FOUNDATIONS OF COMPUTING AND DECISION SCIENCES
Vol. 38 (2013) No. 2

DOI: 10.2478/fcds-2013-0004 ISSN 0867-6356
e-ISSN 2300-3405

DATA MANAGEMENT FOR FINGERPRINT RECOGNITION ALGORITHM BASED ON CHARACTERISTIC POINTS' GROUP

Michał SZCZEPANIK*, Ireneusz JÓŹWIAK*

Abstract. In this paper authors present data management solutions for new fingerprint recognition based on minutes groups. They compared existing fingerprint recognition data store methods and their own solutions. Authors proposed a new algorithm based on distribution minutiae' groups using selective attention algorithms.

Keywords: biometric, fingerprint, minutes group, directing attention algorithms

1. Introduction

Fingerprint recognition is one of the most popular biometric techniques. In addition to comparing the quality of fingerprint algorithm, is also a very important way to store fingerprint data. The first biometric systems retain the original image, this meant that the normally low encrypted data to be stolen and used to force the system. The main problem of data management in biometric systems is structure of a pattern image. It must be encrypted and also ensure that the decryption will prevent reconstruction of the original fingerprint.

The biggest problem of fingerprint recognition systems is usability. Every day, people are exposed to cuts, wounds and burns, therefore it is important that the algorithms were resistant to this type of damage. Current fingerprint recognitions systems for mobile devices are usually use one of the algorithms like:

- Minutiae Adjacency Graph (MAG),
- Elastic minutiae matching (EMM),
- Delaunay Triangulation (DT),
- Pattern-Based Templates (PBT).

Most algorithms based on distribution of minutiae, which are major features of a fingerprint for example: ridge ending and ridge bifurcation.

Most popular algorithms based on local and global structures are represented by graphs like in MAG. In this type of algorithm first local structures are used to find corresponding

* Wrocław University of Technology, Institute of Informatics, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

points to align feature vector, then global structures are matched [4] This type of algorithm was used by Ross et al. [16] He and Ou [6]. They also use thin-plate spline (TPS) model to build an average deformation model from multiple impressions of the same finger. Owing to iteratively aligning minutiae between input and template impressions, a risk of forcing an alignment between impressions originating from two different fingers arises, and leads to a higher false accept rate. Typically, minutiae matching has two steps:

- registration aligns fingerprints, which could be matched, as well as possible
- evaluation calculates matching scores using a tolerance box between every possibly matched points (minutiae) pairs

In MAG algorithm each minutiae is described by 3 or 4 parameters $v=(x, y, T,\Theta)$, where x and y are the coordinate, T is an optional parameter specifying the type and Θ which determines orientation of minutiae. In addition, defined edges connecting the two points representing the minutiae of each edge is defined as follows $e=(u, v, rad, r_c, \Theta)$, where u, v are nodes (minutiae) initial and marginal, *rad* it is Euclidean distance between minutiae, r_c determines the distance by the number of ridges between minutiae, and Θ is the angle between the edge and the axis x.

The EMM algorithm typically uses only global matching, where each point (minutia) which has a type, like end point or bifurcation needs to match to with a related point in second finger print image. Base on elastic deformations which are used to tolerate minutiae pairs that are further apart because of plastic disrotations, and therefore to decrease the False Rejection Rate, so in most popular algorithms authors increase the size of bounding boxes [13] to reduce this problem, but they get higher False Acceptation Rate (FAR) as a side effect. In this type of algorithm for elastic match also TSP [1] can be used, which provides better performance than only one parameter of deformation. In EM algorithm data, about characteristic points, are storage in analogical way like in MAG algorithm. Each minutiae is represented by $p=(x, y, T, \Theta)$, where x and y are the coordinate, T is an optional parameter specifying the type and Θ which determines orientation of minutiae.

The Delaunay Triangulation algorithm [14] based on triangulation connects neighboring minutiae to create triangles, such that no point (minutia) in P is inside the circumcircle of any triangle in $DT(P)$. In this algorithm, the characteristic point information is stored in the same way as in the EMM, the only difference is the method of processing which is based on triangulation. Unfortunately, just as minutiae adjacency graph algorithm is not resistant to injury of physical fingerprint.

Pattern based algorithms [3] compare the basic fingerprint patterns (like arch, whorl, and loop between a previously stored template and a candidate fingerprint. Those algorithms require that the images be aligned in the same orientation and in the same scale. To do this, the algorithm finds a central point in the fingerprint image and centers on that, and after that, scales to the same size of fingerprint's ridge. In a pattern-based algorithm, the template contains the type, size, and orientation of patterns within the aligned fingerprint image. The candidate fingerprint image is graphically compared with the template to determine the degree to which they match. Due to the storage of the original picture for algorithm there is a high risk that this image can be read from the memory card reader or fingerprints database.

2. Fingerprint recognition algorithm based on minutes groups

The proposed solutions, in contrast to other algorithms, are more resistant to damage.

2.1. Fingerprint recognition algorithm based on minutes groups

For older low-resolution readers it is required to detect the areas of correct scanning of the fingerprint. First step of image analysis is the search for the imprint area including the exclusion of areas containing significant damage. Fingerprint image is represented by a gray scale image that defines the area of forced application fingerprint for the reader.

$$I_{fp}(i,j) := < 0,255 > \tag{1}$$

The operation that converts a grayscale image into a binary image is known as binarization. We carried out the binarization process using adaptive thresholding. Each pixel is assigned a new value (1 or 0) according to intensity mean in a local area and the parameter t_g which excludes poorly read fingerprint areas from the analysis.

$$\mathrm{B_{fp}}(\mathrm{i,j}) = \begin{cases} 1 \text{ for } \mathrm{I_{fp}}(\mathrm{i,j}) \geq t_g \\ 0 \text{ for } \mathrm{I_{fp}}(\mathrm{i,j}) < t_g \end{cases} \tag{2}$$

The last step is creating the fingerprint mask based on the binarized image. The Mask for the area of a square *(X, Y)*, which size is 2.5 wide edges, is determined by two parameters p_{lo}, which is a limitation that excludes areas with an insufficient number of pixels describing the image, and p_{hi} excludes blurred areas, such as moist.

$$F_{fp}(X,Y) := \begin{cases} I_{fp}(X,Y) for \sum_{i \in X} \sum_{j \in Y} B_{fp}(i,j) \leq p_{hi} \wedge \sum_{i \in X} \sum_{j \in Y} B_{fp}(i,j) \geq p_{lo} \\ 0 \quad otherwise \end{cases} \tag{3}$$

Created mask is used for finding the most damages area in the fingerprint image [19].

2.2. Detecting features and leveling of the damage in segmentations

Standard leveling of damage is carried out by calculating the variance of points and the analysis of brightness. Based on these two parameters, the frequency of furrows is calculated. Which is used for each fingerprint image.

Figure 1. Fingerprint divided into segments. (Source: own work)

After applying Gabor filter [12][17][18] to highlight the pits and valleys, it uses segmentation, in accordance with its size 2.5 width of segment furrow, the image is redrawn.

After that process fingerprints are continuous and lint. In contrast to the literature, the algorithm does not require additional transformations to find the minutiae, such as converting all the width of 1px furrows. It does not require information about the orientation of minutiae, it only requires the data about its position. Therefore, the resulting image is used to find the edge - the minutiae are located at the intersection of the edge of the furrows.

The problem of fingerprint recognition is a complex process, even in laboratory conditions, therefore, if used as the system to control access to the mobile devices, it should be insensitive to certain natural changes or damages in physical structure of fingerprints, which can include: incomplete fingerprint, fingerprint parts which can be injured or burned, rotation, blurred or partly unreadable.

In order to detect areas most sensitive to damage, we use neural network with selective attention technique . This type of neural network is more like an analysis done by a human. This allows us to create a mask of areas vulnerable to damage.

We created 15 different masks, broken down by the type of fingerprint's core also known as fingerprint patterns (arch, whorl and loop) and the type of finger (thumb, index

finger, middle finger, ring finger, small finger). Basing on this mask we created a filter which we use to compare fingerprints where specific minutiae are weighted in the decision process and their score is based on the location on the fingerprints.

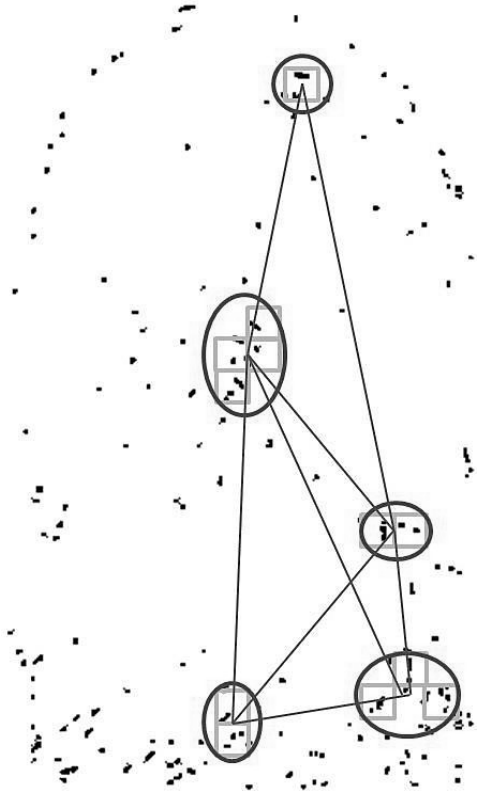

Figure 2. Detecting relations between minutiae groups. (Source: own work)

3. Detecting features and leveling of the damage in fingerprint image

Minutiae image is divided into segments, each segment corresponding to minutiae's group is described by parameters (x, y, nom), where x and y are the coordinates, and nom determines the number of minutiae in the group. Additionally, one implementation uses an additional parameter specifying the probabilities of damage in a given segment which is estimated by a neural network. Based on the distribution of areas rejected by the mask described by the formula. The last step is to create a matrix of Euclidean distances between the groups.

When comparing the use of two parameters: *dx* - the distance, the difference between groups in the pattern and fingerprint test *px* - the threshold probability of damage (determined by whether the group is under consideration in the analysis). When comparing, the groups are divided according to the weight that defines the number of minutiae in the

group and selective attention (SA) algorithms [7], which are based on probabilities of damage in a group segment. This provides quick verification of whether the analysed fingerprint is consistent with the pattern.

4. The quality of the algorithms

For test authors used real fingerprints. Due to the nature of work performed by a group of testers, they were exposed to frequent damage fingerprints.

All four algorithms are compared using the fingerprint database of 120 different fingerprints which had 8 samples of each fingerprint. The database contained 10% of the fingerprints with damage which were mostly cuts and burns, so it simulated the most frequently encountered damage types in daily life. The first test compares the existing algorithms with the proposed one.

Table 1. The result of experiment using real fingerprints

	FAR	FRR
MAG	1.58%	0.95%
EMM	2.35%	2.65%
PBTA	0.35%	8.52%
MGM64	8.45%	0.10%
MGM32	3.10%	0.10%
MGM32 SA	0.30%	0.10%

Most algorithms are almost immune to physical damage of fingerprints. Also, the proposed one has proven to have a very dangerous level of False Acceptation Rate. After applying the selective attention algorithm, fingerprint recognition algorithm improved their performance and reliability. The Proposed algorithm has been developed in such a way, that it uses the property of a damage map, so its results have improved the most.

Table 2. The result of experiment using FVC2004 database

	FAR	FRR
MAG	0.82%	0.65%
EMM	1.23%	1.15%
PBTA	0.15%	1.73%
MGM64	6.90%	0.65%
MGM32	3.66%	0.42%
MGM32 SA	0.35%	0.12%

Second test was done using FVC2004 [12] fingerprints databases. For each of four database a total of 120 fingers and 12 impressions per finger (1440 impressions) were gathered. Unfortunately, most of the publicly available database of fingerprints does not include the problem of physical damage, so additionally on each sample have been generated small damage such as cuts and burns.

5. Data management

Developed algorithm is based on minutiae groups where each group is basically represented by the coordinates - *x, y* and the number of minutiae - *nom* contained in the group. Group covers an area equal to 2.5 the width of the furrow, its coordinates are in the middle of the square which blundering this area. Number of minutiae in the group determines its priority, additionally stored parameter describing the probability of damage - p_d in the area represented by the group. In conclusion the group is defined as follows:

$$M_{group} : \{x, y, nom, p_d\} \tag{4}$$

Based on these data creates a matrix of Euclidean distances between the groups. Data on the characteristic point is limited to its weight (nom) and the probability of damage p_d. Finally we obtain:

$$M_{group}(I) : \{nom_I, (p_d)_I\} \tag{5}$$

$$M_{group}(I, J) : dist(M_{group}(I), M_{group}(J) \tag{6}$$

Where $dist(M_{group}(I), M_{group}(J))$ is Euclidean distances between the group *I* and *J*.

Data stored for analysis to prevent reproduction of the original fingerprint image. Additional storage parameters to estimate the damage Allows you to better match fingerprints in the event of damage.

6. Conclusion

The proposed algorithm enables the identification of fingerprints in places where they are exposed to frequent damage. Way to store fingerprint data makes it impossible to recreate the original structure. It provides information about the minutiae clusters and their frequency. In the future work, data management will be extended by storing information about events of damages which were recognized by identified fingerprints algorithm.

Acknowledgements

This experiment and publication is co-financed by the European Union under the European Social Fund.

References

[1] Bazen A.M., Gerez S.H. (2003) Fingerprint matching by thinplate spline modelling of elastic deformations, *Pattern Recognition*.

[2] Bebis G., Deaconu T., Georgiopoulos M. (1999) Fingerprint Identification Using Delaunay Triangulation, IEEE ICIIS: 452-459
[3] Cappelli R., Lumini A., D. Maio and D. Maltoni (1999) Fingerprint Classification by Directional Image Partitioning, *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol.21, no.5:402-421
[4] Chikkerur S., Govindaraju V., Cartwright E. N.(2006) K-plet and coupled bfs: A graph based fingerprint representation and matching algorithm. *LNCS*: 309 – 315
[5] Grzeszyk C. (1992) *Forensic fingerprint examination marks*. (in Polish), Wydawnictwo Centrum Szkolenia Policji, Legionowo
[6] He Y., Ou Z.(2005) Fingerprint matching algorithm based on local minutiae adjacency graph, *Journal of Harbin Institute of Technology* 10/05, pp. 95-103
[7] Huk M., Szczepanik M. (2011) Multiple classifier error probability for multi-class problems. *Maintenance and Reliability* 3: 12-17
[8] Hicklin A., Watson C. Ulery B. (2005) How many people have fingerprints that are hard to match, *NIST Interagency Report* 7271.
[9] Hong L., Wan Y., Jain A. K. (1998) Fingerprint image enhancement: Algorithm and performance evaluation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*:777-789
[10] Indovina M., Uludag U., Snelick R., Mink A., Jain A. (2003) Multimodal Biometric Authentication Methods: *A COTS Approach, Proc. MMUA*.
[11] Jain A. K., Ross A., Nandakumar K. (2011). Introducing to biometrics, Spinger.
[12] Maltoni D., Maio D., Jain A.K., Prabhakar S. (2009) *Handbook of Fingerprint Recognition*, 2nd Edition, Springer
[13] Pankanti S, Prabhakar S., Jain A.K. (2001) On the individuality of fingerprints, *Proceedings of Computer Vision and Pattern Recognition (CVPR*).
[14] Parziale G., Niel A. (2004) A fingerprint matching using minutiae triangulation. *Proc. of ICBA*.
[15] Ratha N. K., Govindaraju V. (2007*) Advances in Biometrics: Sensors, Algorithms and Systems*, Springer.
[16] Ross A., Dass S.C., Jain A.K. (2005) A deformable model for fingerprint matching, *Pattern Recognition* 38(1): 95–103
[17] Ross, A., Nandakumar K., Jain, A.K. (2011) *Handbook of Multibiometrics* (International Series on Biometrics), Springer.
[18] Shimooka T., Shimizu K. 2004. *Artificial Immune System for Personal Identification with Finger Vein Pattern*, Springer.
[19] Szczepanik M., Szewczyk R. (2008) *Fingerprint identification algorithm* (in Polish). KNS, vol. 1: 131 -136
[20] Wayman J.L., Jain A.K., Maltoni D., Maio D. (2005) Biometric Systems. Technology, *Design and Performance Evaluation*, 1st Edition, Springer

Presented at the 16th East-European Conference on Advances in Databases and Information Systems September, 17-20, 2012, Poznań, Poland