Pattern Anal Applic (2004) 7: 94–113
DOI 10.1007/s10044-003-0201-2

ORIGINAL ARTICLE

Neil Yager · Adnan Amin

Fingerprint verification based on minutiae features: a review

Received: 16 June 2003 / Accepted: 21 November 2003 / Published online: 14 February 2004

Abstract Fingerprints have been an invaluable tool for law enforcement and forensics for over a century, motivating research into automated fingerprint-based identification in the early 1960s. More recently, fingerprints have found an application in biometric systems. Biometrics is the automatic identification of an individual based on physiological or behavioural characteristics. Due to its security-related applications and the current world political climate, biometrics is presently the subject of intense research by private and academic institutions. Fingerprints are emerging as the most common and trusted biometric for personal identification. The main objective of this paper is to review the extensive research that has been done on automated fingerprint matching over the last four decades. In particular, the focus is on minutiae-based algorithms. Minutiae features contain most of a fingerprint's individuality, and are consequently the most important fingerprint feature for verification systems. Minutiae extraction, matching algorithms, and verification performance are discussed in detail, with open problems and future directions identified.

Keywords AFIS · Biometrics · Fingerprint distortions Fingerprint verification · Orientation fields · Ridge detection

Introduction

Fingerprints have been used as a means of personal identification for over a century[1]. Traditionally, the driving force behind advancements in fingerprint technology has been law enforcement agencies and forensic scientists. Using fingerprints lifted at a crime scene to identify suspects can be a crucial step during a criminal investigation. Consequently, massive fingerprint databases have been collected by law enforcement agencies around the world. For example, the FBI maintains the world's largest fingerprint database, containing more than 200 million prints[2] [1]. The administration and querying of these large databases relies heavily on automated systems, thereby motivating the early research efforts in the field.

Another application of fingerprint-based identification that has emerged more recently is biometric systems. Biometrics is the automatic identification of an individual based on his or her physiological or behavioural characteristics. The ability to accurately identify or authenticate an individual based on these characteristics has several advantages over traditional means of authentication such as knowledge-based (e.g., password) or token-based (e.g., key) authentication [2]. Example applications of biometric systems include building access systems, ATM authentication and welfare disbursements. Due to its security-related applications and the current world political climate, biometrics has become the subject of intense research by both private and academic institutions.

There are several human characteristics that can be used as the basis for biometric systems [2]. For example, a person's face, retina, or voice can all be used to identify that individual with a high degree of accuracy. However, the use of fingerprints has several advantages over these other methods. The uniqueness of fingerprints has been studied and it is well established that the probability of two fingerprints matching is extraordinarily small [3]. Furthermore, unlike faces and voice prints, fingerprints are persistent with age and cannot be

N. Yager (✉) · A. Amin
School of Computer Science and Engineering,
University of New South Wales,
2052 Sydney, NSW, Australia
E-mail: nyager@cse.unsw.edu.au

[1] In Australia, the first official fingerprint branch was opened in 1903.

[2] The database maintained by Australian law enforcement agencies contains around 2.1 million prints.

easily disguised. Therefore, fingerprinting is one of the most researched and mature fields of authentication.

Due to the continuing needs of law enforcement and interest from the developers of biometric systems, efficient automated fingerprint identification systems (AFISs) are becoming increasingly widespread and are being extensively researched by the pattern recognition community. Given two fingerprint images as input, a fingerprint matching algorithm attempts to determine whether or not they were captured from the same finger. It is a common misconception that this is a solved problem. In actual fact, despite significant research efforts over the past four decades, state of the art fingerprint matching algorithms are nowhere near the theoretical upper bound on their performance [3]. Most fingerprint matching algorithms are based on matching small fingerprint details known as minutiae. The main goal of this paper is to review the fingerprint matching algorithms that are based on minutiae matching. Also discussed are reasons why automated fingerprint matching is a difficult problem, which approaches seem to have the most promise and possible future directions in the field.

The section Background begins with some background, including the history of fingerprint matching, fingerprint definitions and structure and automated fingerprint identification systems. The section Feature extraction discusses feature extraction for fingerprint matching, in particular the extraction of minutiae points. The section Fingerprint matching takes a close look at the problem of minutiae matching. The performance of some fingerprint matching algorithms is presented in the section Performance, while the section Conclusions closes with some concluding remarks and speculation regarding the future of automated fingerprint identification.

Background

History

With the introduction of the Habitual Criminals Act in England in 1869, the need for a widespread, accurate system of personal identification became apparent [4]. The Act gave lenient sentences for first-time offenders and harsher ones for repeat offenders. Hardened criminals quickly realised that they could receive reduced prison sentences by giving false names to the court. Initially, the courts relied on police officers and prison guards to recognise repeat offenders, but soon realised the limitations and unreliability of this method. In the early 1880s, France adopted a method of identification known as anthropometrics, which identified people based on a number of physical measurements such as arm length and head size. There was some initial success with this method, but taking the measurements was tedious and could be inaccurate if they were not taken by a trained professional.

Around the same time the anthropometric system was implemented in France, preliminary scientific investigations into the use of fingerprints for identification were being conducted. The exact origin of the use of fingerprints for identification is unclear. There is some evidence that fingerprints were used in ancient times [5]; however, there is little indication that anyone recognised their full potential as a means of personal identification. The first reliable record comes from Sir William Herschel [6]. In 1858 he was an employee of the East India Company and stationed in India. While preparing a contract with a local man for building materials, he decided to take an imprint of the individual's palm instead of using the more conventional signature. His motivation was to "frighten [the contractor] out of all thought of repudiating his signature hereafter." Herschel soon recognised the potential of using fingerprints as a means of personal identification and studied the issue as a hobby for years after his initial experiment. The first scientific publication proposing the use of fingerprints for identification was written by Henry Faulds in 1880 [7], and in the late 1880s Sir Francis Galton began a rigorous study of fingerprint-based identification [8]. The first wide-spread adoption of a fingerprint-based identification system was in India in 1893. Edward Henry, another prominent early fingerprint researcher, supplemented the anthropometric system being used with thumb prints [9]. Encouraged by the success of the system, Britain implemented a similar scheme the following year. Soon the anthropometric measurements were dropped entirely, and law enforcement agencies worldwide were using identification schemes based entirely on fingerprints.

There are several advantages of fingerprints over anthropometrics. Unlike the intricate set of measurements required for anthropometrics, fingerprints could be recorded quickly and easily by people with little training. The discrimination of anthropometrics is also not as high as fingerprints. Fingerprints were also discovered to have the remarkable property that they are captured frequently during the course of day-to-day life. When a finger makes contact with a surface, a nearly invisible copy of the fingerprint is often left in its place. This is known as a *latent* print. The implications of this discovery for criminal investigations were staggering. In 1902 fingerprints were first used as evidence in a British courtroom, contributing to the conviction of a burglar [4]. Fingerprint evidence continues to this day to be one of law enforcement's most powerful tools and is an integral part of modern criminal investigations.

Fingerprint structure

The skin on the palm and fingers of the human hand along with the skin on the sole and toes of the foot has the unique property of being corrugated by a pattern of narrow ridges and valleys (also known as furrows or ravines) [5]. These skin ridges have several functional purposes. For example, they increase the friction between the skin and other surfaces, thereby reducing slipping. For this reason the skin is often known as

friction skin. Because of the wide variety of ridge orientations on the surface, the ridges are able to increase the drag in all directions. The corrugation also benefits the sense of touch by increasing sensitivity and helping to distinguish different textures. The ridges on human skin become elevated during fetal development by around the eighteenth week of pregnancy, and the pattern of ridges remains unchanged throughout an individual's life. This is known as *permanence*. The overall pattern of ridges and valleys is largely determined by genetic factors. However, during their formation minor developmental disturbances create local ridge irregularities. These disturbances are consequences of the fetus' unique development environment in the womb, so all ridge valley patterns are unique when examined sufficiently closely. This is even true for identical twins who share the exact same DNA. The global ridge patterns for twins are often very similar, yet minute irregularities can be used to distinguish the individuals [10]. This property of friction skin individuality is known as *uniqueness*.

Of particular interest are the ridges and valleys at the tips of fingers, known as fingerprints.

Figure 1a shows a sample fingerprint and Fig. 1b contains a magnified view of the ridges and valleys from a small area of a fingerprint. Due to the permanence and uniqueness properties mentioned above, fingerprints can be used to uniquely identify individuals throughout their life.

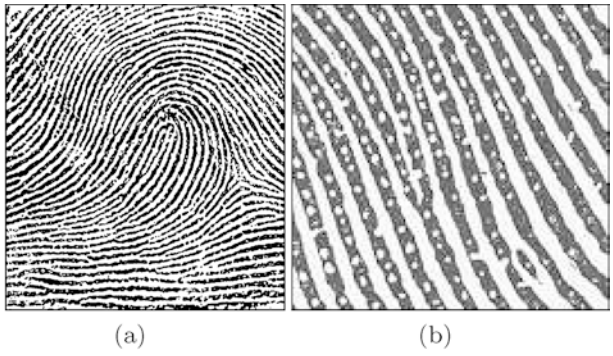


Fig. 1 a A fingerprint; **b** A magnified view of fingerprint ridges and valleys

There are three levels of structure in a fingerprint. The first level of structure is the global pattern of ridges and valleys. For example, some of the ridges in Fig. 1 enter from the bottom-left of the image, loop around a common centre point, and exit on the left. This is the global pattern of the fingerprint ridges, and most fingerprints fall clearly into one of several pattern classes. The most common classification scheme is known as Henry's classification and is shown in Fig. 2 [9].

Henry's classification (and others based on it) are known as *exclusive* because they partition fingerprints into mutually exclusive categories. The primary application of fingerprint classification is indexing. Early fingerprint collections were stored on physical cards and efficient filing and retrieval systems needed to be developed. Using Henry's system, the cards were filed based on the sequence of classes from each of the ten fingerprints. To identify an anonymous suspect, it was only necessary to compare his or her prints against cards with the same sequence of classes. These cards were easy to locate and were typically only a very small percentage of the total number of cards on file.

The distribution of fingerprint classes in nature is not uniform. Central pockets, twin loops and accidentals are very rare so they are often ignored for classification purposes. The probabilities of the other classes are approximately 0.037 (arch), 0.338 (left loop), 0.317 (right loop), 0.029 (tented arch) and 0.279 (whorls) [11]. Note that left loops, right loops and whorls are the most common, making up 93.4% of all fingerprints. Therefore, fingerprint classes have a very limited ability to distinguish individual prints from each other.

Automated fingerprint classification has been studied for many years and continues to be a challenging

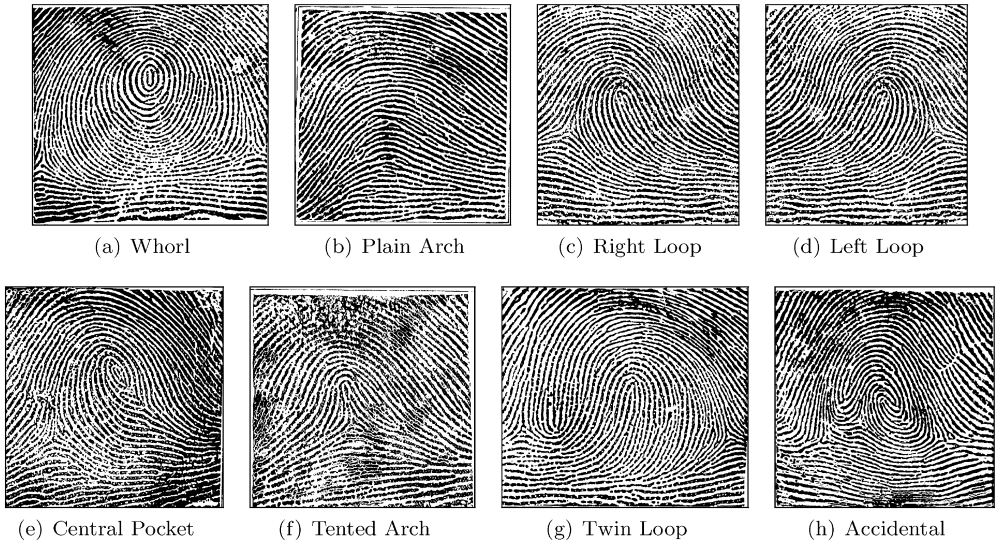


Fig. 2 Henry's fingerprint classes

research problem. However, to limit the scope of this paper, it will not be discussed.

The second level of structure occurs in local fingerprint regions. Notice that near the centre of Fig. 1b one of the ridges splits into two distinct ridges. Also note that near the bottom right of the image there is a very short ridge. These local ridge discontinuities, known as a *minutiae* (or minutia in the less common singular form), have little effect on the global ridge-valley pattern. However, it is the existence and locations of these minutiae that embody much of a fingerprint's individuality. For this reason, they are the most important and common discriminating feature used by human experts.

There are two basic types of minutiae: ridge endings and bifurcations. Ridge endings are places where ridges terminate and bifurcations are locations where a single ridge separates into two ridges (see Fig. 3).

There are other types of minutiae, but they are combinations of ridge endings and bifurcations. A typical fingerprint contains up to 80 minutiae; however, far fewer will be present in a latent print or a print captured from a small scanner.

The third level of fingerprint structure includes low-level features such as sweat pore locations (these are visible in Fig. 1b) and ridge shapes. These features are sometimes used by human experts when comparing prints. However, they are rarely used in automated systems because they require very high resolution scans for reliable feature extraction.

Singularities are another important fingerprint structure that have both global and local properties. Globally, a singularity is a region of a fingerprint where the ridge pattern makes it visually prominent. There are two types of fingerprint singularities: cores and deltas. Locally, a core is the turning point of an inner-most ridge and a delta is a place where two ridges running side-by-side diverge. Core and delta points are best illustrated by examples (see Fig. 4).

Singularities are useful for determining a fingerprint's class. For example, left loops (as in Fig. 4) have one core point near the centre of the print and one delta point to the lower right. Singularities also have other uses, such as fingerprint alignment and as a coarse discriminating feature.

A natural question regarding fingerprint structures concerns the amount of discriminatory information they hold. In particular, what is the probability that two different prints will match? Pankanti et al. have conducted a study on the individuality of fingerprints based on minutiae features [3]. They present a model of fingerprint individuality that is particularly appropriate for automated matching because it takes into account some of the challenges faced by matching algorithms. Pankanti et al. present a variety of results based on the number of minutiae present in a fingerprint and the number of correspondences required to consider two prints a "match". An example result is their estimate that the probability a fingerprint image containing 36 minutiae will match 12 minutiae with a different fingerprint also containing 36 minutiae is 6.10×10^{-8} . In other words, a system whose prints contain 36 minutiae on average should be able to distinguish about 16 million fingerprints based on 12 corresponding minutiae pairs.

AFISs

The first publication on the automation of fingerprint matching appeared in 1963 [12]. In the subsequent four decades there has been a considerable effort by law enforcement, private, and academic institutions to develop efficient AFISs.

There are currently two main applications of automated fingerprint identification technologies; those in the criminal and civil sectors. The systems used in these respective fields do have some differences. For example, the databases for existing criminal applications (e.g., the one maintained by the FBI) are much larger than those used for typical biometric systems (e.g., a system to control access to a secure building). Another difference is the expected quality of the fingerprint images. For criminal applications there are two sources of prints. Prints obtained from suspects in custody are of an arbitrarily high quality (depending on the procedures being used to capture them), while latent prints lifted from crime scenes have a wide range of qualities (depending on a number of factors such as the properties of the capturing surface). On the other hand, for a biometric system the prints are all captured from the

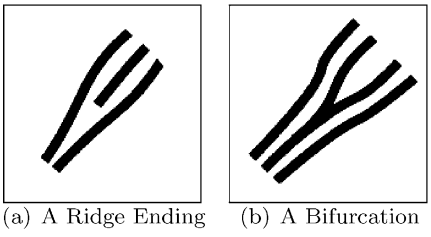


Fig. 3 Fingerprint minutiae

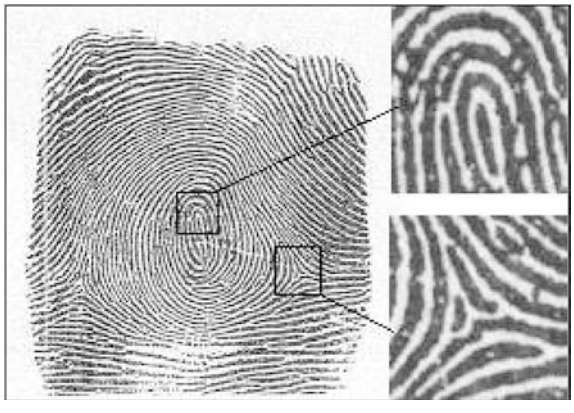


Fig. 4 Core *top* and Delta *bottom* points of a fingerprint

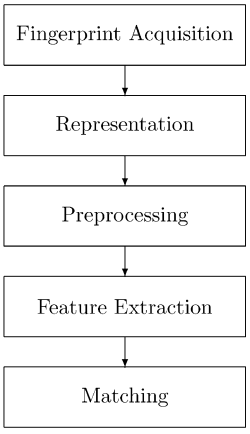


Fig. 5 The stages of a typical AFIS

same type of scanner and will tend to have a consistent quality.

Criminal and civil AFISs share many aspects in common despite their differences. In particular, they follow the basic stages for identification, and researchers from both fields face similar challenges.

Fingerprint matching stages

As mentioned above, most AFISs follow the same basic stages for fingerprint identification. These are illustrated in Fig. 5. The emphasis placed on a particular stage and the techniques used within that stage will vary depending on the requirements of the AFIS.

Fingerprint acquisition: The first issue that must be addressed is how the fingerprints will be acquired in a digital format. Latent fingerprints are prints that are ''lifted'' from crime scenes. The latent prints are developed using fingerprint powders (or more sophisticated chemical processes), photographed and scanned using a high-resolution scanner. The traditional method for police to obtain prints of a suspect in custody is to dab the finger tips in ink and then roll them onto fingerprint cards[3]. This is known as ''inked'' (or ''offline'') acquisition. For use in automated systems, this print would then be scanned into a digital format. Obviously this method is not feasible for biometric systems that perform matching in real-time. Such systems use hardware devices specifically designed for scanning fingerprints. This is known as ''ink-less'' (or ''live-scan'') acquisition[4]. There is currently a wide variety of live-scan fingerprint technologies available [13], and the overall trend is towards devices that are small, fast and increasingly inexpensive.

[3] Australian law enforcement agencies also acquire palm prints from offenders. Palms are covered by the same type of skin as finger tips, and can therefore be used for identification as well.

[4] Australian police are in the process of updating their facilities to make high-end live-scan units available at most major police branches around the country.

Representation: Once a fingerprint has been acquired in a digital form, an AFIS must select an appropriate storage representation. Considering the immense size of some fingerprint databases (such as the one maintained by the FBI), it may be very important to store the prints in a compact and efficient manner. However, it is equally important to select a representation that maintains the fingerprint's discriminatory information. The FBI has developed an image-based representation that is based on wavelet compression [14]. An alternative to image-based representations is to store only the extracted features. One advantage of this approach is that the pre-processing and feature extraction stage only needs to be applied once for each image, as opposed to extracting the features before every match. This can lead to efficient algorithms appropriate for systems performing real-time identification. Furthermore, storing only feature vectors can lead to compact representations since only relevant information is stored. The main disadvantage of this approach is that only partial fingerprint information is stored, so any changes in the pre-processing or feature extraction stages require the fingerprints to be recaptured. Therefore, this approach is not feasible for law enforcement systems which may not be able to obtain new copies of prints on demand. However, small-scale biometric systems with hard-coded matching algorithms may benefit from the storage and computational advantages of a feature-based representations. Of course, it is also possible for systems to store both sample images and extracted feature values.

Pre-processing: A very important component of AFISs is the pre-processing of fingerprint images. Fingerprint impressions are often obtained in uncontrolled environments, leading to a wide range in the quality of the prints. This is especially relevant for latent prints lifted from crime scenes. Many of a fingerprint's discriminating features are minute ridge irregularities that are very difficult to extract for low-quality images. The performance of a fingerprint matching algorithm is closely related to the reliability of its feature extraction stage. Consequently, there has been a lot of research into the problem of fingerprint pre-processing and feature enhancement. In a local area, the ridges and valleys of a fingerprint have a well-defined frequency and orientation. Therefore, it is natural to use frequency analysis tools to enhance ridge information, such as Fourier transforms [15, 16, 17, 18], Gabor filters [19, 20, 21] and wavelets [22, 23]. Other useful pre-processing steps are histogram equalisation [24, 25] and Laplacian filtering [26]. Orientation field estimation, binarisation and thinning are common and important pre-processing steps as well (especially for minutiae extraction), and are discussed in the section Minutiae extraction from ridge skeletons. A review fingerprint image enhancement techniques has recently been conducted by Ailisto, Lindholm and Tikkanen [27].

Feature extraction: The feature extraction stage is concerned with finding and measuring important properties of the fingerprint that will be used to match it

against others. Most fingerprint identification systems are based on matching minutiae, and these algorithms are the main subject of this paper. A detailed discussion of minutiae extraction is presented in the section Feature extraction.

Matching: The final goal of any AFIS is to find (or confirm) the identity of the person whose fingerprint has been submitted to the system. Ultimately, this involves comparing the features extracted from two prints and determining the likelihood that they have been captured from the same finger. Fingerprint matching algorithms are presented in the section Fingerprint matching.

One vital feature of AFISs that has not been mentioned is security, and this is particularly important considering the nature of most applications (both civil and criminal). The goal of AFISs is to identify individuals with a high degree of confidence, and the consequences of attacks can be very serious; especially those leading to false matches. Research has been conducted into the security concerns of AFISs (and biometric systems in general) [28, 29], and these issues should be carefully considered when designing an identification system.

Fingerprint matching challenges

There are several domain-specific challenges that must be addressed by AFISs [13]:

Inconsistent and irreproducible contact: every time a finger is pressed against a surface, it is applied with a certain amount of pressure at a well defined angle. The actual amount of pressure used and the contact angle will vary from time to time, resulting in a different portion of the print being captured. This is especially problematic for fingerprint scanners with a small scanning surface. Due to inconsistent contact, fingerprint verification algorithms should not be sensitive to translations or rotations of fingerprints. Furthermore, a robust verification system should be able to match two prints that only have a small area of overlap.

Incomplete ridge structure: there are several reasons why the entire ridge structure of a fingerprint may not be captured. If the skin is dry, sweaty, diseased or injured, some parts of the ridges may not make contact with the capturing surface, while some valleys may touch the surface. It is important for a verification system to be robust against incomplete ridge structures, and this issue is usually addressed during the pre-processing stage.

Noise: even under ideal conditions, noise will be present to some degree in all fingerprint images. This is an inevitable consequence of taking discrete measurements of the physical environment. Some form of noise removal is usually performed during the pre-processing stage.

Elastic distortions: when a fingerprint is captured, a 3D finger is being mapped to a 2D image. This introduces nonlinear deformations due to elastic distortion of the fingerprint skin. This is particularly troublesome for algorithms based on matching minutiae points as the location and orientation of minutiae can be altered.

Feature extraction

Most fingerprint verification algorithms rely on minutiae information to some degree, and these algorithms can only be as robust as the underlying minutiae information. Therefore, reliable minutiae extraction is vital to a system's performance. There are two main approaches to minutiae extraction. The first approach uses a thinned representation of the binary ridge structure, known as its skeleton. The second approach attempts to extract the minutiae locations from the grey-scale image itself. Some AFISs include a post processing stage to confirm that valid minutiae have been extracted, and this is known as minutiae verification.

Minutiae extraction from ridge skeletons

The most popular method for minutiae extraction is to use a binarised and skeletonised representation of the fingerprint. These algorithms generally consist of three main pre-processing stages: orientation field estimation, ridge detection and thinning. The minutiae are then extracted directly from the skeleton.

Orientation field estimation: orientation fields (also known as directional fields) contain information about the local average directions of fingerprint ridges. They are stored as a discrete matrix whose elements are vectors tangent to the fingerprint ridges in the corresponding region. An example of a fingerprint and its orientation field is shown in Fig. 6.

Orientation fields are widely used in fingerprint systems and have several applications including pre-processing, fingerprint classification and ridge detection. There are several techniques that can be used to calculate orientation fields. One common approach is based on image gradients, and another is based on a mask that is convolved with the image.

In a digital image, a *gradient* points in the direction of the maximum rate of change. For an image $f(x,y)$ the gradients are defined as follows:

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{1}$$

In a fingerprint image the gradient at a pixel on a ridge will point towards a valley. Similarly, the gradient for a valley pixel will point towards the neighbouring ridge. Therefore, the elements of an orientation field are normal to the gradients in the local area. However, gradients are defined at the pixel level so there will be a large variation of values. One obvious method of calculating the orientation field for a given region is to set its orientation perpendicular to the average direction

Fig. 6 a A fingerprint image; **b** The orientation field of **a**

of its gradients. However, some care must be taken when calculating the average gradient direction because two gradients on different sides of the same ridge will point in opposite directions and cancel each other out if averaged (even though they represent the same ridge orientation). One solution to this problem is to double the angles of the gradient vectors before averaging [30].

Jain et al. present another way to calculate orientation fields based on gradients [31]. Their algorithm is a modified version of one originally proposed by Rao [32]. Rao's algorithm works as follows. Firstly, the image is divided into blocks of size $W \times W$ and the gradients G_x and G_y are computed for each pixel. The orientation of the block is then estimated using the formula:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{\sum_{i=1}^{w}\sum_{j=1}^{w} 2G_x(i,j)G_y(i,j)}{\sum_{i=1}^{w}\sum_{j=1}^{w}\left(G_x^2(i,j) - G_y^2(i,j)\right)}\right) \quad (2)$$

If the fingerprint image is of poor quality, additional processing is necessary to improve the quality of the estimated field. Jain et al. propose a hierarchical method to accomplish this. A *consistency level* is calculated based on the variation of orientations in a block. If the consistency level is below a given threshold, the orientations for this region are re-estimated at a lower image resolution. This hierarchical process is continued until the consistency level is above the threshold. This method has the advantage that local areas heavily affected by noise will be smoothed without using excessive averaging for the rest of the image.

The ridge-valley algorithm is a popular way to obtain orientation fields that does not require calculating gradient values. The algorithm is based on one originally proposed for the binarisation of fingerprint images [33]. The first step is the convolution of the mask shown in Fig. 7 with the image. The mask is comprised of 8 slits, numbered 1 though 8. The slit sum s_i for $i = 1,...,8$ is equal to the sum of the pixel intensity values at all spots that contain an i in the mask. The next step is to classify each pixel as belonging to a ridge or a valley, and this is done by comparing the slit sums. Let s_{min} be the lowest slit sum at a pixel and s_{max} be the highest. The pixel is classified as a "valley" pixel (white) if:

7		8		1		2		3
6		7	8	1	2	3		4
		6				4		
5		5		C		5		5
		4				6		
4		3	2	1	8	7		6
3		2		1		8		7

Fig. 7 The ridge valley filter mask

$$\frac{1}{2}(s_{\min} + s_{\max}) > \frac{1}{8}\sum_{i=1}^{8} s_i \quad (3)$$

The intuition behind this is that if the pixel belongs to a valley, one of the slits will lie along that valley, and consequently have a high sum (recall that valleys are white, and white pixels have a high intensity). The other slits will cross ridges and valleys, giving them similar (and lower) slit sums. Therefore, the average of s_{min} and s_{max} is expected to be higher than the average of all the slit sums. If Eq. 3 is false, the pixel is classified as a "ridge" pixel. Finally, an orientation is assigned to each pixel. Valley pixels are assigned the orientation of s_{max} because the slit with the highest sum is the most likely to lie along the valley. Similarly, ridge pixels are assigned the orientation of s_{min}. This process assigns each pixel an orientation (quantised to 8 possible directions). Local area orientations can be averaged to reduce the effect of noise. This method of computing the orientation field is used widely because it is simple and fast. However, the estimated field is coarse due to the limit of 8 possible directions. Systems requiring a greater accuracy should use a gradient-based method.

Ridge detection: in order to find the minutiae in a fingerprint image, it is necessary to first locate the ridges. A *ridge map* is a fingerprint image in which black pixels correspond to ridges and white pixels correspond to valleys (see Fig. 8a). Given a grey scale fingerprint image, the most straightforward approach to creating its ridge map would be to use a simple thresholding algorithm. Since valley pixels are brighter than ridge pixels, any pixel values greater than a given threshold can be classified as valleys, and all others as ridges. However, due to noise, the output of this will be very poor. Therefore, additional information from surrounding pixels is often used to improve the accuracy of the ridge detection. (Figure 9 shows various ridges with holes).

After finding the orientation field, Jain et al. convolve the original input image with two masks that accentuate the grey level values on fingerprint ridges [13]. These

masks exploit the observation that grey level values on ridges attain their local maximum in the direction normal to the ridge orientation. After applying the masks, thresholding is used to determine whether a pixel belongs to a ridge or valley.

The ridge-valley algorithm presented above for orientation field estimation can also be used for ridge detection. In fact, it was originally proposed for ridge detection, and was later adopted for estimating orientation fields [17].

Chang and Fan have taken a close look at the ridge detection problem, and have proposed a sophisticated algorithm based on grey level histogram decomposition [34]. Their approach is more complex than the ones mentioned above, but has a higher performance on low-quality images.

Ridge map thinning: a skeleton is a line representation of an object that is one pixel thick and preserves the topology of the object, and is created by thinning a binary image. An example of a ridge map skeleton is shown in Fig. 8b. Thinning is a common pre-processing step in image processing applications, and many thinning algorithms have been proposed in the literature. Thinning algorithms are often based on mathematical morphology. For example, Fitz and Green present morphological operators specifically for fingerprint thinning [35].

Since a skeleton preserves the ridge map's topology, special care must be taken to ensure that a ridge map is free from artefacts. For example, an accidental hole in a ridge will lead to a skeleton that appears to have two bifurcations. Similarly, a speckle due to noise will create two ridge endings in the skeleton. Therefore, it is important to remove noise from the ridge map before thinning. An examination of the fingerprint thinning problem is presented by Rao [36].

Minutiae extraction: In the ideal case, extracting minutiae from a thinned ridge map is a trivial task. Any black pixel that has only one black (eight-connected) neighbour is a ridge ending, and any black pixel with more than two black neighbours is a ridge bifurcation. This is illustrated in Fig. 10. However, in reality it is much more complicated. Imperfections in the input image will lead to undesired spikes and broken ridges in the fingerprint skeleton. Spikes and broken ridges will create spurious ridge endings and bifurcations, and if left untreated the number of minutiae extracted can be an order of magnitude higher than the number of true minutiae [37]. Because of the importance of reliable minutiae extraction, it is necessary to perform some additional filtering to eliminate false minutiae. For example, single-neighbour pixels around the border of the image should be ignored because they are not caused by true minutiae.

Chen and Kuo suggest several heuristics to eliminate spurious minutiae [38]. To remove spikes, any ridges shorter than a given threshold are deleted. To correct broken ridges, small gaps between ridge endings are connected. Finally, if many minutiae are detected in a small area, they are discarded because they are likely caused by noise. Ratha et al. use morphological operators to detect and remove spikes [39], and various other post processing steps can be found in the literature [40, 41, 37]. However, even after performing filtering steps such as these, it is still common for spurious minutiae to be extracted. Therefore, some AFISs use minutiae verification algorithms to validate candidate minutiae (see the section Minutiae verification).

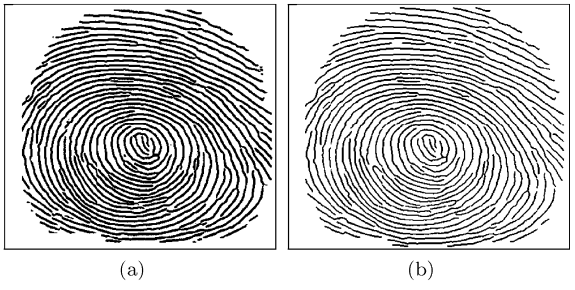


Fig. 8 A fingerprint ridge map and its skeleton [31]

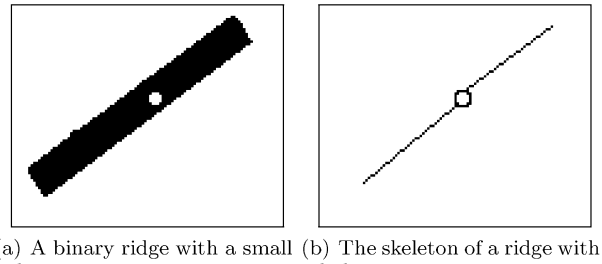


Fig. 9 a A binary ridge with a small hole; **b** The skeleton of a ridge with a hole

Minutiae extraction from grey scale images

The previous section looked at various methods of extracting minutiae from a fingerprint skeleton. The advantage of using this approach is the simplicity of extracting and labelling the minutiae when an accurate

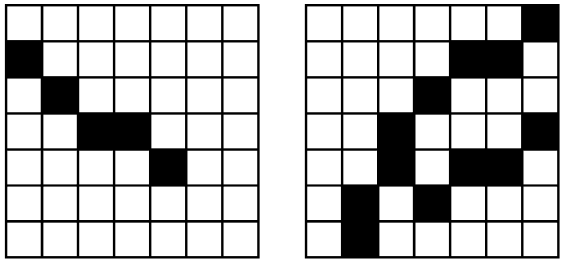


Fig. 10 Minutiae points in a thinned ridge map

ridge skeleton can be found. However, it often happens that many false minutiae are detected due to spikes and broken ridges in the skeleton, and therefore careful processing is necessary to prevent recording these as true minutiae. Furthermore, calculating the orientation field, binary ridge map and skeleton are computationally expensive procedures. Yet, despite extensive processing, the results for low quality images can still be very poor. This is partly due to the fact that a lot of fingerprint information is lost through binarisation and thinning. Another approach to minutiae extraction bypasses these problems by extracting the minutiae directly from grey scale images.

Maio and Maltoni propose a minutiae extraction technique based on following ridge lines in the grey scale image [42]. Ridges are traced with the aid of the orientation field, and the location of ridge endings and bifurcations are recorded. Jiang et al. have developed a modified version of Maio and Maltoni's system using oriented filters [43]. One advantage of their system is that it is adaptive to the bending level of the ridge being traced, making the computation faster and more accurate. Another method using ridge line tracing is presented by Lui et al. [44]. Instead of tracing only ridges, the authors propose to trace both ridges and valleys, and observe their structural relationship. A ridge and its two neighbouring valleys are traced, and if the two valleys join a ridge ending has been found. Similarly, if the distance between the two valleys grows, a bifurcation has been found.

The application of fuzzy logic to minutiae extraction from grey scale images has been investigated by Sagar et al. [45]. A small window is scanned over the fingerprint image, and fuzzy rules based on the pixel intensities are used to determine if the window contains a minutiae. A similar approach is described by Sagar and Koh, which incorporates a neural network for classification [46].

Minutiae verification

Several post processing steps for filtering false minutiae from a thinned ridge map are mentioned in the section Minutiae extraction from ridge skeletons. Most of these are heuristics based on visual observations of commonly occurring false minutiae. However, even after applying these heuristics it is common for spurious minutiae to still exist. Therefore, some AFISs use an additional stage known as minutiae verification. Minutiae verification is a final round of filtering aimed at eliminating false minutiae by using a trained classifier. Maio and Maltoni use a neural network to validate the candidate minutiae output by their minutiae extraction algorithm [42, 47]. The features used for classification are the pixels intensities in the local neighbourhood of a candidate minutiae. The authors report a significant performance increase, but these results are based on a rather small test set.

Using fingerprint skeletons to extract minutiae is appealing because it is conceptually simple. However, large numbers of spurious minutiae are common because useful information is lost during the binarisation and thinning stages. Methods that extract minutiae from the grey level image work well, but are complex and difficult to implement. One approach to fingerprint extraction is to use the skeleton-based algorithms to obtain candidate minutiae, and then use the grey level image for verification. This approach is investigated by Prabhakar et al. [48]. The local area of the candidate minutiae in the grey level image is processed to enhance ridge clarity. A learning vector quantiser (LVQ) is trained using positive and negative samples, and is used to classify candidate minutiae as either true or false minutiae. Strong results are reported by the authors, suggesting that the extra processing required by a minutiae verification stage is justified for minutiae-based AFISs requiring high matching accuracy.

Non-minutiae features

One reason why minutiae features have traditionally been used in AFISs is that human experts use them extensively. Therefore, it was natural for early fingerprint identification systems to rely on the same techniques already in practice. Furthermore, minutiae-based results can easily be validated by a human expert, thereby gaining confidence in a system's robustness. However, there are some drawbacks of systems based strictly on minutiae features. In particular, minutiae are computationally expensive to extract, unreliable for low-quality images, and vulnerable to nonlinear deformations. Consequently, there has been some research into features not based on minutiae.

The cyclic structure of local fingerprint regions [49], shape signatures of fingerprint ridges [50] and directional micropattern histograms [51] have been proposed as alternative fingerprint features. Wavelets [52, 53, 54] and Gabor filters [55, 56, 57] have also been investigated as tools for feature extraction. Furthermore, experiments based on image verification [58, 59, 60] and optical processing techniques [61, 62, 63] have also been conducted. The discriminating power of these features varies widely. For example, wavelet coefficients are known to be sensitive to translations and rotations, so the simple approach taken by Tico et al. [54] will have a limited ability to distinguish fingerprints. On the other hand, Gabor filters are well suited to extracting ridge features, and have yielded encouraging results [55]. In general, it is known that minutiae features contain much of a fingerprint's individuality, so they will continue to be the primary feature for large scale, high performance AFISs. However, non-minutiae features are suitable for biometric systems that use relatively small databases. In these cases, time consuming minutiae extraction algorithms may not be viable. Furthermore, non-minutiae features can be valuable as supplementary features,

especially for low-quality images. Prabhakar and Jain [64] have designed a system that combines the results from three minutiae-based matching algorithms and one texture (extracted using Gabor filters) based algorithm. The results from the combined classifier are significantly higher than any classifier in isolation.

Fingerprint matching

There are two distinct (but related) fingerprint matching problems. Fingerprint *identification* is the problem of matching a query fingerprint against a database to identify its owner. Fingerprint *verification* (or authentication) is concerned with determining whether or not two prints are impressions from the same finger. For a database containing N prints, fingerprint identification can be achieved by performing N 1-to-1 fingerprint verifications. However, since fingerprint verifications are computationally expensive, classification or indexing techniques are generally used to reduce the number of verifications performed [65, 66, 67, 68].

The input to the verification problem is two fingerprints: one is the *test* (or *query*) print whose identity is to be verified, and the other is the *reference* (or *template*) print that is known to belong to a particular individual. The output from the verification is YES or NO, and usually some measure of similarity or confidence.

Minutiae pattern matching

The output of the minutiae extraction stage is typically a list of minutiae locations and orientations, represented by the 3-tuple (x,y,θ). It should be noted that most fingerprint matching algorithms do not differentiate between ridge endings and bifurcations when comparing two fingerprints. One reason for this is that noise or excess pressure can cause a bifurcation to look like a ridge ending, and vice versa. However, it was recently observed that by taking minutiae types into consideration when matching, a system's accuracy can be increased [48]. Therefore, minutiae types should be considered when high-quality images are being used and the minutiae extraction algorithm is reliable.

Aligning the minutiae sets is known as *registration*, and is essentially a point pattern matching problem (see Fig. 11). Point pattern matching is a well known problem that often arises in pattern recognition. The goal is to find a translation and rotation (and possibly scaling) that aligns two point sets. However, in the case of minutiae matching there are several complications. Firstly, the number of minutiae in each set may be different due to different regions of the fingerprint being captured. In this case, one must only align the overlap of the two sets. Secondly, missing and spurious minutiae must be taken into consideration. In other words, the matching algorithm must accommodate points in one set that do not have a corresponding point in the other set

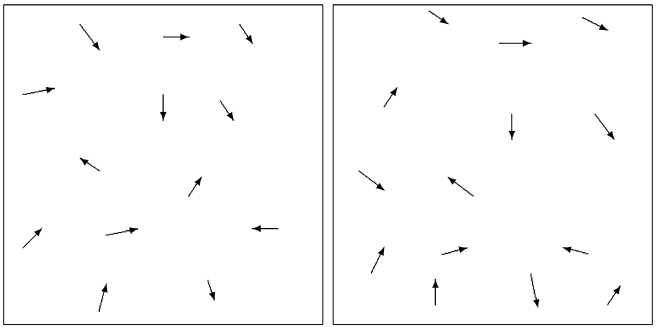


Fig. 11 The minutiae pattern matching problem. Have these minutiae sets been extracted from the same fingerprint?

(even within the overlapped region). The third main difficulty is the nonlinear deformation of point sets. If the deformations are not explicitly modelled, a perfect alignment of the point sets will not be possible. In this case, the alignment algorithm must try to find the optimal alignment according to some criteria (such as minimising the sum of squared errors). Finally, when being used for biometric systems, the alignment algorithm must be very efficient. The reason for this is that the calculations must be performed in real-time by an embedded system with limited computational resources. Numerous point pattern matching algorithms have been proposed, but due to the domain-specific problems just mentioned, many of them are not well-suited to fingerprint matching.

After alignment, a matching score for the two fingerprints is calculated. This is normally done by tabulating the number of corresponding minutiae and normalising by the total number of minutiae. A typical matching score formula [31] is as follows:

$$\text{Matching score} = \frac{100N_{\text{pair}}}{\max\{M,N\}} \tag{4}$$

where N_{pair} is the number of corresponding minutiae, M is the number of minutiae in the reference set, and N is the number of minutiae in the test set. The score required for fingerprints to be considered a match can be an adjustable parameter of the system. Bolle et al. point out that designing an appropriate matching score function is non-trivial, and present an interesting discussion of the issues involved [69].

Ratha et al. [70] estimate the rotation and translation parameters using the generalised Hough transform. The Hough transform is a common tool in image analysis that is normally used for detecting lines in point sets, but it can be generalised for point pattern matching. The space of all possible transformations[5] is discretised into a finite set of values, and this is known as the parameter

[5] In our case, we assume the images are the same resolution and consider only translation and rotation, but Ratha et al. also consider scaling.

space. For each pair of potentially matching minutiae (one from the test set, and one from the reference set) the translation and rotation necessary to align them is calculated. Evidence for this translation and rotation is accumulated in the parameter space. After testing all possible matching minutiae pairs, the parameter space is used to select the most likely translation and rotation parameters (i.e., the one with the most accumulated evidence).

There are two main drawbacks of this approach. One problem is that it is very computationally expensive. As the sizes of the minutiae sets grow, the number of potentially matching minutiae pairs to be considered increases exponentially. Another drawback of this approach is that the transformation parameters are rigid, so nonlinear distortions will not be modelled.

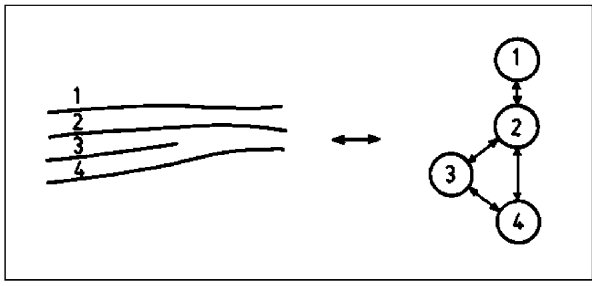
Structural matching

Structural information in pattern recognition encodes the interrelationship of low-level features. This is a natural approach for minutiae matching because the relationship between a fingerprint's minutiae contains much of a fingerprint's individuality.

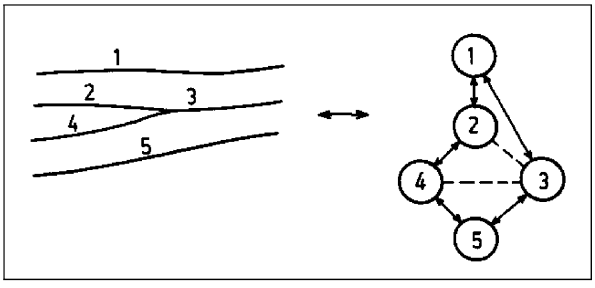
Graphs can be used to represent structural relationships, and are consequently a common tool for structural pattern recognition. Using graphs to represent the topological relationship between fingerprint features has several appealing advantages, including being invariant to fingerprint translations, rotations and distortions. Isenor and Zaky have developed a graph representation for fingerprints in which ridges are nodes and edges connect the nodes for neighbouring ridges [71]. Minutiae are not explicitly extracted from the image, but are represented by known substructures in the graph. This is illustrated in Fig. 12. There are some limitations of this fingerprint representation. Firstly, two graphs from the same fingerprint will not necessarily be isomorphic. This is due in part to different regions of the print being captured, and also due to errors in the ridge structure due to noise (e.g., broken ridges). Therefore, inexact graph matching algorithms are necessary. Secondly, graph matching is known to be a computationally difficult problem in itself, making it inappropriate for real-time systems. One of advantage of the graph representation is that it is tolerant of distortions. However, in a sense this approach is overly tolerant. Absolute distances between minutiae are lost, so care must be taken to ensure that fingerprints are not being matched on the basis of severe distortions.

Fan et al. have developed a fuzzy bipartite weighted graph model of fingerprints [72]. A node is a cluster of neighbouring minutiae, and the cluster features are characterised by fuzzy values. Verification is performed using a fuzzy bipartite weighted graph matching algorithm.

Hrechak and McHugh present an alternative representation of fingerprints that is based on local structural



(a) A ridge ending



(b) A bifurcation

Fig. 12 Minutiae graph representations [71]

relationships among minutiae [73]. For each minutia, the number and types of other minutiae within a given radius is recorded. This information can be used to find potential matches in another minutiae set when the local structure is distinctive. Figure 13 illustrates an example of the local structure of a minutiae. Within the given radius, there are three other minutiae. Additional information such as the angels between the minutiae can also be stored.

Chen and Kuo have developed a fingerprint matching algorithm that is based on local structural features [38]. In addition to the number of minutiae in the local neighbourhood, the number of ridges between them and relative orientations are also recorded. This information is used to find candidate minutiae from each set that are potential matches. Starting from a minutia in one set, a tree is grown in a breadth-first fashion with minutiae as the tree nodes. After an edge is added, the other minutiae set is examined to see if a similar edge can be added. Following this procedure a tree is grown for both minutiae sets until the tree is sufficiently large or no more edges can be added to both sets. If the process fails, it is repeated for the other candidate minutiae pairs. This algorithm is potentially very powerful. Like other graph-based matching algorithms, it is invariant to translations and rotations. It also has the advantage that it can match prints even if they only have a small area of overlap. Furthermore, if appropriate data structures are used for storing the structural information, it can be implemented very efficiently. Finally, this representation can handle nonlinear distortions very well because it can accommodate large global distortions while keeping local distortions small. However, to ensure that large global errors are not accumulated in the wrong

direction, the algorithm should be modified with a verification stage. The main drawback of this approach is that it does not handle missing and spurious minutiae very well. In the likely event that a real minutiae is not present in one of the sets, the algorithm may fail. At the expense of increased computational complexity, the algorithm could be modified to be more tolerant of missing/spurious minutiae, creating a powerful matching algorithm.

Several other approaches based on local structural features can be found in the literature [26, 25, 74, 75, 76]. In general, these algorithms use local structures for finding an initial alignment, and global features are used to refine the alignment and to calculate the matching score. Incorporating both local and global features is a useful tool for dealing with the minutiae matching problem.

Incorporating supplementary fingerprint information

Due to the large number of possible translations, rotations and distortions, aligning fingerprints is a computationally expensive problem. One approach used to deal with this complexity is to exploit local structural information to find potential minutiae matches (see the section Structural matching). Other algorithms incorporate different information from the fingerprint in order to reduce the degrees of freedom of the alignment.

Jain et al. use ridge information as an aid for alignment [13, 31]. When extracting the minutiae, the shape and location of its associated ridge is also stored. For each possible minutiae pair (one from each point set), if the associated ridges are similar the minutiae sets are translated so that the candidate minutiae pair are at the same place, and then rotated so that the associated ridges are aligned. This provides a coarse alignment of the minutiae sets, and is illustrated in Fig. 14. Next, a string representation of both minutiae sets is constructed based on the positions of the minutiae. Each character in the string corresponds to a single minutia, and they are ordered by increasing polar coordinates (the candidate minutiae pair are defined to be at the origin of the coordinate system). A dynamic-programming string matching algorithm is used to define an "edit distance" between the strings corresponding to the test minutiae set and reference minutiae set. Since we can not expect a one-to-one correspondence of the minutiae points, the string matching algorithm allows for insertions and deletions of characters. The string with the minimum edit distance from the reference string is considered to correspond to the most likely alignment of the point sets. Finally, a matching score is calculated for that alignment. Since the fingerprint distortions are not modelled, it is necessary to allow for some tolerance when matching minutiae pairs, especially for those pairs far away from the alignment origin. This is accomplished by using bounding boxes whose sizes grow as the distance from the reference minutiae increases. Minor modifications of this matching algorithm have been suggested by other researchers [77, 78]. This is a powerful matching algorithm, but there are a few drawbacks. One weakness is that it has a high computational complexity because many potential alignments are considered. Another potential problem is that local deformations will distort the ridge shapes. Since the ridges are used to align the point sets, ridge distortions will result in a small local alignment error, which will be amplified for distant minutiae pairs. If this is the case, an alignment may be rejected even if it is based on a true correspondence. This could possibly be addressed by using other minutiae pairs to refine the alignment.

Another system that uses additional features to supplement minutiae information was developed by Germain et al. [79]. Originally developed as a method to index large fingerprint databases, it employs a clever method to align minutiae sets. For each fingerprint in the database, keys are generated based on minutiae triplets. Given three minutiae, a rotation and translation invariant key is constructed using the distance between minutiae pairs, ridge-counts between minutiae pairs and minutiae orientation angles. For a query fingerprint, the same keys are generated and used to find similar minutiae triplets in the fingerprints database. For each pair of similar keys, the rigid transformation parameters needed to align them are calculated, and evidence for these transformations is accumulated. This process is known

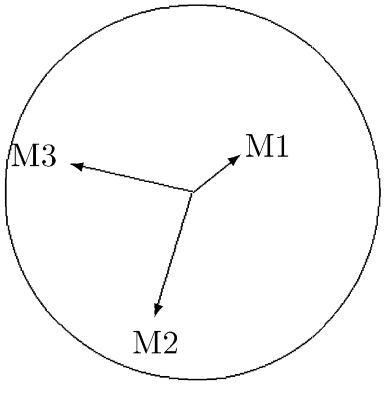


Fig. 13 Local structural relationships of a minutia within a given radius

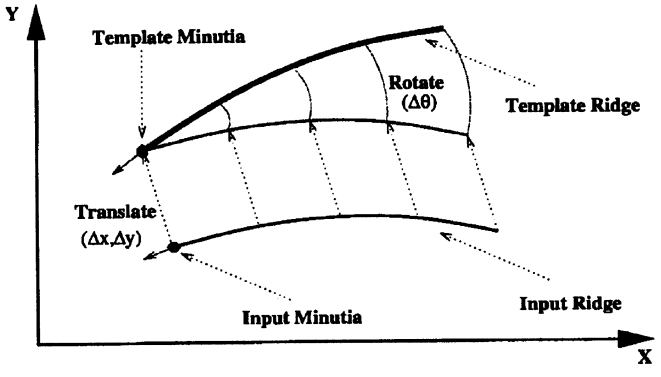


Fig. 14 An alignment using ridges [13]

as transformation parameter clustering and is similar to the generalised Hough transform discussed previously. The novelty of this approach is in the way it generates keys based on minutiae triplets by incorporating supplementary fingerprint information, and how these are used for database queries. Human experts often use ridge counts between neighbouring minutiae as a feature for identification. However, ridge counts are rarely used in automated systems. They are known to have a high discriminating power, yet may be difficult to reliably extract in the presence of noise.

A fingerprint verification system presented by Kovacs-Vanja uses the pixel intensities from the local area of a minutia to help find minutiae matches [80]. Minutiae regions from the test and reference images are compared to find potential correspondences. The main drawback of this approach is that the pixel values around a minutiae point can vary for several reasons. First of all, the grey levels and contrast will vary due to the different conditions under which the prints were taken. This can be mostly accounted for by normalising the pixel values to a standard mean and variance. Secondly, the region may be distorted. However, since only a small area is under consideration, this should have a relatively minor effect. Thirdly, the relative rotation of the regions is unknown and must be accounted for. Kovacs-Vajna rotates one of the regions at increments of 5°, and compares the regions at each increment. This approach requires several comparisons, none of which are guaranteed to be correct (the rotation error can still be as high as 2.5°). A better approach would be to align the regions based on the local ridge orientations. Although this would require additional computation to compute the local orientation around each minutiae, some computation would be saved because only one comparison is necessary. Furthermore, it would be much more accurate and reliable. A final problem with comparing pixel values is the presence of random noise. Careful preprocessing should help minimise this source of error. However, considering all of these possible sources of discrepancies, comparing local pixel intensities does not appear to be a very robust approach to finding similar minutiae. After some minutiae correspondences are found, Kovacs-Vanja's verification system proceeds by using triangular matching. The proposed triangular matching algorithm has the advantage that it allows for small local deformations that can be accumulated to account for large global deformations. A final verification stage is used to validate the tentative alignment found by the triangular matching. To accomplish this, additional fingerprint information is incorporated into the system. Kovacs-Vanja uses dynamic time warping to compare the ridge-valley structures between minutiae. These ridge-valley structures are similar to ridge counts, and are a powerful discriminating feature.

In the section Fingerprint structure the concept of a fingerprint singularity is defined. Figure 4 contains an example of a core point. All fingerprint classes have at least one core point except for a plain arch (see Fig. 4). Since core points are common, they can be used as aid for fingerprint alignment. Zhang and Wang have explored this possibility [81]. First the core points from the two fingerprint images are detected using a multi-resolution algorithm, and are then used as landmarks for registration. Basically, if a corresponding core point from each image can be found, it determines the translation parameter. Structural features of minutiae close to the core point are computed and are used to calculate the rotation parameter. This is followed by a global matching stage to calculate a matching score. This is a powerful and fast approach to fingerprint alignment, assuming that core points can be reliably extracted from fingerprint images. In some cases (such as plain arches) there will be no existing core point, so the definition of a core point must be broadened to include a well-defined position in plain arches. Furthermore, core extraction algorithms are less accurate when applied to low-quality fingerprints. A more serious problem with this approach occurs when the capture surface is small and does not include the true core point at all. Therefore, a robust fingerprint matching system should not rely solely on core points for fingerprint registration. In other words, alternate methods of registration should be available when core points can not be detected with a certain degree of confidence.

Modelling fingerprint distortions

The alignment methods discussed so far have all been rigid in the sense that the same transformation parameters are applied to all of the minutiae. However, it is well known that fingerprints are deformed nonlinearly when pressed against a surface. Therefore, these alignment algorithms find an alignment that minimises errors, as opposed to an alignment with no errors. Since it is known that the alignment is not perfect, it is necessary to allow for some displacement of the minutiae during the matching stage. This is often accomplished using bounding boxes. The disadvantage of this approach is that large bounding boxes are necessary when large distortions are present, increasing the probability of a false match. An alternate approach to fingerprint matching is to explicitly model the nonlinear distortion after registration. If this is done with a high degree of accuracy, the bounding box can be much smaller, decreasing both false rejects and false accepts [82]. For example, Cappelli et al. have developed a model of the elastic distortion in fingerprints that could be incorporated into a minutiae matching algorithm [83].

Almansa and Cohen propose the use of a thin-plate spline (TPS) for modelling fingerprint distortion [84]. Thin-plate splines transform coordinates in a way that minimises the "bending energy" of the transformation [85]. Almansa and Cohen present a two-step iterative minimisation algorithm for elastic matching.

Another application of thin-plate spline models to fingerprint matching is suggested by Bazen and Gerez

[82]. Their algorithm first uses local structures (see the section Structural matching) to find possible minutiae matches from the two fingerprints. These possible matches are used to find a rigid transformation that produces a global alignment consistent with a large number of minutiae. However, due to elastic distortions, it is possible that true minutiae matches are not aligned close to each other. The thin-plate spline model is used to represent the fingerprint deformation. The initial landmarks for modelling the distortion are local minutiae that have a high degree of correspondence. Several iterations are used to refine the model, incorporating new landmark minutiae that have become sufficiently close together. These iterations are continued until the model converges to its final state. Finally, a matching score is calculated based on the number of matching minutiae within a tight bounding box. This is a very powerful fingerprint matching algorithm that has produced impressive results (see Fig. 15). The authors note that the results may possibly be improved further by using more sophisticated minutiae matching algorithms. However, it is unclear how well the algorithm will perform when fingerprint images are of poor quality or share a relatively small area of overlap.

Although the fingerprint distortion is not explicitly modelled, a procedure to reduce its effect is suggested by Lee et al. [86]. Their system is based on the assumption that the average ridge frequency is close to constant throughout a non-distorted fingerprint. Therefore, the average ridge frequency can be used to normalise the distance between two minutiae in a distorted area of a captured print. After normalising the images, the displacement between corresponding minutiae should be small. This approach has potential, but a more careful study of the effect of nonlinear distortions to local frequencies compared to the global average frequency is necessary.

Hao et al. use error propagation to account for fingerprint distortions during matching [87]. An initial alignment is found using ridge information and a Hough transform. Based on this alignment, a set of minutiae is determined that have a high degree of correspondence. Using minutiae pairs from this matched set, local distortions are estimated and used to find new minutiae pairs in the local neighbourhood. Several iterations of this process are executed until no new matches are found. One clever step in this algorithm is to verify that the displacement errors of neighbouring minutiae are consistent. For example, if two close minutiae are ''stretched'' in opposite directions, it is unlikely that they are true matches. This is a useful post processing step that could be incorporated into any algorithm that does not explicitly model fingerprint distortions.

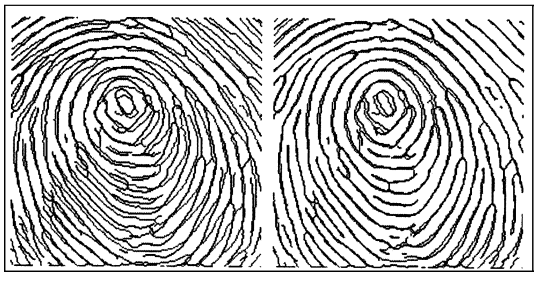


Fig. 15 A ridge map alignment before and after the nonlinear distortion is modeled using thin-plate splines [82]

Alternate matching techniques

Fingerprint matching algorithms have been proposed that are not based on directly comparing minutiae sets. Obviously, systems based on non-minutiae features (see the section Non-minutiae features) rely on alternate methods to compare prints. However, even systems using minutiae features may not base the matching procedure on pairing minutiae from each set. For example, algorithms based on machine learning or statistical pattern recognition approaches have been suggested. Willis and Myers have developed a fingerprint feature vector based on minutiae counts and frequency information [18]. Identification is accomplished by ''classifying'' fingerprints using traditional pattern classification techniques. In this case, there is one ''class'' for each individual in the database, and the classifier must assign unlabelled samples one of these classes. Willis and Myers evaluate the performance of eight different neural network and statistical classifiers. It is difficult to assess their results because very few details about the experiments are provided (e.g., image quality, database size, evaluation measures, etc.). However, it is clear that this approach will only be feasible for small fingerprint databases.

The application of neural networks has been investigated by several other researchers as well. Quek et al. trained a fuzzy neural network to detect similarities between two fingerprint based on minutiae and directional features [88]. Other systems using neural network classifiers can be found in the literature [50, 51, 59, 60]. In general, the performance of systems based on neural network classifiers is not as high as those based on minutiae matching techniques. One of the reasons for this is that they generally rely at least partially on features that do not have as high a discriminating power as minutiae. Furthermore, the processing time necessary to train (and possibly retrain) neural networks can make them infeasible for real-time systems. Fingerprint verification is a highly specialised problem, and most traditional all-purpose classifiers perform poorly compared to domain specific matching algorithms designed specifically for the problem.

Performance

Minutiae extraction

The reliability of the minutiae extraction stage for an AFISs is crucial to the system's overall performance. In

other words, the verification stage can only be as robust as its underlying minutiae data. However, despite its importance, there is a distinct lack of empirical data on the performance of minutiae extraction algorithms. This is partly due to the difficulty of obtaining labelled test data. Hand-labelling minutiae in fingerprint images is a tedious and time-consuming process that should be done by someone with expertise in the field. Therefore, even when empirical results are provided, they are usually based on very small sample sets. Also, it is difficult to directly compare results based on different datasets. This is because an algorithm's performance is closely related to image quality, which varies widely between test sets.

There is no standard metric for evaluating minutiae extraction results. Some papers simply count the number of missing and spurious minutiae, while others use a more complicated evaluation including other information such as image quality [39]. In order to ease comparisons between algorithms, results are reported using *correctness* and *completeness* values. Correctness is the percentage of extracted minutiae that are true minutiae and completeness is the percentage of true minutia that are extracted. Results for several algorithms are presented in Table 1. In most cases the average correctness and completeness values were calculated from the raw data. It should also be noted that minutiae type are not taken into consideration.

Maio and Maltoni have conducted a series of experiments comparing the skeleton approach to the grey scale approach [42]. They implemented four different extraction algorithms based on binarisation and thinning and compared the performance to their method of tracing ridges in grey scale images (see the section Minutiae extraction from grey scale images). The test set was 14 images from a variety of sources. Unlike the results in Table 1, these algorithms were tested on the same data, so a fair comparison is possible. The results show that a similar number of true minutiae were missed by both approaches; however, the grey scale method extracted far fewer spurious minutiae. Furthermore, the authors also found that the average computation time of their method was less than that of other methods. These results show that grey scale techniques are powerful from both an accuracy and efficiency point of view.

Due to problems obtaining labelled testing data, some minutiae extraction algorithms use alternate means of evaluation. For example, Farina et al. tested their algorithm using 500 images from the NIST special database 4 [37]. They report the reduction in the number of minutiae after various post processing steps. On average, 645 minutiae are extracted from binary skeletons if no filtering techniques are used. This number is reduced to 70 after validating the candidate minutiae. These are promising results because their testing database is large and contains many low quality images. However, it is not known how many of these remaining minutiae are spurious, and how many minutiae have not been detected. Jiang et al. evaluate their minutiae extraction algorithm indirectly by observing changes in accuracy at the verification stage [43].

The section Minutiae verification deals with the topic of minutiae verification. Minutiae verification algorithms are evaluated by the number of false minutiae that are eliminated. Maio and Maltoni used a neural network to filter minutiae, and conducted tests on 30 labelled images [47]. They report that the total extraction error is reduced from 49.1% to 35.6%, which is a significant reduction. Xiao and Raafat tested their filtering techniques on 10 sample fingerprints [40]. Initially, 46% of the extracted minutiae were spurious. However, this was reduced to only 4% after a minutiae classification stage. These results prove that minutiae verification is effective, and should be incorporated in high performance AFISs.

Fingerprint verification

There is an unfortunate lack of high-quality, accurate data published on the performance of fingerprint verification algorithms. This is mostly due to the obstacles involved in comparing fingerprint verification algorithm results. Regrettably, it is common for researchers to test their systems using non-standard datasets. First of all, different datasets contain images of different quality, and a system's verification accuracy is closely related to the input quality. Therefore, it is unfair to compare results obtained from different datasets if confidence intervals are not provided. Furthermore, the datasets vary widely in size from a few dozen images to ten of thousands of images. This is problematic because the fingerprint identification problem is more difficult for larger fingerprint databases. Another consideration is the resource constraints of a system. A system designed for off-line verification can use large amounts of memory and unbounded computation time, whereas an online system has much tighter time and memory restrictions. Once again, this factor should be taken into account when comparing the accuracy of matching algorithms.

Table 1 The performance of minutiae extraction algorithms

Author and year	Extraction approach	Correctness	Completeness	Images
Ratha et al. (1995) [39]	Binary skeleton	66%	81%	100
Maio and Maltoni (1997) [42]	Ridge line following in greyscale	93%	96%	14
Sagar and Koh (1999) [46]	Fuzzy logic and NN in greyscale	20%	42%	1
Liu et al. (2000) [44]	Ridge/valley tracing in greyscale	92%	97%	10

Another issue to consider when comparing algorithms is the criteria being used for declaring a match. For example, when matching is based on minutiae correspondences, one system might require 12 matching pairs for a positive verification, while another might require 16. Obviously, it would not be fair to directly compare the results from these systems. This can be addressed by evaluating systems with ROC curves. An ROC curve plots the relationship between a system's false match rate (FMR) and false non-match rate (FNMR) as the matching score threshold is varied. One point of interest is the point where FMR = FNMR. This is known as the equal error rate (EER) and can be used to summarise an algorithm's performance. However, it should be noted that the EER does not necessarily give a good indication of a system's performance in practice. For high security applications in particular, false non-matches are not as serious as false matches. False non-matches are inconvenient, but false matches could potentially lead to security breaches. In these cases, the matching threshold would be set to ensure a low FMG, but tolerate a higher FNMR.

There are two main publicly available fingerprint databases: the NIST special databases and the fingerprint verification competition databases.

NIST special databases

The National Institute of Standards and Technology (NIST) has created several large fingerprint databases, including special databases 4, 9, 14, 24, 27, 29 and 30 [89]. These databases are publicly available and are sometimes used to evaluate fingerprint matching algorithms. Most of the databases contain scanned images of prints acquired using ink. Several obstacles to the direct comparison of verification algorithms are mentioned above, and these are not adequately addressed by using the NIST databases. The databases are typically very large (special database 9 contains 13,500 fingerprint image pairs), and so testing is only conducted on subsets. These subsets are often handpicked based on image quality. Therefore, the size and image quality of the datasets can vary. Furthermore, when left to their own devices, researchers do not always calculate an ROC curve, instead presenting FMR/FNMR pairs for only a few threshold values. Finally, the experiments are conducted in a wide range of environments, so computational comparisons are also difficult. Since they can be misleading, results published using NIST databases will not be discussed. However, it should be noted that the NIST databases are still a valuable aid for designing and testing new algorithms. In particular, the databases are useful test sets for large-scale AFISs.

Fingerprint verification competition

In 2000 the Fingerprint Verification Competition (FVC2000) was established to address the need for a fair and unbiased way to compare the results from state-of-the-art fingerprint matching algorithms [90]. Originally a competition, the FVC2000 also makes its fingerprint databases publicly available as a benchmark for new algorithms to publish their performance results. The FVC addresses all of the issues discussed above concerning the comparison of fingerprint verification algorithms. Most importantly, ROC curves are calculated for all algorithms using the exact same datasets. Furthermore, the algorithms are executed in the same computational environment, and the computation times are recorded. Unlike the NIST databases, the focus of the FVC is biometric applications of fingerprint identification systems. Due to the success of the first competition in 2000, another was held in 2002, and at the time of this writing, participants are registering for the 2004 competition.

For the 2000 competition, four databases were collected, each containing 880 fingerprints from 110 different fingers. 80 images were made available before the competition for training purposes, and 800 were used for testing. 800 fingerprints is a reasonably database size that one might expect for a midsize biometric system. Three of the databases were obtained using fingerprint scanners, and one was synthetically generated. There were 11 participants in the competition, and the results varied widely. The average EER over all participants and all databases was around 14%. The top result was an EER of 1.73% from the French company SAGEM, a leading provider of AFIS technology to forensic departments worldwide[6]. The top academic entry was third place, submitted by researchers from the Centre for Signal Processing at Nanyang Technological University in Singapore. Their algorithm achieved an average EER of 5.19% and is based on minutiae features [43, 74].

In 2002, four new databases were acquired with the same sizes as the originals. This time 31 matching algorithms were tested, and the results were much better with the average EER being around 7%. The top EER was 0.19%, an impressive results achieved by Bioscrypt from the USA. Bioscrypt is a company that produces high security biometric access systems. The top academic result was an ERR of 3.76% from the Biometrics System Lab of the Beijing University of Posts and Telecommunications.

The biggest drawback of the FVC competitions is that most systems are commercial and do not publish their algorithms. For example, of the 31 participants in 2002, only six were academic. Private companies invest huge resources into algorithm development, and are understandably reluctant to share their techniques. Unfortunately for academia, this means that little information is gained about the most successful approaches to fingerprint verification. Of particular relevance to this paper, it is not known how heavily the top algorithms rely on minutiae features. However, the

[6] Australian police use an AFIS developed by SAGEM

competition does give a very good indication of the performance of state-of-the-art algorithms, which is invaluable for all researchers in the field.

After each FVC competition, the databases are made available for testing new systems. He et al. have developed a verification algorithm based on minutiae matching [78] that is a modified version of the system presented by Jain et al. [31] (see the section Incorporating supplementary fingerprint information). This approach uses ridge information for aligning the minutiae sets. When tested on all four of the FVC2000 databases, an average EER of 5.14% was obtained. This is a decent result in comparison with the other FVC2000 participants, and is likely representative of results from other state-of-the-art academic systems based on minutiae matching. Bazen and Gerez use thin-plate splines to model the nonlinear deformations of fingerprints (see the section Modelling fingerprint distortion). They have evaluated their system in database 2 from FVC2000 and obtained an EER of 6%.

Unfortunately, the number of papers publishing results using the FVC datasets is still rather limited. Since the FVC databases are currently the best source of high-quality test sets, it is beneficial for the entire research community for new algorithms (both commercial and academic) to be evaluated using the FVC standard.

Conclusions

Automated fingerprint matching has been studied by the pattern recognition community for several decades, yet the performance of state-of-the-art algorithms is still very poor when compared to theoretical upper bounds [3]. In fact, most modern academic systems struggle to obtain high recognition accuracies for databases containing only a few hundred prints.

There are several areas where future research in the field should be focused. Minutiae extraction is a crucial stage in most matching algorithms, yet current extraction algorithms produce a considerable amount of missing and spurious minutiae. Minutiae-extraction from grey scale images has achieved results comparable to skeleton-based algorithms despite only a fraction of the attention. Further investigation into grey scale techniques may yield algorithms superior to any currently proposed in the literature. Minutiae verification algorithms have also proved to be a useful post processing stage, so further study is encouraged. For example, there is some potential for using grey scale ridge tracing techniques to verify minutiae extracted from binary skeletons, and this is a possibility has yet to be explored.

The focus of this review has been on minutiae features for matching. However, several shortcomings of these features have been identified. Future verification algorithms should incorporate more non-minutiae features to supplement the minutiae information. This will be particularly beneficial when the images are of poor quality. However, it should be noted that large-scale systems will continue to rely heavily on minutiae information due to its high discriminating power. Hybrid systems incorporating several independent feature sets can be used to increase robustness, and this approach will likely become more common. Similarly, systems that combine the results from independent classifiers have great potential [64].

Accurately modelling nonlinear fingerprint distortions is an area of fingerprint research that has the potential to produce very high-performance fingerprint matching algorithms, regardless of the features being used for matching. Given high-quality fingerprint alignment and distortion parameters, the matching problem becomes much easier and more accurate. This is an area that has not yet been extensively explored, so it would benefit from further investigation.

Recently identification based on DNA has become a very high-profile type of physical evidence in criminal trials. However, DNA testing is expensive, time-consuming and laborious. On the contrary, fingerprint-based identification is relatively inexpensive and fast. Furthermore, biometric systems will likely become ubiquitous within the coming years. It is estimated that biometrics will be a multi-billion dollar (US) industry worldwide by 2005, and fingerprints are emerging as the preferred biometric for identification. Considering both the immense interest in the field and the numerous opportunities for innovations and advancements, automated fingerprint verification continues to be an important and exciting area of research.

Originality and contributions

The main contribution of this article is in its presentation of the state of the art in automated fingerprint recognition. Fingerprint recognition is currently a hot research topic in the image analysis and pattern recognition communities. However, despite increasing attention from both private and academic institutions, practitioners in this field still face many challenges. This article surveys the research that has been conducted into the problem over the last few decades, identifying the strategies that have proven to be successful, and those that have not. It also conducts an analysis of current approaches, and indicates important future directions in the field. The focus of the article is on minutiae-based techniques, which have traditionally been the most widely used fingerprint feature. The accuracy and efficiency of minutiae based methods are discussed thoroughly, and there is a particular emphasis on the limitations of this approach. To our knowledge, this is the first survey to approach the topic from this angle. The review will be valuable for readers with no background in fingerprint recognition, and those wishing to acquaint themselves with the latest developments in the field.

About the authors

Adnan AMIN presented his Doctorate D'Etat (D. Sc.) in Computer Science at the University of Nancy, France, in 1985. From 1981 to 1985, Dr. Amin was Maitre Assistant (Assistant Professor) at the University of Nancy II. Between 1985 and 1987 he

worked at INTEGRO (Paris) as head of the Pattern Recognition Department. From 1987 to 1990, he was an Assistant Professor at Kuwait University and joined the School of Computer Science and Engineering at the University of New South Wales, Australia, in 1991 as a senior lecturer. Dr. Amin's research interests are pattern recognition, document image analysis and recognition, neural networks, and machine learning. He has authored more than 100 technical papers in these areas, and is an associate editor of the International Journal on Document Analysis and Recognition, the journal Pattern Analysis and Applications, the International Journal of Pattern Recognition and Artificial Intelligence, and the International Journal of Image and Graphics. He has been on the program committee of many conferences and served as a referee for numerous journals and scientific organisations. He is a member of the IEEE, ACM and IAPR societies.

Neil Yager was born in Sydney, Australia in 1978. He moved to Canada at a very early age, but has since returned to Australia. Neil is currently a PhD student in the School of Computer Science and Engineering at the University of New South Wales in Sydney. His primary research interests are in the fields of image analysis and pattern recognition. In particular, his emphasis is on biometrics and automated fingerprint identification. Neil obtained a BSc in Computer Science from the University of Victoria in Victoria, Canada in 2000, and a MEngSc from the University of New South Wales in 2002.

Acknowledgements The authors would like to thank Sergeant Russell Plummer of the New South Wales Police's Criminal Identification Specialist Support Branch for valuable information and a fascinating discussion on the use of latent fingerprints for criminal investigations in Australia. Furthermore, we thank the anonymous referees for valuable comments and suggestions.

References

1. Federal Bureau of Investigation (FBI) (2003) http://www.fbi.gov. Cited October 2003
2. Jain A, Hong L, Pankanti S (2000) Biometrics: promising frontiers for emerging identification market. Comm ACM Feb:91–98
3. Pankanti S, Prabhakar S, Jain A (2002) On the individuality of fingerprints. IEEE Trans Patt Anal Mach Intell 24(8):1010–1025
4. Beavan C (2002) Fingerprints, murder and the race to uncover the science of identity. Fourth Estate, London
5. Cummins H, Midlo C (1961) Finger prints, palms and soles, an introduction to dermatoglyphics. Dover Publications, Mineola, NY
6. Herschel W (1916) The origin of finger-printing. Oxford University Press, London
7. Faulds H (1880) On the skin-furrows of the hand. Nature 22(574):605
8. Galton F (1892) Finger prints. McMillan, London
9. Henry E (1900) Classification and uses of finger prints. Routledge, London
10. Jain A, Prabhakar A, Pankanti A (2002) On the similarity of identical twin fingerprints. Patt Recog 35(11):2653–2663
11. Wilson C, Candela G, Watson C (1993) Neural network fingerprint classification. J Artif Neur Ntwks 1(2):203–228
12. Trauring M (1963) Automatic comparison of finger-ridge patterns. Nature 197:938–940
13. Jain A, Hong L, Pankanti S, Bolle R (1997) An identity-authentication system using fingerprints. Proc IEEE 85(9):1365–1388
14. Federal Bureau of Investigation (FBI) (1993) WSQ gray-scale fingerprint image compression specification. Document IAFIS-IC-0110v2
15. Sherlock B, Monro D, Millard K (1994) Fingerprint enhancement by directional fourier filtering. IEEE Proc Vis Imag Sig Process 141(2):87–94
16. Kamei T, Mizoguchi M (1995) Image filter design for fingerprint enhancement. Proc ISCV 109–114
17. Candela G, Grother P, Watson C, Wilkinson R, Wilson C (1995) PCASYS—a pattern-level classification automation system for fingerprints. NISTIR 5647, National Institute of Standards and Technology, Gaithersburg, MD
18. Willis A, Myers L (2001) A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips. Patt Recog 34:255–270
19. Hong L, Wan Y, Jain A (1998) Fingerprint image enhancement: algorithm and performance evaluation. IEEE Trans Patt Anal Mach Intell 20:777–789
20. Saatci E, Tavsanoglu V (2002) Fingerprint image enhancement using CNN gabor-type filters. In: Proceedings of CNNA 2002, Frankfurt, Germany, 22–24 July 2002, pp 377–382
21. Kim B, Kim H, Park D (2002) New enhancement algorithm for fingerprint images. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 3:879–882
22. Zhang W, Wang Q, Tang Y (2002) A wavelet-based method for fingerprint image enhancement. Proc Mach Learn Cybern 4:1973–1977
23. Hsieh C, Lai E, Wang Y (2003) An effective algorithm for fingerprint image enhancement based on wavelet transform. Patt Recog 36:303–312
24. Greenberg S, Aladjem M, Kogan D, Dimitrov I (2000) Fingerprint image enhancement using filtering techniques. In: Proceedings of ICPR 2000, Barcelona, Spain, 3–8 August 2000, 3:322–325
25. Wahab A, Chin S, Tan E (1998) Novel approach to automated fingerprint recognition. IEEE Proc Vis Imag Sig Process 145(3):160–166
26. Mital D, Teoh E (1996) An automated matching technique for fingerprint identification. In: Proceedings of EFTA 1996 1:87–92
27. Ailisto A, Lindholm M, Tikkanen P (2003) A review of fingerprint image enhancement methods. Int J Imag Graph 3(3):401–424
28. Ratha NK, Connell JH, Bolle RM (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Sys J 40(3):614–634
29. Ratha NK, Connell JH, Bolle RM (2003) Biometrics break-ins and band-aids. Patt Recog Lett 24:2105–2113
30. Kass M, Witkin A (1987) Analyzing oriented patterns. Comp Vis Graph Imag Proc 37(3):362–385
31. Jain A, Hong L, Bolle R (1997) On-line fingerprint verification. IEEE Trans Patt Anal Mach Intell 19(4):302–314
32. Rao A (1990) A taxonomy for texture description and identification. Springer, Berlin Heidelberg New York
33. Stock RM, Swonger CW (1969) Development and evaluation of a reader of fingerprint minutiae. Cornell Aeronautical Laboratory 1969, Technical Report CAL No. XM-2478-X-1:13–17, Ithaca, NY
34. Chang J, Fan K (2001) Fingerprint ridge allocation in direct gray-scale domain. Patt Recog 34:1907–1925
35. Fitz AP, Green RJ (1996) Fingerprint classification using a hexagonal fast fourier transform. Patt Recog 29(10):1587–1597
36. Rao T (1976) Feature extraction for fingerprint classification. Patt Recog 8:181–192
37. Farina A, Vajna Z, Leone A (1999) Fingerprint minutiae extraction from skeletonized binary images. Patt Recog 32:877–889
38. Chen Z, Kuo C (1991) A topology-based matching algorithm for fingerprint authentication. In: Proceedings of the 25th Annual IEEE International Carnahan Conference on Security Technology, Taipei, Taiwan, 1–3 October 1991, pp 84–87
39. Ratha N, Chen S, Jain A (1995) Adaptive flow orientation based feature extraction in fingerprint images. Patt Recog 28:1657–1672

40. Xiao Q, Raafat H (1991) Fingerprint image postprocessing: a combined statistical and structural approach. Patt Recog 24(10):985–992
41. Hung D (1993) Enhancement and feature purification of fingerprint images. Patt Recog 26(11):1661–1671
42. Maio D, Maltoni D (1997) Direct gray-scale minutiae detection in fingerprints. IEEE Trans Patt Anal Mach Intell 19(1):27–40
43. Jiang X, Yau W, Ser W (2001) Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge. Patt Recog 34:999–1013
44. Liu J, Huang Z, Chan K (2000) Direct minutiae extraction from gray-level fingerprint image by relationship examination. Proc ICIP 2000 2:427–430
45. Sagar V, Ngo D, Foo K (1995) Fuzzy control for fingerprint feature selection. Proc ACCV 1995 3:767–771
46. Sagar V, Koh A (1999) Hybrid fuzzy logic and neural network model for fingerprint minutiae extraction. Proc IJCNN 1999 5:3255–3259
47. Maio D, Maltoni D (1998) Neural network based minutiae filtering in fingerprints. Proc ICPR 1998 2:1654–1658
48. Prabhakar S, Jain A, Pankanti S (2003) Learning fingerprint minutiae location and type. Patt Recog 36(8):1847–1857
49. Hatano T, Adachi T, Shigematsu S, Morimura H, Onishi S, Okazaki Y, Kyuragi H (2002) A fingerprint verification algorithm using the differential matching rate. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 3:799–802
50. Ceguerra A, Koprinska I (2002) Integrating local and global features in automatic fingerprint verification. In: Proc ICPR 2002, Quebec, Canada, 11–15 August 2002, 3:347–350
51. Wang S, Lee C (1999) Fingerprint recognition using directional micropattern histograms and LVQ networks. Proc Info Intell Sys 1999, pp 300–303
52. Lee W, Chung J (1997) Fingerprint recognition algorithm development using direction information in wavelet transform domain. In: Proceedings of the IEEE International Symposium on Circuits and Systems, Hong Kong, China, June 1997, pp 1201–1204
53. Lee S, Nam B (1999) Fingerprint recognition using wavelet transform and probabilistic neural network. Proc IJCNN 1999 5:3276–3279
54. Tico M, Immonen E, Ramo P, Kuosmanen P, Saarinen J (2001) Fingerprint recognition using wavelet features. Proc ISCAS 2001 2:21–24
55. Jain A, Prabhakar S, Hong L, Pankanti S (2000) Filterbank-based fingerprint matching. IEEE Trans Imag Proc 9(5):846–859
56. Lee C, Wang S (2001) Fingerprint feature reduction by principal gabor basis function. Patt Recog 34:2245–2248
57. Ross A, Reisman J, Jain A (2002) Fingerprint matching using feature space correlation. In: Proceedings of the Post-ECCV Workshop in Biometric Authentication 2002, Lecture Notes in Computer Science, vol 2359, pp 48–57, Springer, Berlin Heidelberg New York
58. Seow B, Yeoh S, Lai S, Abu N (2002) Image based fingerprint verification. In: Proceedings of the 2002 Student Conference on Research and Development, Shah Alam, Malaysia, May 2002, pp 58–61
59. Jin A, Chekima A, Dargham J, Fan L (2002) Fingerprint identification and recognition using backpropagation neural network. In: Proceedings of the 2002 Student Conference on Research and Development, Shah Alam, Malaysia, May 2002, pp 98–101
60. Sujan V, Mulqueen M (2002) Fingerprint identification using space invariant transforms. Patt Recog Lett 23:609-619
61. Soifer V, Kotlyar V, Khonina S, Skidanov R (1996) Fingerprint identificaion using the directions field. Proc ICPR 1996 3:586–590
62. Wilson C, Watson C, Paek E (2000) Effect of resolution and image quality on combined optical and neural network fingerprint matching. Patt Recog 33:317–331
63. Alam M, Akhteruzzaman M (2000) Real time fingerprint identification. In: Proceedings of NAECON 2000, Dayton, OH, 10–12 October 2000, pp 434–440
64. Prabhakar S, Jain A (2001) Decision-level fusion in fingerprint verification. Patt Recog 35:861–874
65. Lumini A, Maio D, Maltoni D (1997) Continuous versus exclusive classification for fingerprint retrieval. Patt Recog Lett 18:1027–1034
66. Cappelli R, Lumini A, Maio D, Maltoni D (1999) Fingerprint classification by directional image partitioning. IEEE Trans Patt Anal Mach Intell 21(5):402–421
67. Bebis G, Deaconu T, Georgiopoulous M (1999) Fingerprint identification using delaunay triangularion. In: Proceedings of Information Intelligence and Systems, 31 October–3 November 1999, Bethesda, MD, pp 452–459
68. Bhanu B, Tan X (2003) Fingerprint indexing based on novel features of minutiae triplets. IEEE Trans Patt Anal Mach Intell 25(5):616–622
69. Bolle R, Connell J, Ratha N (2002) Biometric perils and patches. Patt Recog 35:2727–2738
70. Ratha N, Karu K, Chen S, Jain A (1996) A real-time matching system for large fingerprint databases. IEEE Trans Patt Anal Mach Intell 18(8):799–813
71. Isenor D, Zaky S (1986) Fingerprint identification using graph matching. Patt Recog 19(2):113–122
72. Fan K, Liu C, Wang Y (2000) A randomized approach with geometric constraints to fingerprint verification. Patt Recog 33:1793–1803
73. Hrechak A, McHugh J (1990) Automated fingerprint recognition using structural matching. Patt Recog 23(8):893–904
74. Jiang X, Yau W (2000) Fingerprint minutiae matching based on the local and global structures. In: Proceedings of ICPR 2000, Barcelona, Spain, 3–8 August 2000 2:1038–1041
75. Ratha N, Pandit V, Bolle R, Vaish V (2000) Robust fingerprint authentication using local structural similarity. In: Proceedings of the Fifth IEEE Workshop on Applications of Computer Vision, Palm Springs, CA, 4–6 December 2000, pp 29–34
76. Qun R, Jie T, Yuliang H, Jiangang C (2002) Automatic fingerprint identification using cluster algorithm. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 2:398–401
77. Luo X, Tian J, Wu Y (2000) A minutia matching algorithm in fingerprint verification. In: Proceedings of the IEEE International Conference on Pattern Recognition, 4:4833–4836
78. He Y, Tian J, Luo X, Zhang T (2003) Image enhancement and minutiae matching in fingerprint verification. Patt Recog Lett 24:1349–1360
79. Garmain RS, Califano A, Colville S (1997) Fingerprint matching using transformation parameter clustering. IEEE Comput Sci Eng 4(4):42–49
80. Kovacs-Vajna Z (2000) A fingerprint verification system based on triangular matching and dynamic time warping. IEEE Trans Patt Anal Mach Intell 22(11):1266–2000
81. Zhang W, Wang Y (2002) Core-based structure matching algorithm of fingerprint verification. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 1:70–74
82. Bazen A, Gerez S (2003) Fingerprint matching by thin-plate spline modelling deformations. Patt Recog 36:1859–1867
83. Cappelli R, Maio D, Maltoni D. Modeling plastic distortion in fingerprint images. In: Proceedings of ICAPR 2001, Rio de Janeiro, Brazil, March 2000
84. Almansa A, Cohen L (2000) Fingerprint matching by minimization of a thin-plate energy using a two-step algorithm with auxiliary variables. In: Proceedings of the IEEE Workshop on Applications of Computer Vision 2000, Palm Springs, CA, 4–6 December 2000, pp 35–40
85. Bookstein F (1989) Principal warps: thin-plate splines and the decomposition of deformations. IEEE Trans Patt Anal Mach Intell 11(6):567–585
86. Lee D, Choi K, Kim J (2002) A robust fingerprint matching algorithm using local alignment. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 3:803–806

87. Hao Y, Tan T, Wang Y (2002) Fingerprint matching based on error propagation. In: Proceedings of ICPR 2002, Quebec, Canada, 11–15 August 2002, 1:273–276
88. Quek C, Tan K, Sagar V (2001) Pseudo-outer product based fuzzy neural network fingerprint verification system. Neur Ntwks 2001 14:305–323
89. National Institute of Standards and Technology (NIST) (2003) Special databases. http://www.nist.gov/data/. Cited October 2003
90. Maio D, Maltoni D, Cappelli R, Wayman J, Jain A (2002) FVC2000: fingerprint verification competition. IEEE Trans Patt Anal Mach Intell 24(3):402–412