International Journal of Computer Science and Software Engineering (IJCSSE), Volume 7, Issue 3, March 2018
ISSN (Online): 2409-4285 www.IJCSSE.org Page: 47-51

Fingerprint Verification System for Cross-sensor Matching based on LBP and SIFT Descriptors and Score level Fusion

Helala AlShehri[1], Muhammad Hussain[2], Hatim Abo AlSamh[3] and Mansour Zuair[4]

[1, 2, 3, 4] College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

[1]hm.shehri@gmail.com, [2]mhussain@ksu.edu.sa, [3]hatim@ksu.edu.sa, [4]zuair@ksu.edu.sa

ABSTRACT

Fingerprint verification is a popular biometric modality for person authentication. However, existing fingerprint verification systems have been designed with the assumption that the same sensor is used for both enrollment and verification. The performance of these systems significantly declines when different sensors are adopted for enrolment and query (cross matching, fingerprint sensor interoperability problem). In this paper, we propose an automatic fingerprint verification system to solve this problem. Two different sensors result in fingerprints of the same finger, which vary in scale and resolution, but have the same ridge patterns. Based on this observation, the proposed system combines two image-based descriptors, local binary pattern (LBP) and scale-invariant feature transform (SIFT) to address different resolutions and different sizes of fingerprint images. The scores from these two descriptors are fused at score-level. The experimental results show that the proposed system outperforms the state-of-the-art systems such as Minutia Cylinder-Code (MCC) and VeriFinger, which is a commercial Software Development Kit (SDK).

Keywords: *Fingerprint Sensor Interoperability, Fingerprint Cross-matching, Fingerprint Recognition, Local Binary Pattern, Scale-Invariant Feature Transform.*

1. INTRODUCTION

Fingerprint recognition is one of the most active research fields in biometrics and one of the most frequently used systems for authenticating the identities of users. Existing approaches to fingerprint verification are effective when the same type of sensor is used for both enrollment and authentication. However, verifying the fingerprints of an individual enrolled with a different sensor is still an important concern due to the widespread adoption of fingerprint systems in governments, agencies, and organizations of all shapes, sizes, and types. Sensors are also increasingly inexpensive due to widespread implementation, advancements in technology, and the increasing rate of adoption.

Both governments and businesses have huge databases of fingerprints obtained using particular sensors, but sometimes different sensors are used at the point of authentication or verification. This gives rise to fingerprint sensor interoperability problem (cross-matching). Different technologies are used to scan fingerprints, such as ultrasound, optical, and thermal [1]. These technologies each uniquely introduce their own form of distortions which make the cross-matching problem more challenging.

Recent studies have highlighted the importance of examining the effect of employing multiple fingerprint sensors in fingerprint matching approaches [2, 3]. Some research has also examined the scaling of fingerprints [4–6], non-linear distortion [7, 8], and the fusion of existing fingerprint recognition systems [3]. Despite these efforts, cross-matching is still a challenging problem.

In this work, we propose a novel fingerprint verification method to solve the problem of cross-matching. When fingerprints are captured from the same finger using different sensors, they differ in scales and resolution but have the same ridge patterns. To address these issues, the features of a fingerprint are locally extracted from it by decomposing the fingerprint into overlapping windows. We adopted two descriptors, the well-known local binary patterns (LBP) and the scale-invariant feature transform (SIFT) that encode the local ridge patterns, to extract the local features. We argue that adopting these two descriptors will produce both rotation-invariant and scale-invariant features, as the invariant LBP represent the basic structures of the fingerprints, such as edges, while SIFT has also been proven to be a more accurate descriptor, as it is scale-invariant and rotation invariant. Score-level fusion is applied to fuse the scores resulting from each descriptor. Experiments conducted on a public domain database Multisensor Optical and Latent Fingerprint (MOLF) confirm the effectiveness of fusing

LBP with SIFT in mitigating the problem of cross-matching.
The main contributions of the paper are as follows:

- A fingerprint verification system that mitigates the problem of fingerprint sensor interoperability.
- A technique for fusion of two fingerprint descriptors that encode the basic structures of the fingerprint, such as ridge patterns.
- A thorough evaluation of the proposed method using a benchmark dataset and standard evaluation tools to compare the proposed method with state-of-the-art systems.

The rest of the paper is organized as follows: Section 2 presents the details of the proposed approach, while Section 3 describes the experiments and their results. Section 4 concludes the paper.

2. PROPOSED FINGERPRINT VERIFICATION SYSTEM

The framework of the proposed fingerprint verification system is shown in Fig.1

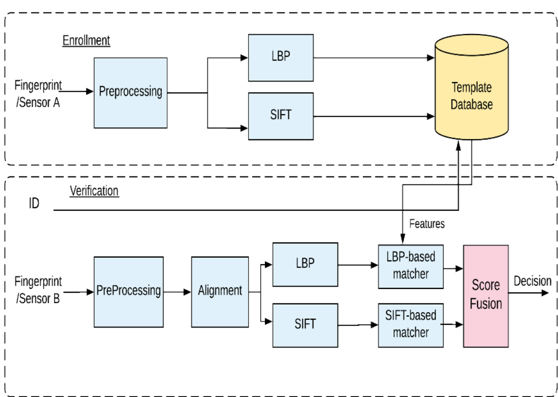

Fig. 1. Framework of The Proposed Method.

The enrollment phase begins with the preprocessing of gallery fingerprints obtained using sensor A to enhance the contrast and reduce noise of the fingerprint. Then, LBP and SIFT features are extracted from the enhanced fingerprint and subsequently stored in a template database.
During the verification step, the preprocessing technique is applied to a probe fingerprint obtained with Sensor B, and then the probe and gallery fingerprints are aligned, and the two types of features are extracted. The process of matching the features is performed separately for each individual feature. Finally, the resulting matching scores are fused together to produce the final matching scores.

In the following subsections, we provide the detail of different algorithms used for preprocessing, alignment, feature extraction, and matching steps.

2.1 Preprocessing

The preprocessing step aims to enhance the contrast of the fingerprint and reduce the noise. The enhancement of the fingerprint is performed using the short-time Fourier transform (STFT), as proposed by Chikkerur et al. [9]. This method starts by dividing the fingerprint into small, overlapping windows and applies STFT on each window. Following this, the frequency of ridges, orientation of ridges, and block energy are computed using the Fourier spectrum. Finally, the fingerprint is enhanced using contextual filtering.

2.2 Alignment of probe and gallery fingerprint

The alignment is performed using the method of Tico et al. [10], which is a minutiae-based method. To extract the minutiae, this method uses the binarization and thinning approach proposed by Hong et al. [11]. To align two sets of minutiae, this method employs minutiae-based orientation descriptors to compare a pair of minutiae. For the minutiae of each probe and each gallery, the method computes the probability of the two minutiae points being similar. The minutiae are then sorted in descending order according to their probability values. The probe minutiae are transformed so that the probability values are maximized for all corresponding pairs in the probe and the gallery fingerprints. Then, a greedy algorithm identifies corresponding minutiae pairs.

2.3 Feature extraction

For extracting discriminative features, it is important to take into account the spatial location of local features in a fingerprint. To introduce the location information about local features, each enhanced fingerprint image is divided into overlapping square blocks. Each block is of fixed dimension D×D, where D = 25 pixels with 50% overlap. Then, the LBP and SIFT features are extracted from each block B_i (i = 1, 2, …, n) of the enhanced image, which are represented as a feature vector v_{bi}. After computing the feature vectors from each block, descriptors are constructed by concatenating them $v = [v_{b1}, v_{b2}, \ldots, v_{bn}]$. In the following subsections we give the detail about how LBP and SIFT features are computed.

2.3.1 Rotation-invariant local binary patterns

LBP is a local texture descriptor that is used to extract local features from a grayscale image. LBP has been

proven to be an efficient and powerful discriminative descriptor in many applications because it has low computational complexity and is invariant to monotonic transformations in grayscale.
The original LBP operator encodes each pixel with decimal numbers that represent the local structure around each pixel. The LBP is applied to each pixel x considering the set of its neighboring P pixels in a circular pattern with radius R, denoted as $LBP_{(P,R)}$. It results in a binary number consisting of P bits, which are encoded to a decimal value. Fig.2 contains an example of LBP calculation. Each gray-level value of a pixel is compared to those of its neighboring pixels by subtracting the value of the center pixel; the resulting positive values are encoded with 1 and the others with 0. Starting from the top left and moving in a clockwise direction, a binary number is constructed by concatenating these encodings, and the binary number is converted into an equivalent decimal value, which is used to label the central pixel. Finally, the histogram of LBP codes is computed to define the LBP descriptor.

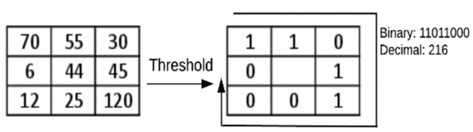

Fig. 2. An example of the basic LBP operator.

By construction LPB is not rotation invariant. It can be made rotation invariant by circularly shifting the bits within binary number a number of times and choosing the smallest value. The original LBP operator was extended by Ojala et al. [12] to produce a rotation-invariant descriptor and to reduce the dimension of the feature space, resulting in what they call uniform patterns. This extension was motivated by the consideration that some binary patterns occur more frequently than others. If the binary pattern includes two or fewer bitwise transitions, from 1 to 0 or vice versa, this binary pattern is called uniform LBP. In the calculation of the uniform LBP histogram, the histogram assigns a separate bin to each uniform pattern, and all non-uniform patterns are assigned the same bin. This reduces the size of the LBP histogram from 256 to 59. The basic structures of the image, such as edges and lighting spots, are characterized by a uniform pattern
As ridge patterns occur in fingerprints captured with different sensors at different scales, so to accommodate the variation of scales in cross-sensor matching scenario, we extract four LBP descriptors with four different combinations of the following parameters: $R \in \{1,2\}$ and $P \in \{8,16\}$, which are concatenated together.

2.3.2 Scale-invariant feature transform

SIFT is a scale-invariant descriptor of an image proposed by David Lowe [13]. It has been widely used in many Computer Vision applications because of its interesting characteristics. It is not only scale invariant but also rotation invariant and small distortion invariant. Because of these characteristics, it is a good choice for cross-sensor fingerprint matching.

In the computation of SIFT descriptor, the first step is to generate different scales of the image and to detect the location and scale of interest points, also called keypoints, in an image. Then, a refining step is performed for the scale and location of the keypoints, and the orientation of each keypoint is determined. Finally, the keypoint descriptor is computed by considering a neighborhood of 16 × 16 around the keypoint and dividing it into sub-regions of size 4 × 4. For each cell, an eight-bin orientation histogram is constructed and, in total, the bins consist of 128 values.

2.4 Matching

An LBP descriptor is basically a histogram, so a histogram dissimilarity measure can be used for it. In view of this, the matching score between two fingerprints is computed using the Euclidean distance.
A SIFT-based descriptor is matched using a Euclidean-distance-based nearest neighbor approach, as proposed in [13]. This involves identifying the first and the second neighbor for each keypoint, based on the Euclidean distance. Then, the ratio of the distances between the first neighbor and the second neighbor is considered, and if the ratio is greater than 0.8, the keypoint will be rejected as a false match. The final score is calculated by aggregating the distances of the correctly matched keypoints.

3.5 Score-level fusion

The method proposed in this study involves two types of descriptors, which result in different matching scores. To enhance the matching performance, a score-level fusion is used to combine the scores.
Before fusing the scores, a normalization technique is applied to each one, as the scores have different ranges. In this paper, a min-max normalization is applied using the following formula:

$$nS_i = \frac{S_i - min_i}{max_i - min_i}, \quad (1)$$

where the min and max are the minimum and maximum scores of the matcher i, s_i is the actual score, and nS_i is the normalized score.
Let nS_{LBP} and nS_{SIFT} be the normalized scores resulting from LBP matching and SIFT matching, respectively. Then, the final score is computed by adopting the sum rule:

$$S = \beta nS_{LBP} \ + (1 - \beta)nS_{SIFT}\text{ , where } \beta \in [0,1] \qquad (2)$$

Based on our experiments, in this work, we set $\beta = 0.5$ for the reported experimental results.

3. EXPERIMENTS AND RESULTS

The fingerprints used for our experiments were taken from a public domain benchmark cross-sensor database called Multisensor Optical and Latent Fingerprint (MOLF) [14]. The MOLF database comprises three subdatasets captured with three optical sensors: Lumidigm Venus IP65 Shell, CrossMatch L-Scan Patrol, and Secugen Hamster-IV. The resolution of the fingerprints captured with CrossMatch, Lumidigm, and Secugen is 500 dpi each, while the sizes are 1600 × 1500, 352 × 544, and 258 × 336 pixels, respectively. For each sensor, there are 1,000 fingerprint classes, with four impressions for each. The total number of fingerprints in each dataset is 4,000. Fig. 3 shows an example of three impressions of the same finger captured with three different sensors. The impressions vary in quality, resolution, and noise due to the different technologies of the sensors.

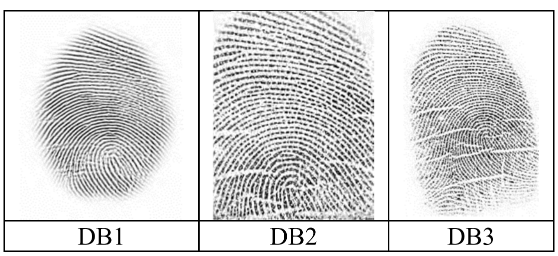

Fig. 3. Three Impressions Of The Same Fingerprint From The MOLF Database

The performance of a matching system can be measured by examining the scores that result from two scenarios of matching: regular matching and cross-matching. Regular matching (also called intra-device or native-device matching) involves comparing two fingerprints captured with the same sensor to produce a native equal error rate (EER). Cross-matching (inter-device or cross-device matching) involves comparing two fingerprints obtained from different sensors to produce a cross- or interoperable-EER.

It should be noted that this study was conducted using MATLAB 2016 running on a machine with an Intel Core i7-4702MQ CPU at 2.2 GHz, 16 GB of RAM, and a 64-bit Windows operating system.

The performance of the proposed method is evaluated by comparing it with state-of-the-art methods: Minutia Cylinder-Code (MCC) [15] and VeriFinger [16]. MCC is a state-of-the-art minutiae-based matching algorithm, and VeriFinger is a well-known commercial matching algorithm developed by Neurotechnology. Various researchers consider MCC and VeriFinger the baseline for comparisons of cross-matching and regular matching methods [5, 17].

Table 1 shows the performance of the proposed method compared to that of VeriFinger and MCC for each pair of datasets (both native- and cross-device) in the MOLF database. In general, the results show that the native-EER is lower than the cross-EER for each method. The overall EER of MCC is higher than those of the other methods. The results of VeriFinger are better than those of MCC, but its results are poor in the case of cross-matching. The results show that the proposed method outperforms VeriFinger and MCC when both the gallery and probe fingerprints are captured with the same sensor (native-device). For cross-matching, the proposed method outperforms the MCC, and its overall results are better than VeriFinger, except for experiments DB2 vs. DB3 and vice-versa. These results indicate the ability of the proposed method to provide a discriminant feature for fingerprint analysis.

Table 1. The EER computed for the three methods for all pairs of datasets

Gallery dataset	Probe dataset	VeriFinger	MCC	Proposed Method
DB1	**DB1**	3.16	11.14	**2.85**
	DB2	6.46	18.48	5.0679
	DB3	6.42	20.80	5.5547
DB2	DB1	6.47	18.48	5.0679
	DB2	3.20	16.82	**3.02**
	DB3	3.94	22.74	6.885
DB3	DB1	6.42	20.81	5.5547
	DB2	3.94	22.74	5.0679
	DB3	3.51	13.83	**3.36**

Fig. 4 shows the average cross-EERs of MCC, VeriFinger, and the proposed method for each dataset of either gallery or probe fingerprints. Among all datasets, MCC produces the highest average cross-EER. The average cross-EER of VeriFinger is better than that of MCC, and the proposed method outperforms VeriFinger overall, considering the average cross-EERs.

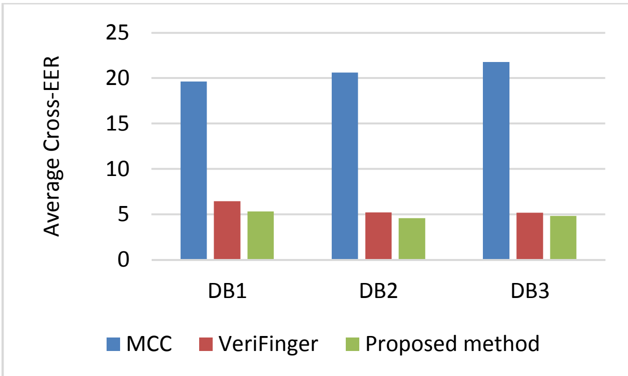

Fig. 4. Average cross-EER obtained using the three systems with the MOLF database.

4. CONCLUSION

In this paper, we introduced an automatic fingerprint verification method to minimize the impact of fingerprint sensor interoperability problem. The proposed method uses two types of descriptors: invariant-LBP and SIFT, and fuses them at the score-level. Experiments concerning the problem of fingerprint sensor interoperability were conducted on public domain benchmark cross-sensor MOLF database, and the proposed method was compared with two state-of-the-art methods: MCC and VeriFinger. The results show that our method outperforms both VeriFinger and MCC.

REFERENCES

[1] A. Ross and A. Jain, "Biometric Sensor Interoperability : A Case Study In Fingerprints," Proc Int. ECCV Work. Biometric Authentication BioAW, vol. 3087, no. May, pp. 134–145, 2004.
[2] L. Lugini, E. Marasco, B. Cukic, and I. Gashi, "Interoperability in fingerprint recognition: A large-scale empirical study," in Proceedings of the International Conference on Dependable Systems and Networks, 2013.
[3] F. Alonso-Fernandez, R. N. J. Veldhuis, A. M. Bazen, J. Fierrez-Aguilar, and J. Ortega-Garcia, "Sensor interoperability and fusion in fingerprint verification: A case study using minutiae-and ridge-based matchers," in 9th International Conference on Control, Automation, Robotics and Vision, 2006, ICARCV '06, 2006.
[4] Y. Zang, X. Yang, X. Jia, N. Zhang, J. Tian, and X. Zhu, "A coarse-fine fingerprint scaling method," in Proceedings - 2013 International Conference on Biometrics, ICB 2013, 2013.
[5] Y. Zang, X. Yang, X. Jia, N. Zhang, J. Tian, and J. Zhao, "Evaluation of minutia cylinder-code on fingerprint cross-matching and its improvement with scale," in Proceedings - 2013 International Conference on Biometrics, ICB 2013, 2013.
[6] C. X. Ren, Y. L. Yin, J. Ma, and H. Li, "Fingerprint scaling," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2008, vol. 5226 LNCS, pp. 474–481.
[7] A. Ross and R. Nadgir, "A thin-plate spline calibration model for fingerprint sensor interoperability," in IEEE Transactions on Knowledge and Data Engineering, 2008, vol. 20, no. 8, pp. 1097–1110.
[8] A. Ross, S. Dass, and A. Jain, "A deformable model for fingerprint matching," Pattern Recognit., vol. 38, no. 1, pp. 95–103, 2005.
[9] V. G. Sharat Chikkerur, Chaohang Wu, "A Systematic Approach for Feature Extraction in Fingerprint Images," in First International conference Biometric Authentication (ICBA), 2004, pp. 1–23.
[10] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 8, pp. 1009–1014, 2003.
[11] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 8, pp. 777–789, 1998.
[12] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pp. 971–987, 2002.
[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.
[14] A. Sankaran, M. Vatsa, and R. Singh, "Multisensor Optical and Latent Fingerprint Database," IEEE Access, vol. 3, pp. 653–665, 2015.
[15] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia Cylinder-Code: A new representation and matching technique for fingerprint recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 12, pp. 2128–2141, 2010.
[16] Nuerotechnology, "VeriFinger." 2016.
[17] X. Jia and X. Yang and Y. Zang and N. Zhang and J. Tian, "A cross-device matching fingerprint database from multi-type sensors," in Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), 2012, pp. 3001–3004.