JOURNAL OF MEDICAL INFORMATICS & TECHNOLOGIES Vol. 13/2009, ISSN 1642-6037

minutiae points, matching score, fingerprint matching

Łukasz WIĘCŁAW*

A MINUTIAE-BASED MATCHING ALGORITHMS IN FINGERPRINT RECOGNITION SYSTEMS

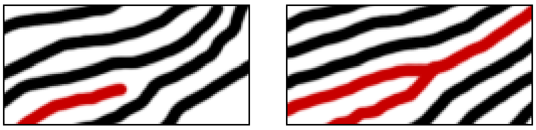
This study presents advantages of the most important methods of minutiae-based matching algorithm in fingerprint recognition systems. Minutia matching is the most popular approach to fingerprint identification and verification. Fingerprint matching usually consist of two procedures: minutia extraction and minutia matching. The performance mostly depends on the accuracy of the minutia extraction procedure. Minutiae matching designate the time complexity of applied solution.

1. INTRODUCTION

Fingerprints are the most used biometrics technique for personal identification. There are two main applications involving fingerprints: fingerprint verification and fingerprint identification [1]. While the purpose of fingerprint verification is to verify the identity of a person, the goal of fingerprint identification is to establish the identity of a person. In the past three decades, automatic fingerprint verification is being more widely than other techniques of biometrics such as face identification and signature identification. Usually associated with criminal identification, now has become more popular in civilian applications, such as financial security or access control.

Many fingerprint identification methods have appeared in literature over the years [1, 5, 7]. The most popular matching approach for fingerprint identification is usually based on lower-level features determined by singularities in finger ridge patterns called minutiae. In general, the two most prominent used features are ridge ending and ridge bifurcation (Fig. 1). More complex fingerprint features can be expressed as a combination of these two basic features.

Minutiae matching essentially consist of finding the best alignment between the template (set of minutiae in the database) and a subset of minutiae in the input fingerprint, through a geometric transformation.



a) b)

Fig. 1. Example of a) ridge ending and b) bifurcation.

2. MINUTIAE EXTRACTION

Typically each detected minutiae m_i is described by four parameters:

$$m_i = (x_i, y_i, \theta_i, t_i) \tag{1}$$

where:

- x_i, y_i – are coordinates of the minutiae point,
- θ_i – is minutiae direction typically obtained from local ridge orientation,
- t_i – is type of the minutiae point (ridge ending or ridge bifurcation),

The position of the minutiae point is at the tip of the ridge or the valley and the direction is computed to the X axis (Fig 2).

* Institute of Informatics, University of Silesia, Będzińska 39 St., 41-200 Sosnowiec, Poland, lukasz.wieclaw@us.edu.pl

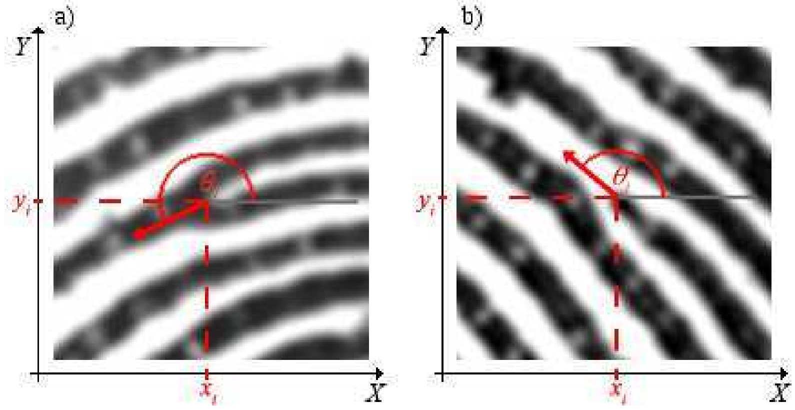


Fig. 2. Parameters of minutiae a) bifurcation and b) ridge ending type.

2.1. FEATURE EXTRACTION

Two approaches of minutia extraction process can be found. The simplest and most used method is based on binarization and ridge thinning stage. Due to a problem of the false minutiae introduced by thinning, some authors proposed direct grey-scale minutiae extraction.

2.2. RIDGE THINNING METHOD

The most commonly used method of minutiae extraction is the *Crossing Number* (*CN*) concept [2, 3, 4]. The binary ridge image needs further processing, before the minutiae features can be extracted. The first step is to binarizate and further to thin the ridges, so that they are single pixel wide (Fig. 3). A large number of skeletonization methods are available in the literature, due to important role in many recognition systems. Rata, Chen and Jain [6] adopted a technique included in HIPS library. One of the most tolerant on irregularity of binary images is method proposed by Pavlidis [7].

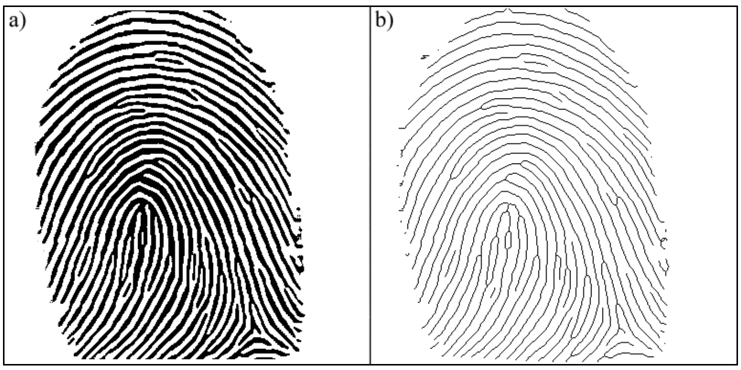


Fig. 3. Fingerprint image a) binarization and b) skeletonization.

The minutiae points are determined by scanning the local neighbourhood of each pixel in the ridge thinned image, using a 3×3 window (Fig. 4).

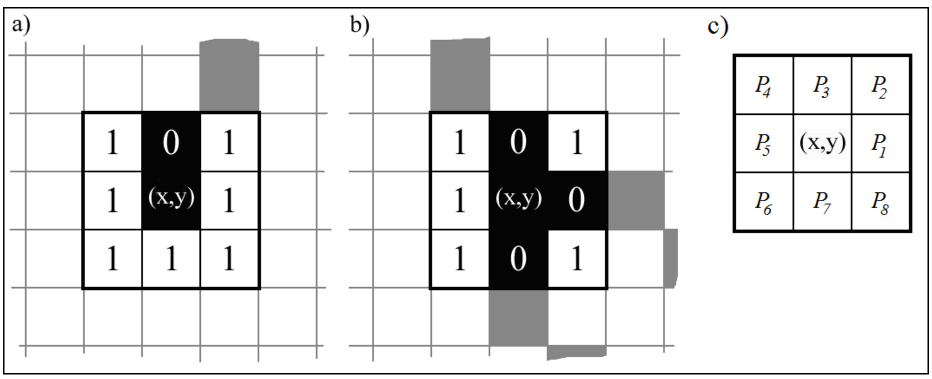


Fig. 4. a) Ridge ending and b) bifurcation in c) 3×3 window.

The *CN* value is then computed, which is defined as half the sum of the differences between pairs of neighbouring pixels p_i and p_{i+1} [8]:

$$CN_{(x,y)} = \frac{1}{2}\sum_{i=1}^{8}|p_i - p_{i+1}|, \ \ p_1 = p_9 \qquad (2)$$

Using the properties of the *CN* as shown in Table (Fig. 5), the ridge pixel can be then classified as a ridge ending, bifurcation or non-minutiae point.

CN	Property
0	Isolated point
1	**Ridge ending**
2	Continuing ridge
3	**Bifurcation**
4	Crossing

Fig. 5. Properties of the *Crossing Number*.

The main problem, in the minutiae extraction method using ridge thinning processes, comes from the fact that minutiae in the skeleton image do not always correspond with true minutiae in the fingerprint image. In fact, a lot of false minutiae are extracted because of undesired spikes, breaks, and holes.

For this reason, time-consuming enhancement algorithms are required prior to thinning stage [9].

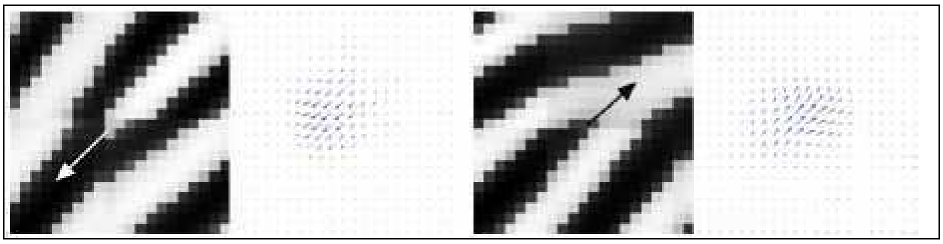
2.3. DIRECT GREY-SCALE METHOD

Minutiae extraction approaches, that work directly on the grey-scale images, without binarization and thinning, was induced by these consideration [9,10]:

- enhancement algorithms are time-consuming,
- a significant amount of information may be lost during the binarization process,
- skeletonization may introduce a large number of false minutiae
- unsatisfactory results when applied to low quality images.

Maio and Maltoni [10] proposed a direct-grey scale minutiae extraction technique. Their basic idea is ridge tracing, by sailing according to the local orientation. The ridge line algorithm attempts to locate at each step, the local maxima, relative to a section perpendicular to the local ridge direction. The algorithm avoids revisiting the same ridge, by keeping track of the points traced so far. They also compared their method to binarization and thinning approaches and concluded that ridge following, significantly reduce computation time.

Nilsson and Bigun [11] proposed using Linear Symmetry (LS) filter in the minutiae extract approach, based on the concept that minutiae are local discontinuities of the LS vector field. Two types of symmetries - parabolic symmetry and linear symmetry are adapted to model and locate the points in the grey-scale image, where there is lack of symmetry (Fig. 6).



a) b)

Fig. 6. Symmetry filter response in the minutiae point. a) ridge bifurcation, b) ridge ending (from [12]).

Finally, Govindaraju, Schneider and Shi [13] proposed a new algorithm based on chain code contour following. Chain codes have been used in computer vision to describe the shapes of object boundaries and in this case they are loss-less representation of ridge contours, at the same time yielding a wide range of information about the contour such as curvature, direction, length etc [13]. As the contour of the ridges is traced consistently in a counter-clockwise direction, the minutiae points are encountered as locations, where the contour has a significant turn. Specifically, the ridge end occurs as significant left turn and the bifurcation as a significant right turn in the contour (Fig. 7). Analytically, the turning direction may be determined by considering the sign of the cross product of the incoming and outgoing vectors at each point.

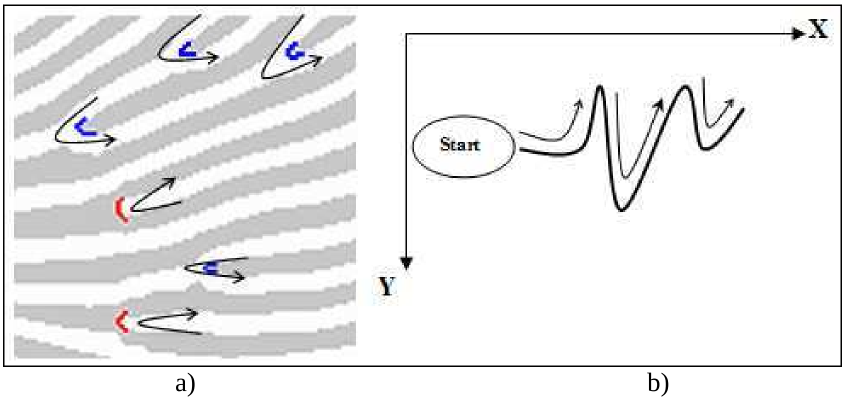


Fig. 7. a) Minutiae marked by significant turn in the contour,
b) the contour extracted by tracing the ridge boundaries in a counter clockwise direction.

2.4. ORIENTATION ESTIMATION

Fingerprint images can be considered as an oriented texture pattern. The orientation field of a fingerprint image prescribes the local orientation of the ridges, contained in the fingerprint. Therefore orientation field define the direction of the minutiae. There have been several approaches to estimate the orientation field of a fingerprint image. Approaches based on pixel alignment relative to a fixed number of reference orientation [14, 15] do not provide very accurate estimates (Fig 8b).

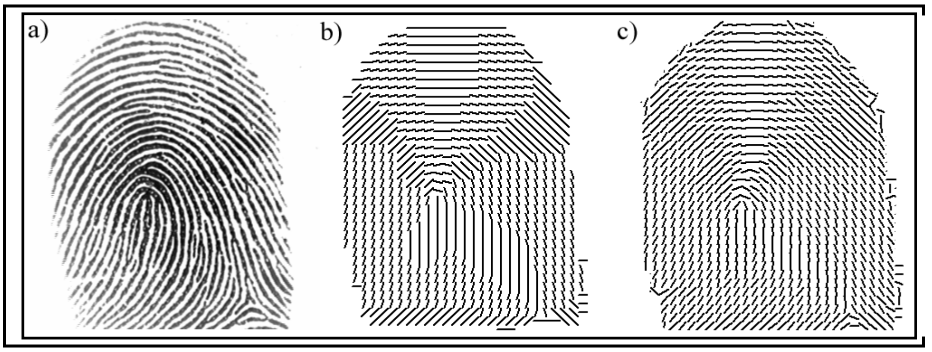


Fig. 8. a) Fingerprint image, b) discrete orientation field, c) orientation field estimated by last mean square method.

The simplest and most natural approach for orientation field estimation is based on computation of gradients in the fingerprint image. The least mean square estimation method employed by Hong [16] is most popular.

The local orientation at pixel (i, j) can then be estimated using the following equations:

$$V_x(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} 2\partial_x(u,v)\partial_y(u,v), \tag{3}$$

$$V_y(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} \partial_x^2(u,v)\partial_y^2(u,v), \tag{4}$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_y(i,j)}{V_x(i,j)}, \tag{5}$$

where:

$\theta(i,j)$ - is the least square estimate of the local orientation at the block centred at pixel (i, j),

∂_x, ∂_y - are the gradient magnitudes (the Sobel operator) in the *x* and *y* directions.

Further, orientation field need to be smoothed in a local neighbourhood using a Gaussian filter.

3. MATCHING SCORE

In a good quality rolled fingerprint image, there are about 70 to 80 minutiae points and in a latent fingerprint the number of minutiae is much less (approximately 20 to 30) [5].

A minutiae-based fingerprint matching system usually returns the number of matched minutiae on both query and reference fingerprints and uses it to generate similarity scores. According to forensic guidelines, when two fingerprints have a minimum of 12 matched minutiae, they are considered to have come from the same finger [3].

Matching algorithm compares two minutiae sets: template $T = \{m_1, m_2, ..., m_j\}$ from reference fingerprint and input $I = \{m_1, m_2, ..., m_i\}$ from the query and returns similarity score $S(T, I)$.

The minutiae pair m_i and m_j are considered to be match only if difference in their position and directions are lower then tolerance distances:

$$sd(m_i, m_j) = 1 \Leftrightarrow \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \le r_0 \tag{6}$$

$$dd(m_i, m_j) = 1 \Leftrightarrow \min\left(|\theta_i - \theta_j|, 360 - |\theta_i - \theta_j|\right) < \theta_0 \tag{7}$$

In addition, it should be noted, that in some cases the bifurcation and ridge ending points can be difficult to distinguish between each other. Hence, in practice, most fingerprint identification systems do not make a difference between bifurcations and ridge endings, when matching minutiae points [11].

4. MATCHING ALGORITHMS

For matching regular sized fingerprint images, a brute-force matching, which examines all the possible solutions, is not feasible since the number of possible solutions increases exponentially with the number of feature points on the prints [3].

Transformation of input minutiae set, is the most important step, in order to maximize the value of similarity score.

Let *map* be transformation function that maps the minutia set from *I* to *I'* according to given geometrical transformation. Then, matching problem can be formulated as:

$$S(T, I) = \max\left(\forall_m \sum_{i=1}^{n} md\left(m_i, map_m(m_i')\right)\right) \tag{8}$$

$$md(m_i, m_j) = sd(m_i, m_j) \cdot dd(m_i, m_j) \tag{9}$$

where:

n - is the number of minutiae points in *I* input set,

m - is the number of transformation equal to the number of minutiae in *T* template set,

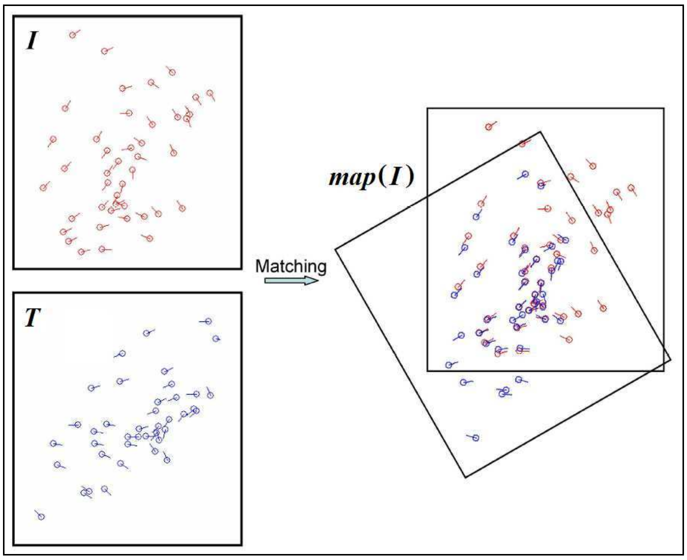


Fig. 9. Minutiae based matching.

Ratha [17] proposed a method that searches the geometric transformation parameters in four-dimensional Hough space. By specifying scale, rotation and shift parameters, a Hough transform was conducted on a minutiae set. A score can be obtained by specifying these three parameters.

Jain [18] proposed an alignment-based matching method, in which adopted the associated ridge to align the input minutiae with the template minutiae. Good performance is reported to overcome deformation. However, Tong noticed [19], that if only short part of a ridges is saved, the algorithm may results in inaccurate alignment. Furthermore, sometimes it is difficult to find long ridges in a thinned fingerprint image.

Jiang and Yau [20] proposed a minutia matching method, using both global and local structure of features. In this technique, local structure was used to find the correspondence pair of minutiae and the global structure was used to compute similarity score. However, if less neighbourhood minutiae is used, false reject rate may arise in case of the presence of false minutiae.

Tico and Kuosmanen [21] adopted an Feature-based minutia descriptor for minutiae matching, and good performance are reported. However, ridge count are not included in the descriptor, which has been widely used and reported good performance [20,22].

Lee [23] proposed a local alignment method. In this method, ridge frequency value was used to minimize distance error, by normalizing the distance between minutiae. But the minimizing distance by frequency makes the algorithm more time-consuming.

5. CONCLUSIONS

In this paper, a minutia matching systems has been described. A minutiae-based fingerprint verification system is divided in two main blocks: the feature extraction block and the matching block. Main problem in feature extraction section is quality of fingerprint image. Low quality areas of fingerprint occurs large number of false minutiae point. Most important in matching stage, is selection of tolerance distances and transformation method. When tolerance values are increasing, then false accept rate is also rising. When transformation of input minutiae set is not precise, then false reject rate value is high.

BIBLIOGRAPHY

[1] BEBIS G., DEACONU T., GEORGIOPOULOS M., Fingerprint Identification Using Delaunay Triangulation, Proc. of Int. Conf. on Information Intelligence and Systems, pp. 452-459, Washington, DC, USA, 1999.

[2] AMENGUAL J., JUAN A., PREZ J., PRAT F., SEZ S., VILAR J., Real-time minutiae extraction in fingerprint images, Proc. of the 6th Int. Conf. on Image Processing and its Applications, pp. 871–875, Ireland, 1997.

[3] MEHTRE B. M., Fingerprint image analysis for automatic identification, Machine Vision and Applications 6, 2, pp. 124–139, India, 1993.

[4] BOASHASH B., DERICHE M., KASAEI S., Fingerprint feature extraction using block-direction on reconstructed images, IEEE region TEN Conf., digital signal Processing applications, TENCON pp. 303–306, Australia, 1997.

[5] GOVINDARAJU V., JEA T., Minutiae-based partial fingerprint recognition, Pattern Recognition, Vol. 38, pp. 1672-1684, USA, 2005.

[6] CHEN S., JAIN A., RATHA K., Adaptive Flow Orientation-Based Feature Extraction in. Fingerprint Images, Pattern Recognition, Vol. 28, No. 11, pp. 1657-1672, USA, 1995.

[7] PAVLIDIS T., A thinning algorithm for discrete binary images. Computer Graphics and Image Processing, Vol. 13, pp.142–157, 1980.

[8] TAMURA H., A comparison of line thinning algorithms from digital geometry viewpoint. Proc. of the 4th Int. Conf. on Pattern Recognition, pp. 715–719, 1978.

[9] MALTONI D., MAIO D., JAIN A.K., PRABHAKAR S., Handbook of Fingerprint Recognition. Springer, New York, 2003.

[10] MAIO D., MALTONI D., Direct Gray-Scale Minutiae Detection In Fingerprints, IEEE Trans. Pattern Anal. Machine. Intell., vol 19, pp. 27-40, USA, 1997.

[11] BIGUN J., NILSSON K., Using linear symmetry features as a pre-processing step for fingerprint images, Conf. Audio and Video Based Biometric Person Authentication, pp.247–252, Sweden, 2001.

[12] BIGUN J., HARTWIG FRONTHALER K., Local feature extraction in fingerprints by complex filtering, Advances in Biometric Person Authentication, LNCS, Vol. 3781, pp.77–84, 2005.

[13] GOVINDARAJU V., SCHNEIDER J., SHI Z., Feature Extraction Using a Chaincoded Contour Representation, Int. Conf. on Audio and Video Based Biometric Person Authentication, UK, 2003.

[14] KAWAGOE M., TOJO A., Fingerprint pattern classification, Pattern Recogntion, Vol 17, No. 3, pp. 295-303, USA, 1987

[15] PORWIK P., Fast fingerprint recognition method based on reference point location, Proc. of the IEEE Workshop on Signal Processing. Poznań, 29 September, pp. 13-22, 2006.

[16] HONG L., JAIN A., WAN Y., Fingerprint image enhancement: algorithm and performance evaluation, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.20, No.8, pp.777-789, 1998.

[17] CHEN S., JAIN A., KARU K., RATHA K., A real-time matching system for large fingerprint databases, IEEE Trans. Pattern Anal. Machine. Intell, Vol. 18, pp. 799-813, USA, 1996.

[18] BOLLE R., HONG L., JAIN A., On-line fignerprint verification, IEEE Trans. Pattern Anal. Machine. Intell, Vol. 19, No. 4, pp. 302-314, USA, 1997.

[19] TONG X., HUANG J., Tang X., Shi D., Fingerprint minutiae matching using the adjacent feature vector, Pattern Recognition Letters, Vol. 26, No. 9, pp. 1337-1345, 2005.

[20] JAING X., YAU W., Fingerprint minutiae matching based on the local and global structures. ICPR2000, Vol. 2, pp. 1042-1045, 2000.

[21] TICO M., KUOSMANEN P., Fingerprint matching using an Feature-based minutia descriptor. IEEE Trans. Pattern Anal. Machine. Intell, Vol. 25, No. 8, pp. 1009-1014, 2003.

[22] BOLLE R., PANDIT V., RATHA K., VAISH V., Robust fingerprint authentication using local structural similarity. Fifth IEEE Workshop on Applications of Computer Vision, pp. 29-34, 2000.

[23] CHOI K., LEE D., KIM J., A robust fingerprint matching algorithm using local alignment. International Conference on Pattern Recognition, Vol. 3 pp. 11-15, 2002.