Distortion Analysis on Binary Representation of Minutiae Based Fingerprint Matching for Match-on-Card

Cynthia Sthembile Mlambo
Council for Scientific and Industrial Research
Information Security
Pretoria, South Africa
smlambo@csir.co.za

Meshack Bafana Shabalala
Council for Scientific and Industrial Research
Information Security
Pretoria, South Africa
mshabalala@csir.co.za

Abstract— **The fingerprint matching on the smart card has been developed and recognized. Now these applications perform well in terms of security than fingerprint matching from the computer or large capacity systems. There has been much research and activities concerned with improving the accuracy, time efficient implementations, security and efficient space of the match on card. In this paper presented is the survey on the methods used to improve the accuracy in matching and memory usage by representing minutiae points in binary. However, distortion is a major challenge in binary representation of minutiae points. Therefore this paper includes the methods used to deal with fingerprint distortion while representing minutiae points as binary vectors. This survey will assist on the new developments of match on card applications, in improving the accuracy and memory usage while dealing with the problem of fingerprint distortion.**

I. Introduction

Fingerprint recognition is one of the most well-known and published systems that can be used for both identification and verification of an individual [1]. During identification a fingerprint of unknown ownership is compared against a database of known fingerprints, and during verification a claimer fingerprint is compared against the enrolled fingerprint corresponding to the claim [1]. A fingerprint is a pattern made up of the ridges and valleys on the surface of a finger [2]. Ridges are the upper layer and valleys are the lower layer of the skin of the finger, as shown in Figure 1.

Fingerprints are the most widely used biometric modality because of its distinctiveness as well as their low cost and maturity of their fingerprint solutions [2]. The distinctiveness of an individual's fingerprints is determined by local ridge characteristics and their relationship [3]. Most widely used characteristics of fingerprints are minutiae point features. This is because; forensic examiners have relied on minutiae to match fingerprints for more than a century and expert testimony about suspect identity based on mated minutiae is admissible in a court of law [3]. In addition, the memory space required to allocate minutiae points, generally, is less compared to other fingerprint features.

Minutiae points are locations on the fingerprint where a ridge either terminates or divides to form two ridges; these are known as a ridge ending and a ridge bifurcation respectively and are shown in Figure 1 [4]. Each ridge ending and ridge bifurcation are characterized by the row and column value of the pixel where they occur on the fingerprint image, together with the orientation of the ridge associated with the minutiae point as illustrated in Figure 2 (a) and (b).

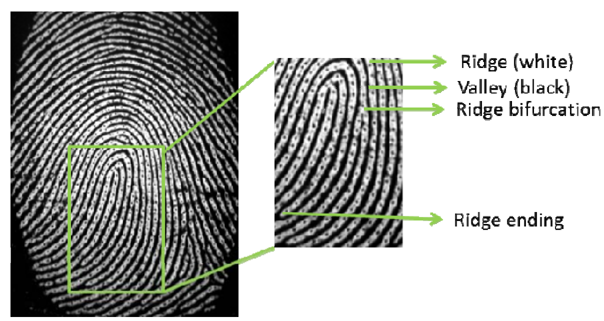

Fig. 1. A region of a fingerprint image with the ridge, valley, ridge bifurcation, and ridge ending marked

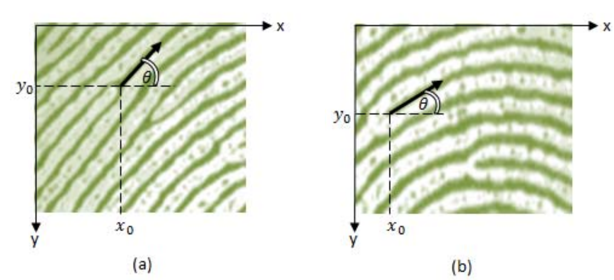

Fig. 2. In (a) and (b) x_0 (column) and y_0(row) represent the coordinate of minutiae and θ represent the orientation of the ridge associated with minutiae [4].

However, there is challenge encountered when matching minutiae points that are extracted from different fingerprints of the same finger but captured in different instances [4]. This can

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.59

be caused by the position and direction of the finger changing towards the surface of the scanner. Other causes can be from the twisting of the finger while it is scanned, different fingerprint levels of dryness and moisture of the finger, dirt on the scanner, and different levels of pressure applied to the scanner. Shown in Figure 3 are different impressions of the same finger captured in different times. The images in Figure 3 were captured using a Futronic surface scanner. It can be seen on the highlighted regions of fingerprints that, minutiae points changes in locations and the locations in relative to each other changes. Some minutiae are missing or new ones introduce on different fingerprint impressions. This change in location and number of minutiae points has a huge impact when representing fingerprints as minutiae points.

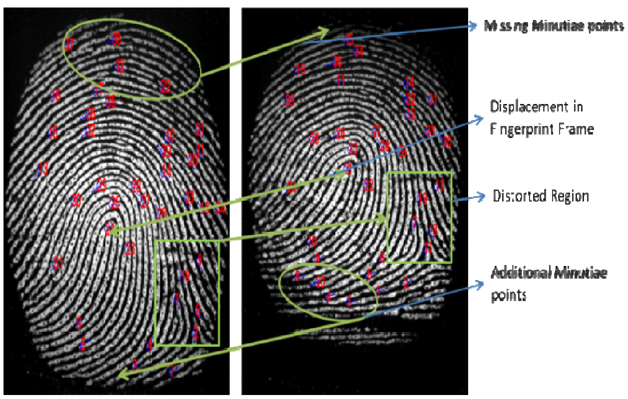

Fig. 3. Two fingerprint impressions of the same finger captured in different instances.

In this paper, different methods are studied on how minutiae points are presented into binary representation and how the problem of distortion is attempted. In Section 2 discusses a brief introduction to match-on-card, and considers different algorithms which perform fingerprint matching using binary representation of minutiae points. Section 3 are the findings on this research and summary of the paper.

II. BINARY REPRESENTATION OF MINUTIAE POINTS

A Match-On-Card is a process of matching fingerprints or other characteristics of a human being (e.g. iris, hands, etc.) inside the smart card [1], [5]. In addition, the smart card stores the information of a human being with the matching algorithms that are used for identification or verification using the information of a card holder and stored information. However, the smart card have very limited resources and computationally demanding [6]. Therefore, when considering information and matching algorithm to store in a smart card, the memory usage becomes a very important issue. In addition, an accurate matching algorithm is important that will require and use efficient memory on the smart card [6]. Different methods have been implemented in the literature to reduce the memory usage on the card. Such methods are based on representing fingerprint features as binary strings.

Earlier works in [7] – [9] presented minutiae points in binary with other fingerprint features. Farooq et al [7] presented a new method of computing histogram of minutiae combination. Minutiae points are grouped in N tuples. However, the combination of minutiae into triples or more increases the memory requirements for the length of the vector to store N tuples of minutiae points. In [8] an alternative method was proposed that construct the binary feature vector from floats by using the spectral representation of minutiae points. This method firstly computes a grid of minutiae points, and performs encoding and decoding in order to deal with the problem of distortion, displacement and missing or additional minutiae. Another binary representation was presented in [9]; by computing local cuboids from the locations of minutiae. However, due to that fingerprints get affected by non-linear distortions, this method requires pre-alignment of minutiae points.

In 2010 Bringer and Despiegel [10] presented a new method for binary feature vector fingerprint representation from minutiae vicinities. Given a set of minutiae points, this method converts the set of minutiae points into a fixed length of binary feature vector. This is performed by first constructing vicinities of each minutia in a given set. Each minutia is set as a center of its nearest minutiae points (as shown in Figure 4). The neighborhood of a central minutia is defined as the set of minutiae points that are the closest by a defined distance. The new coordinate system is then defined from the location and orientation of the central minutia (as shown in the second block of Figure 4).

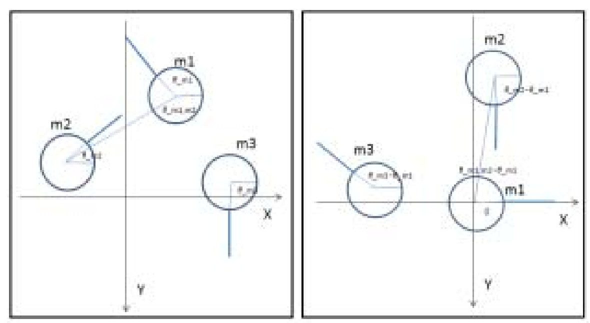

Fig. 4. Vicinity coordinate system [18]

To construct the binary representation from vicinities, all constructed vicinities in a feature vector are compered to all representative of a fingerprint. Then similarity scores are determined whether there is vicinity similar to the fingerprint representative. If there is a higher score in the collection of scores for vicinity of minutia, it indicates that there is common central minutia. The vicinity component of the higher score is then set to bit “1”, else it is set to zero. An overview of this process is illustrated in Figure 5.

The problem of distortion is attempted by encoding fingerprint according to its distances computed in relative to several representative minutiae vicinities. The main advantage of this method is that all the computation efforts are concentrated on the feature extraction process whereas the matching process is almost reduced to matching two fixed size binary feature vectors [10]. This is of great importance for increasing security on the smart card by integrating into cryptographic protocols [10]. Furthermore, a fixed length of binary representation is constructed in this method, and as the approach uses local relationship of minutiae points, it enables to deal with described fingerprint distortions.

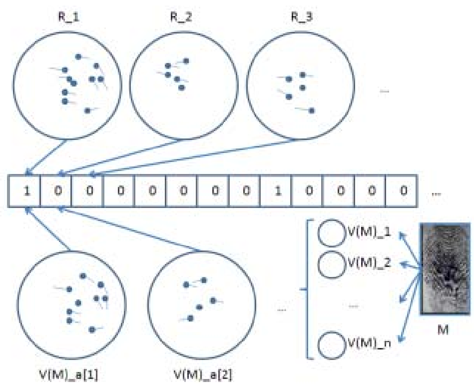

Fig. 5. Binary feature vector construction.[18]

In 2011 Yang et al [11] introduced a method for keyed scalable minutiae coding. This method improves on Bringer and Despiegel's [10] work. This method converts N nearest minutiae triplets into N equivalent binary feature vectors. Where N is the number of acceptable nearest minutiae to the central minutiae, and N can be adjusted. When N is adjusted from low to high value leads to the increase in global information of the central minutiae, as a result improves the performance in fingerprint matching. The difference in this method from Bringer and Despiegel [10] method is that instead of using three neighboring minutiae, only two neighboring minutiae are used. As a result, the vicinity of each minutia is represented as minutiae triplet that includes; central minutiae and two nearest minutiae. This method attempts to solve the challenge of missing minutiae points, by considering a few number of nearest minutiae. In addition, to convert a real-value vector to a binary vector, unary representation is used in alternative to the base-2 numeral representation used in [10]. This is because unary representation has equal weights on each bit to represent a real value; therefore the efficient Hamming distance is used to compare the triplet vector of binary representation [11].

Further improvements on the idea of minutiae vicinities were proposed by Binger and Vincent [12] in 2011. Authors in [12] improved form their earlier work presented in [10], by extending the construction of binary vector and include additional minutiae information. In this method, characteristics of vicinity are used that represent a global feature that provides information of where minutiae are located. The information is computed from the location and orientation of the ridge that represent the central minutiae point of the vicinity (as minutiae orientations are shown in Figure 4).

The construction of binary feature vector is performed in a similar way as in [10], and then the location and orientation of each central minutia is added. The additional information is constructed from the closest minutiae of the central minutiae in the vicinity. The use of position and orientation information leads to improved performance in terms of complex binary representation. This provides more privacy and security on the representation of fingerprint information.

Lately, Benhammadi and Bey [13] proposed binary representation of neighborhood minutiae using one minutiae as the reference point in the whole fingerprint. This method is different form the work of Binger and Despiegel [12], because instead of using nearest neighbors of minutiae, all minutiae in the fingerprint image are deemed as neighbors of the reference minutiae. This method uses circular tessellation to encode fingerprint features by considering the binary localization of minutiae. As shown in Figure 6. The construction of the binary vector is computed from the thinned fingerprint image with extracted minutiae points. This performed by first selecting the reference minutia that will represent the center of the tessellation, and then tessellation is applied on the entire fingerprint image starting from the reference minutiae. Shown in Figure 6 is the circular pattern that consists of 32 sectors, with the radii distance defined to be the maximum length from the reference minutiae to the border of the image. The sectors are distributed equally along the circumference of the circle.

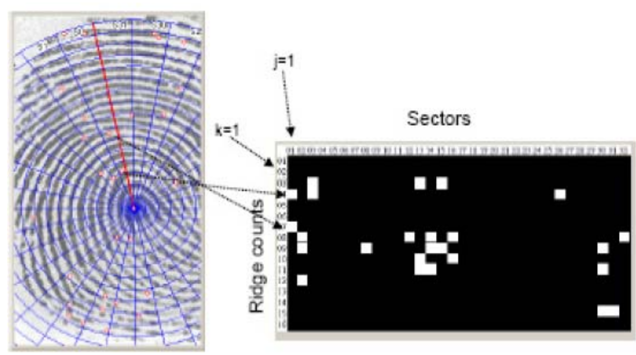

Fig. 6. Representation of circular tessellation on a fingerprint image [13]

The distortions affect this method especially when minutiae locations of different impressions of the same finger changes. As a result, binary codes for minutia localization of the same minutiae from different impressions become different. This situation involves a binary shift by one or two positions in the binary code of minutia. The shift is either to the left or right and downwards or upwards. To deal with this shift, this method involves a bit shift method, which assumes that the ridge counts errors are insignificant and that the tessellation into sectors is larger [13]. This method involves the process of switching the vicinity logical bits, from "1" to "0" or from "0" to "1". This method showed successful implementation on the smart card with efficient memory usage on the card, while dealing with fingerprint distortion problem.

Bourgeat et al [14] improved on the method presented in [10], by attempting to prevent false matches of similar vicinities from different fingerprints of different fingers. This is performed by introducing second-order vicinities. This method allows the extraction of more information from grouping of vicinities by manipulating the information present in the relative distances that separates neighborhoods of central minutiae. To compute a feature binary vector that represent a fingerprint, number of representative vicinities are constructed, depending on how many minutiae points are in a fingerprint [14].

The method of second-order vicinities is referred to as vicinities of vicinities [14]. The first-order vicinities are constructed as defined in [10], and then new representation of vicinities is computed by replacing each vicinity by its barycenter. The second order vicinity involves the position of vicinity with respect to the corresponding parent vicinities. In order to keep vicinities pairwise, vicinities that have central minutiae belonging to other vicinities are deleted. During fingerprint matching, both first-order and second-order vicinities must match to show that two fingerprint images matches. The experimental results showed that with proper selection of the size of nearest minutiae, the accuracy improves under different environments of distortion.

An alternative method was introduced by Vij and Namboodiri [15] by proposing a fixed length illustration for fingerprints that includes exact alignment between the features. This method constructs local minutiae structure by first capturing the complete geometry of minutiae points that are nearest to the central minutiae. This method is applied on the given fingerprint database, firstly, arrangement structures are extracted to collect different structures into a high dimensional structure space. The k-means clustering method is then used to group collected arrangement structures in a dimensional space [15]. The difference between this method and the one presented in [10], is that it involves the relative geometric features nearby the neighborhood of a minutia point because these features are not highly affected by the distortions. These features are: the ratio of the relative distance between central minutiae and two nearest minutiae, with the ratio of their relative orientation with respect to the central minutiae, and the ratio of the angles in the structure of grouped minutiae. The feature vector representation encompasses the idea of object representation using groups of words into groups of minutiae vicinities. The experimental results of this method showed that the representation is invariant to distortions and displacements of minutiae points [15]. In addition, a fixed length binary vector leads to efficient memory usage for the storage on smart card.

III. Classification of Minutiae Binary Representation

The representation of minutiae binary vectors can be classified into three, namely, minutiae vicinity, tessellation, 2nd order vicinity, and minutiae patterns. In minutiae vicinity methods, binary vectors are computed according to the relationship of the nearest minutiae [10] – [12]. Given a set of minutiae points represented with their locations and orientation, each fingerprint is encoded according to the distance of nearest minutiae. The second order minutiae vicinity is the improvement of work presented in [10 – 13], which forms vicinities of vicinities in a minutiae set [14]. A binary vector is generated from the arrangements of vicinities computed from each minutiae and its neighborhood.

Tessellation methods convert a pixel for minutiae points in to binary '1' and other pixels as binary zero [13]. A binary vector is computed from reference point and its nearest minutiae. The information in each sector in the tessellation represents rows and columns of the final finger-code of given set of minutiae.

The last method forms different patterns of minutiae and represent those patterns into binary vector [15]. This method uses relative geometric features around the locality of each minutiae point. This involves relative distance, relative orientation, and angles between three minutiae when one is set as a reference point.

IV. Discussion and Analysis

In this section discussed are the findings from the study according to different distortion environments. Table 1 below shows different types of distortions and their causes, considered during this research.

TABLE I. Distortion Conditions

Distortion	*Cause*
1. Unequal ridge size	Uneven Pressure on the Surface of the scanner
2. Breakage of ridges	Dryness of the finger
3. Connected ridges	Wet and too moisturized fingers
4. Displacement of minutiae	Pressure, different orientation and location of the finger
5. Unequal number of minutiae	Missing or additional minutiae from different regions of a finger

It can be seen in Figure 7 that the minutiae vicinity methods presented in [10]-[12] and [14] get affected when the fingerprint is highly distorted due to uneven pressure applied during the acquisition. This is because when the size of the ridge changes, the locations and orientations of minutiae points changes. As a result, the neighbourhoods of the same minutiae points can differ when different pressure is applied.

Minutiae vicinity and second order vicinity are also affected when there were ridge breakages and some ridges connected, because some feature extractors detect false minutiae points. This lead to unequal number of minutiae with some missing and new ones introduced. However, when the number of minutiae is unequal due to the different region of fingerprint captured, these algorithms are able to construct good binary feature vectors for corresponding fingerprint regions.

Minutiae pattern methods are much affected when minutiae points are displaced. The displacement occurs when a fingerprint is non-linearly distorted and it was skewed when it was captured on the scanner. As a results minutiae points are not removed in the fingerprint but their relative distance and orientation changes.

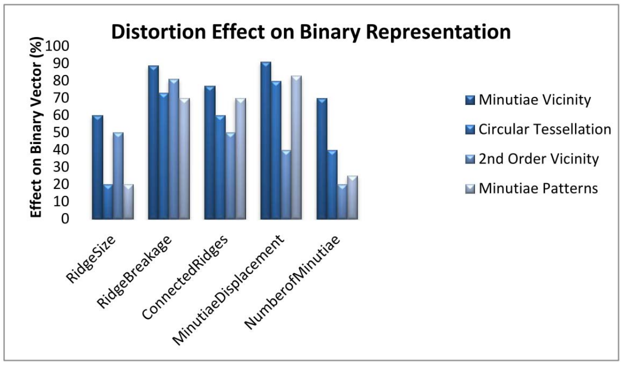

Fig. 7. Analysis of Minutiae-Based Fingerprint Binary Representation

V. Conclusion and Feature works

We have discussed recent and most important methods on presenting minutiae points as binary feature vectors and strings. These methods have been proposed for increasing the accuracy and memory usage in fingerprint matching, while dealing with the problem of distortion, missing or additional minutiae points. The consideration of minutiae neighborhoods lead to the reduction of the effects of distortions in fingerprint matching results. These include principles which can be used on match-on-card algorithms to deal with distortion on minutiae based binary representation of fingerprints and reduce memory required to store fingerprint template of the card owner. In future work we will investigate and develop these methods to find the best of them all, as well as considering error correction methods in solving the problem of distortion on binary representations of fingerprints.

Acknowledgment

Authors like to acknowledge the DST (Department of Science and Technology).

References

[1] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. Handbook of fingerprint recognition. Springer Science & Business Media, 2009.

[2] D.B. Fogel. Biometrics: Theory, Methods, and Applications. IEEE Press Series on Computational Intelligence, 2010. ISBN 978-0470-24782-2.

[3] C. Barral, J. S. Coron, and D. Naccache. "Externalized fingerprint matching." In Biometric Authentication, pp. 309-315. Springer Berlin Heidelberg, 2004.

[4] S. Yang, and I. M. Verbauwhede. "A secure fingerprint matching technique." In Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications, pp. 89-94. ACM, 2003.

[5] S. Bistarelli, F. Santini and A. Vaccarelli, "An Asymmetric Fingerprint Matching Algorithm for Java Card," in Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (5th), pp. 279–288, 2005.

[6] M. Govan and T. Buggy, "A Computationally Efficient Fingerprint Matching Algorithm for Implementation on Smartcards," in Proc. Int. Conf. on Biometrics: Theory, Applications, and Systems (BTAS 07), pp. 1–6, 2007.

[7] F. Faisal, R. M. Bolle, T.Y. Jea, and N. K. Ratha. Anonymous and revocable fingerprint recognition. In CVPR. IEEE Computer Society, 2007.

[8] X. Haiyun, R. N.J. Veldhuis, T. A.M. Kevenaar, A. H.M. Akkermans, and A. M. Bazen. Spectral minutiae: A fixed-length representation of a minutiae set. Computer Vision and Pattern Recognition Workshop, 0:1–6, 2008.

[9] N. Abhishek, S. Rane, and A. Vetro. Alignment and bit extraction for secure fingerprint biometrics. In SPIE Conference on Electronic Imaging 2010, 2010.

[10] J. Bringer, and V. Despiegel, "Binary feature vector fingerprint representation from minutiae vicinities," in [Biometrics: Theory, Applications, and Systems, 2010. BTAS'10. IEEE 4th International Conference on],(2010).

[11] B. Yang, and C. Busch. "Keyed Scalable Minutiae Coding." In Hand-Based Biometrics (ICHB), 2011 International Conference on, pp. 1-5. IEEE, 2011.

[12] J. Bringer, V. Despiegel, and M. Favre. "Adding localization information in a fingerprint binary feature vector representation." In SPIE Defense, Security, and Sensing, pp. 80291O-80291O. International Society for Optics and Photonics, 2011.

[13] F. Benhammadi, and K. B. Bey. "EMBEDDED FINGERPRINT MATCHING ON SMART CARD." International Journal of Pattern Recognition and Artificial Intelligence 27, no. 02 (2013).

[14] T. Bourgeat, J. Bringer, H. Chabanne, R. Champenois, J. Clément, H. Ferradi, M. Heinrich, P. Melotti, D. Naccache, and A. Voizard. "New Algorithmic Approaches to Point Constellation Recognition." In ICT Systems Security and Privacy Protection, pp. 80-90. Springer Berlin Heidelberg, 2014.

[15] A. Vij, and A. Namboodiri, Learning Minutiae Neighbourhoods: A New Binary Representation for Matching Fingerprints. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on (pp. 64-69). IEEE.