Optical, Swipe vs. Swipe and Optical vs. Swipe manual databases

The receiver operating characteristic (ROC) curves of Verifinger and the proposed algorithm for optical vs. optical, swipe vs. swipe and swipe vs. optical databases are plotted in Fig. 8(a), (b), and (c) and their corresponding average EER values are reported in Table 3.

Table 3: equal error rate (EER) for verifinger vs. Proposed method with optical vs. Optical, swipe vs. Swipe and swipe vs. Optical databases (sensor databases)

| Matching Method | EER for Optical vs. Optical | EER for Swipe vs. Swipe | EER for Swipe vs. Optical | Average EER |
|---|---|---|---|---|
| Verifinger | 2.914% | 2.833% | 7.022% | 4.256% |
| Proposed | 2.983% | 3.271% | 2.904% | 3.052% |

The overall average EER values for Verifinger matching (traditional procedure) without ridge distance averaging and the proposed matching with ridge distance averaging with FVC and Sensor databases are tabulated in Table 4.

Table 4: overall average EER values for verifinger matching method vs. proposed matching method.

| Matching Method | Average EER for FVC databases | Average EER for Sensor databases | Overall average EER |
|---|---|---|---|
| Verifinger | 2.866% | 4.256% | 3.561% |
| Proposed | 2.963% | 3.052% | 3.007% |

The EER is defined as the equal error rate when the FAR and the FRR are equal. According to R. Cappelli [28] the average EER categorizes the performance between open and light categories. If the lower level of the average EER is 2.0%, then it is an open category; and if the EER is >3.5%, then it is light category. For the fingerprint matching analysis, the average EER lies between 2% to 3.5%. The average EERs estimated for the FVC databases are found to be 2.866% and 2.963% for the Verifinger and

proposed matching method, respectively. This indicates that our proposed method is close to the Verifinger matching method.

Similarly, the estimated EER for the databases of the sensors are given in Table 3. The EERs for the optical vs. optical, swipe vs. swipe, and swipe vs. optical databases are found to be 2.914%, 2.833% and 7.022% and 2.983%, 3.271% and 2.904% for Verifinger and proposed matching method, respectively. Our proposed matching method for swipe vs. optical database gives the best performance of 2.904% for the EER. The average EERs estimated for the sensors-databases are found to be 4.256% and 3.052% for the Verifinger and proposed matching method, respectively. It clearly indicates that our method gives a better matching performance even on the sensor interoperability issues. From Table 4, it is found that the average EERs estimated for the FVC database and sensor-databases are 3.561% and 3.007% for the Verifinger and proposed matching method. From the overall performance comparison of ROC curves and the EER values, it is clear that the proposed algorithm is designed to overcome the many of the sensor interoperability problems or change in sensors and gives the best result in the overall EER, which is evident from the performance comparison study.

The ridge distance averaging method and Verifinger matching method were developed using C++ and the approximate time of execution of the two algorithms were found to be 368 ms and 318 ms for the proposed and Verifinger matching methods. Even though our method consumes more time, our matching performance is better. An Intel Xeon processor operating at a clock frequency of 1.6GHz was used in our work.

## IV. CONCLUSION

We have succeeded in designing an algorithm which will extract and match all types of fingerprint images, particularly change in sensors. The proposed algorithm gives a high matching performance by adjusting the values of the pixels after the thinning process. The proposed algorithm will not affect the quality of the image as well as false acceptance and false rejection ratio. Good EER values of 3.007% were obtained using the proposed algorithm. We have tested with FVC databases as well as with optical and swipe sensor images. This algorithm may be adaptable for other sensor (thermal, capacitive, etc.) interoperability issues.

Whatever the capturing device, our proposed algorithm is very useful to extract and match the fingerprint images easily with a good matching score.

In this study, we have considered only two different sensors. It can be easily being extended with multiple sensors using the proposed algorithm.

REFERENCES

[1]   P. Komarinski, "Automated Fingerprint Identification System (AFIS)", *Elsevier,* New York, 1998.
[2]   D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, "Handbook of Fingerprint Recognition", *Springer-Verlag, second ed.*, New York.
[3]   D. J. Ohana, L. Phillips, L. Chen, "Preventing Cell Phone Intrusion and Theft using Biometrics", *IEEE Security and Privacy Workshops,* 2013, pp-173-180.
[4]   Z. Shihai, L. Xiangjiang, "Fingerprint Identification and its Applications in Information Security Fields", *International Conference of Information Science and Management Engineering (ISME),* Vol. 1, 2010, pp- 97-98.
[5]   S. Shigematsu, H. Morimura, Y. Tanabe, T. Adachi, K. Machida, "A Single-Chip Fingerprint Sensor and Identifier", *IEEE J. Solid-State Circuits,* Vol. 34, 1999, pp-1852-1857.
[6]   X. Xia, L. O'Gorman, "Innovations in fingerprint capture devices", *Pattern Recognition Society, Pattern Recognition*, Vol. 36, pp-361-367.
[7]   D. Gafurov, P. Bours, B. Yang, C. Busch, "GUC100Multisensor Fingerprint Database for In-House (Semipublic) Performance Test", *EURASIP J. Inform. Sec., Article ID 391761*, 2010, pp-1-7.
[8]   T. Feng, V. Prakash, W. Shi, "Touch Panel with Integrated Fingerprint Sensors Based Identity Management", *IEEE International Conference on Technologies for Homeland Security (HST)*, 2013, pp-1-3.
[9]   J. Jang, S. J. Elliott, H. Kim, "On Improving Inter-operability of Fingerprint Recognition Using Resolution Compensation Based on Sensor Evaluation*", Lecture Notes on Computer Science,* Vol. 4642, 2007, pp-456-460.
[10]  K. Cao, X. Yang, X. Chen, Y. Zang, J. Liang, J. Tian, "A novel ant colony optimization algorithm for large-distorted fingerprint matching", *Pattern Recognition*, Vol. 45, 2012, pp-151-152.
[11]  A. Ross, A. Jain, "Biometric Sensor Inter-operability: A Case Study in Fingerprints", *Proc. of International ECCV Workshop on Biometric Authentication (BioAW),* Vol. 3087, 2004, pp-134-144.
[12]  FVC2000, [*online*], http://bias.csr.unibo.it/fvc2000
[13]  FVC2002, [*online*], http://bias.csr.unibo.it/fvc2002
[14]  FVC2004, [*online*], http://bias.csr.unibo.it/fvc2004
[15]  Y. Li, Y. Yin, G. Yang, "Sensor-oriented Feature Usability Evaluation in Fingerprint Segmentation", *Optical Engineering*, Vol. 52, 2013, pp-067201-2-067201-4.
[16]  G. Yang, Y. Li, Y. Yin, Y-S. Li, "Two-Level Evaluation on Sensor Inter-operability of Features in Fingerprint Image Segmentation", *Sensors,* Vol. 12, 2012, pp-3188-3194.
[17]  G. Yang, G-T. Zhou, Y. Yin, X. Yang, "K-Means Based Fingerprint Segmentation with Sensor Inter-operability", *EURASIP J. Adv. in Signal Processing, Article ID 729378*, 2010, pp-2-5.
[18]  A. Ross and R. Nadgir, "A Thin-Plate Spline Calibration Model for Fingerprint Sensor Interoperability*" IEEE Transactions on Knowledge and Data Engineering,* Vol. 20, No. 8, 2008, pp-1097-1109.
[19]  J. Addepalli, A. Vasudev, "Fingerprint Sensor and Blackfin Processor Enhance Biometric-Identification Equipment Design", *Analog Dialogue,* Vol. 42, 2010, pp-1-5.
[20]  R. K. Rowe, K. A. Nixon, P. W. Butler, "Multispectral Fingerprint Image Acquisition" article, in: K. Nalini, V. Govindaraju (Eds.), *Advances in Biometrics, Sensors, Algorithms and Systems, Springer-Verlag, London,* 2008, pp. 3-8.
[21]  J-W. Lee, D-J. Min, J. Kim, W. Kim, "A 600-dpi Capacitive Fingerprint Sensor Chip and Image-Synthesis Technique", *IEEE J. Solid-State Circuits,* Vol. 34, 1999, pp-460-474.
[22]  N. Bartlow, N. Kalka, B. Cukic, "A. Ross, Identifying Sensors from Fingerprint Images*", IEEE Computer Society Workshop on Biometrics at the Computer Vision and Pattern Recognition (CVPR),* 2009, pp-1-4
[23]  T. Uz, G. Bebis, A. Erol, S. Prabhakar. "Minutiae-based template synthesis and matching for fingerprint authentication", *Computer Vision and Image Understanding,* Vol. 113, 2009, pp-983-984.
[24]  Verifinger, [*online*], http://www.neurotechnology.com
[25]  N. Liu, Y. Yin and H. Zhang, "A Fingerprint Matching Algorithm Based On Delaunay Triangulation Net*", IEEE Proceedings of the Fifth International Conference on Computer and Information Technology (CIT'05),* 2005.
[26]  Y. Yin, J. Tian, X. Yang, "Ridge Distance Estimation in Fingerprint Images Algorithm and Performance Evaluation", *EURASIP J. Applied Signal Processing,* Vol. 4, 2004, pp-495–502.
[27]  Zs. M. KovaHcs-Vajna, R. Rovatti, M. Frazzoni, "Fingerprint ridge distance computation methodologies", *Pattern Recognition*, Vol. 33, 2000, pp-69-80
[28]  R. Cappelli, D. Maio, D. Maltoni, J. L. Wayman, A. K. Jain, "Performance Evaluation of Fingerprint Verification Systems", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 28, 2006, pp-10-12.