# Matching Fingerprints

*Delft, July 2005*

*Written by:*
Ankie van der Zanden
Business Mathematics and Informatics
Free University, Amsterdam

*Supervised by:*
Dr. Elena Marchiori
Free University, Amsterdam

# Preface

This paper has been written as the last assignment of my study Business Mathematics and Informatics (BMI) at the Free University in Amsterdam. The objective is to investigate the available literature in reference to a topic that should cover at least two out of the following areas: business management, mathematics and computer science.

After my internship at the Police my interest for criminological research has grown. That is the main reason why I chose for the topic of matching fingerprints. It caught my attention by the way fingerprints are matched, namely through very small details like the ending or splitting of a line (ridge). Besides that, it is very special that each fingerprint is unique. There is no second person with exactly the same fingerprint.

For writing this work I would like to express my appreciation and gratitude to two persons. First I would like to thank dr. Elena Marchiori for supervising this research. Second, my boyfriend Erdoğan Taşkesen who explained me the secrets of image processing.

Ankie van der Zanden

Delft, July 2005

# Table of Contents

# 1. Introduction

A fingerprint is believed to be unique to each person (and each finger). Even the fingerprints of identical twins are different. The pattern is quite stable trough out our lifetime, in case of a cut, the same pattern will grow back. The features of a fingerprint depend on the nerve growth in the skin's surface. This growth is determined by genetic factors and environmental factors such as nutrients, oxygen levels and blood flow which are unique for every individual. [19] Fingerprints are one of the most full-grown biometric technologies and are used as proofs of evidence in courts of law all over the world. Fingerprints are, therefore, used in forensic divisions worldwide for criminal investigation.

The performance of matching techniques relies partly on the quality of the input fingerprint. In practice, due to skin conditions, sensor noise, incorrect finger pressure and bad quality fingers like from elderly people and manual workers, a significant percentage of fingerprint images is of poor quality. This makes it quite difficult to compare fingerprints.

The central question that will be answered in this report, is:

*"How are fingerprints compared to each other?"*

Based on this, the following questions can be set up:
- Which steps are to be taken before the actual matching can take place?
- Which techniques are there to match fingerprints and which one is most used in practice?
- How does this most used matching technique work?

In chapter 2 the whole process of fingerprint matching is globally described. It contains five steps, all further explained in the succeeding chapters. Chapter 3 contains the first step of that process: preprocessing the image. A few enhancement techniques are discussed. After that, there is a classification step, described in chapter 4. In chapter 5 the actual matching is explained. The method specified is minutiae-based matching, a method that is well known and widely used. Finally the conclusions are to find in chapter 6.

# 2. The process of matching

There are multiple approaches found in literature to match fingerprint images. The one that is well known and used often is called minutiae-based matching. In this chapter is first described the process of fingerprint matching, in case of minutiae-based matching is used.

Fingerprints are unique, there are no two individuals that have exactly the same pattern. In principle, every finger is suitable to give prints for authentication purposes. However, there are differences between the ten fingers. There is no clear evidence as to which specific finger should be used for identification. The thumb provides a bigger surface area but there is not much association of the thumb with criminality. Forefingers have been typically used in civilian applications. In most cases one can assume that the index finger obtains the best performance. Since the majority of the people is right handed, the best choice would be to take the right hand index finger. [25, 26]

After capturing the fingerprint, for example in crime, it is compared to other fingerprints in the database to find a matching pair. Nowadays this whole process goes automatically via identification marks. A fingerprint has various identification marks. On the global level there are the ridges that make a particular pattern. Moreover there are singular points to detect, like the delta and core. At the local level you find minutiae details. The two most occurring are a ridge ending and the bifurcation (splitting of ridges). At the very fine level there are sweat pores. These can only be used at images from very high quality and are not discussed in this paper.

The comparing of two prints can only take place if the fingerprints are of reasonable quality and have values that are measured the same way and mean the same thing. A few preparations must precede before the actual matching takes place. These steps are called preprocessing. Preprocessing improves the quality of the fingerprint (mostly a digital image) and removes scars and other noise in the image. It also makes the image better readable for the matching step by making the image black and white for instance.

The matching of fingerprints has been studied a lot, resulting in multiple approaches. One of these approaches is minutiae matching. It is the most well known and often used method that makes use of small details in the fingerprint, called minutiae, as ridge endings and split points. Each minutia has its own information, like the angle and position. Extracting this information is one of the steps of the minutiae matching approach, matching these extracted minutiae is another problem. This matching of minutiae can be seen as a point matching problem. A suitable algorithm to solve this problem is the Hough transform-based algorithm. It calculates the optimal transformation for matching minutiae. If there exists a matching fingerprint in the database, the template with the most matching minutiae is probably the same as the input.

Between the extraction of minutiae points and the matching, there is often a post-processing step to filter out false minutiae, caused by scars, sweat, dirt or even by the preprocessing step. In the end, using the minutiae can lead to a unique match of fingerprints.

The procedure described above can be extended with one important time saving step, namely classification. Classification is used in cases where the database containing fingerprints is so large that matching a fingerprint with all the images is too time-consuming. Fingerprints can be categorized by their patterns. The classification is based on the following patterns: loops, whorls and arches. In this way an input fingerprint does not have to be matched with all the fingerprints in the database, but only a part of it.

The steps to take for the whole process of matching an input fingerprint with one of the templates from the database, is depicted in figure 2-1. The assumption is made that the templates in the database have already gone through this process and so only the input fingerprint has to be prepared for matching.

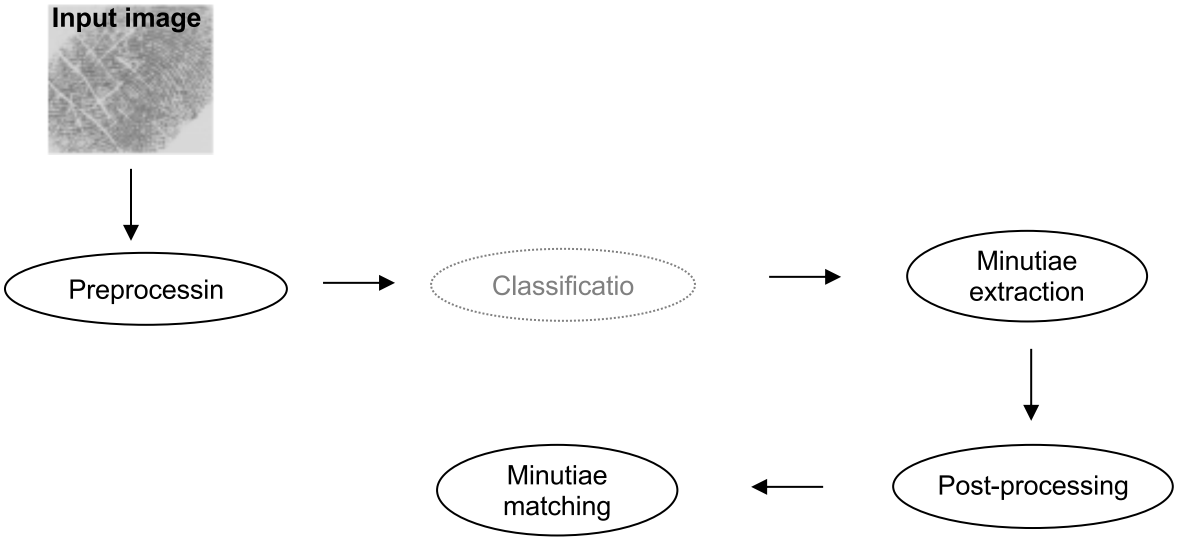



*Figure 2-1: The process of matching an input fingerprint with a template from the database*

# 3. Preprocessing

When a fingerprint image is captured, nowadays through a scan, it contains a lot of redundant information. Problems with scars, too dry or too moist fingers, or incorrect pressure must also be overcome to get an acceptable image. Therefore, preprocessing, consisting of enhancement and segmentation is applied to the image.

It is widely acknowledged that at least two to five percent of target population has fingerprints of poor quality. These fingerprints that cannot be reliably processed using automatic image processing methods. This fraction is even higher when the target population consists of older people, people doing manual work, people living in dry weather conditions or having skin problems, and people who have poor fingerprints due to their genetic and racial attributes. [13]

A fingerprint can contain regions of different quality:

- *a well-defined region*, where ridges are clearly differentiated from each another;
- *a recoverable region*, where ridges are corrupted by a small amount of gaps, creases and smudges, but they are still visible and the neighboring regions provide sufficient information about their true structure;
- *an unrecoverable region*, where ridges are corrupted by such a severe amount of noise and distortion that no ridges are visible and the neighboring regions do not allow them to be reconstructed. [17]

## *3.1 Steps*

A critical step in automatic fingerprint matching is to automatically and reliably extract minutiae from the input fingerprint images. However, the performance of a minutiae extraction algorithm relies heavily on the quality of the input fingerprint images. To ensure that the performance of an automatic fingerprint identification system will be robust with respect to the quality of the fingerprint images, it is essential to implement a fingerprint enhancement algorithm in the minutiae extraction module. [28]

In the literature there are several methods to improve the quality of an image and make it ready for matching details. The steps that are present in almost every process are:

1) normalization,
2) filtering,
3) binarization,
4) skeletonization.

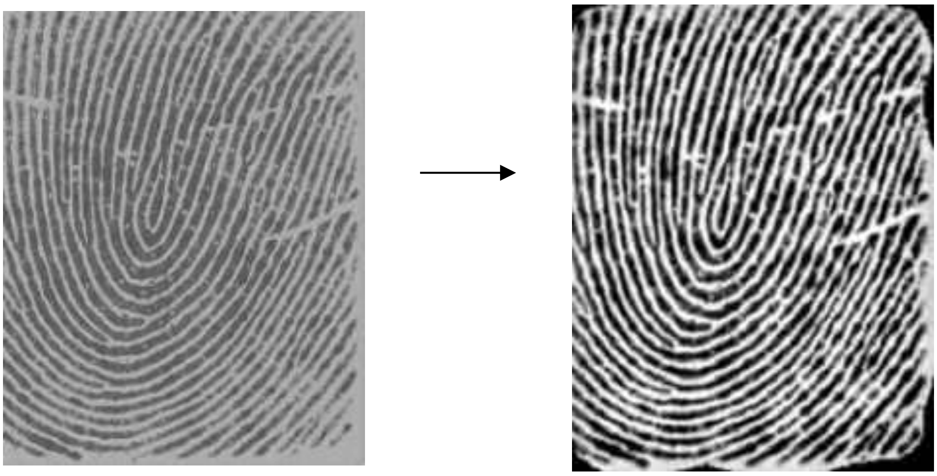
In the first step the input image from the sensor is normalized. This is important since image parameters may differ significantly with varying sensors, fingers and finger conditions. By normalizing an image, the colors of the image are spread evenly throughout the gray scale. The filtering step is the one that changes the most. There are a lot of filters to smooth the ridges, take away scars, noise and irrelevant segments. Low pass filters are used, just as Gaussian masks, Gabor filters or orientation filters. The third step is making the image binary; transform the gray scale image into a binary image (black and white). The ridges are then made thinner from five to eight pixels in width down to one pixel, for precise location of endings and bifurcations.

### 3.1.1 Normalization

Normalization is a good first step for improving image quality. To normalize an image is to spread the gray scale in a way that it is spread evenly and fill all available values instead of just a part of the available gray scale, see figure 3-1. The normal way to plot the distribution of pixels with a certain amount of gray (the intensity) is via a histogram. To be able to normalize an image, the area which is to normalize within, has to be known. Thus it is necessary to find the highest and the lowest pixel value of the current image. Every pixel is then evenly spread out along this scale. Equation (1) represents the normalization process.

$$I_{norm}(x,y) = \frac{I(x,y) - I_{\min}}{I_{\max} - I_{\min}} \times M \,, \tag{1}$$

where $I$ is the intensity (gray level) of the image. $I_{min}$ is the lowest pixel value found in the image, $I_{max}$ is the highest one found. $M$ represents the new maximum value of the scale, mostly $M = 255$, resulting in 256 different gray levels, including black (0) and white (255). $I_{norm}(x, y)$ is the normalized value of the pixel with coordinates $x$ and $y$ in the original image $I(x,y)$. When images have been normalized it is much easier to compare and determine quality since the spread now has the same scale. Without the normalization it would not be possible to use a global method for comparing quality. [2, 8]



*Raw image from sensor* *Normalized image*

*Figure 3-1: The normalization step*

## 3.1.2 Filtering

It is important to filter out image noise coming from finger consistency and sensor noise. For that purpose the orientation of the ridges can be determined so that it is able to filter the image exactly in the direction of the ridges. In figure 3-2 an orientation field overlayed on a fingerprint is shown.

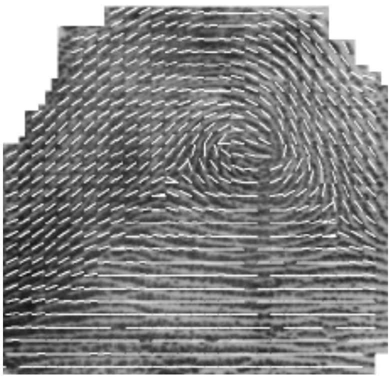
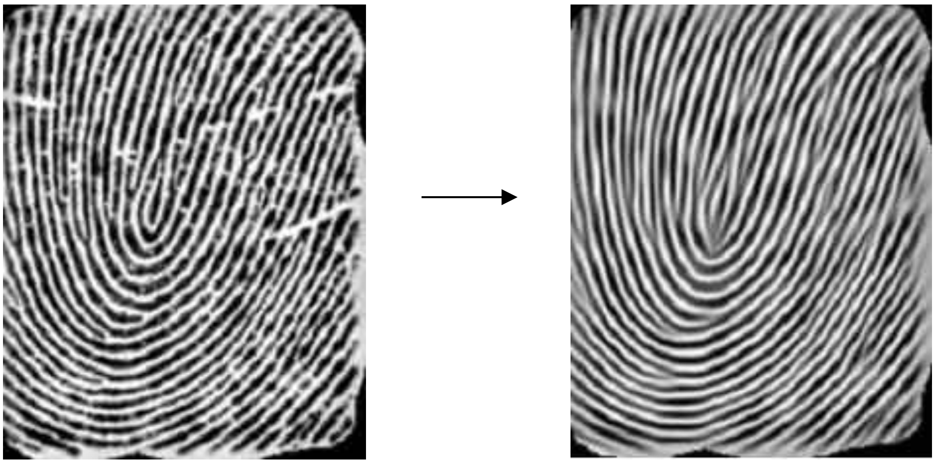


*Figure 3-2: An orientation field overlayed on a fingerprint*

By this filter method the ridge noise is greatly reduced without affecting the ridge structure itself, see figure 3-3. One approach to ridge orientation estimation relies on the local image gradient. A gray scale gradient is a vector whose orientation indicates the direction of the steepest change in the gray values and whose magnitude depends upon the amount of change of the gray values in the direction of the gradient. The local orientation in a block can be determined from the pixel gradient orientations of the block. [3, 13, 27]

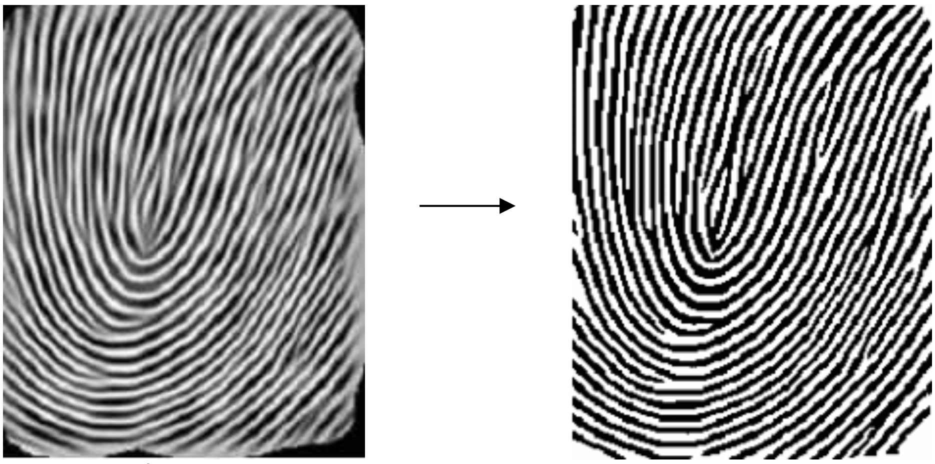


*Normalized image* *Directionally filtered image*

*Figure 3-3: The filtering step (in this case an orientation field filter)*

### 3.1.3 Binarization

Binarization can be seen as the separation of the object and background. It turns a gray scale picture into a binary picture. A binary picture has only two different values. The values 0 and 1 are represented by the colors black and white, respectively. Refer to figure 3-4 for a binarized image. To perform binarization on an image, a threshold value in the gray scale image is picked. Everything darker (lower in value) than this threshold value is converted to black and everything lighter (higher in value) is converted to white. This process is performed to facilitate finding identification marks in the fingerprints such as singularity points or minutiae (see chapter 4 and 5).
The difficulty with binarization lies in finding the right threshold value to be able to remove unimportant information and enhance the important one. It is impossible to find a working global threshold value that can be used on every image. The variations can be too large in these types of fingerprint images that the background in one image can be darker than the print in another image. Therefore, algorithms to find the optimal value must be applied separate on each image to get a functional binarization. There are a number of algorithms to perform this, the most simple one uses the mean value or the median of the pixel values in the image. This algorithm is based on global thresholds.
What often are used nowadays are local thresholds. The image is separated into smaller parts and threshold values are then calculated for each of these parts. This enables adjustments that are not possible with global calculations. Local thresholds demand a lot more calculations but mostly compensate it with a better result. [2, 8, 27]



*Directionally filtered image* *Binarized image*

*Figure 3-4: The binarization step*

### 3.1.4 Skeleton modeling

One way to make a skeleton is with thinning algorithms. The technique takes a binary image of a fingerprint and makes the ridges that appear in the print just one pixel wide without changing the overall pattern and leaving gaps in the ridges creating a sort of “skeleton” of the image. See figure 3-5 for an example of skeletonization. The form is used as structural element, consisting of five blocks that each present a pixel. The pixel in the center of that element is called the origin. When the structural element overlays the object pixels in its entirety, only the pixels of the origin remain. The others are deleted.

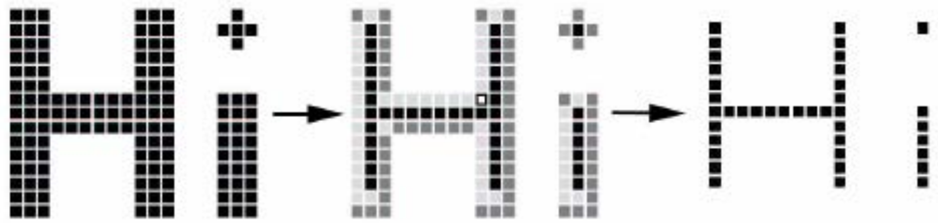
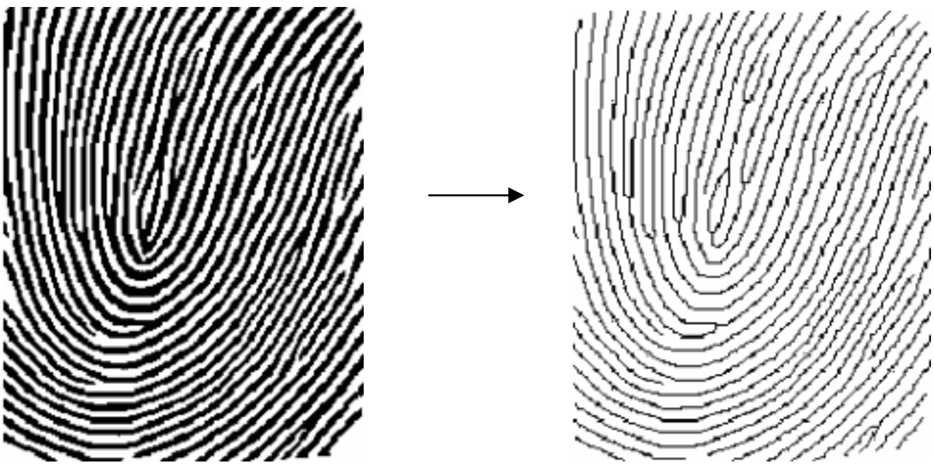


*Figure 3-5: An example of skeletonization*

Skeleton modeling makes it easier to find minutiae and removes a lot of redundant data, which would have resulted in longer process time and sometimes different results. There are a lot of different algorithms for skeleton modeling that differ slightly. The result of a skeletonized (or thinned) fingerprint is shown in figure 3-6. [2, 8, 27]



*Binarized image* *Skeletonized image*

*Figure 3-6: The skeletonizing step*

## *3.2 Discussion*

Binarization reduces the information in the image from gray scale to black and white. While this simplifies for further algorithms to decode the fingerprint image into information useful for identification, it also reduces the complexity of the image and removes information that might have been necessary for the identification of the fingerprint. Some authors have proposed minutiae extraction approaches that work directly on the gray-scale images without binarization and skeleton modeling. This choice is motivated by these considerations [17]:

- a significant amount of information may be lost during the binarization process;
- binarization and thinning are time consuming;
- thinning may introduce a large number of spurious minutiae;
- in the absence of an a priori enhancement step, most of the binarization techniques do not provide satisfactory results when applied to low-quality images.

A reduction of information is not necessarily a negative thing though.
Still the binarization is a necessary step for many of the algorithms used for minutiae analysis. Advanced algorithms like skeletonization will only work if this process is performed.

The thinning performed in skeleton modeling enables point identification via simple counting of the nearby pixels. When this process has been performed, a mapping of available minutiae in the image is made, used for minutiae-based matching. It is now comparable to earlier stored templates. However, if pattern recognition is used instead, it is not necessary to perform binarization (or even normalization). Instead every pixel direction is calculated and a direction field of the total image is created. [2]

# 4. Classification

The fingerprints have been traditionally classified into categories based on information in the global pattern of ridges. A recognition procedure consists in retrieving one or more fingerprints in a large database corresponding to a given fingerprint, whereas a classification procedure consists in assigning a fingerprint to a pre-defined class. In this chapter it is described how this works.

## *4.1 Why classification*

Fingerprint recognition is the basic task of the identification systems of the most famous policy agencies. If all the fingerprints within the database are a priori classified, the recognition procedure can be performed more efficiently, since the given fingerprint has to be compared only with the database items belonging to the same class.
The first (semi-)automatic systems for the fingerprint recognition were developed in the ‘70s by the US Federal Bureau of Investigation (FBI) in collaboration with the National Bureau of Standards, Cornel1 Aeronautics Laboratory and Rockwell International Corporation. Since then the volumes of the fingerprint databases and the amount of requests of identification increased constantly, so that it was necessary very soon to classify the fingerprints to improve the recognition efficiency. The FBI now has about forty to fifty million fingerprints stored in their database.

Although the first approaches to automatic fingerprint classification were proposed many years ago [9, 10], policy agencies still go on performing classification manually with a great expense of time. On the other hand, the problem of fingerprint classification is very hard: a good classification system should be very reliable (it must not misclassify fingerprints), it should be selective (the fingerprint database has to be partitioned in a number of non-overlapping classes with about the same properties) and it should be efficient (each fingerprint must be processed in a short time). [16]

## *4.2 Patterns*

A fingerprint can be looked at from different levels; the global level, the local level, and the very fine level. At the global level, you find the singularity points, called core and delta points, see figure 4-1. These singularity points are very important for fingerprint classification, but they are not sufficient for accurate matching.

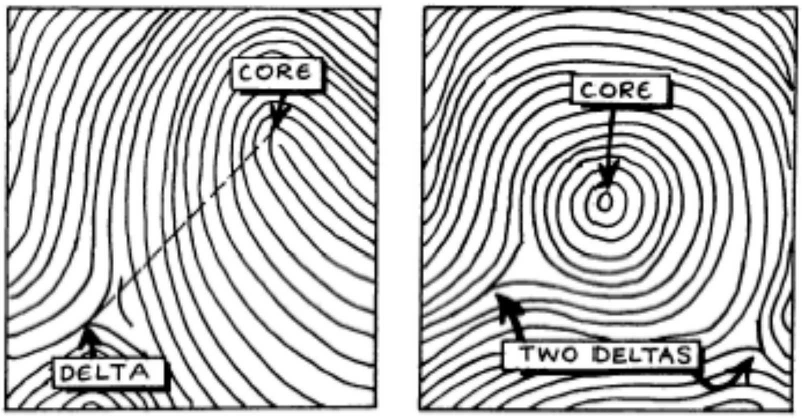



*Figure 4-1: Core and delta points marked on sketches of the two fingerprint patterns loop and whorl*

The core is the inner point, normally in the middle of the print, around which whorls, loops, or arches center. It is frequently characterized by a ridge ending and several curved ridges. Deltas are the points, normally at the lower left and right hand of the fingerprint, around which a triangular series of ridges center. [28]

Fingerprint classification nowadays is usually based on the Galton-Henry classification scheme. Galton divided the fingerprints into three major classes (arches, loops and whorls) and further divided each category into subcategories [7]. Edward Henry refined Galton's classification by increasing the number of classes [11]. The five most common classes of the Galton-Henry classification scheme are: plain arches, tented arches, left loop, right loop, and whorl. Of all fingerprint patterns 65% consist of loops, whorls make up about 30%, and arches the remaining 5%.

## Loops

The loop is the most common fingerprint pattern. They are usually separated into right loops and left loops. The difference between these two is the direction that the ridges turn to. If the ridges turn to the left it is a left loop and vice versa. Figure 4-2 shows a right loop.

In a loop pattern, one or more of the ridges enter on either side of the fingerprint, recurve, touch or cross the line running from the delta to the core and terminate or tend to terminate on or toward the same side of the fingerprint from which such ridge or ridges entered. There is one delta. [2, 23]

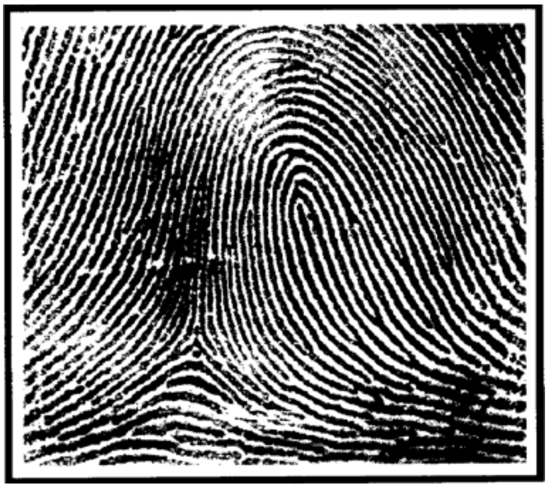


*Figure 4-2: A right loop*

## Whorls

*Figure 4-3: A whorl*

Whorls are the second most common pattern. Here the ridges form circular patterns around the core. Most often they form to spirals, but they can also appear as concentric circles.

In a whorl some of the ridges make a turn through at least one circuit. There are two loops (or a whorl) and two deltas, see the red arrows in figure 4-3. [2, 23]

## Arches

Arches are more uncommon than loops and whorls. In the arch type the ridges run from one side to the other, making no backward turn. Usually arches are classified into plain (simple) and tented (narrow) arches, see figure 4-3. A plain arch does not have loops or deltas. The tented arch often has a loop and a delta point below. [2, 23]

*Figure 4-4:*
*A plain arch*

*A tented arch*

## 4.3 Classification techniques

It is important to note that the distribution of fingers into the five classes is highly skewed. A fingerprint classification system should take that into account and should be invariant to rotation, translation and elastic distortion of the skin.

Although a wide variety of classification algorithms has been developed for this problem, a relatively small number of features extracted from fingerprint images have been used by most of the authors. In particular, almost all the methods are based on one or more of the following features: ridge line flow, orientation image, singular points and Gabor filter responses. Ridge line flow is usually represented as a set of curves running parallel to the ridge lines. An orientation image is used in most classification approaches because it contains all the information required for the classification. Usually the orientation image is registered with respect to the core point, one of the singular points, explained in paragraph 4.2. A Gabor filter is a common directional filter which has both frequency-selective and orientation-selective properties.

Several approaches have been developed for automatic fingerprint classification. The best approaches can be broadly categorized into the following categories [17]:

- *Rule-based*
  These approaches mainly depend on the number and position of the singular points of the fingerprint. This is the approach commonly used by human experts for manual classification. A plain arch has no singular points. A tented arch, left loop and right loop have one loop and one delta. A whorl has two loops (or a whorl) and two deltas. The result of this technique is a scheme to follow, which tells in which class the input image belongs to. [12, 13, 15]
- *Structural*
  Structural approaches are based on the relational organization in the structure of the fingerprints. They are often based on the orientation field. Cappelli et al [4] proposed partitioning the orientation image into homogeneous regions and subsequently classify the image based on the pattern of the lines between the regions. See also Maio and Maltoni [16].
- *Statistical*
  In statistical approaches, a fixed-size numerical feature vector is derived from each fingerprint and a general-purpose statistical classifier is used for the classification. One of the most widely adopted statistical classifiers is the *k*-nearest neighbor. Many approaches directly use the orientation image as a feature vector. [14]
- *Neural network-based*
  Most of the proposed neural network approaches are based on multilayer perceptrons and use the elements of the orientation image as input features. The success of neural networks was mainly in the 1990s. [14]
- *Multi-classifier*
  Different classifiers offer complementary information about the patterns to be classified, which may improve performance. This approach is especially studied in recent years. For example Jain,

Prabhakar and Hong [14] adopt a two-stage classification strategy: a *k*-nearest neighbor classifier is used to find the two most likely classes. Then a specific neural network, trained to distinguish between the two classes, is exploited to obtain the final decision.

The techniques have an error rate between 6.5 and 11.9% when fingerprints are assigned to five different classes. [17] Many errors are due to the misclassification of tented arch fingerprints as plain arch. That is the reason why some authors only use four classes; tented arch and plain arch as one. The classification errors drop to between 5.1 and 9.4%.

# 5. Fingerprint matching

This chapter describes how an input fingerprint is compared with one of the template fingerprints, stored in a database. In the first paragraph some methods for fingerprint matching are described. Subsequently, the most well known method of these, minutiae-based matching, is described in detail.

## *5.1 Methods*

Many algorithms have been developed to match two different fingerprints and they can be divided into the following groups: [17, 28]

*Correlation-based matching*: Two fingerprint images are laid on top of each other and the correlation between corresponding pixels is computed for different alignments (various displacements and rotations). This technique has some disadvantages. Correlation-based techniques require the precise location of a registration point and are affected by non-linear distortion.

*Minutiae-based matching*: As described in section 5.2, a fingerprint pattern is full of minutiae points, which characterize the fingerprint. In minutiae-based matching, these points are extracted from the image, stored as sets of points in the two-dimensional plane, and then compared with the same points extracted from the template. With this technique, it is difficult to extract the minutiae points accurately when the fingerprint is of low quality.

*Ridge feature-based matching*: This matching can be conceived as a superfamily of minutiae-based matching and correlation-based matching, since the pixel intensity and the minutiae positions are themselves features of the finger ridge pattern. The matching method uses features like local orientation and frequency, ridge shape, and structure information. Even though minutiae-based matching is considered more reliable, there are cases where ridge feature-based matching is better to use. In very low-quality fingerprint images, it can be difficult to extract the minutiae points, and using the ridge pattern for matching is then preferred.

Minutiae-based matching is certainly the most well known and widely used method for fingerprint matching, thanks to its strict analogy with the way forensic experts compare fingerprints and its acceptance as a proof of identity in the courts of law in almost all countries. For this reason specific attention is paid to this method.

## *5.2 Minutiae extraction*

At the local level, a total of 150 different local ridge characteristics, called minutiae details, have been identified. Most of them depend heavily on the impression conditions and quality of fingerprints and are rarely observed in fingerprints. The seven most prominent ridge characteristics are shown in figure 5.1. [17]

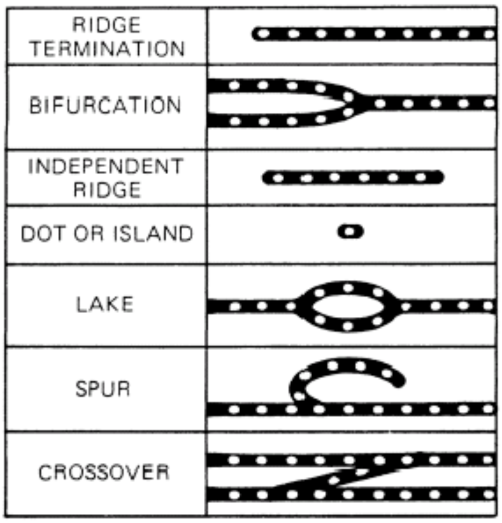


*Figure 5.1: Minutiae details*

The measured fingerprint area consists in average of about thirty to sixty minutiae points depending on the finger and on the sensor area. These can be extracted from the image after the image processing step (and possibly the classification step) is performed. The point at which a ridge ends, and the point where a bifurcation begins, are the most rudimentary minutiae, and are used in most applications. Once the thinned ridge map is available, the ridge pixels with three ridge pixel neighbors are identified as ridge bifurcations, and those with one ridge pixel neighbor identified as ridge endings. However, all the minutiae detected are no facts yet because of image processing and the noise in the fingerprint image. For each extracted minutia a couple of features are stored: the absolute position (*x,y*), the direction ($\theta$), and if necessary the scale (*s*). [3]

The location of the minutiae are commonly indicated by the distance from the core, with the core serving as the (0,0) on an *x,y*-axis. Some authors use the far left and bottom boundaries of the image as the axes, correcting for misplacement by locating and adjusting from the core. In addition to the placement of the minutia, the angle of the minutia is normally used. When a ridge ends, its direction at the point of termination establishes the angle. This angle is taken from a horizontal line extending rightward from the core.

At the very fine level, intra-ridge details can be detected. These are essentially the finger sweat pores whose position and shape are considered highly distinctive. However, extracting pores is usable only in high-resolution fingerprint images of good quality and therefore this kind of representation is not practical for most applications. [17]

## *5.3 Post-processing*

Minutiae localization begins with a preprocessed image. At this point, even a very precise image will have distortions and false minutiae that need to be filtered out. For example, an algorithm may search the image and eliminate one of two adjacent minutiae, since minutiae are very rarely adjacent. Irregularities caused by scars, sweat or dirt appear as false minutiae, and algorithms locate any points or patterns that do not make sense, such as a spur on an island (probably false) or a ridge crossing at right angles to two or three others (probably a scar or dirt). A large percentage of false minutiae are discarded in this post-processing stage. [28]

In figure 5.2 several examples of false minutiae can be observed. In clockwise order: interrupted ridges, forks, spurs, structure ladders, triangles and bridges are depicted in the figure. The interrupted ridges are two very close lines with the same direction. Two lines connected by a noisy line compose a fork. The spurs are short lines whose direction is orthogonal to ridges direction. The structure ladders are pseudo-rectangle between two ridges. The triangles are formed by a real bifurcation with a noisy line between two ridges. Finally, the bridge is a noisy line between two ridges. All these characteristics generate several false minutiae. The algorithm is divided into several steps, executed in a pre-arranged order: elimination of the spurs, union of the endpoints, elimination of the bridges, elimination of the triangles, elimination of the structure ladders. [6, 24]

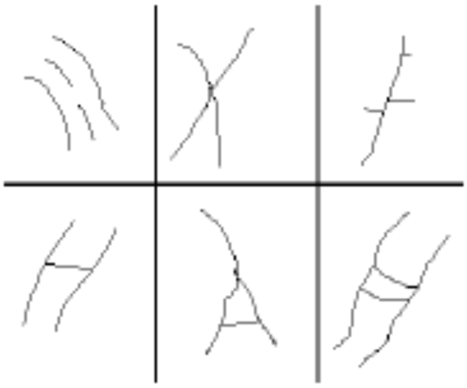


*Figure 5-2: False minutiae: interrupted ridges, forks, spurs, structure ladders, triangles and bridges, in clockwise order*

## 5.4 Minutiae matching

Minutiae-based techniques first find minutiae points and then map their relative placement on the finger in order to match the minutiae with the template fingerprint minutiae. A general approach for matching minutiae is described in paragraph 5.3.1 and an algorithm for transformation-based minutiae matching is subsequently described in paragraph 5.3.2. The minutiae matching is mainly based on the book by D. Maltoni, D. Maio, A. K. Jain and S. Prabhakar [17].

### 5.4.1 General approach

Let T and I be the representation of the template and input fingerprint, respectively. This representation is a feature vector whose elements are the fingerprint minutiae:

T = $\{m_1, m_2, \ldots, m_m\}$
I = $\{m'_1, m'_2, \ldots, m'_n\}$,

where $m$ and $n$ denote the number of minutiae in T and I, respectively.

Each minutia may be described by a number of attributes, including its location in the fingerprint image, orientation, type (e.g. termination or bifurcation), a weight based on the quality of the fingerprint image in the neighborhood of the minutia, and so on. Most common minutiae matching algorithms consider each minutia as a triplet $m = \{x, y, \theta\}$ that indicates the coordinates ($x$, $y$) of the absolute location of the minutia and the minutia angle $\theta$:

$m_i = \{x_i, y_i, \theta_i\}$ $\quad i = 1..m$
$m'_j = \{x'_j, y'_j, \theta'_j\}$ $\quad j = 1..n$

A minutia $m'_j$ in I and a minutia $m_i$ in T are considered matching, if the spatial distance between them is smaller than a given tolerance $r_0$ and the direction difference between them is smaller than an angular tolerance $\theta_0$:

spatial_distance($m'_j$, $m_i$) = $\sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \le r_0$ (1)

and

direction_distance($m'_j$, $m_i$) = $\min\left(\left|\theta'_j - \theta_i\right|, 360° - \left|\theta'_j - \theta_i\right|\right) \le \theta_0$. (2)

This last equation takes the minimum of the two because of the circularity of angles (the difference between angles of $2°$ and $358°$ is only $4°$). The tolerances $r_0$ and $\theta_0$ are necessary to compensate for the unavoidable errors made by feature extraction algorithms.

In order to match the fingerprints, there has to be done a displacement and rotation and possibly some other geographical transformations as well. When the fingerprints are from two different scanners, the resolution may vary. So the scale has to be considered. Also the prints can be damaged or affected by distortions. There has to be a mapping function to deal with these problems. See figure 5-3 for the transformation of a minutia point in two fingerprints. [21]

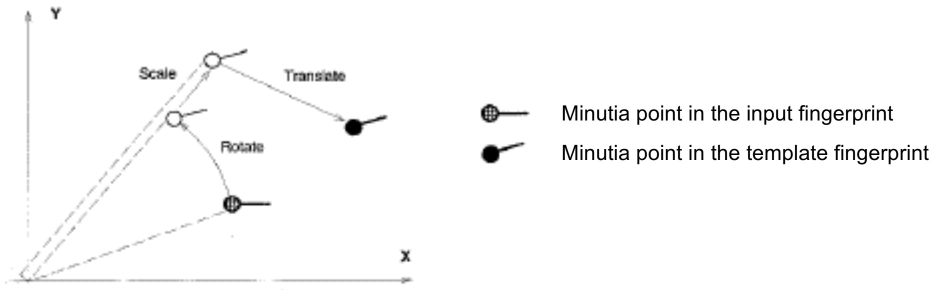



*Figure 5-3: Applying a transformation to a minutia point*

In the absence of noise and other deformation, the rotation and displacement between two images can be completely determined using two corresponding point pairs. In the ideal scenario, the true alignment can be estimated by testing all possible pairs of minutiae for correspondence and then selecting the best correspondence. [18]

Let map() be the function that maps a minutia $m'_j$ (from I) into $m''_j$ according to a given geometrical transformation. For example, by considering a displacement of $[\Delta x, \Delta y]$, and a counterclockwise rotation $\theta$ around the origin:

$$\mathrm{map}_{\Delta x,\Delta y,\theta}(m'_j = \{x'_j, y'_j, \theta'_j\}) = m''_j = \{x''_j, y''_j, \theta'_j + \theta\}, \text{ where}$$

$$\begin{bmatrix} x''_j \\ y''_j \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}. \tag{3}$$

A pair of fingerprints that are most alike will have the maximum number of matching minutiae. Let mm() be an indicator function that returns 1 in the case where the minutiae $m''_j$ and $m_i$ match according to the spatial distance and direction distance:

$$\mathrm{mm}(m''_j, m_i) = \begin{cases} 1 & \text{spatial\_distance}(m''_j, m_i) \le r_0 \quad \text{and} \quad \text{direction\_distance}(m''_j, m_i) \le \theta_0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Then the problem can be formulated as:

$$\max_{\Delta x,\Delta y,\theta,P} \sum_{i=1}^{m} mm(map_{\Delta x,\Delta y,\theta}(m'_{P(i)}), m_i), \tag{5}$$

which indicates the maximum number of matched minutiae. The function P(i) determines the paring between I and T minutiae. In particular, each minutia has either exactly one mate in the other fingerprint or has no mate at all. See figure 5-4 for an example of pairing. If $m_1$ were mated with $m''_2$ (the closest minutia), $m_2$ would remain unmated. However, pairing $m_1$ with $m''_1$, allows $m_2$ to be mated with $m''_2$, thus maximizing equation (5).

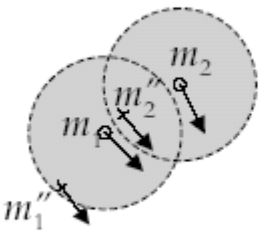



*Figure 5-4: An example of pairing minutiae*

The point-matching problem is studied a lot, inducing many approaches. The Hough transform-based approach [1, 21] is one that is quite popular and will be described in the next paragraph.

## 5.4.2 The Hough transform-based algorithm

Many transformations for minutiae matching are based on the Hough transform-based approach. It is an algorithm with embedded fingerprint alignment in the minutiae matching stage, as proposed in Ratha 1996 [21]. It discretizes the set of all allowed transformations, and for each transformation, the matching score is computed. The transformation with the maximal score is believed to be the correct one. It consists of three major steps:

1) Estimate the transformation parameters $\Delta x$, $\Delta y$, $\theta$, and $s$ between the two representations, where $\Delta x$ and $\Delta y$ are translations along $x$- and $y$-directions, respectively, $\theta$ is the rotation angle, and $s$ is the scaling factor.
2) Align the two sets of minutiae points with the estimated parameters and count the matched pairs within a bounding box.
3) Repeat the previous two steps for the set of discretized allowed transformations. The transformation that results in the highest matching score is believed to be the correct one.

The space of transformations consists of quadruples ($\Delta x$, $\Delta y$, $\theta$, $s$), where each parameter is discretized (denoted by the symbol $^+$) into a finite set of values:

$\Delta x^+ \in \{\Delta x^+_1, \Delta x^+_2, \ldots, \Delta x^+_a\}$ $\quad$ $\theta^+ \in \{\theta^+_1, \theta^+_2, \ldots, \theta^+_c\}$

$\Delta y^+ \in \{\Delta y^+_1, \Delta y^+_2, \ldots, \Delta y^+_b\}$ $\quad$ $s^+ \in \{s^+_1, s^+_2, \ldots, s^+_d\}$

A four-dimensional array A, with one entry for each of the parameter discretizations, is initially reset and the following algorithm is executed:

```
For each mi, i = 1..m                          //for each template minutia point
For each m'j, j = 1..n                         //for each input minutia point
For each θ+ ∈ {θ+1, θ+2, …, θ+c}               //for each discretized rotation
  If direction_distance(θ'j + θ+, θi) < θ0        //if the directions difference
                                                  //after rotation is small
    For each s+ ∈ {s+1, s+2, …, s+d}           //for each discretized scale
    {
```

$$\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix} = \begin{bmatrix}x_i\\ y_i\end{bmatrix} - s^+\begin{bmatrix}\cos\theta^+ & -\sin\theta^+\\ \sin\theta^+ & \cos\theta^+\end{bmatrix}\begin{bmatrix}x'_j\\ y'_j\end{bmatrix} \qquad \text{//compute displacement}$$

```
      Δx+, Δy+ = quantization of Δx, Δy to the nearest bin
      A[Δx+, Δy+, θ+, s+] = A[Δx+, Δy+, θ+, s+] + 1
                                               //count matched pairs
    }
}}}
(Δx*, Δy*, θ*, s*) = arg max A[Δx+, Δy+, θ+, s+]
//where the count of matched pairs is highest, give the optimal displacement, rotation and
scale
```

This maximum gives the transformation that is believed to be the right one. If there exists a matching fingerprint in the database, the template with the most matching minutiae is probably the same as the input. [17, 20, 21]

## 5.4.3 Pre-alignment

An intuitive more logical choice would be to pre-align all the templates in the database and the input image before the matching procedure. In this way the alignment takes place only once for every image. A great advantage is that it significantly reduces the computational time. Pre-alignment cannot compare images to one another so only depends on the properties of itself. The most common pre-alignment technique convert the fingerprint according to the position of the core point. Unfortunately, reliable detection of the core is very difficult in noisy images and in arch type patterns. For adjusting the rotation the shape of the silhouette, the orientation of the core delta segment, the average orientation in some regions around the core, and the orientations of the singularities can be used. But still this is a complex problem and causes often errors in the matching procedure. That is the main reason why embedded alignment in the minutiae matching stage is often used. [17]

# 6. Conclusion

The process described in this paper and shown in figure 2-1 (page 5) consists of the main steps that a automatic fingerprint identification system should take when performing minutiae-based matching. However, some experts repeat steps or speed the process up by carrying out multiple steps at a time. The preprocessing and classification step can partly be executed at the same time for instance. The reason for this is that some preprocessing steps are necessary for the classification and some are necessary for minutiae extraction. As said before, weather to carry out the classification step depends on de size of the database and the time in which the identification has to be fulfilled.

Almost each author has his own preprocessing steps. Especially the filtering step varies by author. Moreover, as said in section 3.2, some steps of the preprocessing can be skipped to shorten the processing time and even improve quality in some cases.

For other matching algorithms, like correlation-based and ridge feature-based matching, the processes can be different. The preprocess is adjusted especially for the kind of matching that takes place further in the process, in this paper the minutiae-based matching.

**Improvement**
Still there is a lot to improve in fingerprint identification technology. The improvement in performance is mainly to achieve in the preprocessing step. Fingerprints of poor quality, as shown in figure 6-1, are still difficult to classify and match. Most of the misclassified or mismatched fingerprints are the result of bad quality. The preprocessing can improve the quality, but also discard useful information. Improvements are still achieved in this area.

*Figure 6-1: Fingerprint image of poor quality*

Furthermore there can be improvements in time. A user does not want to wait for his results. Identification has to be real-time when going through security at an airport. Recently a new way of classification is developed, that speeds up the process considerably. In stead of using the classes of the Galton-Henry scheme, the images are classified based on the angle of the ridges. Each image is divided into four main directions: 0, 45, 90 and 135 degrees. There arise four scatter plots of the fingerprint, where the gray scale indicates the concentration of present angles. A white spot on the plot of 45 degrees indicates only angles of 45 degrees, black spots indicate no angles of 45 degrees. The result is a very fast classification. This goes about 100 times more rapidly than so far possible, and the error percentage is smaller than 0.5%. [31]

**Future of fingerprint identification**
A dirty skin, scars, sweat and bruises can easily distort fingerprint recognition. A false matching can be the result of this. This factor does not play a role in iris recognition, a very reliable form of biometrics. Experts say that iris scans are therefore superior to fingerprint recognition systems. Also because an iris scan will produce results more quickly than a scan of a fingerprint. A check against 100.000 iris codes in a database takes two seconds. A fingerprint search will take fifteen seconds to perform the same task [30]. On the other hand, the fingerprint is the most practical biometric recognition method. It only takes a simple sensor and a smart link to a database. These necessities can easily be built in in small equipment like PDA’s and mobile phones. Iris detection on the contrary needs a high resolution camera. Besides, fingerprints are used in criminology where iris scans can not be used. Fingerprint identification will therefore be indispensable in the future.

# References

[1] D.H. Ballard - *Generalizing the Hough Transform to Detect Arbitrary Shapes*, in Pattern Recognition vol. 3, no. 2, pp. 110-122, 1981

[2] J. Blommé - *Evaluation of biometric security systems against artificial fingers*, Master's thesis LITH-ISY-EX-3514-2003, LinkÄoping University, 2003

[3] Al Bovik - *Handbook of Image and Video Processing*, Academic Press, 2000

[4] R. Cappelli, A. Lumini, D. Maio, D. Maltoni, - *Fingerprint classification by directional image partitioning*, in IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 21 no. 5, pp. 402–421, 1999

[5] S. A. Cole - *Suspect Identitites: A History of Fingerprinting and Criminal Identification,* Harvard University Press, 2001

[6] Vincenzo Conti, Giovanni Pilato, Salvatore Vitabile, Filippo Sorbello - *Verification of Ink-on-paper Fingerprints by Using Image Processing Techniques and a New Matching Operator*, in Proceedings of AI, 2002

[7] F. Galton - *Finger prints*, London: McMillan, 1892

[8] Rafael C. Gonzales, Richard E. Woods - *Digital Image Processing*, Prentice Hall, 2002

[9] A. Grasselli - *On the Automatic Classification of Fingerprint - Some consideration of the Linguistic Interpretation of Pictures,* in Methodologies of Pattern Recognition, 1969

[10] W. J. Hankley, J. T. Tou - *Automatic Fingerprint Interpretation and Classification via Contextual Analysis and Topological Coding,* in Pictorial Pattern Recognition, pp. 41 1-456, 1968

[11] E. R. Henry - *Classification and Uses of Fingerprints*, London: Routledge, pp. 54-58, 1900

[12] Lin Hong, Anil K. Jain - *Classification of Fingerprint Images*, in 11th Scandinavian Conf. Image Analyses, 1998

[13] Anil K. Jain, Sharath Pankanti - *Automated Fingerprint Identification and Imaging Systems,* in Advances in Fingerprint Technolofy. Elsevier, New York, second edition, 2001

[14] Anil K. Jain, Salil Prabhakar, Lin Hong - *A Multichannel Approach to Fingerprint Classification*, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, 1999

[15] Anil K. Jain, Salil Prabhakar, Arun Ross - *Fingerprint Matching: Data Acquisition and Performance Evaluation*, MSU Technical Report TR99-14, 1999

[16] Dario Maio, Davide Maltoni - *A Structural Approach to Fingerprint Classification*, in Proceedings of ICPR, 1996

[17] Davide Maltoni, Dario Maio, Anil K. Jain, Salil Prabhakar - *Handbook of Fingerprint Recognition*, Springer, 2003

[18] Karthik Nandakumar, Anil K. Jain - *Local Correlation-based Fingerprint Matching*, in Proceedings of ICVGIP, 2004

[19] S. Pankant, S. Prabhakar, A.K. Jain - *On the Individuality of Fingerprints*, 2001

[20] Nalini K. Ratha, Ruud Bolle - *Automatic Fingerprint Recognition Systems*, Springer, 2004

[21] Nalini K. Ratha, Kalle Karu, Shaoyun Chen, Anil K. Jain - *A Real-time Matching System for Large Fingerprint Databases*, in IEEE Transactions on Pattern Analysis and Machine Learning Intelligence vol. 18, 1996

[22] G. Stockman, S. Kopstein, S. Benett – *Matching Images to Models for Registration and Object Detection via Clustering*, in IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 4, no. 3, pp. 229-241, 1982

[23] Victoria Forensic Science Centre Fingerprint Branch (Victoria Police) - *Fingerprint Pattern Defenitions*, FPB Training Module 4, 2002

[24] Qinghan Xiao, Hazem Raafat - *A Combined Statistical and Structural Approach for Fingerprint Image Postprocessing*, in Pattern Recognition vol 24, pp. 985-992, 1991

[25] Qinzi Zhang, Kai Huang, Hong Yan - *Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges*, in Conferences in Research and Paractice in Information Technology, Vol. 11, 2002

[26] http://www.bromba.com/faq/fpfaqe.htm#beweisen

[27] http://www.bromba.com/technole.htm

[28] http://www.biometricsinfo.org/fingerprintrecognition.htm

[29] http://www.nifs.com.au/FactFiles/Fingerprints/how.asp?page=how&title=Fingerprints

[30] http://www.biometricgroup.com/in_the_news/miami_herald.html

[31] http://www.utwente.nl/nieuws/pers/archief/2002/cont_02-042.doc