Grenze International Journal of Engineering and Technology, Jan 2020

A Robust Fingerprint Alignment and Matching using Multiple Features

Umarani Jayaraman[1], Vandana Dixit Kaushik[2] and P. Gupta[3]
[1]Indian Institute of Information Technology Design & Manufacturing, Chennai, India
Email: umarani@iiitdm.ac.in
[2]Harcourt Butler Technical University, Kanpur, India
Email: vandanadixitk@yahoo.com
[3]National Institute of Technical Teachers' Training & Research, Kolkata, India
Email: phalgunigupta@nitttrkol.ac.in

***Abstract*— In this paper, a novel fingerprint alignment and matching is introduced. It uses multiple features to align fingerprints and then matching. The proposed system uses non-minutiae features such as SURF and BRISK features for alignment and minutiae features for matching. Fingerprint alignment has been done with the help of SURF key points for initial alignment followed by BRISK key points to get the final alignment. Minutiae features are used to find the similarity score between two fingerprints which has helped to check the correctness of the alignment. To verify the proposed alignment approach, a novel matching score calculation method has been presented. This approach is different from the traditional fingerprint alignment approaches which are rely heavily on minutiae features. The proposed algorithm is robust, efficient and accurate. Further, it has been compared with well known alignment approaches on FVC2002 dataset. The proposed approach is found to be superior and has the Equal Error Rate (EER) of 0.035%.**

***Index Terms*— Fingerprint alignment, matching, minutiae, SURF and BRISK.**

I. INTRODUCTION

Large number of research has been carried out in fingerprints since the last decade, yet fingerprint recognition remains an open problem. The fingerprint images are usually of poor quality. Matching of extracted features from these images with the database remains a challenge [14]. Fingerprint alignment is fundamental for fingerprint based recognition system. So, researchers have studied it in great detail [21, 23]. If a fingerprint is traslated, rotated or partially occluded by another fingerprint, then the pairing of minutiae becomes a challege. For example, there are approximately about 50-70 minutiae, then there may be a maximum of 70x70 possible pairs of minutiae. Aligning two fingerprints before matching may greatly reduce the possible number of minutiae pairs and would, therefore, reduce the computational cost [10]. The indexing of fingerprint dataset using minutiae features [9] and finger-knuckle-prints dataset using non minutiae feature [8] has been studied to minimize the cost of searching. In the most accurate types of fingerprint recognition, there are two main stages, namely alignment and matching. Alignment is essential to obtain good accuracy in the matching algorithm. During acquisition of fingerprint, the finger impressions can possibly become misplaced or rotated. This makes the fingerprint alignment a major challenge as it becomes difficult to find

Grenze ID: 01.GIJET.6.1.503

similarity between two fingerprints even of the same person. Figure 1 shows two fingerprints which are captured from the same person.

It can be observed that two impressions are not properly placed. So, proper alignment of the two fingerprints is needed before it can be processed at the matching stage. Every fingerprint has small details called minutiae. Many algorithms use these minutiae to match between two fingerprints [22]. Minutiae can be classified in to two types namely, i) ridge endings and ii) bifurcations. Ridge endings are the points where rides terminate and bifurcations are places where a ridge splits into two. Extracting minutiae from noisy or low resolution images is a hard task. This adds the difficulty in minutiae-based algorithms. Errors in extraction of minutiae propagate into alignment and then at matching stage. This makes the algorithm dependent on the quality of the extracted minutiae. To overcome this dependency, pixel intensities [2] or filterbanks [7, 16] have been used as non-minutiae features. Filterbank extracts orientation information and frequecy from fingerprint ridges as features. This feature set does not have discriminatory information and so may not be useful in fingerprint recognition. The proposed method uses several features to align and to match two fingerprints. It uses non-minutiae features such as SURF and BRISK features for *alignment* and minutiae features for *matching*. There are two stages in the proposed system, namely, *fingerprint alignment* and *matching*. The alignment stage can be done in two steps. Key points from SURF (first step) and BRISK (second step) are used to get the intial alignment and final alignment respectively. In order to verify the correctness of the alignment step, *fingerprint matching* can be done using minutiae features to find the similarity score. To verify the proposed algorithm, a new matching score calculation is used.

Figure 1: Example of misplaced fingerprint impressions during acquisition

The paper is organizsed as follows. Section 2 presents the fingerprint alignment stage. Fingerprint matching score calculation is given in Section 3. Section 4 describes the experimental study. Final section presents conclusion and future scope of work.

II. Fingerprint Alignment

There are two stages in the proposed system namely, *fingerprint alignment* and *matching*. Figure 2 depicts the block diagram of the proposed system. There are four steps for alignment namely, 1) detect features 2) extract features 3) match features and 4) registration. In this section, the alignment of two fingerprints is discussed step by step.

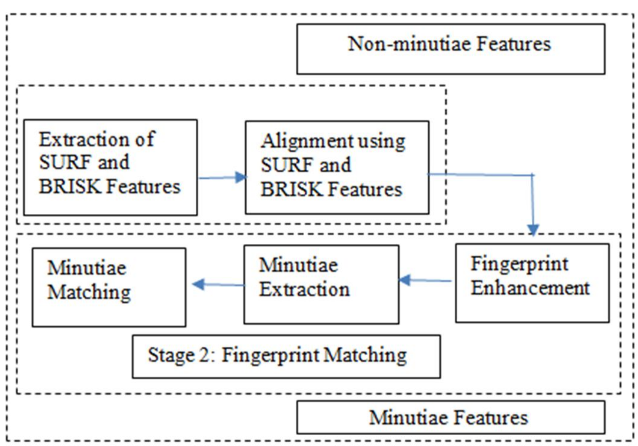

Figure 2: Block Diagram of the Fingerprint Recognition System

a) **Feature Detection:** Rotated and translated test fingerprint with respect to the template fingerprint should be aligned or registered before matching. The first step in alignment is to detect *non-minutiae* features. It is done using Speeded Up Robust Feature (SURF) algorithm [1] followed by Binary Robust Invariant Scalable Keypoints (BRISK) algorithm [12]. SURF algorithm finds blob features as keypoints in a gray scale images while BRISK algorithm is used to detect multiscale corner features. These multiple features are combined to estimate the geometric relationships between the two fingerprints. Unlike the traditional method where minutiae features are heavily used for both alignment and then matching, in this method, non-minutiae features such as SURF and BRISK are used for alignment and then minutiae features are used for matching. The non-minutiae features are robust to scale, translation, rotation and occlusion and hence the alignment with this is more efficient as compared to minutiae features alone.

b) **Feature Extraction:** In this step, it extracts descriptors from a region around each keypoint from a gray scale image. It returns extracted feature vectors as descriptors and their corresponding locations. If the region lies outside of the image, it cannot compute a feature descriptor for that point. When the keypoint lies too closed to the edge of the image, it cannot compute the feature descriptor. In these cases, the algorithm ignores the keypoint. These points are not included in the keypoints set. Descriptors are derived from pixels around a key point. Both SURF and BRISK algorithms set the orientation of the extracted features in radians.

c) **Feature Matching:** A *Threshold* is used to determine whether two feature vectors are matched based on the distance between them. If the distance value is less than the threshold, the vectors are said to be matched. Otherwise, they do not match. Figure 3 shows the SURF features extracted from two fingerprint images (template and test) and their matched features (keypoints). One can observe that from Figure 3, two fingerprints of the same person, but one is distorted with another. The test fingerprint (Red in color) is distorted and translated with template fingerprint (light green in color). In order to align these two fingerprints, first SURF features are detected which are marked as red 'o' and blue '+' on template and test fingerprints respectively. Since only partial image of test fingerprint is available, very few features points are getting matched using SURF features.

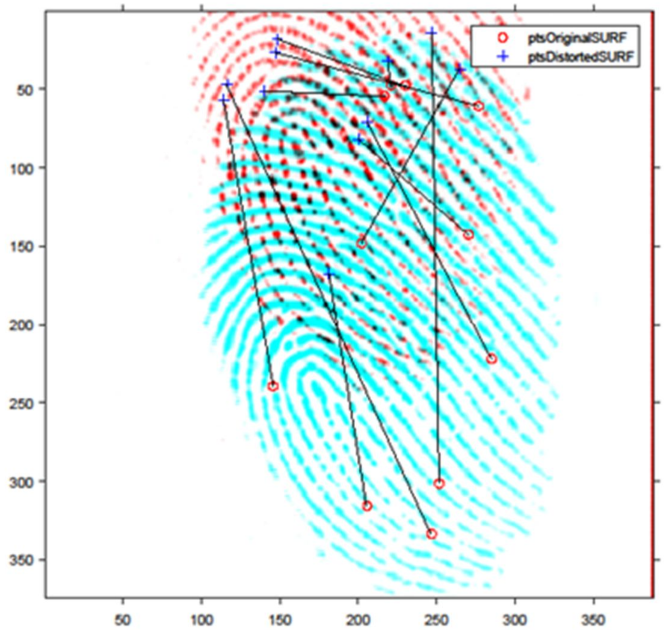

Figure 3: Matching Points using SURF

As BRISK features are more robust and efficient than SURF, it could be used for fine or accurate alignment. It can extract more features than SURF. These more features are used for final alignment. Result of matching is shown in Figure 4. The light green color ridge lines are of the original template image and the red color ridge lines are of distorted test image. BRISK extracted features for two fingerprints and their matched keypoints are shown in Figure 4. Figure 5 shows the same with an addition of SURF extracted features.

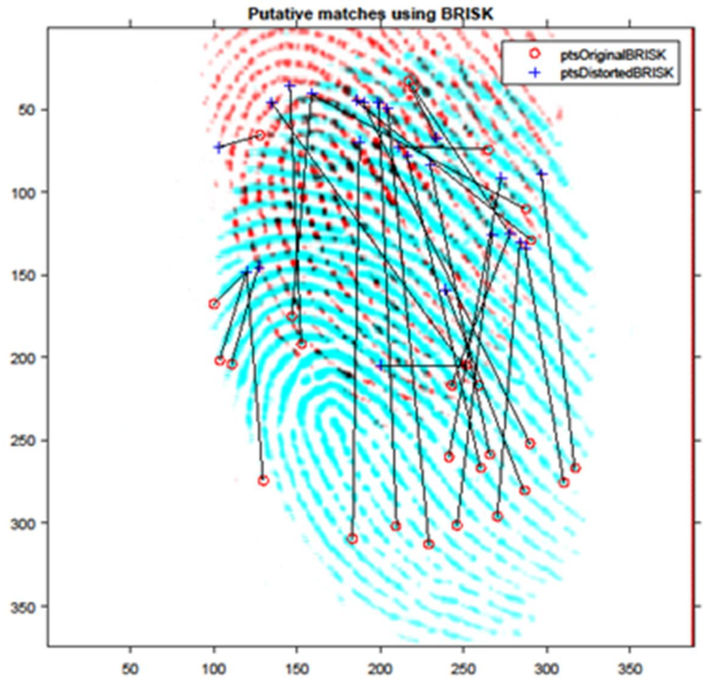

Figure 4: Matching Points using BRISK

d) **Registration:** Fingerprint registration is performed using SURF and BRISK features which are scale and rotation invariant keypoint detectors and descriptor algorithms which exhibit great performance under a variety of image transformations. Given two matchedpoints1 and matchedpoints2, it returns 2-D geometric transform by mapping of inliers from matchedPoints1 to matchedPoints2. Outliers are excluded by the usage of M-estimator SAmple Consensus (MSAC) algorithm. It is a modification of RANdom SAmple Consensus (RANSAC) algorithm. MSAC is randomized in nature and therefore, results may not be similar.

III. Fingerprint Matching

The *matching* stage is done using minutiae features and it consists of 3 steps 1) fingerprint enhancement, 2) minutiae extraction and 3) minutiae matching. In order to verify the correctness of alignment between two fingerprints (test and template), it is proposed the new matching score calculation.

a) **Fingerprint Enhancement:** Fingerprint is enhanced using Gabor filter [5]. Fingerprint enhancement consists of stages like Normalization, Segmentation, Orientation Estimation, Ridge frequency Estimation, Gabor filtering, Binarization and Thinning.

b) **Minutiae Extraction:** Most of the fingerprint matching algorithms are based on minutiae. The minutiae information is regarded as highly significant features for fingerprint matching. Crossing number (CN) concept [15] is used to detect minutiae from a fingerprint.

c) **Minutiae Matching using Proposed Matching Score:** Minutiae matching is done using proposed matching score method. Let the number of minutiae features extracted from each fingerprint image be represented by $M= \{m^1, m^2, ..., m^o\}$. Each minutiae m^i can be represented as a 4-tuple (x^i, y^i, θ^i, T^i) consisting of its x and y co-ordinates, direction and minutiae type. Let A and B be the template and test fingerprint respectively. For the template fingerprint (A), let minutiae set $A = \{m^1_A, m^2_A, m^3_A ,...,m^p_A\}$ where each m_B^i (x^i_A, y^i_A, θ^i_A, T^i_A) and $m_A^i \in \{m^1_A, m^2_A, m^3_A ,...\}$, $1 \leq i \leq p$. Again, for the test fingerprint (B), let minutiae set $B = \{m^1_B, m^2_B, m^3_B ,...,m^q_B\}$where $m_B^j= (x^j_B, y^j_B, \theta^j_B, T^j_B)$ and $m_B^j \in \{m^1_B, m^2_B, m^3_B ,...\}$, $1 \leq j \leq q$. After the minutiae extraction, the matching score is calculated between two fingerprints (template and test images) based on the distance measure. In minutiae based matching, minutiae points from a template fingerprint minutiae set A is mapped to a test minutiae set B. The mapping is one-to-one mapping or pairing of minutiae between the sets A and B. Minutiae pairs between set A and B are formed using $\pi(k)$ as a mapping permutation of pairs from set A to set B. For finding minutiae matching pairs, geometric constraints are determined and it includes geometric distance,

$$dist(m_A^i, m_B^j) = \sqrt{(x_A^i - x_B^j)^2 + (y_A^i - y_B^j)^2} \leq r_\delta$$

and minutiae angle difference

$$dist_\theta(m_A^i, m_B^j) = \min(|\theta_A^i - \theta_B^j|, 360^o - ||\theta_A^i - \theta_B^j||) \leq r_\theta$$

where r_δ and r_θ are thresholds for geometric distance and angular differences, respectively.

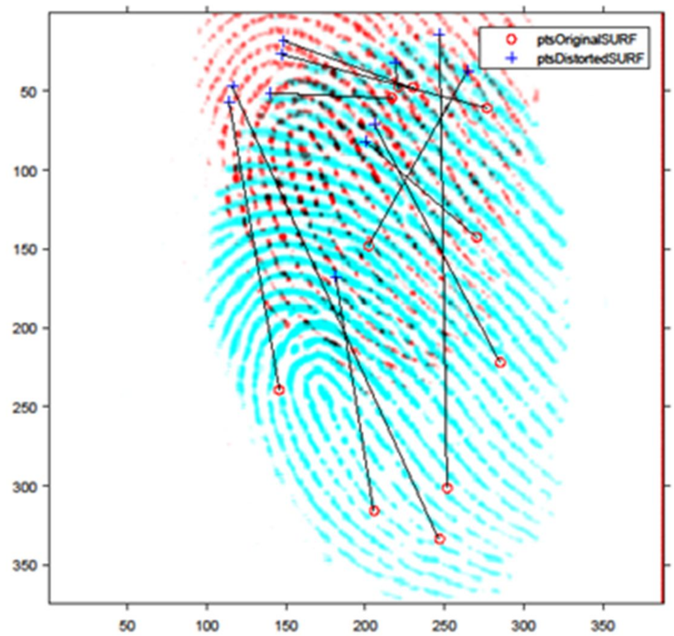

Figure 5: Matching Points using SURF and BRISK

After finding the minutiae pairs, the similarity between each pair needs to be calculated. This puts a need for a similarity score which clearly describes the equality of the images. The proposed similarity matching score is given as follows

$$S = \frac{V}{ZA * ZB}$$

where V is the square of the number of matched minutiae pairs in both template A and test fingerprints B, and ZA and ZB are the number of minutiae in the overlapped regions (Region of Interest (ROI)) of the two images A and B.

The minutiae features along with the orientation of both template and test fingerprints plotted (minutiae features from template and test fingerprint is overlapped on same graph) and is shown in Figure 6. It can be observed that blue and red colored lines are the orientation plot of the template and test fingerprints respectively. The number of overlapped minutiae (red and blue) are the candidate set which is used to calculate the matching score.

Example: let the number of matched minutiae pairs be 10 and the number of minutiae in the overlapped region of both template and test fingerprints be 13 and 12 respectively. Therefore, the matching score computed for the two images is 0.64.

Algorithm: Fingerprint Matching (A,B)

Input: Template fingerprint $A = \{ m^1_A, m^2_A, m^3_A, ..., m^p_A \}$
Test Fingerprint $B = \{ m^1_B, m^2_B, m^3_B, ..., m^q_B \}$
Output: Matching Score, S
Step 1: Compute Euclidean distance, $dist(m_A^i, m_B^j)$
Step 2: Compute angle difference, $dist_\theta(m_A^i, m_B^j)$
Step 3: If $dist(m_A^i, m_B^j) \leq r_\delta$ and $dist_\theta(m_A^i, m_B^j) \leq r_\theta$

then these minutiae are candidate set to compute the matching score

Step 4: Matching Score, $S = \frac{V}{ZA*ZB}$

Step 5: If $S \geq T$hreshold then two fingerprints are match

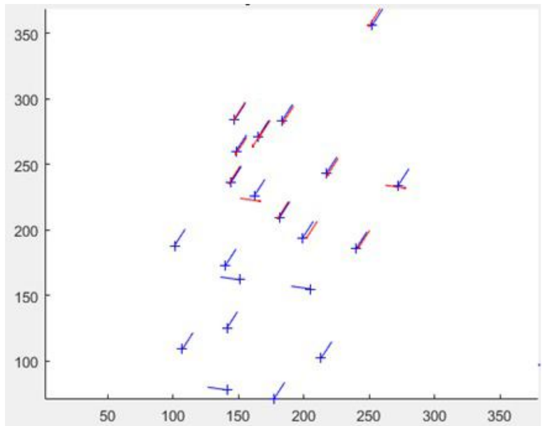

Figure 6: Computing Similarity Score

IV. Experimental Results

The proposed algorithm has been tested on publicly available database known as Fingerprint Verification Competition (FVC 2002). The following metrics are used to measure the performance of the proposed approach, namely; 1) False Acceptance Rate (FAR), 2) False Rejection Rate (FRR) and 3) Equal Error Rate (EER).

Database: Fingerprint Verification Completion (FVC2002) database consists of DB1, DB2, DB3 and DB4 datasets. Out of these, some of the fingerprints are handpicked at randomly (few are ignored because of very low quality) and form the new datasets. There are 9 different fingers, each with 7 impressions per finger which adds up to a total of 63 fingerprints. Out of 7 impressions, one image from each person is considered as a test images (9 different fingerprints as test dataset). The rest fingerprints are considered as the template dataset (9x6=54 images as a template dataset).

The similarity score is generated by considering this template (9x6=54 fingerprints) and test datasets (9 different fingerprints). It is calculated by varying the threshold between '0' to '1' (interval of [0, 1]) where threshold '0' means 100 percent match (everyone matches) and threshold '1' means zero match (no one matches).

Selection of Threshold and Results: FAR-FRR diagram Vs threshold is used in practice so that the best threshold value can be selected for which the error rates are minimum. Imposter scores exceeding and falling below a certain threshold are captured by FAR and FRR respectively. Figure 7 shows FAR (blue color) and FRR (red color) plots against various thresholds. Let us assume threshold = 0.7. One impostor having score exceeding the threshold of '0.7' is accepted while 52 genuine scores falling below the threshold of '0.7' are rejected. Hence, FAR = 0.19% with FRR = 82% which is very high.

When FAR is equal to FRR, they meet at point of intersection called Equal Error Rate (EER). The EER for the proposed approach is found to be 0.035% which is far better than any other well known approaches. Therefore, the threshold of 0.09 is selected for which FAR and FRR are equal (EER=0.035%). Figure 8 shows the matching score distribution. The dotted line (red color) indicates genuine scores while the solid line (blue color) indicates impostor scores. It can be observed that most of the imposters (blue) have low matching score, *i.e* less than 0.1, as compared to the genuine person. The matching score for the genuine person varies in between 0.1 to 1. Also, there is not much overlapping between imposters and genuine persons.

Comparison with existing alignment approaches: The proposed approach has been tested on FVC2002 DB1 dataset and is compared with well know approaches. In Table 1, values of EER which should be as low as possible in any biometric system, are shown in descending order. The proposed approach achieved an EER rate of 0.035 on FVC 2002 DB1 dataset.

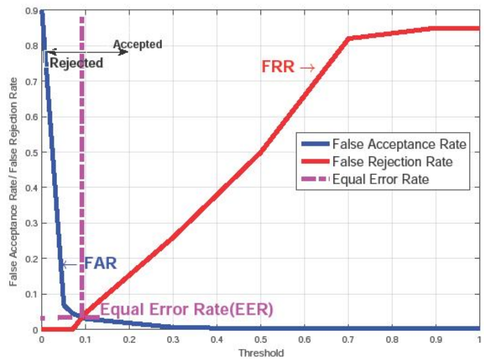

Figure 7: FAR and FRR Curve

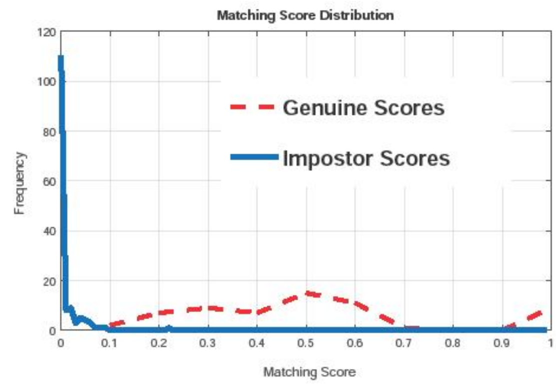

Figure 8: Matching Score Distribution

TABLE I. ERR FOR EACH ALGORITHM ON FVC 2002 DB1

Author	EER Results in %
Jain et al. [7]	EER=27.40
Jiang and Yau [11]	EER=12.40
Jain et. al. [6]	EER=5.20
Xi et. Al. [21]	EER=4.60
Ratha et al. [17]	EER=4.50
Benhammadi et. al [3]	EER=4.27
Lumini and Nanni [13]	EER=4.20
Goa et. al [4]	EER=3.50
Li Wang et. al [20]	EER=2.61
Neil Yager and Adnan Amin [22]	EER=2.60
Wahby Shalaby and Ahmad [18]	EER=2.57
Neil Yager and Adnan Amin [23]	EER=1.60
Wang et. Al [19]	NA
Proposed Approach	EER=0.035

V. CONCLUSIONS

This paper has presented an efficient fingerprint alignment and matching algorithm. It has used multiple features to align fingerprints and then matching. It considered non-minutiae features such as SURF and BRISK features for *alignment* and minutiae features for *matching*. The proposed alignment approach has been verified by a novel matching score calculation method. This approach is different from traditional fingerprint alignment approaches which rely heavily on minutiae features for alignment. Further, it has been compared with well known alignment approaches on FVC2002 dataset. The proposed approach has EER of 0.035% and is found to be superior to is found to be superior.

Most of the existing approaches to solve the problem of fingerprint alignment are based on minutiae features or ridge orientation. One can think to use different possibilities of other features instead of using minutiae features and ridges orientation. Usages of device to capture the fingerprint images should be free from rotation, translation and occlusion. Many existing digital fingerprint devices are having the above challenges. But this problem of alignment is considered to be so important when we do match between two scanned fingerprints, especially in the banking applications where fingerprints are taken from the illiterate people. Hence, more work on the fingerprint alignment needs to be done in future.

REFERENCES

[1] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). Comput. Vix. Image Underst., 110(3):346-359.

[2] Bazen, A. M., Verwaaijen, G. T., Gerez, S. H., Veelenturf, L. P., and van der Zwaag, B. J. (2000). A correlation-based fingerprint verification system. In Proceedings SAFE, ProRISC, SeSens2000, pages 205-213. STW Technology Foundation.

[3] Benhammadi, F., Amirouche, M.N., Hentous, H., Bey Beghdad, K., and Aissani, M. (2007). Fingerprint matching from minutiae texture maps. Pattern Recogn., 40(1):189-197.

[4] Gao, Z., You, Z., Zhou, L., and Zeng, W. (2011). A novel matching technique for fingerprint recognition by

graphical structures. In Wavelet Analysis and Pattern Recognition (ICWAPR), 2011 International conference on, pages 77-88. IEEE.

[5] Hong, L., Wan, Y., and Jain, A. (1998). Fingerprint image enhancement: Algorithm and performance evaluation. IEEE transactions on pattern analysis and machine intelligence, 20(8):777-789.

[6] Jain, A., Hong, L., and Bolle, R. (1997). On-line fingerprint verification. IEEE Trans. Pattern Anal. Mach. Intell., 19(4):302-314.

[7] Jain, A. K., Prabhakar, S., Hong, L., and Pankanti, S. (2000). Filterbank-based fingerprint matching. Trans. Img. Proc., 9(5):846-859,

[8] Jayaraman, U., Gupta, A. K., and Gupta, P. (2014a). Boosted geometric hashing based indexing technique for finger knuckle-print database. Information Sciences, 275:30-44.

[9] Jayaraman, U., Gupta, A. K., and Gupta, P. (2014b). An efficient minutiae based geometric hashing for fingerprint database. Neurocomputing, 137:115-126.

[10] Jayaraman, U., Prakash, S., and Gupta, P. (2013). Use of geometric features of principal components for indexing a biometric database. Mathematical and Computer Modelling, 58(1-2): 147-164.

[11] Jiang, X. and Yau, W.-Y. (2000). Fingerprint minutiae matching based on the local and global structures. In Pattern recognition, 2000. Proceedings. 15th international conference on, volume 2, pages 1038-1041. IEEE.

[12] Leuteneggeer, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: binary robust invariant scalable keypoints. In Proceedings of the 2011 International Conference on Computer Vision, ICCV'11, pages 2548-2555.

[13] Lumini, A. and Nanni, L. (2006). Two-class fingerprint matcher. Pattern Recogntion, 39(4): 714-716.

[14] Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S. (2009). Handbook of Fingerprint Recognition. Springer Publishing Company, Incorporated, 2nd edition.

[15] Mehtre, B. M. (1993). Fingerprint image analysis for automatic identification. Machine vision and Applications, 6(2-3): 124-139.

[16] Park, C., Lee, J., Smith, M. J. T., Park, S., and Park, K. (2004). Directional filter bank-based fingerprint feature extraction and matching. IEEE Trans. Circuits Syst. Video Techn., 14(1):74-85.

[17] Ratha, N. K., Karu, K., Chen, S., and Jain, A. (1996). A Real-time Matching System for Large Fingerprint Databases. IEEE Trans. Pattern Analysis and Machine Intell., 18(8):799-813.

[18] Wahby Shalaby, M. A. and Omair Ahmad, M. (2013). A multi-level structural technique for fingerprint representation and matching. Signal Process., 93(1):56-69.

[19] Wang, C. and Gavrilova, M. L. (2005). A Novel Topology-Based Matching Algorithm for Fingerprint Recognition in the Presence of Elastic Distortions.

[20] Wang, L., Battacharjjee, N., and Srinivasan, B. (2012). A method for fingerprint alignment and matching. In Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia, MoMM'12. Pages 297-301.

[21] Xi, K., Hu, J., and Han, F. (2011). An alignment free fingerprint fuzzy extractor using near-equivalent dual layer structure check (nedlsc) algorithm. In 2011 6th IEEE Conference on Industrial Electronics and Applications, pages 1040-1045.

[22] Yager, N. and Amin, A. (2004). Fingerprint verification based on minutiae features: A review. Pattern Anal. Appl. 7(1):94-113.

[23] Yager, N. and Amin, A. (2006). Fingerprint alignment using a two stage optimization. Patter Recognition Letters, 27(5):317-324Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. J. Mol. Biol. 147, 195–197 (1981)