Pattern Anal Applic (2004) 7: 77–93
DOI 10.1007/s10044-004-0204-7

ORIGINAL ARTICLE

Neil Yager · Adnan Amin

Fingerprint classification: a review

Received: 24 April 2003 / Accepted: 2 February 2004 / Published online: 9 March 2004

Abstract Biometrics is the automatic identification of an individual that is based on physiological or behavioural characteristics. Due to its security-related applications and the current world political climate, biometrics is currently the subject of intense research by both 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 fingerprint classification over the last four decades. In particular, it discusses the fingerprint features that are useful for distinguishing fingerprint classes and reviews the methods of classification that have been applied to the problem. Finally, it presents empirical results from the state of the art fingerprint classification systems that have been tested using the NIST Special Database 4.

Keywords Biometrics · Fingerprint classification · Gabor filter · Orientation fields · Heuristic approach

Introduction

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 [1]. Due to its security-related applications and the current world political climate, biometrics has recently 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 [1]. For example, a person's face, retina, or voice can all be used to identify that individual with a high degree of accuracy. The use of fingerprints has several advantages over the other methods, and therefore is one of the most researched and mature fields of authentication. The uniqueness of fingerprints has been studied and it is well established that the probability of two fingerprints matching is vanishingly small [2, 3]. Furthermore, unlike faces and voice prints, fingerprints are persistent with age and can not be easily disguised.

Fingerprint matching and fingerprint classification are two conceptually separate problems. For the fingerprint matching problem the input is two fingerprint images and the output is the probability that the fingerprints were captured from the same finger. A comprehensive review of fingerprint matching algorithms has been recently conducted by Yager and Amin [4]. The current state of the art algorithms for fingerprint matching are computationally very expensive. Therefore, given the large size of present day fingerprint databases (the FBI fingerprint database contains over 200 million prints), it is often necessary to employ methods to reduce the number of one-to-one fingerprint comparisons performed when executing a fingerprint query. The most common way to do this is to partition the fingerprint databases into mutually exclusive classes (or bins). After classifying (or binning) the query fingerprint, it is only necessary to match that print against others of the same type. This drastically reduces the number of one-to-one matches performed.

Many schemes have been presented for the automatic classification of fingerprints over the last four decades and this paper presents the first extensive review of this research. The section History begins with an overview of the history of fingerprint classification. The section Fingerprint classes presents commonly used fingerprint

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

classes and the section Fingerprint classificatin challenges discusses some issues that make the fingerprint classification problem difficult. The section Feature extraction looks at the fingerprint features that are useful for classification and discusses some methods used for extracting them. The section Classification performance presents the techniques used for classification. Finally, the Conclusioncloses with some concluding remarks.

Fingerprint classification

History

The exact origin of the use of fingerprints for identification is unclear. There is some evidence that fingerprints were used in ancient times; however there is no indication that anyone at the time recognised the full potential of fingerprints as a means of personal identification. The first reliable record comes from Sir William Herschel [5]. 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. After noting that fingerprints taken from the same individual 32 years apart were unchanged, Herschel discovered the immutability of fingerprints. The persistence of fingerprints over time is vital to their applicability as a means of personal identification.

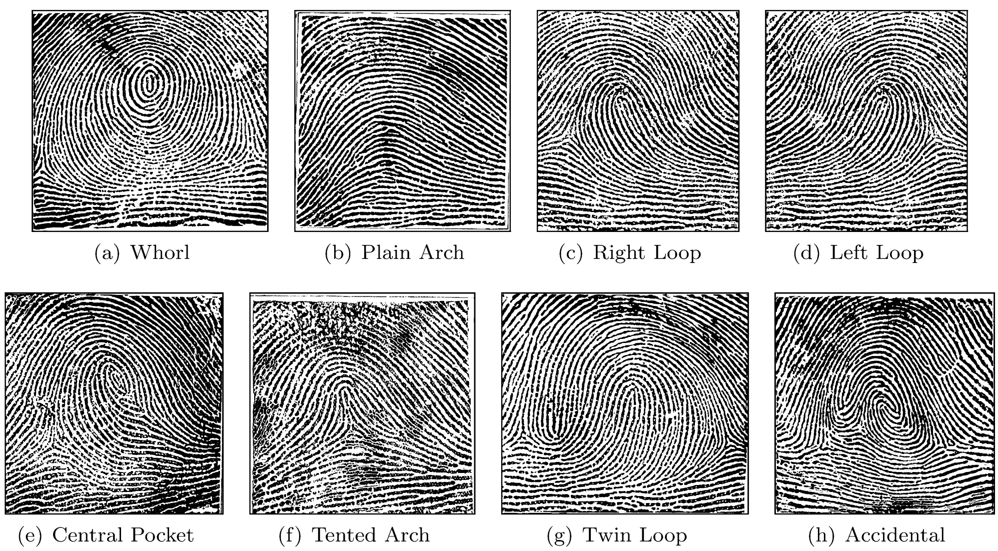
The first scientific publication proposing the use of fingerprints for identification was written by Henry Faulds in 1880 [6]. In the late 1880s Sir Francis Galton began the first rigorous study of fingerprint-based identification [7]. Among many contributions to the field, his work contained the first system for fingerprint classification. Classification was introduced as a means of indexing fingerprints in order to facilitate searching for a particular fingerprint within a collection of many prints. He proposed three basic fingerprint classes: the arch, the loop and the whorl. Galton's other major contribution was the first study into the uniqueness of fingerprints. In addition to permanence, uniqueness is the other necessity for fingerprints to be a viable method of personal identification.

Several years later Edward Henry continued Galton's work on fingerprint classification [8]. Henry subdivided the three main classes into more specific subclasses. He also introduced the concept of fingerprint "core" and "delta" points and used them as aids for fingerprint classification. Henry's classification scheme constitutes the basis for most modern classification schemes.

The first publication on the automation of fingerprint matching appeared in 1963 [9]. The subject received increased attention during the subsequent decades and continues to be an active area of research to this day.

Fingerprint classes

Fingerprints can be categorised based on their global pattern of ridges and valleys. Galton originally proposed the use of three fingerprint classes: the arch, the whorl and the loop [7]. Henry subdivided the arch category to include the plain arch and the tented arch. He also defined two types of loops: ulnar and radial, also known as right loops and left loops respectively. He also added classes that are combinations of arches, loops and whorls: central pocket loops, twin loops and accidentals. Henry used the accidental class to describe



(a) Whorl (b) Plain Arch (c) Right Loop (d) Left Loop

(e) Central Pocket (f) Tented Arch (g) Twin Loop (h) Accidental

Fig. 1 Henry's fingerprint classes

the small number of fingerprints that are ambiguous and do not fall clearly into any of the other categories. These eight categories are known as "Henry's Classification" and examples of each are shown in Fig. 1.

The distribution of the 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, 0.338, 0.317, 0.029 and 0.279 for the arch, left loop, right loop, tented arch and whorl, respectively [10]. Note that left loops, right loops, and whorls are the most common, making up 93.4% of all fingerprints.

Henry's classification (and others based on it) are known as *exclusive* because they partition fingerprints into mutually exclusive categories. Many large-scale fingerprint verification systems are designed to only match a print against others from the same class. This has several disadvantages [11]. First of all, there are some fingerprints that are ambiguous and can not be classified even by a human expert. In these cases it is unclear which fingerprint classes the ambiguous print should be matched against. Furthermore, there is always the possibility of misclassification due to noise or an error in the system. In these cases, if no match is found in the erroneous class the entire database will need to be searched. Finally, even assuming the classification is robust, this does not always significantly reduce the number of one-to-one matches. A large majority (almost 95%) of fingerprints fall into only three classes. When an individual is being identified using all ten of their fingerprints (known as ten-print based identification), knowing the class of all ten fingers greatly reduces the number of matches that are necessary. However, when only a single fingerprint is available (such as latent fingerprints) the print will almost certainly need to be matched against a large segment (usually about 1/3) of the database.

Continuous classification addresses some of the shortcomings of exclusive classification. Instead of being represented by a single class, fingerprints are represented by a feature vector containing important distinguishing characteristics of the print. The similarity of two fingerprints can be defined as the Euclidean distance between them in the feature space and query fingerprints are matched against all other prints that fall within a given radius. This immediately resolves the problem of which prints should be matched against an ambiguous fingerprint. Furthermore, the matching radius can be adjusted as a system parameter allowing the system operator to dynamically adjust the expected accuracy of the system. Relatively little research has been conducted into continuous fingerprint classification, and therefore it is not a focus of the present work. However, one continuous system is described in the section Graph matching. Fingerprint database indexing is a closely related problem to continuous classification [12, 13, 14].

Fingerprint classification challenges

There are several challenges to the classification of fingerprints that are specific the fingerprint domain. These are important issues that need to be addressed by any fingerprint classification system.

Prints from the same finger will be slightly different every time they are captured for several reasons, and classification systems must be designed to be robust when dealing with these variations. There are several causes for this variation. For example, every time a finger is pressed against a surface, it is applied with a certain amount of pressure at a well defined angle, and these parameters vary from time to time, resulting in a different portion of the print being captured. Consequently, fingerprint classifiers should not be sensitive to translations or rotations of fingerprints. Random noise and other effects caused by the skin conditions (e.g., dry, sweaty, dirty, diseased, etc.) can cause errors in the fingerprint image. It is important for classification systems to recover the original ridge patterns, and therefore preprocessing is usually conducted to enhance the fingerprint image.

Another major challenge is related to fingerprint class variation. The majority of classification schemes use 5 classes (see the section Fingerprint classes). However, there is a wide variety of possible patterns within each class. Furthermore, in some cases prints from one class can appear very similar to prints from another class. In other words, there is large intraclass variation and small interclass variation. This is the most significant factor that makes the fingerprint classification problem so difficult, and is one of the motivations for developing continuous classification schemes.

Ambiguous fingerprints are a related issue. In some cases, a fingerprint will have properties from more than one class. A fingerprint classification system must devise a method for dealing with these prints, such as having an "anomalies" class or rejecting them outright.

Feature extraction

The goal of feature extraction in a general pattern recognition system is to extract information from the input data that is useful for determining its category. In the fingerprint domain there are several features that have been found to be useful for this purpose.

The classification of a fingerprint is based on its global pattern of ridge and valleys. Therefore, features based directly on the fingerprint ridges are a natural choice. There are many different ways to extract and represent ridge information and these will be discussed in detail in the section Ridge features. Orientation fields are a convenient way to summarise the ridge-valley patterns of a fingerprint and are presented in the section Orientation fields. Another feature that is often used for distinguishing fingerprint classes is the existence and location of *singularities*. A discussion of singularities and singularity extraction is presented in the section Singularities. Finally, information on structural features can be found in the section Structural features.

For completeness, minutiae points should also be mentioned as an important fingerprint feature. Minutiae points are places where a ridge separates into two ridges (known as a bifurcation) or a ridge terminates (known as a ridge ending). The location and orientation of minutiae points is a very important distinguishing feature for fingerprint matching and a large number of fingerprint matching algorithms are based on comparing the minutiae points from two fingerprint images. Consequently, there has been a tremendous amount of effort put into developing minutiae extraction and matching algorithms [4]. However, minutiae features are not useful for fingerprint classification and will not be discussed.

Ridge features

Several methods have been developed in an effort to extract useful information from fingerprint ridges. Some of these methods examine the frequency and orientation of the ridges, while others develop mathematical models to represent the structure of the ridges.

Frequency

The ridges and valleys in a small area of a fingerprint have a well-defined frequency and orientation. Therefore, valuable information about a fingerprint can be obtained by employing various methods of frequency analysis.

Fourier transform: The Fourier transform is a fundamental tool for frequency analysis. In the context of digital image processing, all images can be represented fully in either the *spatial* domain or the *frequency* domain. In the spatial domain, the image is represented by the intensity of pixel values at every location in the image. In the frequency domain images are represented by their component frequencies.

In Fig. 2a small section of a fingerprint image is shown. Within this region, the ridges form a set of periodic and nearly parallel lines with a constant frequency and orientation. The Fourier transform of this region is shown in Fig. 2b. The bright spots in the Fourier transform represent the dominant frequencies.

Fitz and Green [16] use Fourier transform features as the basis of their classification scheme. They use a Hexagonal Fast Fourier transform to transform the fingerprint image into the frequency domain and use a "wedge-ring detector" to extract features. A wedge-ring detector partitions the frequency domain into non-overlapping regions as shown in Fig. 3.

The values in each segment are summed to give a feature for that region, and classification is then based on an eight element feature vector. The wedge-ring detector overcomes some of the invariance challenges discussed in the section Feature extraction. Translations in the spatial domain will not cause changes in the frequency domain, therefore this representation is translation invariant. It also offers some rotation invariance because the rings are summed along an angular variable. However, there are major limitations of this approach. The features extracted by the wedge-ring detector are strictly global and do not contain very much discriminatory information. For example, the frequencies of ridges and valleys is similar for all fingerprint images of the same resolution. Therefore, summing along the rings of the detector will likely give similar results for all input. Orientation information is important, but the wedge features are too simplistic to capture much useful information about the distribution of orientations that determine a fingerprint's class. Furthermore, the dominant orientations in a fingerprint image depends heavily on the area of the print that has been captured and has little relevance to the fingerprint's class.

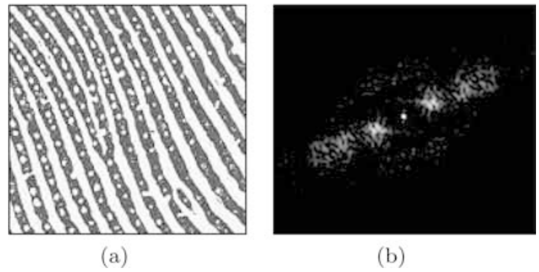


Fig. 2 a Fingerprint ridges;**b** The Fourier spectrum of a

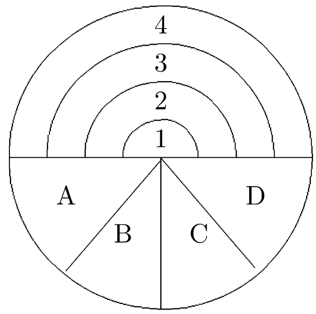


Fig. 3 A wedge-ring detector

Gabor filters: As shown in Fig. 2a, the ridges and valleys in a small region of a fingerprint contain a narrow range of frequency and orientation components. Gabor filters are band-pass filters with adjustable frequency, orientation, and bandwidth parameters. A Gabor function is a sinusoidal waveform that is modulated by a rotated Gaussian envelope, and has the following form in the spatial domain:

$$G(x, y; f, \theta, \sigma) = \exp\left\{-\frac{x''^2 + y''^2}{2\sigma^2}\right\} \cos(2\pi f x'') \qquad (1)$$

$$x'' = x\cos\theta - y\sin\theta \quad (2)$$

$$y'' = x\sin\theta + y\cos\theta \quad (3)$$

where f is the frequency of the sinusoidal plane wave along the direction θ and σ is the standard deviation of the Gaussian envelope [17].

Fig. 4a shows a Gabor filter in the spatial domain and Fig. 4b shows a Gabor filter in the frequency domain, which is simply a shifted Gaussian function.

When properly tuned, a Gabor filter can be used to filter an image, maintaining only regions of a given frequency and orientation. This has profound implications for fingerprint image analysis because the filters can be tuned to the inter-ridge frequency and local orientation. The primary application of this is for fingerprint image enhancement [18]; however, it can also be used for feature extraction.

Jain et al. use Gabor filters in four directions to extract features from fingerprints for classification [17]. The filter frequency f is set to the average ridge frequency and four values for θ are used, being 0°, 45°, 90° and 135°. The 0° -oriented filter accentuates those ridges parallel to the x-axis and smoothes all other ridges. The first step of the feature extraction algorithm is to find a centre point in the fingerprint image, which is defined to be the point of maximum curvature. A circular disc around the centre point is tessellated into 48 sectors, and each sector is filtered in four directions. The variance of each sector after being filtered defines a feature vector with 192 elements, known as a FingerCode. This is a powerful representation of a fingerprint that captures information about ridge orientations around the core of the fingerprint. Since the Gabor filter is tuned to the inter-ridge frequency, it will filter out all irrelevant information from the image. Furthermore, since each fingerprint is registered with respect to a common landmark, the representation is not sensitive to translations. One disadvantage is that finding the centre point can be difficult for noisy images. Another problem with this representation is that it is sensitive to rotations of the fingerprint image. A classification system that uses FingerCode features can be found in the section Hybrid classifiers.

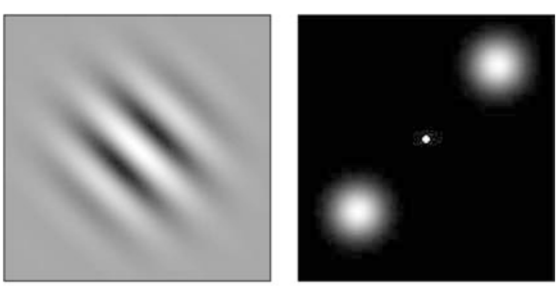
Ridge structure features

There are two main approaches to extracting information about fingerprint ridge structures. One method is to develop a mathematical models of fingerprint ridges and represent the fingerprint using these models. Another approach is to simply record characteristics of the ridges and store this information for classification.

Geometric framework: A geometric framework for analysing the global ridge structure of a fingerprint image has been developed by Chong et al. [19]. Their method approximates the shape of each ridge using a B-spline curve. A B-spline is a generalization of the Bézier curve and is often used for curve fitting in computer graphics. After the individual ridges have been modelled using B-splines, neighbouring ridges with a similar orientation are grouped together. Combining similarly oriented, non-overlapping ridges into composite ridges helps avoid the problem of ridges that have been broken due to noise, making it a robust representation of the overall ridge structure. After grouping, only a few representative ridges are left and they can be used to represent the global geometric shape of the fingerprint. This is illustrated in Fig. 5.

This representation of the global ridge structure is rich, compact and mostly invariant to translation and rotation. Furthermore, these ridge feature are robust against the large intraclass variation of the fingerprint classes. However, this scheme does not handle the small interclass variation very well. In other words, using these features it will be difficult for a classifier to distinguish between two fingerprints that have a similar global shape yet belong to different classes. Furthermore, since the ridges are extracted from the fingerprint image itself, extensive preprocessing and post-processing is necessary to deal with artefacts created by noise. It is possible that this could be avoided by extracting the geometric shapes from a fingerprint's orientation field (see the section Orientation fields). A classification system based on Chong et al.'s geometric framework is presented in the section Global ridge structures.

Hierarchical kernel fitting: Another mathematical model for representing fingerprint ridge structures has been proposed by Jain and Minut [20]. For each fingerprint class a fingerprint kernel is defined that represents the ridge structure of fingerprints belonging to that



(a) (b)

Fig. 4 A Gabor filter

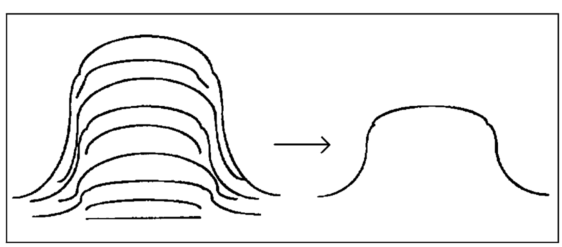


Fig. 5 On the left are the fingerprint ridges of an arch. Similar ridges are grouped to obtain the global geometric shape of the fingerprint [19]

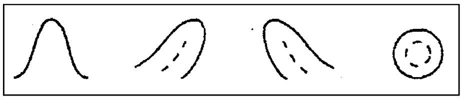


Fig. 6 The kernels (from left to right) for arch, left loop, right loop and whorl [20]

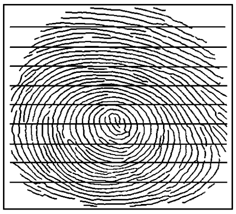


Fig. 7 Fingerprint ridges with horizontal fiducial lines for sampling

class. Some examples kernels are shown in Fig. 6. The fingerprint kernels are defined using polynomial splines. A hierarchy of kernels has been constructed, with kernels at the top representing "ambiguous" classes and the sub-kernels are used for finer distinctions. One advantage of fingerprint kernels is that they are extracted from a fingerprint's orientation field, which can be reliably created even for noisy images (see the Conclusion). For kernel-based classification, the feature extraction stage and classification stage of the system are closely related. The fingerprint is classified by finding the kernel that best fits its orientation field.

Fiducial lines: Senior has developed a novel method for extracting and representing fingerprint ridge features [21]. The first step is to obtain a thinned binary representation of the fingerprint ridges. Next, a set of horizontal lines are arranged across the fingerprint for the purpose of sampling ridge features (see Fig. 7). Senior refers to these as *fiducial lines*. Finally, for each intersection of a fiducial line and a ridge, four features are calculated: (1) the distance since the last intersection, (2) the angle of intersection, (3) the change in angle since the last intersection, and (4) the curvature of the ridge at the intersection. When calculated for each intersection, these features capture information about the orientation, spacing and curvature of the fingerprint ridges. The advantage of this approach over using orientation fields (see the section Orientation fields) is that any number of features can be calculated at each intersection, thereby capturing additional information that may aid classification. However, the primary disadvantage is the assumption that the fingerprint ridge map can be reliably extracted, which is a difficult and computationally expensive operation. A system using these features for classification is discussed in the section Hybrid classifiers.

Ridge Recurrence: Jain et al. store information about the structure of fingerprint ridges for their classification scheme [22]. Each ridge is classified as either non-recurring, recurring, or fully recurring. Non-recurring ridges do not curve very much, recurring ridges curve approximately π and fully recurring ridges curve by more than π. This information alone can not be used to classify fingerprints, but can be used to supplement other features with some information about the overall ridge structure of the fingerprint.

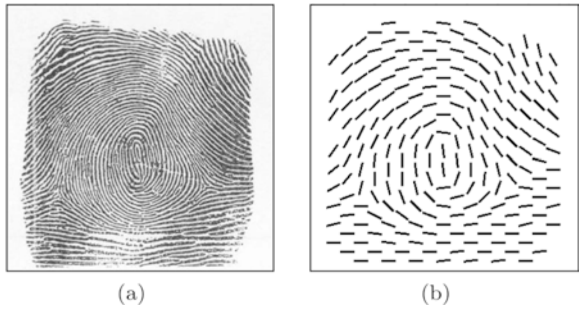


Fig. 8 a A fingerprint Image; **b** The orientation field of a

Orientation fields

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. 8.

Orientation fields are widely used in fingerprint systems and have several applications. One common application is for image enhancement [18] and they can also be used for the detection of singular points (see the section Singularities). One of their most important applications is for fingerprint classification. A fingerprint's class is determined largely by the global pattern of its ridges, and it is exactly this information that is captured by an orientation field. Orientation fields have the additional benefit that they can be calculated reliably even for noisy images.

The calculation of orientation fields

Several methods for calculating orientation fields have been devised. Perhaps the most common method involves averaging gradients in a local area. Image gradients will point in the direction from ridges towards valleys as this will be the direction with the greatest rate of change in greyscale pixel values for a local area. The orientation of the ridges in that area will be perpendicular to the average gradients. Gradient-based techniques are popular due to their ability to create highly accurate results. Several algorithms using this approach can be found in the literature [23, 24, 25, 26].

Another technique for creating orientation fields is based on convolving the fingerprint image with masks designed to detect local ridge orientations. These algorithms have the advantage that they do not require the gradient fields to be calculated, and can therefore be executed very quickly. The ridge-valley algorithm is the most popular mask-based method [27, 28]. A disadvantage of this method is that the orientations are quantised to only eight directions. Most modern systems require a greater degree of accuracy, and therefore rely primarily on gradient based algorithms.

A more thorough discussion of orientation field estimation can be found in [4].

Dimension reduction for orientation fields

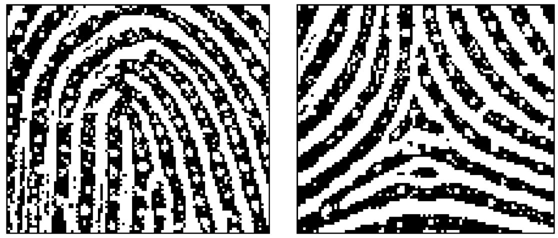
Orientation fields are a very important feature for fingerprint classification. In particular, orientation fields are often used as features for structural and neural network classification approaches (see the sections Structural features and The neural approach, respectively). However, one problem when using the orientation field as a feature vector for classification is its high dimensionality. The actual number of orientation components will vary depending on the implementation. For example, if a fingerprint image is 512×512 pixels and tiles of size 16×16 are used, the orientation field will be comprised of 1024 individual elements. Feature vectors with high dimensions such as this have several practical disadvantages such as large storage requirements, high computational costs, and the "curse of dimensionality". Therefore, it is often beneficial to reduce the dimensionality of orientation field feature vectors before classification.

Candela et al. use the Karhunen-Loève (K-L) transform to reduce the dimension of the feature vector from 1680 elements to 64 elements in their classification system PCASYS [28]. The KL transform has the advantage that it preserves the approximate distance between feature vectors in the feature space. The KL transform, which reduces an original feature vector $\mathbf{u}$ to a vector $\mathbf{w}$ with n elements, is defined by$\mathbf{w} = \Psi^T \mathbf{u}$. Ψ is obtained by first calculating the covariance matrix of some typical feature vectors. Then a diagonalisation routine is used to produce a subset of eigenvectors of the covariance matrix corresponding to the largest eigenvalues. Ψ contains the first n eigenvectors as its columns. By using the K-L transform, PCASYS obtains (approximately) the same classification results as using the original feature vectors, but with much lower computational and storage costs.

Self-organising feature maps (SOMs) have also been used to reduce the dimensionality of feature vectors based on orientation fields, and are discussed in the section The neural approach.

Singularities

A singularity is a local region of a fingerprint where the ridge pattern has special properties making it visually prominent. There are two types of fingerprint singularities: cores and deltas (see Fig. 9). A core is the turning point of an inner-most ridge and a delta is a place where



(a) Core Point (b) Delta Point

Fig. 9 Singularities

two ridges running side-by-side diverge. Henry originally introduced the concept of fingerprint singularities as an aid for fingerprint classification [8]. He noted, for example, that whorl fingerprints have two delta points and one or two core points. Similar observations can be made about the other fingerprint classes, and heuristic rules such as these form the basis of some fingerprint classification systems (see the section Singularities).

Singularities have other applications such as alignment landmarks for fingerprint matching. Because of their extensive use in fingerprint systems, reliable detection and extraction of singular points is very important. Various methods for singularity detection have been proposed. Complex filters designed to detect prominent symmetries in the complex orientation field have been proposed by Nilsson and Bigun [30]. Drets and Liljenström train a neural network to recognise singularities in the orientation field [31], and a syntactic singularity detector is presented by Kawagoe and Tojo [32].

The most common tool for singularity extraction is the Poincaré index. The first application of Poincaré indices to fingerprint images was presented by Kawagoe and Tojo [32]; however, a intuitively similar algorithm was presented earlier [33]. In the context of fingerprint images, the Poincaré index is defined as the rotation of the vectors along a curve in the orientation field. All points of a fingerprint can be classified as either core points, delta points or normal points depending on their Poincaré index.

The Poincaré index is computed by considering a small closed curve around a point in the orientation field and summing the changes in direction of vectors along the curve [34]. Let $\Psi_x(\cdot)$ and $\Psi_y(\cdot)$ represent the x and y coordinates of a closed digital curve with N_Ψ pixels and let $O(x,y)$ be the orientation field vector at position (x,y). The Poincaré index of a pixel (i,j) enclosed by Ψ is:

$$\text{Poincaré}(i,j) = \frac{1}{2\pi} \sum_{k=0}^{N_\Psi} \Delta(k) \tag{4}$$

where

$$\Delta(k) = \begin{cases} \delta(k) & \text{if } |\delta(k)| < \pi/2, \\ \pi + \delta(k) & \text{if } \delta(k) \leqslant \pi/2, \\ \pi - \delta(k) & \text{otherwise.} \end{cases} \tag{5}$$

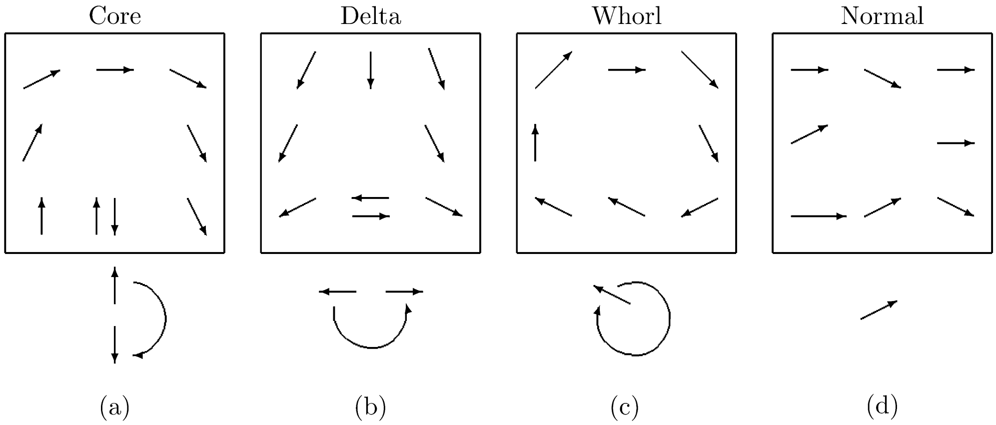


Fig. 10 Singular points and their Poincaré indexes. The indices are as follows: **a** 1/2; **b**-1/2; **c** 1; **d**0

$$\delta(k) = O\big(\Psi_x(i''), \Psi_y(i'')\big) - O\big(\Psi_x(i), \Psi_y(i)\big) \tag{6}$$

$$i'' = (i+1) \bmod N_\Psi \tag{7}$$

Several examples of this are illustrated in Fig. 10. Figure 10a shows an example of typical orientations that would surround a core point. If you start with the vector in the middle of the bottom row and move counter-clockwise around the other vectors, the total change in direction is 180°. This corresponds to a Poincaré index value of $\frac{1}{2}$. Poincaré indices of $-\frac{1}{2}$, and 0 occur for delta points, whorls and normal points respectively.

The size of the enclosure Ψ is very important. If it is too small spurious singularities may be detected, or if it is too big some singularities may be missed. The optimal size for Ψ depends on the resolution of the fingerprint images and can be determined empirically.

Structural features

Structural features record the relationship between low-level elements, and can be useful for fingerprint classification. Maio and Maltoni present a representation of fingerprints based on relational graphs [35] that has also been adopted by several other researchers [36, 37]. Their representation segments the orientation field into regions whose orientation vectors are approximately homogeneous (i.e., they have similar orientations). An example of this is given in Fig. 11a. Each region is represented by a node of a graph and adjacent regions are connected by an edge, as in Fig. 11b. The nodes are labelled with the area of the corresponding region and edges are labelled with the orientation difference of the regions, the distance between the centres of the regions and the length of their common border. This relational graph summarises the topology of the fingerprint and is invariant to translation and rotation.

When segmenting the orientation field the variance of the orientation vectors should be minimised. However, the shape of the regions is also important because highly irregular shapes are likely due to overfitting. In [35] the shape is controlled by simultaneously minimising the

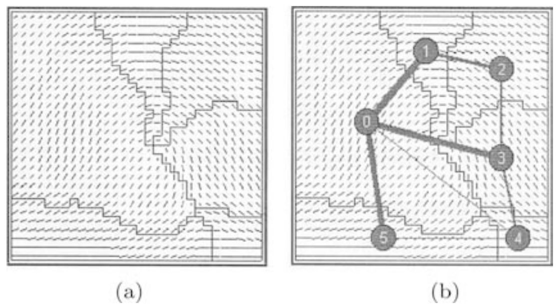


Fig. 11 Structural features using a relational graph [36]

variance of vector orientations and penalising elongated shapes (shapes with a large perimeter and a small area). Segmentation is accomplished using a dynamic clustering algorithm. In [36] it was pointed out that the technique described above suffers from several problems which lead to very different outputs from similar inputs. A new approach is proposed that uses a set of dynamic masks. A dynamic mask is defined for each of the fingerprint classes and is used to guide the segmentation process. Classification systems based on relational graphs are presented in the section Graph matching.

Chang and Fan present an alternate fingerprint representation that captures structural information [38]. The fingerprints are represented as combinations of 10 basic ridge patterns. Examples of the basic patterns are shown in Fig. 12. Chang and Fan claim that all fingerprints can be represented by combinations of these basic types and classification can be performed based on their distribution in the fingerprint (see the section Syntactic pattern recognition). This fingerprint representation provides structural information about the fingerprint

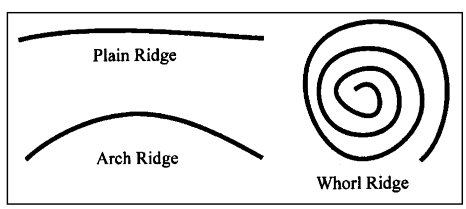


Fig. 12 Three of Chang and Fan's ten basic ridge patterns [38]

and is very powerful if it can be reliably constructed. However, it may be difficult to categorise the individual ridge structures for a noisy image, making this approach only suitable for high-quality input.

Classification techniques

The previous section presented fingerprint features that are commonly used for fingerprint classification. This section discusses some of the classification methods that have been applied to the problem. The section Structural features presents structural classification approaches, namely syntactic pattern recognition and graph matching. Various heuristic approaches based on singularities and ridge structures are discussed in the section The heuristic approach. The section The neural approach surveys the existing applications of neural networks to fingerprint classification and the section Other approaches reviews miscellaneous approaches to classification.

A structural approach

Structural pattern recognition classifies input based on the interrelationships of low-level features. See the section Structural features for a discussion on structural features of fingerprints. Two methods of structural pattern recognition will be discussed here: syntactic classification and graph matching.

Syntactic pattern recognition

In syntactic pattern recognition an analogy is drawn between the structure of the input data's features and the syntax of a language. Complex patterns are decomposed into simple sub-patterns known as *primitives* that comprise the alphabet of the language. The input data is represented by a sequence of primitives, which is considered to be a sentence of a language. Every class has an associated set of rules (or grammar) that describe how to build new sequences (or sentences). Classification is performed by determining which grammar most likely produced a given input sequence. In the context of fingerprint classification, each fingerprint class would have a grammar that generates sequences corresponding to that class.

During the mid-1970s Moayer and Fu published several of the early papers on fingerprint classification [39, 40, 41]. All of these publications make use of syntactic pattern recognition. In [39], a class of context-free languages is used to describe the fingerprint classes. The fingerprint features are extracted from the fingerprint's orientation field (see the section Structural features) whose vectors have been quantised to four directions. For each square block of 4 orientation field elements, the number of possible combinations of directions is $4^4 = 256$. Of these 256 combinations, 56 occur commonly and are used as primitives. A fingerprint can be represented as a sequence of these 56 primitives. Context-free grammars are used to describe the production rules for the fingerprint classes and a top-down parser is developed for classification.

Another classification method using the syntactical approach is presented by Rao and Black [42]. More recently, Chang and Fan developed a classification scheme [38] that uses regular expressions to describe the structure of fingerprint ridges. In the section Structural features a ridge distribution model is presented that represents fingerprints as combinations of ten basic ridge patterns. These basic patterns are the primitives for Chang and Fan's syntactic classification scheme. A "ridge distribution sequence" is obtained by the noting the type of ridges encountered when traversing the fingerprint from the bottom to the top. Regular expressions for generating the ridge distribution sequences for seven fingerprint classes are formulated and a nondeterministic finite automata is constructed to perform classification.

Syntactic methods tend to be robust in the presence of noise and can be designed to be invariant to translations and rotations. However, they struggle with the large intraclass and small interclass variations of fingerprint classes. In other words, the grammars must be able to recognise a wide variety of different sequences as being from the same class, yet still be able to differentiate very similar sequences from different classes. The syntactic classifier's performance is closely related to the features used as fingerprint primitives. Of the schemes discussed above, only the primitives of the ridge distribution model are rich enough to form the basis of a robust syntactic classifier.

Graph matching

Given two graphs as input, graph matching algorithms attempt to determine whether or not the graphs are isomorphic. In the section Structural features a method for representing a fingerprint's topology using relational graphs is presented (see Fig. 11), and Maio and Maltoni have proposed a system that classifies fingerprints based on these graphs [35]. For each fingerprint class a model graph is created that has a structure typical of that class. Since there will inevitably be some variation among graphs from the same class, Maio and Maltoni propose to use an inexact graph matching algorithm. This would allow one to define a distance between two graphs that could be used as the basis for either an exclusive or continuous classification scheme.

Graph matching based techniques have been further researched by Cappelli et al. [36]. As described in the section Structural features, Cappelli et al. use a modified method of partitioning the orientation field by "guiding" the segmentation with the use of dynamic masks. A dynamic mask is defined for five fingerprint classes: the arch, left loop, right loop, tented arch and whorl. For an input image, an application cost is calculated for each of the dynamic masks. Intuitively, the application cost quantifies how well the mask "fits" the input fingerprint. The application costs for each dynamic mask are used to create a feature vector with five elements. If exclusive classification is required, the class with the lowest application cost can be used as the fingerprint's class. A more sophisticated approach to exclusive classification would be to use a neural network or statistical classifier to classify the feature vectors. However, this representation was originally intended to be used as the basis for a continuous classification scheme, and it is more suitable for this purpose. This can be illustrated with an example. The relational graphs for tented arches, left loops and right loops tend to look similar. The strength of this approach is that the degree of similarity with all three classes is recorded. This is valuable discriminatory information, and it is beneficial to exploit it for continuous classification rather than forcing the print into a single arbitrary category.

The heuristic approach

Humans have been classifying fingerprints using their expert knowledge of the domain ever since Henry first proposed his classification scheme in the early 1900s. One approach for automated fingerprint classification is to codify the knowledge of human experts using a system of heuristic rules. Some systems use heuristic rules based on the singularity features, some use ridge features, while others use a combination of singularity and ridge features.

Singularities

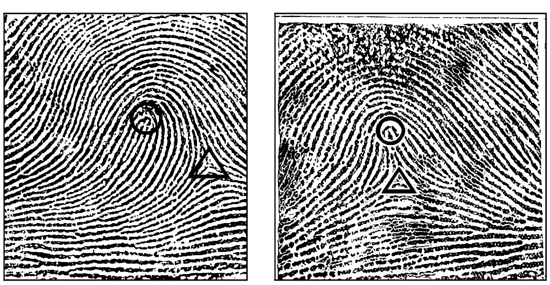
The section Singularitiesdefines singularities (core and delta points) and discusses various methods of detecting and extracting them from fingerprint images. Henry introduced singular points because of their usefulness for classifying fingerprints. For example, he noted that "in loops... there is one delta" [8]. Fig. 13a shows a left loop with one core point and one delta point, which is characteristic of left loops. Having detected a fingerprint's singularities, heuristic rules based on their number and location can be used to accurately classify fingerprints.

Karu and Jain have developed a six class fingerprint classification system based on heuristics [43]. Fingerprints with no core or delta points are classified as arches. Loops and tented arches both contain one core and one delta. They are discriminated by the symmetry of the line connecting the core and delta point: this line is parallel to local orientation vectors for a tented arch, but it transverses the orientation vectors for a loop. This can be illustrated by looking at Fig. 13. In Fig. 13a line connecting the core and delta points transverses the ridges between them, while a line connecting the core and delta points in Fig. 13b is parallel to the ridges between them. Whorls and twin loops have two cores and two deltas and are discriminated using a similar technique that discriminates loops and tented arches. Left loops and right loops are distinguished by examining the direction of local orientation vectors around the core point. Other classification systems based on singularity heuristics can be found in the literature [44, 45, 46].

Systems based on singularity rules work very well if the singularities are accurately detected. They are not sensitive to rotations and translations and fingerprints. Furthermore, they can classify prints from different classes with similar global ridges patterns (small interclass variation) and prints with very different global ridges patterns from the same class (large intraclass variation). However, a major disadvantage of systems that base their classification only on singularity features is that the reliable detection of core and delta points is very difficult. Since singularities are local features they are very sensitive to noise. If the singularities are not extracted (or if spurious singularities are) the systems described in this section perform poorly.

Global ridge structures

Several methods have been proposed to extract and represent the ridge structures that are found in a fingerprint (see the section Ridge features) and it is possible to use these features as the basis for a heuristic classification system. One advantage of this approach is that



(a) Left Loop (b) Tented Arch

Fig. 13 Fingerprints with their core and delta points marked by a *circ* and Δ, respectively

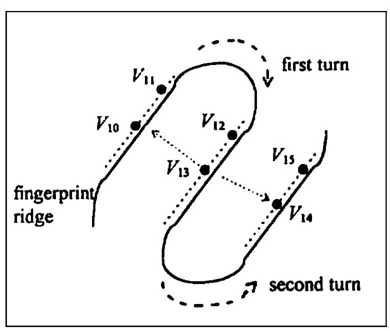


Fig. 14 The global geometry of a twin loop and features that can be used to recognise it [19]

ridge structures can be global features, and therefore can often be reliably extracted even for noisy images.

Chong et al. use the geometric framework described in the section The calculation of orientation fieldsto represent the global geometric shape of fingerprints (see Fig. 5) [19]. It is interesting to note that their classification system uses the five fingerprint classes arch, right loop, left loop, whorl and twin loop, whereas most five-class systems use the tented arch instead of the twin loop. Classification is based on analysing the global geometric shape of the fingerprint. For example, Fig. 14 shows the global geometric shape of the twin loop. Twin loops can be recognised by the fact that they are the only global geometric shape that has two turns with opposite signs. Other (more complex) rules can be constructed that discriminate the other fingerprint classes. The advantages and disadvantages of the geometric representation of fingerprints are discussed in the section Ridge structure features.

Singularities and global ridge structures

As mentioned above, systems that are based solely on singularity features perform very poorly on noisy images since the singularities can not always be extracted reliably. However, a drawback of global ridge structure features is their difficulty dealing with the large intraclass variations and small interclass variations of fingerprint classes. Some classification systems overcome these limitations by using both local (singularities) and global (ridge structures) features. These systems, if designed properly, can accurately classify fingerprints even when some singularities can not be found.

Kawagoe and Tojo use singularity counts to provide a coarse classification of fingerprint images according to Table 1 [32]. Flow-line tracing around singular regions is then used to perform more detailed classification. For example, for a tented arch the flow-line leading from the core point should be straight down, and this can be used to discriminate it from the loop types. A similar approach is taken by Zhang et al. [47].

Ridge shape can also be incorporated with singularity information to aid classification [22, 34]. The fingerprint ridges are divided into three categories based on their curvature. Non-recurring ridges do not curve very much, type-1 recurring ridges curve approximately π and type-2 fully recurring ridges curve by more than π. Classification is based on the number of core points, delta points, type-1 recurring ridges and type-2 recurring ridges. For example, a rule states that if there exists at least one type-2 recurring ridge, two core points, and two deltas points, then classify the fingerprint as a whorl.

Table 1 Kawagoe and Tojo's coarse classification by singularity count [32]. A whorl point contains two close core points and *represents any number

Whorl	Core	Delta	Type
1	0	*	Whorl
0	1	*	Loop, pocketed loop, or tented arch
0	2	*	Twin loop or whorl
0	0	0	Arch

The neural approach

The applicability of neural networks to fingerprint classification began to be researched in the early 1990s [48, 49, 50]. Research has continued since then, and neural networks are now one of the most commonly used classifiers for fingerprint classification systems.

The National Institute of Standards and Technology developed a classification system for the FBI in the early 1990s [29] that uses neural network classification. This research formed the basis for the PCASYS system (Pattern-level Classification Automation System for Fingerprints) whose source code is now publicly available at http://www.itl.nist.gov/[28]. After preprocessing, the first stage in PCASYS is to calculate the fingerprint's orientation field. The directional image is then registered with respect to the centre of the fingerprint image. Registration is used to reduce the translational variation between the orientation fields of different fingerprints. Registration requires a feature that can consistently be extracted from all fingerprint classes. PCASYS uses the core of loops, the upper core of whorls and a well-defined feature of arches and tented arches (which do not have true core points). The dimensionality of the orientation field is reduced using the KL transform (see the section Dimension reduction for orientation fields), reducing the feature vector from 1680 elements to 64 elements. Next, a probabilistic neural network (PNN) is used to classify the feature vector. Finally, an auxiliary classifier known as a *pseudoridge tracer* is used to improve the reliability of classification. The pseudoridge tracer helps to detect whorls, and a simple rule is used to combine the output of the pseudoridge tracer and the PNN. The result is a hybrid classification system (see the section Dimension reduction for orientation fields), although the pseudoridge tracer plays a relatively minor role.

Neto and Borges have developed a neural network classification system that uses wavelet features [51]. Wavelets form the basis of the FBI's fingerprint image compression scheme [52]; however, wavelet-based representations are not very useful for fingerprint classification due to their sensitivity to rotations and translations. A feed-forward neural network with a single hidden layer was trained to classify feature vectors consisting of 64 wavelet coefficients. However, due to the limitations of the feature set, the results from this system are not very impressive.

Halici and Ongun have suggested the use of a neural network known as an SOM [53]. SOMs are based on Kohonen learning and are used for dimensionality reduction (see the section Dimension reduction for orientation fields). The SOM maps n-dimensional input vectors to a lower dimension; usually the 2-D plane. The topology for a typical SOM network has $n$input nodes and $m \times m$ output nodes and each output node has a connection to each input node. During the training stage, a mapping is found such that similar input vectors are situated close to each other in the output plane. This process assigns a class to all regions of the output plane, and classification is performed by mapping input vectors to these regions. Halici and Ongun present a modified version of SOM that includes a *certainty* parameter to handle fingerprints with distorted regions. The features being used for classification are the fingerprint's orientation field and some certainty measures. Several other systems using SOMs have also been proposed [54, 55].

A fuzzy-network classifier is used by Mohamed and Nyongesa to classify fingerprints based on singularity features [56]. The features used include the number of core and delta points, the orientation of core points, the relative position of core and delta points and the global direction of the orientation field. The authors point out that noise and preprocessing errors lead to a wide variation of values for fingerprints within the same class (intraclass variation). Therefore, techniques that provide some flexibility, such as fuzzy-neural classifiers, are desirable. Fuzzy-neural systems combine the advantages of fuzzy logic techniques and neural networks. Fuzzy logic deals with explicit knowledge and high-level reasoning, while neural networks deal with implicit knowledge and offer algorithms for learning and classification. The neural network is used to automatically generate fuzzy logic rules during the training period. In a sense, this scheme is a combination between the neural approaches discussed in this section and the heuristic approaches based on singularities discussed in the section Singularities.

Another fingerprint classification system using artificial neural networks is described by Nagaty [57]. Most of the neural network classification systems discussed so far use the vectors from the orientation field as the features for classification. Nagaty presents a system that uses structural and statistical features. Structural features are extracted from the orientation field using a line tracing algorithm. Prominent flow lines are represented by strings of symbols that encode information about their endpoints and curvature. These are known as characteristic strings, and moments are used to extract statistical features from them. A three-layer feed-forward artificial neural network with six subnetworks (one for each class) is used for classification.

Other approaches

Several fingerprint classification systems use clustering algorithms for classification. The section Frequencydescribed a feature extraction method proposed by Fitz and Green that is based on Fourier transforms. Information from the frequency domain is extracted using a wedge-ring detector [16]. Feature vectors for labelled samples are inserted into the feature space as references, and unlabeled samples are classified according to the label of their nearest neighbour. Using a single nearest neighbour for classification is very simplistic and is not capable of creating complex partitions of the feature space.

The use of a k -Means classifier has been investigated by Wang et al. [58]. Clustering was performed on 500 samples, each labelled as either a whorl, left loop, right loop, or arch. The features used were the orientation vectors in the area surrounding a fingerprint's core. Through experimentation, the authors found that using nine clusters had the best performance, and these clusters were found using a k -Means clustering algorithms. For classification, an unlabelled feature vector is assigned to the most common class of its three-nearest neighbours. Using clustering and three-nearest neighbours is certainly more powerful than simply using a single nearest neighbour, and it still has a low computational complexity. However, the classifier is not as sophisticated as neural approaches or many statistical classifiers.

Cappelli et al. have developed a system that uses a multi-space KL transform as the basis for classification [59]. As mentioned in the section Dimension resduction for orientation fields, the KL transform reduces the dimensionality of a feature space while minimising the average mean-squared error. The Multi-space KL (MKL) transform is a generalisation of the KL transform that uses multiple subspaces for classification. One subspace is trained for each fingerprint class and fingerprints are characterised by their distances to the subspaces. This representation could be used as the basis for a continuous classification scheme, or the fingerprint class could be assigned using a minimum-distance criterion. Using MKL is powerful and has a strong ability to distinguish the fingerprint classes.

Support vector machines (SVMs) are a relatively recent classifier that are based on statistical learning theory [60]. SVMs are a binary classifier that work by finding the optimal separating hyperplane in the feature space [61]. Yao et al. have applied SVMs to the problem of fingerprint classification using the FingerCode representation of the fingerprint (see the section Frequency) [62]. One advantage of SVMs is their strong ability to classify vectors with high-dimensions (such as FingerCodes). Since SVMs are a binary classifier, various methods have been adopted for the application of SVMs to problems with multiple classes. Assuming there are n classes, one approach is to use $n2$ classifiers, each of which distinguishes a single pair of classes (*one-vs-one*). Another approach is to use n classifiers, each of which distinguish one class from all others (*one-vs-all*). Yao et al. performed experiments with both of these approaches, along with a scheme based on error-correcting codes. SVMs are a powerful classifier and encouraging results were presented.

Hybrid classifiers

It is well known in pattern recognition that all classifiers have strengths and weaknesses when compared to each other. In other words, there is no known classifier that consistently outperforms all other classifiers on all problems. Therefore, it is often beneficial to combine classifiers to exploit their relative advantages. When two different classifiers are used in conjunction it is known as a *hybrid* classifier.

SVMss (the section The neural approach) and neural networks are combined for classification by Yao et al. [37]. The features used are a combination of structural features and frequency-based features. For the structural features, the orientation field is segmented into homogeneous regions and represented using relational graphs (see Fig. 11). FingerCode features (see the section Frequency) are used to provide frequency information. The authors performed several experiments to compare the discriminatory ability of these features. In one experiment, the FingerCodes were classified using a multilayer perceptron and the relational graphs were classified using a recursive neural network. The authors found that vector-based features (FingerCodes) had a much higher classification accuracy. In particular, they noted that the structural information is good for classifying whorls and arches, but not very useful for distinguishing left loops, right loops and tented arches. By combining the results from the two classifiers (using a k -nearest neighbour classifier) they were able to obtain better results than either individual classifier in isolation. In another experiment the authors explore the use of SVMs. They compare an SVM's performance when using only FingerCode features with the performance obtained when also incorporating structural features that were extracted using a recursive neural network. Once again, the classification accuracy was improved by incorporating structural features. Furthermore, the SVM results were superior to the ones from using two neural networks for classification.

Senior has proposed a hybrid of a hidden Markov model (HMM) classifier and a decision tree [63]. HMMs are a family of tools for modelling sequential processes in a statistical and generative manner. The HMM classification features were generated from the fiducial lines described in the section Ridge structure features. Hidden Markov models are used to model the features found along the parallel lines. The decision tree classifier used a completely different set of features so that its classification errors were uncorrelated to those from the HMM classifier. The features for the decision tree classifier describe the ridge shapes. Each node of the decision tree is a binary question about the features of a fingerprint. Depending on the answer, a sub-tree is chosen and another question is asked. When a node with no children is found (a leaf), it will have a class (or class probability distribution) associated with it. Senior also considers the classes predicted by the PCASYS classification system (the section The neural approach). Therefore, his system uses three different feature sets and three different classifiers. The results are combined using a backpropagation neural network. Systems such as this one using multiple feature sets and multiple classifiers have the potential to be very powerful because they can exploit the advantages of the different fingerprint representations and classifiers to overcome any individual weaknesses.

Jain et al. have proposed a two-stage classifier based on FingerCodes (see the section Frequency) [17]. The algorithm first uses a k -Nearest-Neighbour classifier to determine the two most likely classes of the fingerprint. These are determined by the two most common classes of the $k$nearest neighbours to the vector in the feature space. During the second stage, the fingerprint's class is determined by a neural network trained specifically to distinguish those two classes. Therefore, if five fingerprint classes are being used $\binom{5}{2}$ individual neural networks are trained. This approach allows the neural networks to be highly specialised to distinguish between only two classes. This is an advantage because the variability between classes can be very small, making it difficult for a single classifier to accurately discriminate all of them. SVMs have been shown to be well suited for classifying FingerCodes [62], so by using SVMs instead of neural networks the accuracy of this system may be improved further.

Classification performance

A common way to represent the results of fingerprint classification is by using a confusion matrix. A confusion matrix has a row for each predicted class and a column for each actual class. Table 2 shows a confusion matrix from [43]. Numbers on the diagonal (shown in bold) are fingerprints that have been correctly classified, while numbers off the diagonal show misclassifications. For example, 197 tented arches were classified as arches.

The comparison of classification algorithms is difficult for several reasons. First of all, different classification systems use different fingerprint classes, and this directly affects the accuracy of a system. For example, a system that categorises fingerprints as only arches, loops, or whorls has a much easier classification task than one that uses all eight of Henry's classes.

Table 2 An example confusion matrix [43]. The left column is the assigned class and the top row is the actual class

	Whorl	Left loop	Right loop	Arch	Tented arch
Whorl	731	35	30	1	10
Left loop	33	780	6	10	79
Right loop	23	3	672	7	7
Arch	5	36	37	912	197
Tented arch	4	11	45	5	321

Table 3 Summary of the classification algorithms evaluated using NIST 4

Author and year	Features	Classification
Wilson et al. (1992) [29]	Orientation field	NN
Karu and Jain (1996) [43]	Singularities	Heuristics
Senior (1998) [21]	Ridge structure	HMM
Jain et al. (1999) [17]	FingerCode	kNN, NN
Hong and Jain (1999) [34]	Singularities and ridge structure	Heuristics
Cappelli et al. (1999) [59]	Orientation field	Multi-space KL
Cappelli et al. (1999) [36]	Relational graph	Inexact graph matching
Yao et al. (2001) [62]	FingerCode	SVM
Senoir (2001) [63]	Ridge structure	HMM, Decision tree, NN
Chang and Fan (2002) [38]	Ridge distribution	Syntactic
Mohamed and Nyongesa (2002) [56]	Singularities	Fuzzy NN
Zhang et al. (2002) [47]	Singularities, ridge tracing	Heuristics
Jain and Minut (2002) [20]	Fingerprint kernels	Kernel fitting
Yao et al. (2003) [37]	Relational graph, FingerCode	SVM, NN

Furthermore, different fingerprint databases are used to evaluate system results. The quality of the fingerprint images is closely tied to the system's performance, and it varies widely between datasets. Fortunately the National Institute for Standards and Technology has made several standard fingerprint databases publicly available. NIST Special Database 4 contains 2000 8-bit greyscale fingerprint pairs (i.e., there are two different fingerprints from 2000 fingers, so there are 4000 images in total) [64]. The prints are labelled as arches (A), tented arches (T), left loops (L), right loops (R), or whorls (W). Some fingerprints are also labelled with alternate classes when ambiguities are present. The database is evenly distributed over the five classes (800 images of each). This distribution does not reflect the actual class distribution found in nature. The NIST Special Databases 9 and 14 contain fingerprints labelled using all eight of Henry's classes plus scar and amputation. The distribution of the classes in NIST 9 and NIST 14 approximates the natural distribution. NIST 4 has become the most common dataset for evaluating fingerprint classification algorithms. Table 3 lists some fingerprint classification systems that have published results using the NIST 4 database. The other classification systems discussed in this paper have not been included in Table 3 because they are evaluated using non-standard databases.

Even though the algorithms listed in Table 3 are tested on the same dataset, there are still some obstacles to performing a direct and fair comparison between them. Some algorithms (such as neural networks and SVMs) need to be trained, and the images used to train a classifier will vary from system to system. Often these algorithms will use a portion of the database for training and a portion for testing. For example, Mohamed and Nyongesa use the entire database for both training and testing [56]. Consequently, their reported accuracy results do not necessarily reflect how well the classifier will perform on unseen data. Some other systems (such as Jain et al. [17]) use half of the NIST 4 database for training and half for testing.

Table 4 A comparison of fingerprint classification algorithm accuracies

Author and year	Classes	Accuracy
Wilson et al. (1992) [29]	5	81.0
Karu and Jain (1996) [43]	5	85.4
Jain et al. (1999) [17]	5	90.0
Hong and Jain (1999) [34]	5	87.5
Cappelli et al. (1999) [59]	5	92.2
Cappelli et al. (1999) [36]	5	87.1
Yao et al. (2001) [62]	5	89.3
Chang and Fan (2002) [38]	5	94.8
Mohamed and Nyongesa (2002) [56]	5	92.4
Zhang et al. (2002) [47]	5	84.0
Yao et al. (2003) [37]	5	90.0
Wilson et al. (1992) [29]	4	86.0
Karu and Jain (1996) [43]	4	91.4
Senior (1998) [21]	4	81.6
Jain et al. (1999) [17]	4	94.8
Hong and Jain (1999) [34]	4	92.3
Senoir (2001) [63]	4	88.5
Jain and Minut (2002) [20]	4	91.3
Yao et al. (2003) [37]	4	94.7

Table 4 shows the accuracy results for the algorithms in Table 3when tested on the NIST 4 database. The "Classes" column is the number of classes used for classification. When five classes are used, they are arch, tented arch, whorl, left loop and right loop. Arch and tented arch are difficult to distinguish and are relatively rare, so they are combined into one class when four classes are used. NIST 4 provides alternate classes for some fingerprints. Most classification schemes only use the first class listed and ignore alternate classes. However, Jain and Minut [20] and Cappelli et al. [59] consider a fingerprint to be correctly classified if it matches any of the alternate labels provided. This results in a higher accuracy than if only the first label is used.

The "Accuracy" column is the percentage of the database that is correctly labelled by the classification system. The accuracy values listed are not weighted to reflect the natural distribution of fingerprint classes. As mentioned above, the NIST 4 database has an equal number of samples from each fingerprint class. In order to get a better estimate of a system's real-world performance, some authors weight the accuracies for each class individually according to the natural distribution before combining the results. Using weighted accuracies will lead to a different overall accuracy. For example, a system that often misclassifies arches (which are rare) will have a lower accuracy when tested on NIST 4 than in the real world because NIST 4 contains 20% arches. In the case of Senior, the accuracy value was calculated from the confusion matrix given in Table 2[63] because only weighted accuracies were presented.

Some algorithms reject images for various reasons. In some cases fingerprints that can not be classified with a certain degree of confidence are rejected in order to increase a system's overall performance. Fingerprints can also be rejected during the preprocessing stage if they are of particularly poor quality. For example, some images in the NIST 4 database do not have FingerCode representations because the core point could not be determined. For some systems the number of images rejected is an adjustable parameter and the authors publish results showing the how the system's performance varies as the rejection rate is increased. Due to space constraints, only accuracies with a 0% rejection rate are shown in Table 4. However, there are two exceptions: for Chang and Fan [38] and Jain and Minut [20] the performance with no rejected input is not reported. The rejection rates for these systems are 5.1% and 1.8%, respectively.

With all of these factors in mind, it is difficult to directly compare the results shown in Table 4. However, it is definitely clear that the accuracies have been improving over time. Also, several of the better results are reported from systems that use combinations of features. For example, Yao et al. [37] have very high accuracies for both the four and five class problem, and they use combinations of feature sets and classifiers.

Conclusions

Automated fingerprint classification is an inherently difficult problem that has yet to be adequately solved. Karu and Jain claim that the FBI requirements for an acceptable fingerprint classification system is 99% accuracy with a 20% rejection rate [43]. Some of the state of the art algorithms listed in Table 4 come very close to satisfying this requirement. For example, Cappelli et al. can achieve 97% accuracy for the five class problem with a rejection rate of 21.6% [59]. This is close to the FBI requirements, however one can not help but question the performance of a system that rejects such a high percentage of its input. Yet, this is the current state of the art of fingerprint classification despite several decades of research.

The next generation of fingerprint classification systems will most likely use combinations of features. The best features for classification are fingerprint singularities. However, core points and delta points can be difficult to detect in noisy images. On the other hand, orientation fields and ridge structure features can be reliably calculated even for noisy images, but these features are not as discriminative as singularities. Robust systems of the future will need to exploit the relative advantages of these different fingerprint representations. Since different classifiers work well with different feature sets, systems with combinations of classifiers will also become more common. Using meta-level classification to combine results from base-level classifiers is a promising approach that would benefit from further investigation.

Future efforts should also explore alternative means of classification. Henry's system was appropriate 100 years ago when classification was performed manually. However, this does not necessarily indicate that it is well-suited as the basis for the automated fingerprint identification systems of today. Continuous classification is one alternative that may be a more natural choice for modern identification systems. However, continuous classification is not compatible with many existing fingerprint databases that are based on Henry's classification, so it will be adopted slowly.

Biometric systems will likely become ubiquitous within the coming years, and fingerprints are emerging as the preferred biometric for identification. In order for real-time identification to be feasible, it is vital that large fingerprint databases execute queries quickly and accurately. These databases will rely heavily on fingerprint classification and indexing to minimise the number of necessary one-to-one fingerprint matches. Considering both the importance of the problem and the need for performance improvements, more research into fingerprint classification is required and there are many opportunities for advancements and innovations in the field.

Originality and contributions

The main contribution of this article is in its presentation of the state of the art in automated fingerprint classification. Fingerprint classification is currently a hot research topic in the image analysis and pattern recognition communities. However, despite increasing attention from both private and academic institutions, the fingerprint classification problem is far from solved. This article surveys the features the are most useful for classification, and identifies the classification strategies that proven successful.

Of particular value for practitioners in the field is a tabulation of results from a wide variety of systems on a standard dataset (i.e., the NIST Special Database 4). This will quickly and conveniently familiarise researchers with the top results in the field. To our knowledge, this is the most extensive survey of fingerprint classification techniques in publication. The review will be valuable for readers with no background in fingerprint recognition as there is little assumed knowledge. Furthermore, it will also be of interest to those with a background in the area wishing to acquaint themselves with the latest developments in the field.

About the authors

Adnan AMIN received his Doctorate D'Etat (D. Sc.) in Computer Science from 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 in INTEGRO (Paris) as Head of the Pattern Recognition Department. From 1987 to 1990, he was an Assistant Professor at Kuwait University and then 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 more than 100 technical papers in these areas. He is an associate editor of the International Journal on Document Analysis and Recognition, Pattern Analysis and Applications, the International Journal of Pattern Recognition and Artificial Intelligence, and the International Journal of Image and Graphics. He was on the program committee of many conferences and served as a referee for numerous journals and scientific organizations. 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, Canada in 2000, and a MEngSc from the University of New South Wales in 2002.

References

1. Jain A, Hong L, Pankanti S (2000) Biometrics: promising frontiers for emerging identification market. Comm ACM Feb:91–98
2. Pankanti S, Prabhakar S, Jain A (2002) On the individuality of fingerprints. IEEE Trans Patt Anal Mach Intell 24(8):1010–1025
3. Jain A, Prabhakar A, Pankanti A (2002) On the similarity of identical twin fingerprints. Patt Recog 35(11):2653–2663
4. Yager N, Amin A (in press) Fingerprint verification based on minutiae features: a review. Pattern Analysis and Applications
5. Herschel W (1916) The origin of finger-printing. Oxford University Press, London
6. Faulds H (1880) On the skin-furrows of the hand. Nature 22(574):605
7. Galton F (1892) Finger prints. McMillan, London
8. Henry E (1900) Classification and uses of finger prints. Routledge, London
9. Trauring M (1963) Automatic comparison of finger-ridge patterns. Nature 197:938–940
10. Wilson C, Candela G, Watson C (1993) Neural network fingerprint classification. J Art Neur Net 1(2):203–228
11. Lumini A, Maio D, Maltoni D (1997) Continuous versus exclusive classification for fingerprint retrieval. Patt Recog 18:1027–1034
12. Germain RS, Califano A, Colville S (1997) Fingerprint matching using transformation parameter clustering. IEEE Comp Sci Eng 4(4):42–49
13. Bhanu B, Tan X (2003) Fingerprint indexing based on novel features of minutiae triplets. IEEE Trans Patt Anal Mach Intell 25(5):616–622
14. Tan X, Bhanu B, Lin Y (2003) Fingerprint identification: classification vs. indexing. In: Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, Miami, FL, July 2003
15. Gonzalex R, Woods R (2002) Digital image processing. Prentice-Hall, New York
16. Fitz AP, Green RJ (1996) Fingerprint classification using a hexagonal fast Fourier transform. Patt Recog 29(10):1587–1597
17. Jain A, Prabhakar S, Hong L (1999) A multichannel approach to fingerprint classification. IEEE Trans Patt Anal Mach Intell 21(4):348–359
18. Hong L, Wan Y, Jain A (1998) Fingerprint image enhancement: algorithm and performance evaluation. IEEE Trans Patt Anal Mach Intell 20:777–789
19. Chong M, Ngee T, Jun L, Gay R (1997) Geometric framework for fingerprint image classification. Patt Recog 30(9):1475–1488
20. Jain A, Minut S (2002) Hierarchical kernel fitting for fingerprint classification and alignment. Proc ICPR 2:469–473
21. Senior A (1997) A hidden Markov model fingerprint classifier. Conference record of the Thirty-First Asilomar Conference on Signals, Systems & Computers 1:306–310
22. Jain A, Prabhakar S, Pankanti S (2001) Matching and classification: a case study in the fingerprint domain. Proceedings of the Indian National Science Academy 67(2):67–85
23. Kass M, Witkin A (1987) Analyzing oriented patterns. Comp Vis Graph Imag Proc 37(3):362–385
24. Jain A, Hong L, Bolle R (1997) On-line fingerprint verification. IEEE Trans Patt Anal Mach Intell 19(4):302–314
25. Rao A (1990) A taxonomy for texture description and identification. Springer, Berlin Heidelberg New York
26. Bazen A, Gerez S (2002) Systematic methods for the computation of the direction fields and singular points of fingerprints. IEEE Trans Patt Anal Mach Intell 24(7):905–919
27. 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
28. Candela G, Grother P, Watson C, Wilkinson R, Wilson C (1995) PCASYS—a pattern-level classification automation system for fingerprints. National Institute of Standards and Technology; NISTIR 5647
29. Wilson CL, Candela GT, Grother PJ, Watson CI, Wilkinson RA (1992) Massively parallel neural network fingerprint classification system. National Institute of Standards and Technology; NISTIR 4880
30. Nilsson K, Bigun J (2002) Prominent symmetry points as landmarks in fingerprint images for alignment. Proceedings of ICPR 3:395–398
31. Drets G, Liljenström H (1999) Fingerprint subclassification: a neural network approach. Intelligent biometric techniques in fingerprint and face recognition. CRC Press, New York
32. Kawagoe M, Tojo A (1984) Fingerprint pattern classification. Patt Recog 17(3):295–303
33. Chang T (1980) Texture analysis of digitized fingerprints for singularity detection. Proceedings of ICPR 1:478–480
34. Hong L, Jain A (1999) Classification of fingerprint images. In: Proceedings of the 11th Scandinavian Conference on Image Analysis, Kangerlussuaq, Greenland, June 1999
35. Maio D, Maltoni D (1996) A structural approach to fingerprint classification. Proc ICPR 3:578–585
36. 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
37. Yao Y, Marcialis G, Pontil M, Frasconi P, Roli F (2003) Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines. Patt Recog 36(2):397–406
38. Chang J, Fan K (2002) A new model for fingerprint classification by ridge distribution sequences. Patt Recog 35(6):1209–1223
39. Moayer B, Fu K (1975) A syntactic approach to fingerprint pattern recognition. Patt Recog 7:1-23
40. Moayer B, Fu K (1976) An application of stochastic languages to fingerprint pattern recognition. Patt Recog 8:173–179
41. Moayer B, Fu K (1976) A tree system approach for fingerprint pattern recognition. IEEE Trans Comp 25(3):262–274

42. Rao C, Black K (1980) Type classification of fingerprints: a syntactic approach. IEEE Trans Patt Anal Mach Intell 2(3):223–231
43. Karu K, Jain A (1996) Fingerprint classification. Patt Recog 29(3):389–404
44. Srinivasan V, Murthy N (1992) Detection of singular points in fingerprint images. Patt Recog 25(2):139–153
45. Cho B, Kim J, Bae J, Bae I, Yoo K (2000) Fingerprint image classification by core analysis. In: Proceedings of the International Conference on Signal Processing, Vancouver, Canada, September 2000
46. Ballan M, Sakarya F (1997) A fingerprint classification technique using directional images. In: Proceedings of the Conference Record of the Thirty-First Asilomar Conference on Signals Systems and Computers 1997
47. Zhang Q, Huang K, Yan H (2001) Fingerprint classification based on extraction and analysis of singularities and pseudoridges. In: Proceedings of the Pan-Sydney Area Workshop on Visual Information Processing, Sydney, Australia, December 2001
48. Hughes P, Green A (1991) The use of neural networks for fingerprint classification. In: Proceedings of the Second International Conference on Artificial Neural Networks 1991, Bournemouth, UK, November 1991
49. Moscinska K, Tyma G (1993) Neural network based fingerprint classification. In: Proceedings of the Third International Conference on Artificial Neural Networks, Brighton, UK, May 1993
50. Kamijo M (1993) Classifying fingerprint images using neural networks. In: Proceedings of the IEEE International Conference on Neural Networks, San Francisco, CA, 28 March–1 April 1993
51. Neto HV, Borges DL (1997) Fingerprint classification with neural networks. In: Proceedings of the IVth Brazilian Symposium on Neural Networks, Goiania, Brazil, 3-5 December 1997
52. Federal Bureau of Investigation (1993) WSQ gray-scale fingerprint image compression specification. Document IAFIS-IC-0110v2
53. Halici U, Ongun G (1996) Fingerprint classification through self-organizing feature maps modified to treat uncertainties. Proc IEEE 84(10):1497−1512
54. Bernard S, Boujemaa N, Vitale D, Bricot C (2001) Fingerprint classification using a Kohonen topologic map. In: Proceedings of the International Conference on Image Processing, Thessaloniki, Greece, October 2001
55. Shalash W, Abou-Chadi F (2000) A fingerprint classification technique using multilayer SOM. In: Proceedings of the Seventeenth National Radio Science Conference, Egypt
56. Mohamed S, Nyongesa H (2002) Automatic fingerprint classification system using fuzzy neural techniques. In: Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, Washington, DC, April 2002
57. Nagaty K (2001) Fingerprint classification using artificial neural networks: a combined structural and statistical approach. Neur Net 14:1293–1305
58. Wang S, Zhang W, Wang Y (2002) Fingerprint classification by directional fields. In: Proceedings of the Fourth IEEE Conference on International Conference on Multimodal Interfaces, Los Alamitos, CA, October 2002
59. Cappelli R, Maio D, Maltoni D (1999) Fingerprint classification based on multi-space KL. In: Proceedings of the Workshop on Automatic Identification Advances Technologies, Summit, NJ, October 1999
60. Vapnik V (1995) The nature of statistical learning theory. Springer, Berlin Heidelberg New York
61. Burges C (1998) A tutorial on support vector machines for pattern recognition. Data Min Know Disc 2:121–167
62. Yao Y, Frasconi P, Pontil M (2001) Fingerprint classification with combinations of support vector machines. In: Proceedings of the 3rd International Conference on Audio and Video Based Biometric Person Authentication, Halmstad, Sweden, June 2001
63. Senior A (2001) A combination fingerprint classifier. IEEE Trans Patt Anal Mach Intell 23(10):1165–1174
64. Watson C, Wilson C (1992) NIST Special Database 4: Fingerprint Database. National Institute of Standards and Technology