Contents

Chapter 1

More Than Two Thousand Years Of Fingerprinting

However the very first research paper on fingerprint identification was published in 1684, the first record of fingerprint on official documents is indeed dated [?] as far back as the third century B.C., by the Chinese who were using it to sign land sales, contracts, loans or acknowledgement of debts[1].

After Dr. Nehemiah Grew published his first paper in 1684, he was closely followed in 1686 by the publication of Professor Marcello Malpighi (*De Extemo Tactus Organo*) [?]. However, this is not before 1798 that the first theory on fingerprint uniqueness and permanence was given by J.C. Mayer. Later, in the years 1896 to 1897, Sir Edward Henry, Inspector General of the Bengal Police in India, developed the *Henry Classification System* [?], [?], which was the first fingerprint matching system to find a worldwide acceptance because its classification achieved a significant increase of performances by reducing the matching time from days or weeks to only few hours.

Even though fingerprint is nowadays considered the most reliable identification method, manual verification has indeed become obsolete because of its obvious tediousness and cost. Therefore, using an automatic fingerprint identification system (AFIS) is the only chance to cope with today's performance requirements for both forensic and civilian applications such as criminal identification, access control or identity verification.

[1]Although historical records also relate the use of fingerprints to establish identity in courts, it is not agreed by historians and researchers wheter or not the Chinese were aware of the uniqueness of fingerprints

Chapter 2

Basis about the AFIS process

The litterature describes that two given fingerprints come from the same person if there is a topological match [?] between their features patterns. However, matching two fingerprints is not an easy task. Issues such *rotation, translation, deformation* and spurious or missing *minutiae* are to be addresses with the use of techniques from image processing and pattern recognition methods.

To introduce to the typical steps carried out within the approaches described in the review, the next two sections will be describing a simple approach of features extraction and matching, as described in [?].

2.1 Minutiae extraction

However *loops, archs* and *whorls* are not being considered for this extraction, it will be useful for the later methods to know that they are described as *global features* while *minutiae* are *local features*. Out of the eighteen different types of local ridges identified by [?], the most noticeable are the *ridge endings* and *ridge bifurcations* (see figure **??**), usually referred to as *minutiae*. Before being matched, the minutiae have to be extracted from a fingerprint image. [?] considers three steps for that purpose :

- Estimation of orientation field
- Ridge detection
- Minutiae detection

The orientation of the *flow-like* ridges can be done using the Rao's [?] algorithm. The image is then enhanced using image processing techniques.

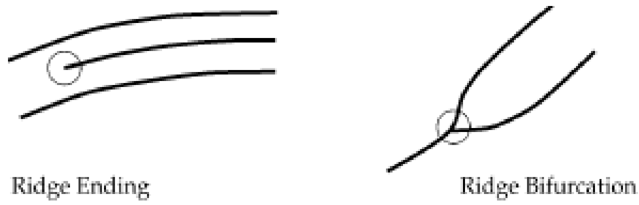


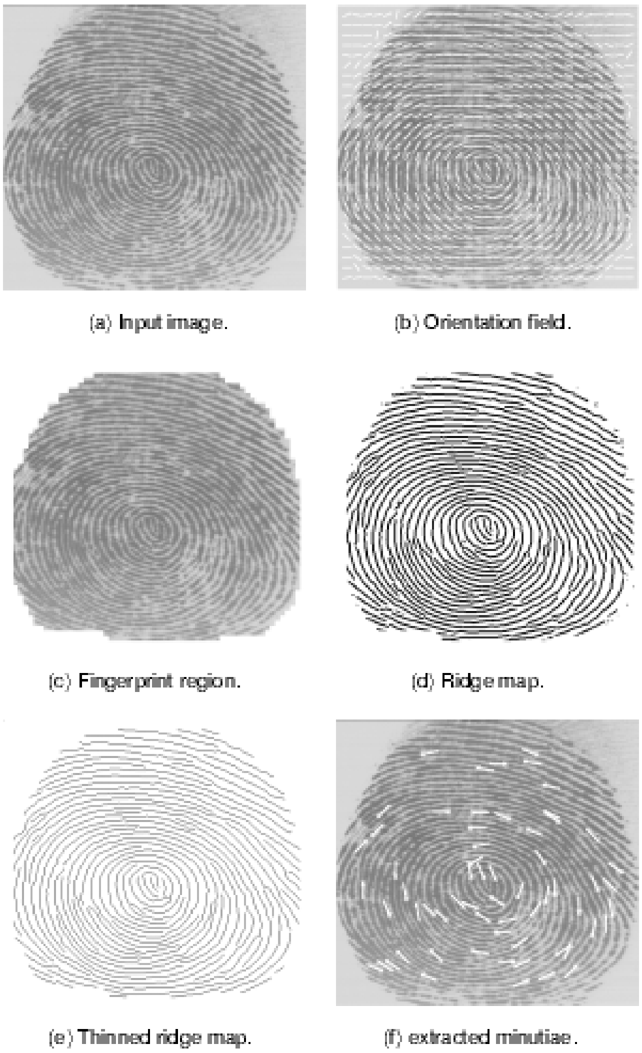
Figure 2.1: Ridge endings and bifurcations

Regarding the detection of ridges, [?] gives the following property: "gray level on ridges attain their local maxima along the normal directions of the local ridges". This property can be applied with the use of convolution masks that take the (previously estimated) orientation in consideration. Still after this image enhancement step, spurious minutiae are left due to noise, breaks and smudges in the fingerprint image. Heuristics and algorithms can then be used to delete spurious minutiae and reconnect broken ridges, thus finally producing a thinned *ridge map* where minutiae shall be easy to detect.
A technique to then locate the minutiae is to assume that a black pixel is a *ridge ending* if there is exactly 1 single black pixel among his 8 neighbooring pixels while a *ridge bifurcation* pixel would require strictly more than 2.

The different steps of the minutiae detection are illustrated in figure ??. Even though the given techniques used during the following steps of the extraction are not the most efficient, they are the basis of most of the later methods.

2.2 Minutiae matching

Once the minutiae have been extracted from the fingerprint image, they can be represented as points. Thus the verification process becomes a *point pattern* matching. Ideally, the points of the template pattern can be aligned completely with those of the input pattern. However, due to noise, deformation, rotation and translation, a perfect alignment of the points is not realistic. Therefore minutiae matching methods must be tolerant (to some extent) and also cope with issues such as rotation and translation.



(a) Input image.
(b) Orientation field.
(c) Fingerprint region.
(d) Ridge map.
(e) Thinned ridge map.
(f) extracted minutiae.

Figure 2.2: Extraction of minutiae

Chapter 3

Fingerprint matching methods

As explained in the previous section, a basic fingerprint matching can be described as a *point pattern matching*. Such patterns can be matched using genetic algorithms [?] and other methods of pattern recognition [?], [?]. Such methods are slow, due to their iterative nature. Numerous research papers propose different approach to fingerprint verification. Indeed, gathering more information about the minutiae such as their type (bifurcation or ending), direction and correlation but also considering the ridges and global features and, finally, combining different methods allow to bring more discrimination and get far better accuracy. The next sections will introduce some of these methods and describe their working, advantages and shortcomings.

3.1 Topology-based matching

An important key to fingerprint matching is the scheme being used to classify the extracted features. The Henry classification system [?], which was the first effective system, is based on the global features of the ridges and has been used during many years [?] as a mean to partition fingerprint databases. Considering the local features of the fingerprint shall substantially improve the classification. However, the fact that distortions of the pattern can not be avoided is a critical issue for such a classification.

With this regard, [?] proposes an topology-based classification where the structure of the fingerprint is represented by a geometrical configuration of its minutiae, thus reflecting the general spatial structure in the neighborhood of each minutiae. After the minutiae have been extracted, a specific neighborhood, called the *minutiae window*, is determined for each of them. In each *minutiae window*, the following information are collected:

- Total number of surrounding minutiae
- Number of ridges between the central minutiae of the *window* and the surrounding ones
- Orientation of the surrounding minutiae
- Angle between the direction of the central minutiae ridge and the ridge of surrounding minutiae
- Distance between the central minutia and the surrounding ones

Afterwards, the *minutiae windows* are ordered according to the number of minutiae they contain. At this point we could argue that this ordering scheme does not take enough characteristics of the *minutiae window* for ordering them (two *windows* may have the same amount of surrounding minutiae). We could also mention that spurious and missing minutiae (due to already cited factors) will further degrade the order. However, it appears that this is not a problem at all when using a tree matching. Furthermore, if noise issues can not be avoided, it must be noticed that the features located within a *minutiae window* do not suffer of rotations or translations. The tree matching can be described as follows:

- On a one-to-one basis, line up the surrounding minutiae in the pair of initial *window*.
- Starting at the root, generate a tree via a breadth-first search scheme.
- Continue the tree expansion if similarly constructed counterpart can be found in the template (classified as describe before), until all minutiae have been used.
- If no counterpart is found, re-iterate until considering a non-match if non sufficiently large tree can be generated.

The scheme relies on the assumption that surrounding features in a close neighborhood are supposely invariant to rotation and translation. However it appears that minutiae associated to global features such as *whorls* or located in noisy regions are quite troublesome. An advantage of this matching method might be his classification scheme. Since features are partitioned in the database, it should increase the efficiency of one-to-many queries in a database. However, the tolerance required to cope with spurious or missing minutiae would substantially affect the efficiency of queries.

3.2 Orientation-based Minutia Descriptor

Keeping in mind that considering only the minutiae location and direction does only exploit a very low amount of the rich information content that is present in fingerprint patterns, [?] introduces a *minutiae descriptor* that relies on describing the orientation information of the fingerprint pattern with regards to each minutiae details. Fingerprint are here classified as "weakly-order textures exhibiting a dominant ridge orientation at each point". To compare with the method in [?], the description around each minutiae is orientation-based, rather than minutiae-based. In other words, not the surrounding minutiae, but the ridge orientation of each of them are considered. Their orientation are even invarient to rotation and translation with noisy inputs, thus bringing a substantial advantage over the minutiae-based method. The *minutiae descriptor* is further said to be "independant with respect to any other minutiae.

The matching is then done by performing the following steps:

- Detect the minutiae of the input pattern and pair as many of them with those of the template.
- For each corresponding pair of minutiae, compute their match (ridges direction distance) to end up with a total matching score normalised with the number of overlapping minutiae.
- The match is then determined by evaluating the score against an empirical treshold.

The experimental results report that the rejection rates increases with factors such as errors in minutiae detection, unreliable orientation. Poor quality images are also an issue because they reduce the number overlapping (because not detected) minutiae to be compared.

3.3 Alignment-based matching

As it has already been explained, due to deformations and consequences of noises, it is not realistic to consider that the minutiae of overlapping patterns will perfectly match. In [?], alignment-based recognition [?] is used in the matching algorithm because of its simplicity, efficiency of discrimination and speed. The algorithm can be divided into two main steps:

- *Alignment*: transformations due to rotation, translation and scaling are estimated and then used to realign the minutiae of the input pattern with the template (see figure **??**)

- *Matching*: conversion of the input and template minutiae into polygons in the polar coordinate system and use of an elastic matching algorithm to match the polygons

Figure **??** illustrates the alignment during the matching process. In the first step the input pattern is aligned with the template by rotation and translation. The elastic point pattern algorithm concatenates each of the minutiae (then represented as polygons in the polar coordinate system) into two strings in the increasing order of radial angles. The distance between the two strings is calculated and used to compute the match score.

However non-linear transformations might have remained after the first step, the elastic algorithm in polar coordinates has the great advantage to achieve a certain amount of tolerance, thus coping with that issue. On the other side, if this scheme is tolerant to inexact localizations and non-linear modifications, it does not indeed compensate (which would be more clever). Also, string matching algorithms are not the most efficient in term of speed and this might be an issue when using the scheme for identification on large databases.

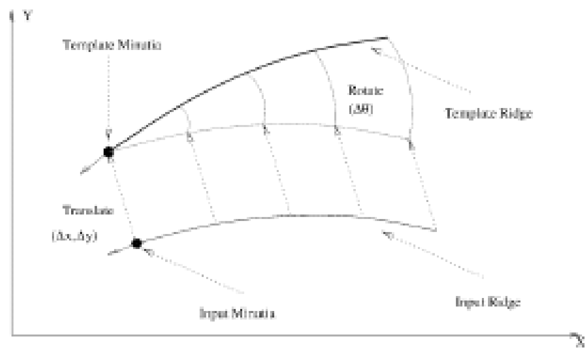


Figure 3.1: Alignment of an input ridge and a template ridge

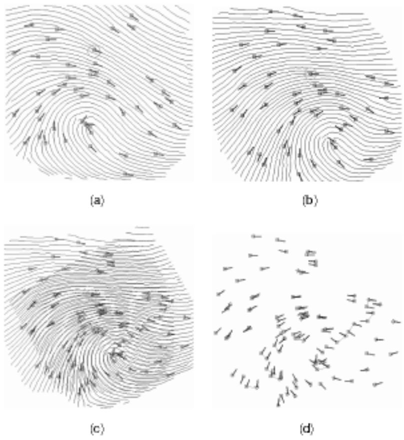


Figure 3.2: (a) Input minutiae set, (b) template minutiae set, (c) alignment, (d) pairing of the minutiae

3.4 Bank of Gabor filtering

[?] proposes a method capturing both the local and global features into a so-called *FingerCode*, using a bank of Gabor[1] filters. The matching is then reduced to finding the euclidean distance between these *FingerCodes* and hence it shall be very fast. The method performs the following steps:

- Determine a *reference frame* for the image and tesselate this region into 8 sectors
- Using the bank of Gabor filters, filter the image in 8 different directions
- Compute the grayscale variance (standard deviation) in each sector and thus define the feature vector of the *FingerCode*

The verification is achieved by measuring the euclidean distance of the query fingerprint and of the one of the template. The flow diagram of the method for fingerprint verification is illustrated in figure **??**.

The first obvious advantage of this method is that the rotation invariance can be very easily achieved. Thanks to its circular shape, the tesselated region (see figure **??**(a)) can be simply rotated like a disc. The *FingerCodes* are 640-dimensional (8 filtered images tesselated into 80 cells). The matching

[1] Gabor filterbank is an efficient technique to perform the capture and decomposition of useful information in a particular bandpass

is very fast because it is done by measuring their euclidean distance. Due to the nature of the data and because the calculation of the euclidean distance involves bit comparison, the matching scheme is suitable for hardware implementation. Furthermore, because of their small size, the templates could easily be embeded in the chip of a smart card.

This methods also brings its collection of shortcomings. The first one is

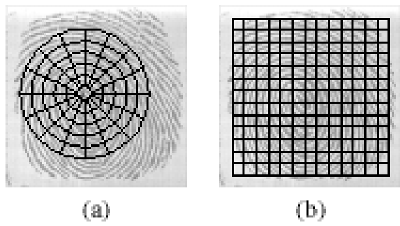


Figure 3.3: Tesselations

the determination of the *reference frame*. The detection of the core point is not a trivial task. The region of interest may also be partially omited during acquisition or it could even happen that the core point (center of the *reference frame*) is not even present in the acquired image. Occlusion and obliteration have the same consequences. It is true that the circular tesselation offers the great advantage to achieve very easily the rotation invariance, but on the other side it does not cover the whole image, thus reducing discrimination and therefore accuracy. This gets even worse if the core point is determined to be close to the boundary, because we obtain not only a partial image but partial image which is partially void. In this case, the features in the *Fingercode* can not very discriminative enough. The experiments also show that about 99% of the process is spent on the Gabor filtering

[**?**] offers a hybrid matching method, developed by members of [**?**] and aiming at addressing these issues. The framework remains basically the same. The changes can be resumed by the following points:

- The fingerprint image is enhanced prior to filtering, thus offering more accurate information on the ridges.
- The tesselation is a square (see figure **??**(b)) that does not consider a landmark point, thus covering the entire image.
- The minutiae are extracted with the method of [**?**] and are indeed used to align the squared tesselation and add more discrimination.

- The Gabor filter computation is achieved in the frequency space.

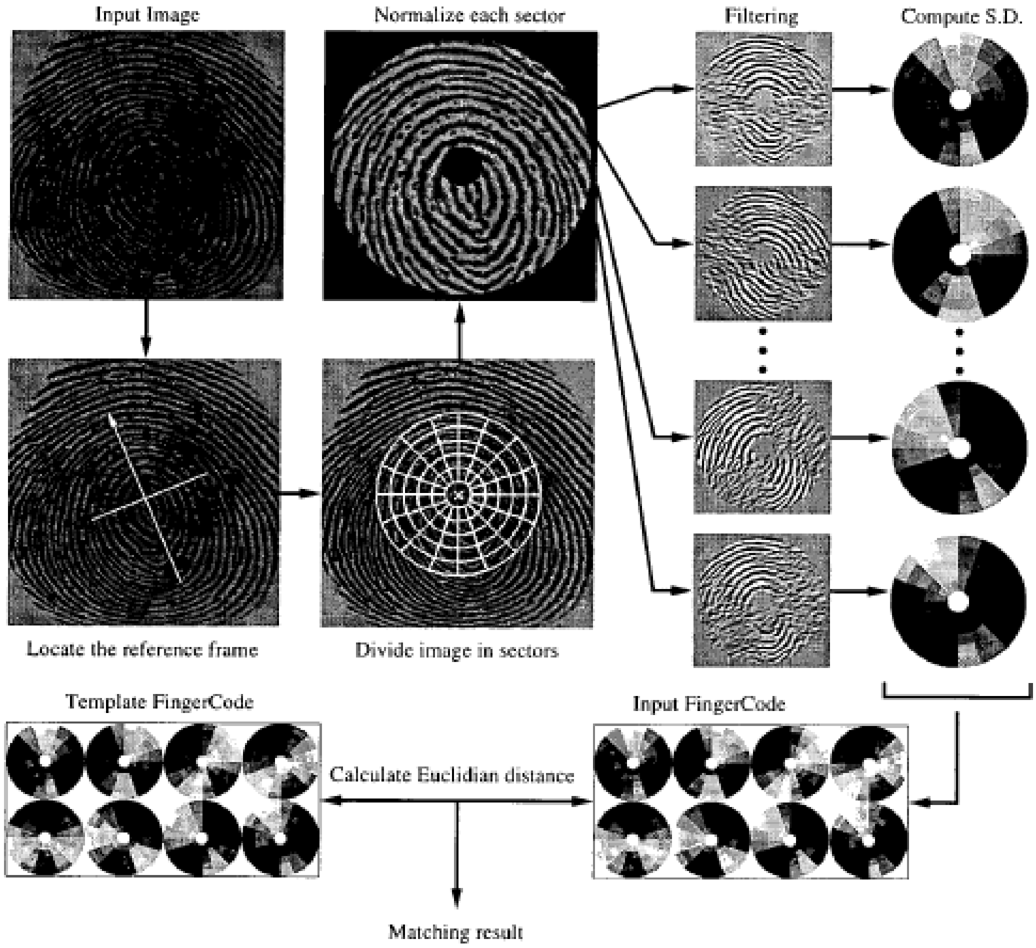


Figure 3.4: Flow diagram of the FingerCode verification system

Covering the entire image rather than only a limited region further to align the tesselation with the extracted minutiae instead od detecting an empirical core point, is far more accurate. The dimension of the feature vector is substantially increased (1352 dimensions for a 13x13 tesselation with 8 Gabor-filtered images). The matching code of the *FingerCodes* (their euclidean distance) and of the minutiae-based are merged into a single matching score. In the previous implementation, the Gabor filtering was the most CPU-expensive step of the process is now fairly reduced by the new computation scheme. Experiments claim better results with this hybrid matching method. However, the method could be improved by first performing a minutiae-based match and only continue the verification is the match score is below a given treshold, thus reducing the processing time.

Chapter 4

Conclusion

We have been through several research papers, presented the working of their approached methods and critically discussed their advantages and shortcomings. Regarding the methods efficiency, it must be noticed that *false rejection* and *false acceptance* rates given in the experimental results section have not been taken in consideration as a comparison because they are not to be considered relevant since experiments have not been performed on the same training and test sets.

Several other approaches to fingerprint matching have been proposed in the litterature such as *optical correlation* [?], [?] or *graph matching* [?]. It appears very clearly that a lot of research is being done to perform better fingerprint verification. However, there is a popular misconception that fingerprint recognition system is a fully solved problem since it has been around for a quite long period of time. In fact, fingerprint recognition remains a very challenging and important pattern recognition problem. Finally, with the increasing needs of identity verification for security purposes in today's world, the number of publications on fingerprint-related topics is obviously not expected to decrease soon.

Bibliography

[NSW Police] The Thin Blue Line, *The History of Fingerprinting.* New South Wales Australia, Unofficial Police Website. Retrieved March 23, 2004, from `http://www.policensw.com/info/fingerprints/finger01.html`

[Moore] G. Moore, *Brief History of Fingerprint Identification.* Fingerprint Consulting Service. Retrieved March 23, 2004, from `http://www.gregandsheila.com/consult/xx4.htm`

[LPE, 2003] Latent Print Examination, *The History of Fingerprints.* November, 2003. Retrieved March 23, 2004, from `http://onin.com/fp/fphistory.html`

[IBG, 2003] International Biometric Group, The Henry Classification System. New York, 2003. Retrieved March 23, 2004, from `http://www.biometricgroup.com/Henry\%20Fingerprint\%20Classification.pdf`

[Henry, 1900] E.R. Henry, "Classification and uses of finger prints", Routledge, London, 1900

[FBI, 1984] Federal Bureau of Investigation, *The Science of Fingerprints: Classification and Uses.* Washington, D.C.: US Government Printing Office, 1984.

[Lee, 1991] H.C. Lee and R.E. Gaensslen, eds., *Advances in Fingerprint Technology.* New York: Elsevier, 1991.

[Jain, 1997] A. Jain, L. Hong and R. Bolle. "On-line fingerprint verification". *IEEE Trans. Pattern Analysis and Machine Intell.*, 19(4):302-3013, Apr. 1997.

[Rao, 1990] A.R. Rao, *A taxonomy for Texture Description and Identification.* New Yor: Springer-Verlag, 1990

[Mallesware, 1976] T.C. Mallesware, “Feature Extraction for Fingerprinting Classification”, Pattern Recognition, Vol 8, 1976, pp 181-192

[Ansari, 1992] N. Ansari, M.H. Chen, and E.S.H. Hou, “A Genetic Algorithm for Point Pattern Matching”, Chapt. 13, B. Soucek and the IRIS Group, eds., *Dynamic, Genetic, and Chaotic Programming*. New York: John Wiley & Sons, 1992

[Ranade, 1993] A. Ranade and A. Rosenfeld, “Point Pattern Matching by Relaxation”, *Pattern Recognition*, vol. 12, no. 2, pp. 269-275, 1993

[Starink, 1995] J.P.P Starink and E. Backer, “Finding Point Correspondance Using Simulated Annealing”, *Pattern Recognition*, vol 28, no. 2, pp. 231-240, 1995

[Tico, 2003] M. Tico, P. Kuosmanen, “Fingerprint Matching Using an Orientation-Based Minutiae Descriptor”, *Trans. on Pattern Analysis and Machine Intelligence*, Vol 25, pp. 1009-1014

[Isenor, 1986] D. K. Isenor, S. G. Zaky, “Fingerprint Identification Using Graph Matching”, *Pattern Recognition*, Vol. 19, pp. 111-112, 1986

[Chen, 1991] Z. Chen, C. H. Kuo, “A Topology-Based Matching Algorithm for Fingerprint Authentication”, *Proc. of 25 Annual IEEE International Carnahan Conference on Security Technology*, pp. 84-87, 1991

[Huttenlocher, 1987] D.P. Huttenlocher und S. Ullman, “Object Recognition Using Alignment”, *Proc. First Int’l Conf. Computer Vision, pp. 102-111, London, 1987*

[Jain, 2000] A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti. Filter-bank based Fingerprint Matching. *IEEE Transactions on Image Processing*, 9(5):846-859, May 2000

[Ross, 2002] A. Ross, A. Jain, J. Reisman. A Hybrid Fingerprint Matcher. *Proc. of International Conference on Pattern Recognition, Quebec City, 2002*

[Wilson, 2000] C. Wilson, C. Watson and E. Paek, “Effect of Resolution and Image Quality on Combined Optical and Neural Network Fingerpritn Matching”, *Pattern Recognition*, Vol 33, no. 2, pp. 317-331, 2000

[Gamble, 1992] F. Gamble, L. Frye and D. Grieser, "Real Time Fingerprint Verification System", *Applied Optics*, Vol 31., no. 5, pp. 652-655, 1992

[Isenor, 1986] D.K. Isenor and S.G. Zaky, "Fingerprint Identification Using Graph Matching", *Patter Recognition*, Vol. 19, no. 2, pp. 113-122, 1986