Ridge-Based Fingerprint Matching: A survey

Technical Report · January 2015

2 authors:

Katy Castillo-Rosado
Centro de Aplicaciones de Tecnologias de Avanzada
8 PUBLICATIONS 2 CITATIONS
SEE PROFILE

José Hernández-Palancar
Advanced Technologies Applications Center, (CENATAV), Cuba
75 PUBLICATIONS 213 CITATIONS
SEE PROFILE

Table of Content

List of Figures

List of Tables

Ridge-Based Fingerprint Matching: A survey

Katy Castillo-Rosado and José Hernández-Palancar

Biometrics Research Team, Advanced Technologies Application Center (CENATAV), Havana, Cuba.
{kcastillo, jpalancar}@cenatav.co.cu

RT_071 Serie Azul, CENATAV
Aceptado: 14 de enero de 2015

Abstract. One feature commonly extracted to perform fingerprint matching, from full and latent fingerprint impressions are minutiae. In many cases, the image quality is very poor (e.g. latent impressions) and just minutiae are not enough to identify the person. In this work it will be studied the different existing matching algorithms in the literature, specially the approaches that work with more features than just minutiae (extended features). An analysis will be made regarding what features are feasible and which are not to work with latent fingerprint images.

Keywords: fingerprint, latent impressions, features, following, extraction, representation, ridge pattern, descriptor, matching.

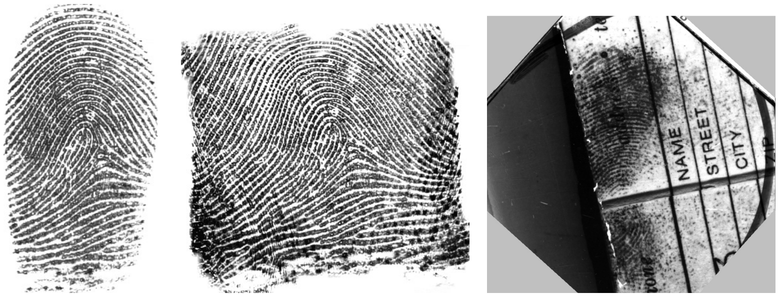
Resumen. Uno de los rasgos más utilizados para la comparación de impresiones dactilares, latentes y no latentes son las minucias. En algunos casos, las imágenes de las impresiones presentan muy mala calidad (ej: impresiones latentes) por lo que la cantidad y calidad de las minucias obtenidas es insuficiente para realizar una identificación efectiva. En este trabajo se estudiarán los diferentes algoritmos de comparación de impresiones dactilares existentes en la literatura, que utilizan otros rasgos de mayor robustez obtenidos del patrón de crestas. El objetivo fundamental de este estudio es realizar un análisis de las características que son factibles para el trabajo con impresiones que no presenten la calidad requerida para el proceso de identificación.

Palabras clave: huella dactilar, impresión latente, rasgos, seguimiento, extracción, representación, patrón de crestas, descriptor, cotejo.

1 Introduction

Biometrics is the study of automated methods for recognizing humans based on one or more behavioural traits or physical intrinsic features. One of the most widely used biometric traits are fingerprints. The uniqueness of fingerprints was accepted since 1893. These are used in civilian and forensic applications (ex. solving crimes) [1].

In an AFIS (Automated Fingerprint Identification System), depending on the acquisition methodology, fingerprints can be classified in: rolled, plain, or latent fingerprint. An example can be seen in figure 1. Rolled fingerprints are those where the finger is rolled from one side to the other ”nail to nail”. Flat fingerprints are obtained by pressing the finger against the surface, these fingerprints contains less information than rolled, but do not present the deformations caused by the rolling process. These two types of fingerprints are normally of good quality, because there are taken in an attended mode, both are equally important for AFIS. Latent fingerprints, are one of the fingerprints used in forensic applications. These

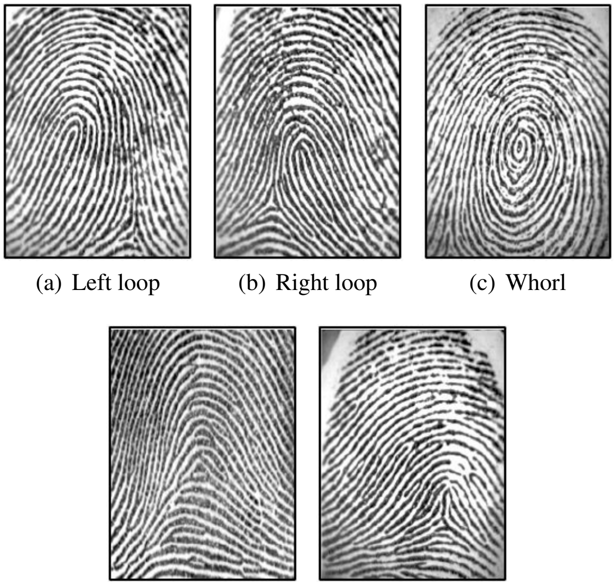


(a) Plain (b) Rolled (c) Latent

Fig. 1. Examples of the three types of fingerprints.

are partial fingerprint images collected from inadvertently touched surfaces at a crime scene and normally have poor quality [1].

To carry out the process of fingerprint comparison, some features are extracted from them. Depending on the scale that the pattern of ridges is analyzed, the extracted features can be classified into 3 levels as follows [1]:



(a) Left loop (b) Right loop (c) Whorl

(d) Arch (e) Tended arch

Fig. 2. Types of fingerprints. Loop and deltas marked.

- Level 1 groups the features extracted at the global level. Singular points, called loop and delta (squares and triangles, respectively), and the ridge line flow presented in figure 2, are features extracted at this level. The distinctiveness of these features is not sufficient for accurate matching, but singular points and coarse ridge line shape are used for fingerprint classification and indexing. Other features like external fingerprint shape, orientation image, and frequency image belong to this set too.
- Level 2 groups the features at the local level. Features extracted at this level are known as minutiae (ridge endings and ridge bifurcations principally) as can be seen in figure 3. A ridge ending occurs

when a ridge ends abruptly. A ridge bifurcation happens when a ridge forks or splits into two ridges. Normally minutiae are stable and robust to the fingerprint conditions. Minutiae-based representation has a high identification character, but the reliable extraction of these features can be very problematic in poor-quality fingerprints because the absence of the ridge structure.

- Level 3 includes the features extracted at the very-fine level, in other words, the intra-ridge details are detected. Some of the features are width, shape, curvature, edge contours of ridges, and others like dots and incipient ridges. Sweat pores are one of the most important features, as their positions and shapes are considered highly distinctive. This detail extraction is possible in high-resolution (1,000 dpi) fingerprint images with good quality [1].

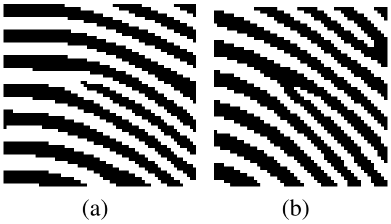


Fig. 3. Minutiae more frequently extracted from fingerprints, ridge ending (3(a)) and ridge bifurcation (3(b)).

Most AFIS use only minutiae and core/delta for matching. A core is defined as “the north most point of the innermost ridge line” [11]. Usually, the core point occurs in the center of the north most loop type singularity [1], but in fingerprints belonging to the arch class that do not contain loop and whorl singularities, it is very difficult to find the core. That is why, the core is associated with the point of maximum ridge line curvature [1]. Human examiners normally exploit a much larger set of fingerprint features. This could be one of the reasons for the superior performance of human examiners over AFIS [12] (especially in cases of difficult latent matches). Other features, besides minutiae and core/delta, are also referred as extended feature set (EFS) and every feature is correctly defined by the ANSI/NIST Committee to Define an Extended Fingerprint Feature Set (CDEFFS) [10]. A summary of the different extended features from all three levels is presented in the table 1. In addition, the relationship between some features can be used.

Obtaining a correct fingerprint matching is extremely difficult as a result of the large intra-class variability. Consequently, fingerprints of different fingers may look the same, while fingerprints of the same finger may look quite different. An example can be seen in figure 4. The fingerprints of figures 4(a) and 4(b) are from the same finger, and fingerprints of figures 4(c) and 4(d) belong to different fingers.

There exist 3 types of fingerprint matching: rolled/plain prints against rolled/plain prints, latent prints against rolled/plain prints, and latent prints against latent prints. Although a great progress in rolled and plain fingerprint matching have been made in literature, latent fingerprint matching (latent to rolled/plain and latent to latent) continues to be a very hard problem. The approaches dedicated to rolled/plain (full) fingerprints use features that are very difficult to extract from poor quality latents. The manual intervention in both, features extraction and matching process is still required. This is because of the poor quality of the images, the small finger area collected and the large non-linear distortions.

In fingerprint verification and identification problems, the feature extraction and matching processes are normally quite similar. The fingerprint identification problem (search a fingerprint in a database of N fingerprints) can be conceived as N one-to-one comparisons (verifications). To reduce the computational cost of identification algorithms, fingerprint classification and indexing are exploited [1].

Table 1. Extended features set [10].

Feature	Description	Level
Pattern Classification	Arch [plain arch, tended arch], whorl [plain whorl, central pocket loop, double loop, accidental whorl], left & right loop.	1
Reference Points	Cores and deltas, core-delta ridge counts, center point of reference.	1
Ridge Quality Map	Ridge quality of each cell in a grid in the region of interest defines the ability to discern detail in this location. (0-Unusable, 1-Poor, 2-Satisfactory, 3-Good)	1
Ridge Flow Map	Direction of friction ridges at various sampling points (uniform sampling frequency) throughout the region of interest.	1
Ridge Wavelength Map	The peak-to-peak distance between ridges at various sampling points (uniform sampling frequency) throughout the region of interest.	2
Degree of distortion	Overall degree of distortion (ridge flow deformation) in the image (1-No obvious distortion, 2-Some distortion, 3-Significant distortion).	1
Distinctive characteristics	Types are presented in table 2.	2
Minutiae ridge counts	Number of intervening ridges between specified minutiae.	2
Dots	A dot is a single or partial ridge unit that is shorter than 0.5mm.	2
Incipient Ridges	A thin ridge unit, substantially thinner than local ridge width.	2
Creases and linear discontinuities	Creases, cracks, cuts, and thin or non-permanent scars.	2
Skeletonized Image	The ridge tracing image, which is black and white. Each ridge is represented as a black single-pixel-wide line.	2
Ridge edge features	Protrusions (abrupt increases in ridge width), Indentations (abrupt decreases in ridge width), and Discontinuities (points where a ridge stops briefly)	3
Pores	Each pore is represented by its center (x; y).	3

Table 2. Types of distinctive features. Taken from [10].

Code	Description
CORE	Unusual core area
DELTA	Unusual core area
SCAR	Scar
DYSPLASIA	Dissociated ridges/ Dysplasia
MINUTIA	Unusual minutia
MINGROUP	Group or cluster of minutiae in close proximity
COMPRESSED	Distorted area with compressed ridges
STRETCHED	Distorted area with stretched ridges
TWISTED	Distorted area with twisted ridges
OVERLAP	Area in which another fingerprint is superimposed over the fingerprint of interest
NEGATIVE	Used if only a portion of the fingerprint image has ridges and valleys inverted so that ridges appear white and valleys appear black
OTHER	Other unusual features not characterized elsewhere

The rest of the article is organized as follow. In section 2 a brief summary of the fingerprint matching techniques is presented. In section 3 different fingerprint features and matching algorithms are described. Finally, the conclusions are given.

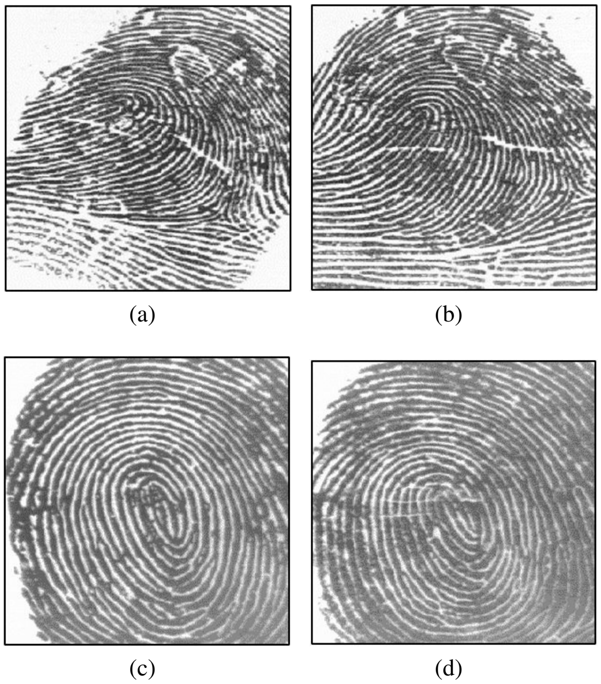


Fig. 4. An example of large intra-class variability and small inter-class variability. Taken from [1].

2 Fingerprint Matching Algorithms

Fingerprint matching is an important task in an AFIS. In this stage two given fingerprints are compared and a degree of similarity (a number between 0 and 1) or a binary decision (mated or non-mated) is returned.

To ensure that two fingerprint are from the same finger, fingerprint experts take into account several factors: 1) both fingerprints must be of the same type (global pattern configuration agreement), 2) the minutiae must be identical (qualitative concordance), 3) at least a minimum amount of corresponding minutiae must be found, it varies across countries (quantitative factor), and 4) the corresponding minutiae must be identically inter-related (corresponding minute details). In practice, rules and complex protocols for manually performing fingerprint matching by experts are defined.

Although there have been several advances in fingerprint matching, this field still allows improvements. Fingerprint latent matching has had less advances, because of the very bad quality of these fingerprints. Latent fingerprints matching algorithms do not have a common guide line. In fact, like the algorithms which use minutiae to perform the fingerprint matching are based in the manual procedure of the human fingerprint examiners, other different approaches have been proposed in all these years, and some are specifically designed to the automation of this process.

The fingerprint matching problem is extremely difficult because of the variability among different impressions of the same finger (large intra-class variations [1]). Impressions of different fingers may look very similar especially in terms of general [overall] structure (small inter-class variations [1]). The principal causes of this problem are: displacement, rotation, partial overlap, non-linear distortion, variable pressure, changing skin condition, noise, and feature extraction errors [1]. All these problems appear with more probability in latent impressions mainly due to the acquisition way, besides only a partial fingerprint is normally obtained.

A possible classification of the different fingerprint matching algorithms could be the next one [1]:

- Correlation-based matching: the correlation of two superimposed fingerprint images for different alignments (e.g. different displacements and rotations).
- Minutiae-based matching: in this case the minutiae (usually endings and bifurcations) are extracted from the fingerprint images and stored. Minutiae matching principally consists on obtain the alignment of minutiae sets (template and input) which maximize the number of minutiae pairings.
- Non-minutiae feature-based matching: when the fingerprint images are of low quality, the minutiae extraction process turns very difficult, while other features of the fingerprint ridge pattern (e.g., local orientation and frequency, ridge shape, texture information) could be extracted more reliably than minutiae. The distinctiveness of these features is usually lower. These algorithms compare the fingerprints using the features extracted from the ridge pattern.

The different ridge-based fingerprint matching strategies are presented below.

2.1 Correlation-Based Fingerprint Matching

The correlation-based fingerprint matching algorithms are the older ones. Correlation between both images is the fingerprint similarity measure of these algorithms. As a result of the inevitable differences of two fingerprint impressions (rotation, displacement), a simple superposition of the images do not resolve the fingerprint matching problem. The final fingerprint similarity will be the highest fingerprint similarity for the different image alignments. By analysing the image globally, the obtained results are not good because of the many cases in which the images are affected by non-linear distortion, skin condition and finger pressure [1]. In order to improve the results, some approaches propose to get the image correlation locally. A set of local regions is extracted from the query image (usually 24×24 or 32×32) and the correlation of each one with the entire original fingerprint image is calculated. There are some different criteria to select the local regions: non-overlapping windows, overlapping windows or just selecting the regions of interest [1].

Since the many problems that have the latent impressions (very poor quality and large distortions) this type of algorithms are not advisable to latent impressions matching.

2.2 Minutiae-Based Fingerprint Matching

Fingerprint matching algorithms based on minutiae are the most used and the best known technique. This occurs because of their similarity with the fingerprint experts work. In most countries, minutiae are accepted as legal proof of identity.

In the case of minutiae-based fingerprint matching algorithms, the fingerprint representation is a feature vector, whose elements are the fingerprint minutiae. Each minutiae can be described by multiple attributes, the most common are: its location, orientation, type, the fingerprint image quality of the minutia neighbourhood, among others.

One important step in the minutiae matching process is the alignment of two fingerprint. To obtain a correct alignment of two fingerprints, the displacement (in x and y) and rotation (orientation) are needed. Also, to compensate other geometrical transformation like scale (when the fingerprints images have different resolutions) or other possible distortion tolerances if the fingerprints are affected by severe distortions is important [1]. The fingerprint alignment step has a large computational cost. To avoid the problems caused by the alignment of minutiae and to better deal with local distortions, some authors perform the minutiae matching locally. Very few authors have tried to perform global fingerprint matching without the alignment step. Local matching provides simplicity, low computational complexity, and high tolerance.

On the other hand, global matching gives a high degree of distinctiveness. Local versus global matching is a trade-off among the provided benefits of each other. To the human fingerprint examiners the number of pairing minutiae is the most important, but in the automated process this must be converted into a similarity score. Several approaches have been presented. To return a correct similarity score is essential because the classification of genuine or impostor match depends on it.

However, when more geometric transformations (besides translation and rotation) are tolerated, more freedom is added to the minutiae matcher. When a fingerprint matching algorithm is designed, the degree of freedom that will be given must be carefully evaluated because the new freedom results in a large number of new possible alignments. This increases the possibility of incorrectly matching two fingerprints from different fingers.

The aim of most minutiae-based algorithms is to compare full fingerprint impressions. To provide an accurate result, these algorithms require a certain number of minutiae. When the fingerprint image have very few usable minutiae (low quality or a small fingerprint area), the performance of these algorithms is very poor (many false positives).

2.3 Ridge-Based Fingerprint Matching

To minimize the false positives, to obtain better alignments, and due to the absence of a considerable number of minutiae, the use of fingerprints ridge pattern features has been increased. This is an important area, especially in the case of latent prints. For example, level 3 features are frequently used by latent print examiners to assist in identification and the idea of *ridges in sequence* which is recommended by SWGFAST [13] is used too. Ridge pattern features can be used to increase the accuracy of the fingerprint matching algorithms.

Hence, another line of fingerprint matching algorithms may be called ridge feature-based fingerprint matching. In this approach, other fingerprint distinguishing features in replace or addition to minutiae are sought. Some of the main reasons that promote this are [1, 14]:

- To improve the accuracy and robustness of the fingerprint matching algorithm, the ridge pattern features are used with minutiae, not instead of them. It is well known that several techniques of this type use the minutiae in the pre-alignment step or to define the anchor points.
- With fingerprint images of very poor quality, to perform a reliable minutiae extraction process turns very difficult. Minutiae have the most fingerprint discriminatory information, but they do not always are the best alternative between accuracy and robustness with fingerprint images of poor quality.
- When the area of the sensor is small, these algorithms may have better performance than the minutiae-based algorithms. In a small area of a fingerprint can exist only 4-5 minutiae, which could provoke that minutiae-based algorithms do not perform well.
- Fingerprint matching algorithms must try to better reflect the way in which the fingerprint human examiners perform manual comparison.

The more commonly used non-minutiae features are [1]:

1. Size of the fingerprint and shape of the external fingerprint silhouette.
2. Number, type, and position of singularities.
3. Global and local texture information.
4. Geometrical attributes and spatial relationship of the ridge lines.
5. Skeleton fingerprint image.
6. Level three features (e.g., sweat pores).

7. Other features: fractal features, shape features derived from the one-dimensional projection of the two dimensional fingerprint image, invariant moments.

Features that are classified as (1) and (2) non-minutiae features, are very unstable. They depend on which part of the fingerprint is analyzed. The features pertaining to (3), (4) and (5) sets, have shown to be useful to the automatic fingerprint matching. Level three features are highly discriminative, but reliable extraction of sweat pores requires high-resolution scanners (1000 ppi) and very robust extraction algorithms [1].

3 Ridge-Based Fingerprint Features and Matching Review

3.1 Feature Extraction Approaches

To extract the fingerprint ridge pattern, in literature, they normally perform image enhancement, binarization and thinning, then the ridges are extracted from the skeleton image [3–5, 7, 15, 16]. This approaches present the problems arising from each previous step. Other approaches follow the ridge pattern directly from the gray-scale image [17–19]. Others extract features from the ridge pattern using image filters [20, 21]. Some of these approaches are presented here.

The algorithms which extract ridge features from the skeleton image, usually perform directional filtering, enhancement, thresholding (binarization) and thinning as principal steps of the entire process. One of the first approaches and very used is the algorithm presented by Jain and Bolle [15]. They proposed an on-line fingerprint verification system. The steps of the minutiae extraction algorithm are: smoothing filter, orientation field estimation, fingerprint region localization, ridge extraction, thinning and finally, minutiae extraction. To calculate the orientation field, a hierarchical implementation of Rao's method [22] is used. If a *consistency level* of the orientation field calculated for an specific region by Rao's method is above a threshold, then the local orientations around that region are re-estimated at a lower resolution level until it is below a certain level. Then, the region of interest from the fingerprint image is located using a segmentation algorithm based on the local variance of gray levels (its assumed that there is only one fingerprint presented in the image). The ridge pixels are detected based on the fact that gray level values on ridges attain their local maxima along the normal directions of local ridges. From the binary image obtained a thinned ridge map is extracted. Because of the noise in the input image, a hole and speckle removal step is applied before the ridge thinning. Jianjiang Feng et al. [3, 4, 16] extract the ridge features with an algorithm developed in their laboratory, that follows the usual steps of a minutiae extraction procedure. Feng's algorithm [16] is similar to the algorithm presented by Jain et al. [15]. Jain et al. [23] obtain the ridge lines from the full fingerprints using the algorithm described by Jain et al. [15], while the ridge lines in the latent prints are manually marked.

Zin Mar Win and Myint Myint Sein [24] presented a fingerprint recognition system for identifying the low quality fingerprint images on Myanmar National Registration Cards (NRCs). The ridge line features of the skeleton image of the fingerprints are compared using Euclidean distance metric. Then, to ensure the validity, the histograms of both fingerprints are extracted and compared. To compute the skeleton image, the Single Pass Thinning algorithm [25] is used, after the region cropping, ridge orientation detection and enhancement stages.

These algorithms have the problems generated from the filtering and enhancement stage which are not reliable for images with very low quality. Spurious ridge pattern and spurious minutiae may be generated and a large percent of genuine minutiae or genuine ridge pattern may be ignored. Also, some specific details are lost. Binarization methods may provoke lost of a lot of information. Both, binarization and thinning processes may introduce false ridge patterns and heavily depend of the enhancement stage. Furthermore, these methods are fundamentally proposed for fingerprint minutiae extraction.

Maio and Maltoni [17] presented an approach to extract minutiae directly on the gray scale fingerprint images. The minutiae extraction is performed while the ridges are followed. At each step, a local maximum relative to a section orthogonal to the ridge direction is computed. By connecting the local maxima consecutively, the ridge line is extracted. The maximum can be computed just by comparing the gray levels of the points belonging to the analized section, but this technique is unsuitable for images with noise and contrast deficiency. To make the determination of the local maximum more reliable, two filters are applied at each step with the aim of regularizing the section silhouette. This process is performed until a ridge ending or a bifurcation is encountered. This algorithm has a high computational cost and is very sensitive to noise and image quality.

Xudong Jiang et al. [18] improved Maio and Maltoni's algorithm [17], by adaptively tracing the gray-level ridge of the fingerprint image. The adaptive oriented filter is applied only in the regions that need to be smoothed. The ridge is approximated with piecewise linear lines. Jianjiang Feng et al. [19] proposed an improvement to the Maio and Maltoni's algorithm with respect to robustness to noise. A stricter stop criterion is proposed. In this case, multiple paths are explored and a criterion is used to choose which is the best of all the explored paths. The enhancement step is eliminated because it has a high computational cost. By exploring all possible paths, the algorithm is not efficient in speed. To improve efficiency, they only explore possible paths until the first path that satisfies the constraint conditions is found. Despite improvements, this algorithm still has a high computational cost. Xiaohui Xie et al. [26] compared two skeleton images directly. The skeleton image is extracted by applying the Maio and Maltoni's method in the high quality zones. In the blurred image zones, during the ridge following process, more paths are searched and a strict stop criterion is used. Then, an algorithm presented by Chen [27] is used to join the broken lines in the skeleton image.

Anil K. Jain et al. [20, 21] presented a hierarchical fingerprint matching algorithm which uses features belonging to the three levels of features (see section 1). The features are extracted from 1000 ppi fingerprint images. Level three features (pores and ridge contours) are extracted using Gabor filters and wavelet transform. In the case of the pores extraction, a linear combination (addition) between the enhanced image using Gabor filter and the Wavelet response ($s = 1.32$) of the original image is calculated. Then the resulting image is thresholded and the pores are extracted. In the case of the ridge contours extraction, a linear combination (substraction) between the Gabor enhanced image and the Wavelet response ($s = 1.74$) of the original image is computed. The resulting image is binarized. The ridge contours are extracted by convolving the binarized image ($f^b(x, y)$) with the filter $H = (0, 1, 0; 1, 0, 1; 0, 1, 0)$ given by equation 1, and (x, y) is a ridge contour point if $r(x, y) = 1$ or 2.

$$r(x, y) = \sum_{n,m} f^b(x, y) H(x - n, y - m). \tag{1}$$

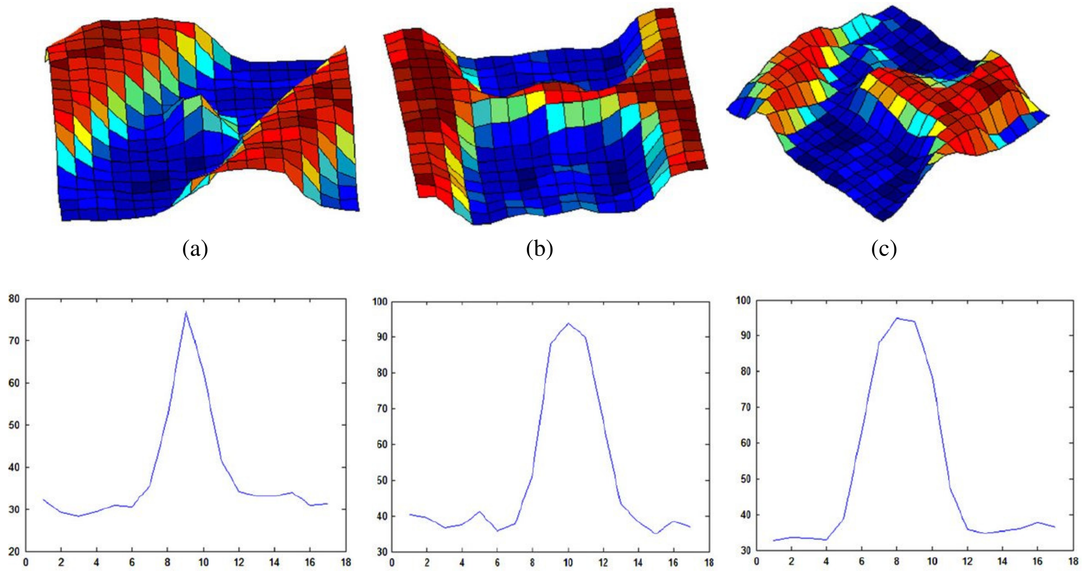
The principal impediments of these filters are given by the high quality and resolution of fingerprints need, which results in an expensive computational cost. This algorithm does not extract correctly the fingerprint open pores. Since the parameters (threshold applied to extract pores, threshold used at binarization step) are experimentally set for a specific database, it cannot adapt itself to different fingerprints.

Mayank Vatsa et al. [28] presented a 3-level feature extraction by using curve evolution with the fast implementation of the Mumford-Shah functional [29, 30]. Mumford-Shah curve evolution is used to segment contours present in the image. After obtaining the final fingerprint contour, the image is scanned using the standard contour tracing technique [31]. The contour information is classified into pores and ridges during tracing. A pore is a blob of size greater than 2 pixels and less than 40 pixels. A pore is approximated with a circle and the center is used as the pore feature. The ridge contour is the edge of the ridge. The ridge feature is represented by the coordinates (x; y) and the direction of the contour at each pixel of the edge. These algorithms have an expensive computational cost. The seed contour decides the algorithm performance and choosing it automatically is not an easy problem.

Shashi Kumar et al. [32, 33] used Block Filter and Strength Factor to propose a Hybrid Fingerprint Matching algorithm (HFMBFS). The minutiae and ridge features are extracted using Block Filter and Hough Transform respectively. The Hough Transform is used to detect straight lines in images, but can be used to extract other shapes that can be expressed as a set of parameters, like curves. Choosing wrong parameters may result in a failure detection of some curves, low accuracy, low speed, and other difficulties [34]. Another problem is that the image must have good quality. The final matching score is a combination of the minutiae matching score and the ridge matching score using the strength factors Alpha (α) and Beta (β).

Zhao et al. [35] presented an adaptive DoG-based (difference of Gaussian) pore extraction algorithm. The fingerprint image is divided into blocks and the scale of the Gaussian filters is adaptively estimated. In the procedure it is assumed that pores are circular features. This assumption is not true for all real fingerprint images and the open pores are not correctly extracted.

Zhao et al. [2, 36] introduced a dynamic anisotropic pore model (DAPM) to perform pore extraction adaptively. The three types of representative pore structures are presented in figure 5. The three types of pores present Gaussian-shaped profiles, but the width of the Gaussian bell is different for each pore. This shows that open pores are not isotropic.


(a) (b) (c)

(d) (e) (f)

Fig. 5. Representative pore structures. 5(a) is a closed pore, 5(b) and 5(c) are open pores. 5(d), 5(e), 5(f) are the intensity pixel values of the pores structures along the ridge orientation. Taken from [2].

In order to represent the different pore types, a model that depends on scale and orientation is proposed. These two parameters are adaptively determined according to the local ridge features (i.e. ridge orientation and frequency). The dynamic anisotropic pore model (DAMP) is defined as follows according to Zhao et al. [2]:

$$\begin{cases} P_0(i,j) = \varepsilon^{(-j^2/2\sigma^2)} cos(\frac{\pi}{3\sigma} i), \\ -3\sigma \leq i,j \leq 3\sigma, \end{cases} \tag{2}$$

$$\begin{cases} P_\theta(i,j) = Rot(P_0,\theta) = \varepsilon^{(-\widehat{j}^2/2\sigma^2)} cos(\frac{\pi}{3\sigma} \widehat{i}), \\ \widehat{i} = icos(\theta) - jsen(\theta), \widehat{j} = isen(\theta) + jcos(\theta), \\ -3\sigma \leq i,j \leq 3\sigma. \end{cases} \tag{3}$$

The reference model is presented in equation 2 (i.e. the zero-degree model), and the rotation model is shown in equation 3. The scale parameter used to control the pore size is σ, determined by the local ridge frequency. To control the direction of the pore model is used θ, estimated by the local ridge orientation.

To estimate θ, the local ridge orientation is obtained by using some existing method [37]. To obtain the σ parameter, with the range of the pore scales, a bank of multi-scale matched filters can be used to extract the pores, but this is very time-consuming. That is why the maximum valid pore scale is used. The pore scales should be restricted by the ridge width because they are located in the ridges. The maximum pore scale is associated with the local fingerprint ridge period by a ratio, $\sigma = \tau/k$, where τ is the local ridge period (the reciprocal of the local ridge frequency) and k is a positive constant (empirically set to $k = 12$). The local ridge frequency is calculated with a projection-based method [38]. The pore extraction procedure consists of five main steps. First the fingerprint image is partitioned into a number of blocks, and each block is classified in a well-defined block, an ill-posed block or a background block. Then the ridge orientation field and the ridge frequency map of the fingerprint image are calculated. The binary ridge map is calculated (the image is enhanced with a bank of Gabor filters, and then it is binarized). This ridge map is used to remove the spurious pores (because the pores are only located on ridges). To locate pores, each foreground fingerprint block is processed. In each well-defined block a local instantiation of the DAPM is defined based on the local ridge orientation and frequency. For each ill-posed block, an adaptive DoG is defined based on the local ridge frequency on the block. After processing all blocks, a binary image is obtained where the candidate pores have value 1 and the other pixels have value 0. The spurious and false pores are removed as final step.

David Zhang et al. [39] studied the necessary resolution that fingerprint recognition algorithms must have to use minutiae and pores. If a sampling period half the size of the smallest pores is assumed, then the minimum resolution required for pore extraction is 700 dpi. By using the minutiae extraction algorithm behaviour and three established criteria that make the system more robust to noise, the resolution chosen as the better option is 800 dpi.

Raoni F. da Silva and Neucimar Jeronimo [40] presented a pore extraction algorithm based on properties of multi-scale morphological transformations (dilation and erosion). This approach attempts to eliminate the extraction problems that suffer other approaches (they do not detect the open pores available in the image). By using the directional field map, the ridge map R (indicates the ridge domain) and a morphological directional erosion, a set of candidate components (A) is defined as follows:

$$A(x,y) = \begin{cases} [max([\epsilon_{g+}(R)](x,y), [\epsilon_{g-}(R)](x,y))]^c & \text{if } R(x,y) = 1, \\ 0 & \text{otherwise}, \end{cases} \tag{4}$$

where c represents the complement of the corresponding element, $\epsilon_{g+}(R)$ and $\epsilon_{g-}(R)$ are the linear erosion of R in the orientation calculated previously.

Then a filtering process is applied to eliminate the spurious components detected in the previous step. This filter depends on the maximum ridge width (calculated using the Euclidean distance) and a positive constant related with the estimated size of the pores (empirically assigned to 5).

3.2 Feature Representation and Matching Approaches

Marana and Jain [41] proposed a ridge based fingerprint matching algorithm using hough transform. For each ridge separately detected from the thinned image, the straight lines are detected using Hough Transform. Each ridge is classified into one of five categories using the straight lines detected (Hough space peaks selected with a threshold). The ridge category is proportional to the ridge curvature (1 for almost straight lines and 5 for almost circular ridges). For each peak a triplet is computed $(\theta_i, \rho_i, \nu_i)$. θ_i is the orientation of the perpendicular to the i^th straight line, ρ_i is the distance of the i^th straight line to the origin and ν_i is the value of the peak. Using the rotation and translation parameters estimated by the Hough Transform for the query and the template, a matching score is computed using a matrix of ridge alignments. The ridge-based matching accuracy is very low, but the combination of this algorithm with a minutiae-based matcher, can get an accurate result.

Feng et al. [16] presented an exact ridge matching algorithm. A ridge is represented by a list of equidistant points, and the correspondences for these points are established. First, the pair most similar is chosen as the initial base pair, and the ridges next to the initial are compared. The new matched pair is used as the base pair, the following ridges are compared, and so on. Multiple pairs of ridges are chosen as initial base pair. The final matching score is the maximum score of all the different comparisons. The ridge matching approach is relatively robust to noise, small overlap area and non-linear deformations. This approach does not work well when it has to analyze difficult impostor matches. This happens because the ridge images from different fingerprint of the same category (arch, whorl, loops, etc), and the local ridge image from different fingerprints may have very similar ridge flow. Feng and Cai [3] represented the fingerprint in a Ridge Coordinate System called RCS. RCS is based on a ridge and an oriented point on the ridge, called $r - axis$ and $origin$ (O) respectively. In this coordinate system, a point includes two components (r, c), analogous to (x, y) in Cartesian coordinates. It is logical the use of a minutiae as $origin$ and the ridge along minutiae as $r - axis$, but any point could be chosen to be the $origin$. More points besides minutiae are chosen as $origin$ to enlarge the region covered by a RCS. An example is presented in figure 6. The order for the neighbours points of the $origin$ is given by its orientation on the $r - axis$. The r coordinate of O is 0. The positive side of the $r - axis$ is the side pointed by O. The r coordinates of the positive side are 1, 2, 3, ..., and in the negative side are -1, -2, -3, ... The orientation of a point p_i is defined as $\overrightarrow{p_i p_{i+1}}$, where i and $i + 1$ are r coordinates or two points.

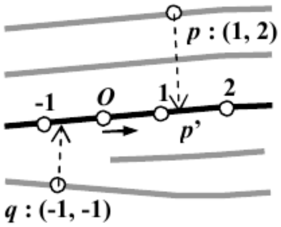


Fig. 6. RCS example. Taken from [3].

A table is computed to record the similarity score between the RCSs of the query fingerprint with the RCSs of the template fingerprint. By using this table, a greedy matching algorithm is implemented.

The problem of choosing which points belongs to each RCS and which does not, is very important. If every pixel is chosen, then the representation size and computational cost of the matching algorithm may increase. However, the selection of very few points could decrease the algorithm performance.

Feng et al. [4] implemented a fingerprint matching algorithm which uses both, minutiae and ridge correspondences. To perform the matching stage, substructures are constructed using the ridges. Each substructure is composed by a minutiae and the ridges around it. In the case of a termination, the ridge that the termination belongs to, and two more adjacent ridges are included (7(a)). In case of a bifurcation, the three ridges that the bifurcation belongs to, and also, two more adjacent ridges are included (7(b)). Each adjacent ridge is split in two sub-ridges. The ridge and sub-ridges are labeled, starting by the ridge that the minutia belongs to, and then the adjacent sub-ridges. In the case of the bifurcation, the three ridges where the minutia belongs to are labeled using the relative direction between the ridges, and the adjacent sub-ridges are labeled using their relative positions and relative directions with respect to the minutia.

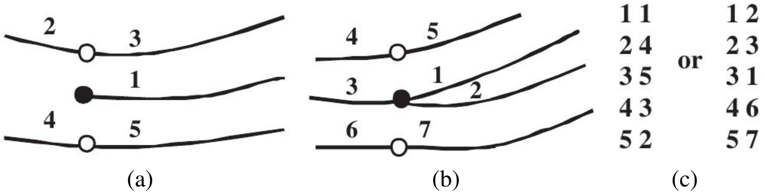


Fig. 7. Example of the substructures. 7(a) substructure for a termination, 7(b) substructure for a bifurcation, 7(c) two possible correspondences for substructures for a termination and a bifurcation. Taken from [4].

To obtain a right match between two substructures, their minutiae have to be of the same type and their ridges must have the same labels, or their minutiae are of different types and the ridges label match on one of the two possible ways shown in figure 7(c).

The top N substructure pairs are extracted as initial pair. With an initial substructure pair, other ridges and minutiae are gradually matched. Feng and Cai [42] proposed an invariant-based indexing scheme. These invariants describe relations between substructures that combine minutiae and surrounding ridges. The substructures are the same used by Feng et al. [4].

Gang Fang et al. [43, 44] proposed to use representative ridge points (RRPs) along minutiae. This points were represented similar to minutiae and can be used easily together with minutiae. Since it is not necessary to use all the ridge points, two requirements are proposed to the ridge point selection scheme: Effectiveness and Efficiency. Effectiveness proposed that in images of the same fingerprint, the same ridge points should be selected. The ridge points are sample from a minutia along the ridge associated to it one by one. Each ridge point has an implicit index. Efficiency demanded that by considering the run-time, the amount of selected ridge points should not be too high. Defining the selection schemes is a very difficult problem because of the requirements that must be satisfied. By trying the problem as a minutiae matching, the algorithms efficiency will decrease due to the considerably increase of the point count.

Xuchu Wang et al. [5] introduced an algorithm to extract two discriminative features that describe macro orientation patterns, micro ridge representation and fingerprint minutiae. Both features (OrientationCodes and PolyLines) are fixed-length, easy to be measured by similarity, and effective in some stages like alignment, minutiae pairing, matching score computation, and matching rates fusion. In the different stages of the matching process, the similarity measure of OrientationCodes and the similarity measure of PolyLines are combined. The OrientationCodes is a feature surrounding each minutiae (or singular point) and describes the orientation flow of a determined area. Similar to FingerCode [45], the section is circularly tessellated through several bands and sectors (e.g. 4 bands and 12 sectors in each band). The band should start from the direction of the minutia to make the feature invariant to rotation and translation alignment. A curve (a ridge) can be sampled with isolated points [15]. These points can be compared in a fingerprint matching algorithm after their associated minutia has been matched. In this case, these points are connected by a PolyLine feature. It is independent from rotation and translation to its associated minutiae. In figure 8, an example of these two features is presented.

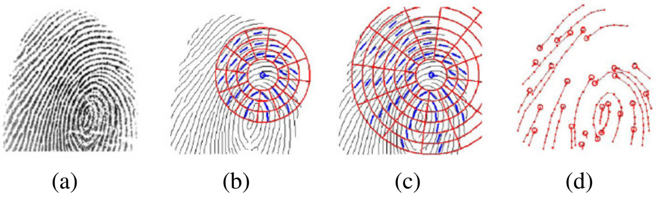


Fig. 8. Example of the feature representation. 8(a) Original image, 8(b) OrientationCode (local), 8(c) OrientationCode (global) and 8(d) PolyLines. Taken from [5].

In cases where the fingerprint quality is bad or the fingerprint area collected is small, the number of extracted minutiae considerably decreases. This can cause that the discriminative behaviour of the OrientationCodes and PolyLines features could be bad.

Anil K. Jain et al. [23] proposed a fingerprint matching algorithm that uses minutiae and ridge information. A ridge is represented by a set of points $R^k = \{R^k(i) = (x_i^k, y_i^k), i = 1, 2, .., N_{R^k}\}$ where (x_i^k, y_i^k) is the i^{th} point in the k^{th} ridge and N_{R^k} is the total number of points in that ridge. The orientation vector can be computed at any point (x_i^k, y_i^k) of a ridge as can be seen in equation 5:

$$\overrightarrow{o} = \frac{(R^k(i+\delta) - R^k(i-\delta))}{|R^k(i+\delta) - R^k(i-\delta)|}, \tag{5}$$

where δ is 10. The ridges are sampled at intervals of 10 pixels and the orientation vectors are computed for these points, for computational efficiency. Using an alignment obtained from the minutiae correspondences, a ridge correlation score is computed. The final matching score is a fusion of minutiae and ridge correlation scores. This representation may have problems with fingerprint images which present very similar ridge patterns.

Xiaolong Zheng and Yangsheng Wang [46] proposed a method based on local ridge similarity. A local rotation angle is estimated from sampling points of ridges that belongs to a minutia (normally the rotation angle between the minutiae sets is estimated using orientation field). Each ridge is represented by a set of points where the first point is the minutia, and the rest are sampled within an uniform interval. If just a few minutiae are detected, then the representation will not be discriminative enough. The optimal local rotation is calculated by fitting the samples using least square method. Using the fitting error the similarity between two ridges is measured. With the smoothed similarity histogram the best reference pair is obtained and by comparing the two minutiae sets according to this reference pair, a final matching score is calculated.

Jain et al. [6] presented a fingerprint matching algorithm (latent to rolled prints) that besides minutiae, it used orientation field and quality map of the fingerprints. A minutia is formed by five attributes (the position x and y, minutiae direction, type and quality). A single orientation and quality values are assigned to each block of size 16 × 16 pixels of the image. The orientation-based descriptor is composed in the following way. A local coordinate system is defined with a minutia as the origin and its direction as the positive axis. The local ridge orientation of a set of fixed sampling points (figure 9) form the orientation descriptor.

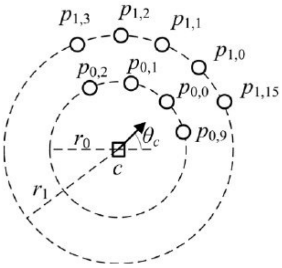


Fig. 9. Set of sampling points of the orientation descriptor. Taken from [6].

The sampling points are distributed equally on 4 circles around the minutiae. The circles radius are 27, 45, 63 and 81, the numbers of sampling points on each circle are 10, 16, 22, and 28, respectively. These parameters are empirically determined [47]. This descriptor depends on the extracted fingerprint minutiae. In cases where the number of minutiae is very small, the false positives increase considerably and the algorithm accuracy decrease. Therefore, in this case, more information could be extracted, maybe in areas where no minutiae where extracted.

Li et al. [48] proposed a matching algorithm that combines the minutiae matching score with the ridge curvature map matching score using a sum fusion rule. A curvature value for every pixel is computed and every value is displayed in another gray scale image which is called Ridge Curvature Map (RCM). To obtain the RCM two steps are performed: coarse curvature value extraction and polynomial model fitting. In the coarse curvature value extraction step, the orientation field is computed using the gradient-based method, then the image is enhanced and binarized. The ridge contours are extracted from the binarized image. Finally, the coarse curvature is computed for the ridge contours image. This is different from other approaches where the ridge curvature is obtained for the thinned image, by using the ridge contour image the effective coarse curvature increase and the minutiae's curvatures can also be computed correctly. The ridge contour line pixel's curvature is computed using the orientation difference of an arc with fixed-number pixels. In the second step a bivariate polynomial function model is used to model the fingerprint ridge curvature. The model's coefficients are calculated using a Least Square (LS) algorithm. In matching stage, a correlation matching score is computed in the frequency domain. The Fourier-Mellin Transform on RCMs is performed and the rotation and translation parameters are calculated. Then, an Inverse Fourier Mellin Transform is performed and a correlation image in the space domain is obtained. The experimental results show that RCM is easily influenced by distortion and image quality.

Jucheng Yang [49], after making a summary of various non-minutiae based fingerprint descriptors, proposed to use the tessellated IM (proposed by Yang and Park [50, 51]) as features, because the effects of noise and non-linear distortions are reduced. The feature vector dimension is high, in order to reduce dimensionality and to select the distinctive features, PCA is used. Then a SVM classifier is used to perform the fingerprint matching procedure.

Heeseung Choi et al. [7] used ridge features and minutiae features (minutiae type, orientation, and position), to increase the fingerprint matching performance against non-linear deformation in fingerprints. The ridge features extracted are ridge count, ridge length, ridge curvature direction (concave shape and convex shape), and ridge type between two minutiae. A ridge-based coordinate system in the skeleton image is defined as shown in figure 10. Each ridge-based coordinate system is determined by a minutiae (the $origin$) and a vertical and horizontal axes starting from it. The vertical axis is defined as the line orthogonal to the $origin$ orientation intersecting this point. The horizontal axes are defined as ridges intersecting the vertical axis. The sign of each axis depends on the orientation vector of the $origin$. The features are extracted using the ridge-based coordinate system. When the image quality is poor or some bad quality areas are presented in the fingerprint, the ridge count feature can not be used. These features describe the relationship among minutiae. In cases where the number of reliable minutiae is low these features can not represent correctly the fingerprint. In the matching process a breadth-first search (BFS) is performed to incrementally detect the matched minutiae pairs. Finally the matching score is the maximum score computed.

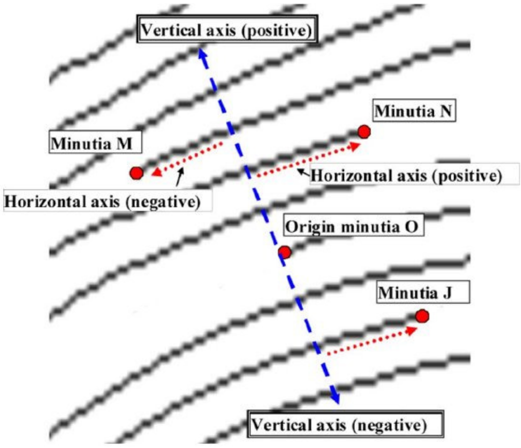


Fig. 10. Example of the ridge-based coordinate system. Taken from [7].

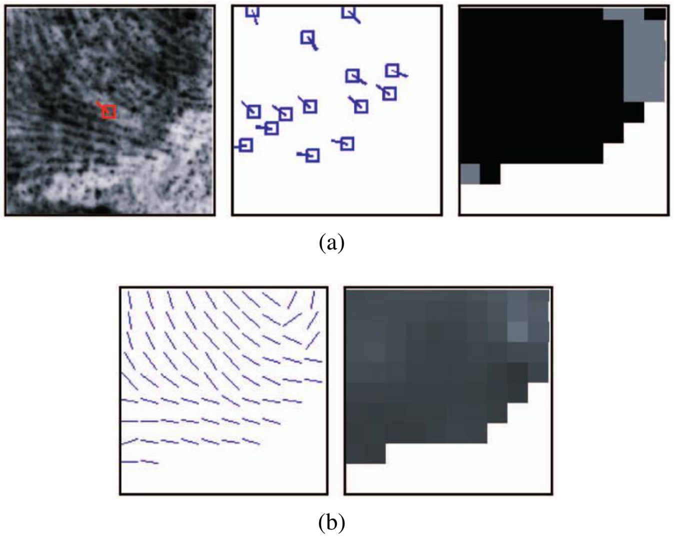
Anil K. Jain and Jianjiang Feng [8] presented a latent-to-rolled/plain matching algorithm that uses minutiae, reference points (core, delta, and center point of reference), overall image characteristics (ridge quality map, ridge flow map, and ridge wavelength map), and skeleton (or skeletonized image). These features were chosen because of their distinctiveness, repeatability, universality, and detectability in 500 ppi fingerprint images. The proposed skeleton matching algorithm is an improvement of the algorithm presented by Feng et al. [4]. The ridge features from latent impressions are manually extracted. This is the more extensive research found about extended features. The ridge quality map, ridge flow map, and ridge wavelength map are used in both, local minutiae matching (minutiae descriptor, see figure 11) and global score calculation (see figure 12).

For presenting the benefits provided by the extended features, the features are aggregated to the matching stage, progressively. The database used was $NISTSD27$, which presents 258 latent impressions and their 258 corresponding rolled impressions. Latent impressions are separately in good, bad and ugly because of the quality impression. The achieved results are shown in figure 13.

Using pores, dots and incipients do not provided a good results. Figure 14 shows the number of pores and dots/incipients presented in latent impressions and the number of them that found mate. Level three features show a low performance probably because of the low number of extracted features from latent impressions.

Hasan and Abdul [52] presented a survey of the fingerprint images enhancement and recognition algorithms. Along the article some approaches of fingerprint verification and identification are described. The review shows that much effort is still needed in latent fingerprint feature representation and matching.

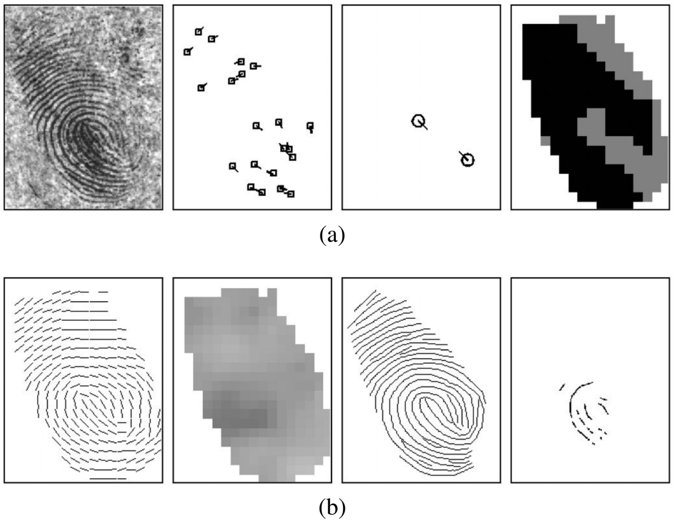
Alessandra A. Paulino et al. [53] presented a latent fingerprint matching approach. Using minutiae and orientation field information, a descriptor-based Hough transform is used to align and measure similarity between fingerprints. The alignment procedure is conducted similar to the proposal of Ratha et al. [54]. By assuming that true mated minutiae pairs will vote for very similar sets of alignment parameters, and the behaviour of non-mated minutiae pairs will be random, the alignment that obtains the highest vote is chosen as result. Ten sets of alignment are considered. The final alignment depends largely on the minutiae



(a)

(b)

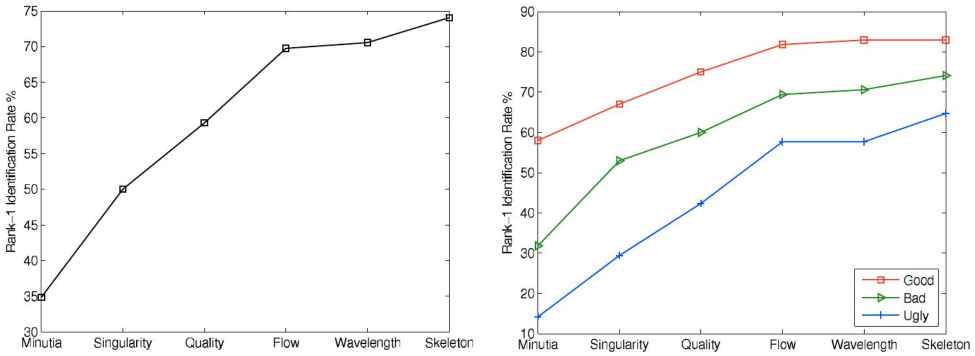
Fig. 11. Minutiae descriptor. In this order, local gray level image, neighbouring minutiae, local ridge quality map, local ridge flow map, local ridge wavelength map. Taken from [8].



(a)

(b)

Fig. 12. In this order, gray level image, extracted minutiae, singular points, ridge quality map, ridge flow map, ridge wavelength map, skeleton and level three features (dot/incipients). Taken from [8].



(a) (b)

Fig. 13. Results achieved for 13(a) all fingerprints and 13(b) good, bad and ugly, in the matching stage aggregating the extended features progressively. Taken from [8].

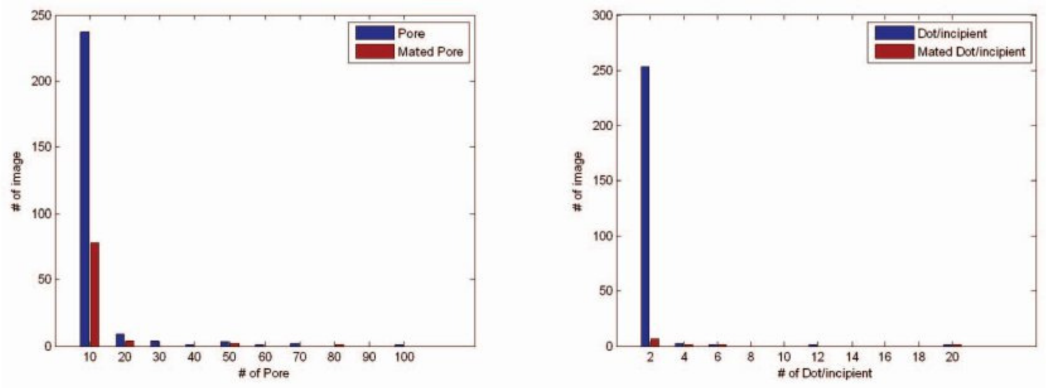


Fig. 14. Number of pores-mated pores and dots/incipients-mated dots/incipients presented in latent impressions. Taken from [8].

descriptor (local neighbourhoods) similarity. In cases where these descriptors are very different as a result of missing minutiae or bad image quality, minutiae are not enough and the algorithm fails. Only minutiae are manually marked (a common practice in latent matching) and the orientation is reconstructed from them. This makes it easy to use in law enforcement applications.

Shalaby and Ahmad [9] proposed a structural technique for fingerprint representation and matching, which is referred as multilevel structural technique for fingerprint fingerprint recognition (MSFR). In this case, the fingerprint is first decomposed into sub-images (regions) based on their global features. Each region is characterized by an orientation value θ_i chosen from the set $\{0, \Delta, 2\Delta, ..., \pi - \Delta\}$ and the region type that could be only one of these: core region (upper and/or lower core), left delta region and right delta region. An example of the fingerprint decomposition algorithm is shown in figure 15.

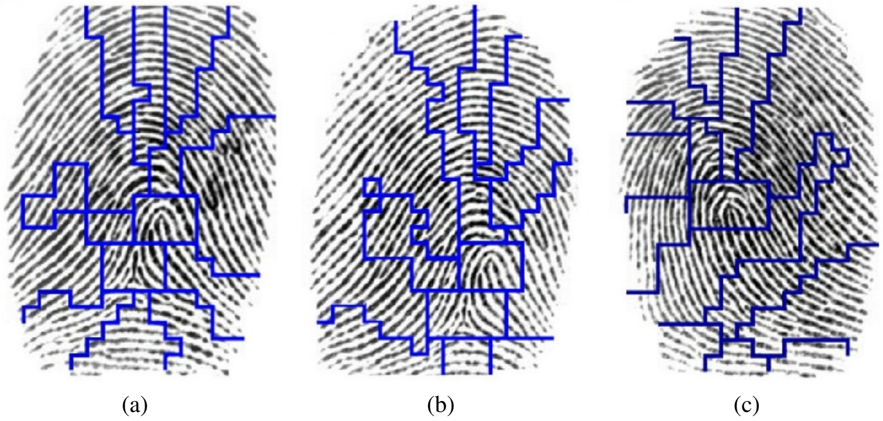


Fig. 15. Examples of decomposition algorithm. 15(a) and 15(b) are images from the same finger while 15(c) is an image from another finger. Taken from [9].

Then for the representation it is proposed a multilevel feature vectors (MFVs) which describes each region and contains all the local and global information of the region. Using this representation, the problem of comparing two fingerprints, is reduced to the problem of finding for each region on the query its correspondence with a region in the reference fingerprint. The final similarity is calculated based on the similarity of matched pairs of regions between the two fingerprints, using local and global features. To decompose the fingerprint image into several regions, orientation image and singular points are used. A region of a fingerprint is represented using three levels:

1. The global features of the region (global features with respect to the core region).
2. The neighbourhood features of the region (relations with its adjacent regions, this is useful when the fingerprint point core cant be detectable).
3. The local characteristics of the region (curvature of ridges and minutiae).

A multilevel feature vector is defined as three feature vectors: FV1, FV2 and FV3 corresponding to the three levels described above. The first level (FV1) of the multilevel feature vector is composed by two components: $i)$ ξ, a number between 1 and 11, representing the current region's type and its relative position to the core point, and $ii)$ Θ_{R-core}, the orientation of this region relative to the core orientation, calculating by ($\Theta_k - \Theta_{core}$), where Θ_k is the orientation of the ridges of the region. Therefore, this value is between $[-\pi, \pi]$. The second level (FV2) is a set of neighbourhood features of the region (k), the components amount is the number of adjacent regions. Each component is composed by a pair of features: $i)$ ρ_{kj}, the position of the adjacent region j relative to that of the region k, and $ii)$ Θ_{kj}, the orientation of the adjacent region j relative to Θ_k. The third level (FV3) is formed by two local features: $i)$ ϕ_k, the curvature of this region, and $ii)$ $Minu_k = \{m_1, m_2, ..., m_L\}$, the minutiae set belonging to the region k. In the matching stage, if the center point does not exit, only FV2 and FV3 are used. If fingerprint images are from the same category, the best corresponding pairs of regions are found, using FV1. Then, a similarity degree is calculated. If the value of this similarity is equal to zero, a non-match is reported. Or else, a second similarity measure is calculated using FV2. If the latter similarity is equal to zero, a non-match is reported. Otherwise, a third similarity measure is estimated by using FV3. The final similarity is obtained by combining the three degrees of similarities.

The principal problem of this algorithm is the high dependence with the core point location. Several times, latent fingerprints do not have the center region of the image, and it is unknown the position of the partial fingerprint with respect to the center region.

Krish et al. [55] present a new approach to improve the identification rate for latent fingerprints. This is an extension of another work presented by them [56], where a correlation based orientation field registration method is presented, and only minutiae are used for orientation field generation. The main idea is to reduce the minutiae search space of the full fingerprint (tenprint) with respect to the latent fingerprint (query) minutiae set. The proposal is to use the orientation field estimated from partial and full fingerprints. The experiments are presented with the orientation field obtained by reconstructing it from their respective minutia sets [57] and also, with the orientation field directly estimated latent and full fingerprint images. The better results were achieved using the orientation field generated from the minutia sets.

Balti and Sayadi [58] proposed a fingerprint identification technique based on the theory of linear algebra called singular value decomposition (SVD) and a set of seven invariant moment features. The problem with these features is that they are not invariant to the texture differences of fingerprint images from different sensors.

González et al. [59] present an analysis of the learning and retrieval capabilities of the diluted metric attractor neural network applied to fingerprint images. By using the dilution, the computational cost of the network decreases, therefore the region of interest can be increased and could cover almost the complete fingerprint. This technique was tested for different noisy configurations, but these noises do not reflect the real distortions that fingerprints may present. The results showed that network topologies with a 2D-Grid arrangement adapt better to the fingerprints spatial structure. To represent the intrinsic spatial structure of the fingerprints an optimal ratio of local connections to random shortcuts was presented. A phase diagram is used to characterize its influence on the retrieval quality.

3.3 Summary of the Different Approaches

There are many descriptors that depend on minutiae. However, the accuracy obtained by the latent fingerprint matching algorithms is not yet the demanded value due to latent fingerprint characteristics. To solve it, some algorithms have emerged using other features besides minutiae.

To improve the performance of matching algorithms an extended feature set has been standardized. Some of these features have not been proved yet by automatic matching algorithms (like scars or creases). Other features (like core, delta, center point of reference, ridge quality map, ridge flow map, ridge wavelength map, and skeleton image) have been used and clearly improve the fingerprint matching accuracy. Nevertheless, some of these feature representations are not yet discriminative enough.

To merge features, the fusion of different matching algorithms is proposed. The merge of several algorithms could provoke an increase of the computational cost. Therefore, a balance between accuracy and efficiency must be found.

The process of minutiae extraction from latent impressions is very time consuming because it is performed manually by experts. Automatic extraction of some features like minutiae reliability and the orientation field, are still computationally complex. This topic must still remain under investigation.

A summary of the different approaches studied in this review is presented in table 3.

Table 3. Summary of studies on ridge fingerprint matching.

Author(s)	Features	Approach	Database
Maio and Maltoni (1997) [17]	Minutiae	Ridge Line Following	14 images (7 from the NIST database [60], 4 from an FBI sample set, 3 from an opto-electronic device)
Jiang et al. (2001) [18]	Minutiae	Adaptive tracing the gray-level ridge	410 fingerprint images of size 300 × 300 and 4000 fingerprint images of size 512 × 512 from NIST Database 4
Feng et al. (2004) [19]	Minutiae	Ridge Following	100 fingerprint images from FVC2002's database
Marana and Jain (2005) [41]	Ridges	Hough Transform	MSU-VERIDICOM fingerprint database
Feng et al. (2005) [16]	Ridges points	Exact ridge matching	DB1_A of FVC2002 (800 fingerprints)
Feng et al. (2006) [3]	Minutiae and Ridges	Ridge Coordinate System (RCS)	db1_a and db3_a from FVC2002 databases
Feng et al. (2006) [4]	Minutiae and Ridges	Ridge correspondences around minutiae	FVC2002 databases
Xie et al. (2006) [26]	Ridges	Relation of a ridge with its neighbour ridges	FVC2002 databases
Fang et al. (2007) [43]	Minutiae and Ridge Points	Representative Ridge Points	FVC2002 databases
Jain et al. (2006) [20]	Minutiae, pores and ridge contours	Level 3 matcher	1640 fingerprint images
Jain et al. (2007) [21]	Minutiae, pores and ridge contours	Hierarchical matcher	1640 fingerprint images
Wang et al. (2007) [5]	Minutiae, local orientation field, ridge lines	OrientationCode and Polylines	FVC2002 and FVC2004 databases
Jain et al. (2007) [23]	Minutiae and Ridges	Fusion of the minutiae an ridge-correlation based matchers	NIST SD27
Zheng and Wang (2008) [46]	Minutiae and Ridges	The selected ridges are those associated with minutiae	DB3 in FVC2002
Jain et al. (2008) [6]	Minutiae, orientation field	Orientation-based descriptor and neighbouring minutiae-based descriptor	NIST SD27
Zhao et al. (2008) [36]	Pores	Adaptive anisotropic pore model	198 fingerprint images (320 × 240)
Vatsa et al. (2009) [28]	Minutiae, ridge contours and pores	Delaunay triangulation with minutiae, eight elements describe each triangle	5500 fingerprint images from 550 classes (10 images per class) captured using an optical scanner
Wang et al. (2009) [61]	Lines/Curves pixels	Mean vector and the standard deviation	16 pairs of real images

Finished on next page

Table 3 (*finished*)

Author(s)	Features	Approach	Database
	gradients	of gradients	(not fingerprints)
Li et al. (2009) [48]	Minutiae and RCM	Sum rule score fusion of minutiae matching and ridge curvature map correlation	FVC2002 and DB1 and DB2 from FVC2004
Zhao et al. (2010) [2]	Pores and Minutiae	Adaptive anisotropic pore model and Score fusion	1480 fingerprint images of 320 × 240 and 1480 of 640 × 480
Kumar et al. (2010) [32, 33]	Minutiae and Ridges	Block Filter and Hough Transform. Matching score fusion with Strength Factors.	FVC2004
Zhao et al. (2010) [35]	Minutiae and pores	Pore-valley descriptor (PVD) (locations and orientations)	210 partial fingerprint images from 35 fingers as the training set. 1480 fingerprint fragments from 148 fingers (including the fingers in the training set) as the test.
Win and Sein (2011) [24]	Ridges	A single pass thinning algorithm and extract features points (ridges points).	300 images from NRC card and FVC2000 database.
Yang (2011) [49]	Gray scale intensities	Invariants moments	FVC2002 databases
Choi et al. (2011) [7]	Minutiae and Ridge Features (ridge count, ridge length, ridge curvature direction, ridge type)	The ridge features are extracted using their proposed Ridge-Based Coordinate System.	FVC2002 (DB1, DB2, DB3) and FVC2004 (DB1) databases
Jain and Feng (2011) [8]	Reference points (singularity), ridge quality map, ridge flow map, ridge wavelength map, minutiae and skeleton. Dots, incipient ridges, and pores	Minutiae matching, orientation field matching, and skeleton matching. Use level-3 only when level-2 fails	NIST SD27, NIST SD4 and SD14 databases as background
Paulino et al. (2013) [53]	Minutiae and orientation field	Minutia Cylinder Code (MCC), and Descriptor-based Hough transform	NISTSD27 and West Virginia University Latent Fingerprint Database (WVU LFD)
Silva and Jeronimo (2013) [40]	Pores	Multi-scale Morphological Transformations	PolyU High-resolution-fingerprint Database
Shalaby and Ahmad (2013) [9]	Minutiae, orientation, singular points, curvature	Multilevel fingerprint region representation	FVC2002 (DB1, DB3, DB4) FVC2004 (DB1, DB2) FVC2006 (DB1)
Krish et al. (2014) [55]	Minutiae	Reduction of the minutia search space of the tenprint	NISTSD27
Balti and Sayadi (2014) [58]	SVD features and seven invariant moments	Singular value decomposition	FVC2002 (DB1, DB2, DB3, DB4)
González et al. (2014) [59]	Optimal ratio of local connections to random shortcuts	Diluted metric attractor neural network	FVC2000 FVC2002 FVC2004

A summary of the reviewed approach accuracies for some of the different databases is presented in tables 4, 5, 6 and 7.

Table 4. Summary of accuracies for NISTSD 27 database.

Author(s)	Features	Rank-1 Accuracy (%)			
		All	Good	Bad	Ugly
Jain et al. (2007) [23]	Minutiae and Ridges	86.0	94.5	86.0	79.0
Jain et al. (2008) [6]	Minutiae and orientation field	79.5	92.0	76.0	73.0
Jain and Feng (2011) [8]	Reference points, ridge quality map, ridge flow map, ridge wavelength map, minutiae and skeleton	74.0	83.0	73.5	64.0
Paulino et al. (2013) [53]	Minutiae and orientation field	53.5	75.0	47.1	37.6
Krish et al. (2014) [55]	Minutiae	-	99	85	82
		contain at least 75% of the matched minutiae			

Table 5. Summary of accuracies for FVC 2002 databases.

Author(s)	Features	Accuracy EER (%)			
		DB1	DB2	DB3	DB4
Feng et al. (2005) [16]	Minutiae and ridges points	1.0	-	-	-
Feng et al. (2006) [3]	Minutiae	1	4	-	-
Feng et al. (2006) [4]	Minutiae and Ridges	0.017	0.014	0.069	0.051
Xie et al. (2006) [26]	Ridges	0.35	0.63	1.45	0.7
Fang et al. (2007) [43]	Minutiae and ridge points	0.7	1.0	7.8	3.4
Wang et al. (2007) [5]	Minutiae, local orientation field, ridge lines	0.46	0.61	3.58	2.04
Li et al. (2009) [48]	Minutiae and RCM	1.1	0.59	3.3	1.0
Yang (2011) [49]	Tessellated IM	1.42	2.23	2.48	3.31
Choi et al. (2011) [7]	Minutiae and ridge features (ridge count, ridge length, ridge curvature direction, ridge type)	1.8	0.8	3.5	-
Shalaby and Ahmad (2013) [9]	Minutiae, orientation, singular points and curvature	2.57	-	6	2.81
Balti and Sayadi (2014) [58]	SVD features and seven invariant moments	2.85	4.62	3.55	2.2

Table 6. Summary of accuracies for FVC 2004 databases.

Author(s)	Features	Accuracy EER (%)
Shalaby and Ahmad (2013) [9]	Minutiae, orientation, singular points and curvature	DB1: 2 DB2: 3.2
Wang et al. (2007) [5]	Minutiae, local orientation field, ridge lines	DB1: 7.49 DB3: 2.83
Li et al. (2009) [48]	Minutiae and RCM	DB1: 7.3 DB2: 5.9
Choi et al. (2011) [7]	Minutiae and ridge features (ridge count, ridge length, ridge curvature direction, ridge type)	DB1: 4.3

4 Conclusions

Definitely, the best way to increase the accuracy to the fingerprint matching algorithms when the number of suitable minutiae is low is by merging features from all levels or by the fusion of different matchers [62, 63]. Using only the ridge paths (skeleton image) is too kindly with respect to the elastics distortions, but

Table 7. Summary of studies accuracies for other databases.

Author(s)	Features	Database	Results (%)
Maio and Maltoni (1997) [17]	Minutiae	14 images (7 from the NIST database [60], 4 from an FBI sample set, 3 from an opto-electronic device)	dropped 4.51 false 8.52 enhanced minutiae 13.03
Feng et al. (2004) [19]	Minutiae	100 fingerprint images from FVC2002's database	dropped minutiae 6.1 spurious minutiae 12.6
Jiang et al. (2001) [18]	Minutiae	410 images of size 300 × 300 and 4000 images of size 512 × 512 from NIST Fingerprint Database 4	90
Marana and Jain (2005) [41]	Ridges	MSU-VERIDICOM fingerprint database	EER 3.03
Jain et al. (2006) [20]	Minutiae, pores and ridge contours	1640 fingerprint images	EER 4.9
Jain et al. (2007) [21]	Minutiae, pores and ridge contours	1640 fingerprint images	EER 3.3
Zhao et al. (2008) [36]	Pores	198 fingerprint images (320 × 240)	EER 4.04
Vatsa et al. (2009) [28]	Minutiae, ridge contours and pores	5500 fingerprint images from 550 classes (10 images per class) captured using an optical scanner	96.37 with FAR: 0.01
Zhao et al. (2010) [2]	Pores and Minutiae	1480 fingerprint images (320 × 240) and 1480 fingerprint images (640 × 480)	EER 0.53
Zhao et al. (2010) [35]	Minutiae	210 partial fingerprint images from 35 fingers as the training set. 1480 fingerprint fragments from 148 fingers (including the training set) as the test.	EER 29.5
Win and Sein (2011) [24]	Ridges	300 images from NRC card and FVC2000 database.	EER 0.05
Silva and Jeronimo (2013) [40]	Pores	PolyU High-resolution-fingerprint Database	true rate detection: 91.2 false rate detection: 19.2
Shalaby and Ahmad (2013) [9]	Minutiae, orientation, singular points and curvature	FVC2006 DB1	EER 5.21

this could provoke an increase of the false positives because there are many ridge patterns that seem quite similar. The merge of several features seems to be the most promising alternative. Automatic fingerprint matching algorithms must be defined more similar to the work procedure of the human fingerprint experts.

Some general problems found in latent fingerprint matching are:

- A robust feature extraction is needed (principally to low-quality fingerprints, e.x. latent impressions).
- To extract features from the fingerprint ridge pattern that best represent fingerprints of low quality (e.g. folds, scars, dots, incipient ridges).
- To define a robust representations for features extracted from the ridge pattern (representations invariant to non-linear distortions).
- Fingerprint classification methods that perform an efficient search in a database are needed. Fingerprint classification algorithms still have low accuracies.
- Automated systems do not perform yet the ridge matching in the same way that fingerprint experts do.

Most matching algorithms which use extended features are implemented by the fusion of different matchers. Human examiners merge the fingerprint features while the comparison is being performed. This is a goal to reach by the automatic fingerprint matching algorithms. All features must be used simultaneously, because all of them are closely related to each other. That is why, *the design of descriptors that properly merge the minutiae and extended features* will be our principal research line. The representation of some extended features (ex. ridge path using sampling points) must be rethought, because it is not discriminatory enough.

References

1. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. Handbook of fingerprint recognition. *Second Edition, Springer*, 2009.
2. Qijun Zhao, David Zhang, Lei Zhang, and Nan Luo. Adaptive fingerprint pore modeling and extraction. *Pattern Recognition*, 43(8):2833–2844, 2010.
3. Jianjiang Feng and Anni Cai. Fingerprint representation and matching in ridge coordinate system. In *ICPR (4)*, pages 485–488, 2006.
4. Jianjiang Feng, Zhengyu Ouyang, and Anni Cai. Fingerprint matching using ridges. *Pattern Recognition*, 39(11):2131–2140, 2006.
5. Xuchu Wang, Jianwei Li, and Yanmin Niu. Fingerprint matching using orientationcodes and polylines. *Pattern Recognition*, 40(11):3164–3177, 2007.
6. nil K. Jain, Jianjiang Feng, Abhishek Nagar, and Karthik Nandakumar. On matching latent fingerprints. In *Proceedings of CVPR Workshops on Biometrics*, pages 1–8, June 2008.
7. H. Choi, K. Choi, and J. Kim. Fingerprint matching incorporating ridge features with minutiae. *IEEE Transactions on Information Forensics and Security*, 6(2):338–345, 2011.
8. Anil K. Jain and Jianjiang Feng. Latent fingerprint matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(1):88–100, 2011.
9. M. A. Wahby Shalaby and M. Omair Ahmad. A multilevel structural technique for fingerprint representation and matching. *Signal Processing*, 93:56–69, 2013.
10. CDEFFS. Cdeffs, data format for the interchange of fingerprint, facial, & other biometric information, working draft version 0.2, January 2008.
11. Edward Richard Henry. *Classification and Uses of Finger Prints*. George Routledge and Sons, London, 1900.
12. Anil K. Jain. Automatic fingerprint matching using extended feature set. Technical Report 2007-RG-CX-K183, Department of Computer Science, Michigan State University, 2011.
13. SWGFAST. Swgfast, memo to mike mccabe (nist) regarding ansi/nist itl 1-2000, Nov. 2005.
14. Dr. C. José Ruiz Shulcloper MSc. Miguel Ángel Medina Pérez. Estado del arte de combinación de algoritmos de cotejo de huellas dactilares. Technical Report RT_017, CENATAV, September 2009.
15. Anil K. Jain, Lin Hong, and Ruud M. Bolle. On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):302–314, 1997.
16. Jianjiang Feng, Zhengyu Ouyang, Fei Su, and Anni Cai. An exact ridge matching algorithm for fingerprint verification. In *IWBRS*, pages 103–110, 2005.
17. Dario Maio and Davide Maltoni. Direct gray-scale minutiae detection in fingerprints. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(1):27–40, 1997.
18. Xudong Jiang, Wei-Yun Yau, and Wee Ser. Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge. *Pattern Recognition*, 34(5):999–1013, 2001.
19. Jianjiang Feng, Fei Su, and Anni Cai. Robust ridge following in fingerprints. In *SINOBIOMETRICS*, pages 424–431, 2004.

20. Anil K. Jain, Yi Chen, and Meltem Demirkus. Pores and ridges: Fingerprint matching using level 3 features. In *ICPR (4)*, pages 477–480, 2006.
21. Anil K. Jain, Yi Chen, and Meltem Demirkus. Pores and ridges: High-resolution fingerprint matching using level 3 features. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):15–27, 2007.
22. A Ravishankar Rao. A taxonomy for texture description and identification. *New York: Springer-Verlag*, 1990.
23. Anil K. Jain, Abhishek Nagar, and Karthik Nandakumar. Latent fingerprint matching. Technical Report MSU-CSE-07-203, National Institute of Justice, December 2007.
24. Zin Mar Win and Myint Myint Sein. An efficient fingerprint matching system for low quality images. *International Journal of Computer Applications*, 26(4):5–12, July 2011. Published by Foundation of Computer Science, New York, USA.
25. Geok See Ng, Ruowei Zhou, and Hiok Chai Quek. A novel single pass thinning algorithm. *IEEE Transaction on System Man and Cybernetics*, 1994.
26. Xiaohui Xie, Fei Su, and Anni Cai. Ridge-based fingerprint recognition. In *ICB*, pages 273–279, 2006.
27. Pei hua Chen and Xiao guang Chen. A new approach to healing the broken lines in the thinned fingerprint image. *Journal of China Institute of Communications*, 25(6):115–119, 2004.
28. Mayank Vatsa, Richa Singh, Afzel Noore, and Sanjay K. Singh. Combining pores and ridges with minutiae for improved fingerprint verification. *Signal Processing*, 89(12):2676–2685, 2009.
29. Andy Tsai, Anthony J. Yezzi Jr., and Alan S. Willsky. Curve evolution implementation of the mumford-shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Transactions on Image Processing*, 10(8):1169–1186, 2001.
30. Tony F. Chan and Luminita A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.
31. Pavlidis T. *Algorithms for Graphics and Image Processing*. Springer, 1982.
32. Shashi Kumar D.R., Kiran Kumar K., K.B. Raja, R. K. Chhotaray, and Sabyasachi Pattnaik. Hybrid fingerprint matching using block filter and strength factors. *International Conference on Computer Engineering and Applications*, 1:476–480, 2010.
33. Shashi Kumar D. R., R. K. Chhotaray, K.B. Raja., and Sabyasachi Pattanaik. Fingerprint verification based on fusion of minutiae and ridges using strength factors. *International Journal of Computer Applications*, 4(1):1–8, July 2010. Published By Foundation of Computer Science.
34. Lei Xu, Erkki Oja, and Pekka Kultanen. A new curve detection method: Randomized hough transform (rht). *Pattern Recognition Letters*, 11(5):331–338, 1990.
35. Qijun Zhao, David Zhang, Lei Zhang, and Nan Luo. High resolution partial fingerprint alignment using pore-valley descriptors. *Pattern Recognition*, 43(3):1050–1061, 2010.
36. Qijun Zhao, Lei Zhang, David Zhang, Nan Luo, and Jing Bao. Adaptive pore model for fingerprint pore extraction. In *ICPR*, pages 1–4, 2008.
37. Asker M. Bazen and Sabih H. Gerez. Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):905–919, 2002.
38. Lin Hong, Yifei Wan, and Anil K. Jain. Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(8):777–789, 1998.
39. David Zhang, Feng Liu, Qijun Zhao, Guangming Lu, and Nan Luo. Selecting a reference high resolution for fingerprint recognition using minutiae and pores. *IEEE T. Instrumentation and Measurement*, 60(3):863–871, 2011.
40. Raoni Florentino da Silva Teixeira and Neucimar Jerônimo Leite. On adaptive fingerprint pore extraction. In *ICIAR*, pages 72–79, 2013.

41. Aparecido Nilceu Marana and Anil K. Jain. Ridge-based fingerprint matching using hough transform. In *SIBGRAPI*, pages 112–119, 2005.
42. Jianjiang Feng and Anni Cai. Fingerprint indexing using ridge invariants. In *no me se en donde*, pages 1–4, 2006.
43. Gang Fang, Sargur N. Srihari, Harish Srinivasan, and Prasad Phatak. Use of ridge points in partial fingerprint matching. In *Proc. of SPIE: Biometric Technology for Human Identification IV*, 2007.
44. G. Fang and State University of New York at Buffalo. *Representative Ridge Points in Fingerprints: A Modified Minutiae Matching Algorithm and Analysis of Individuality*. State University of New York at Buffalo, 2007. ISBN 9781109826173. URL `http://books.google.com.cu/books?id=pSgHtwAACAAJ`.
45. Anil K. Jain, Salil Prabhakar, Lin Hong, and Sharath Pankanti. Filterbank-based fingerprint matching. *IEEE Transactions on Image Processing*, 9(5):846–859, 2000.
46. Xiaolong Zheng and Yangsheng Wang. Fingerprint matching based on ridge similarity. In *ICASSP*, pages 1701–1704, 2008.
47. Marius Tico and Pauli Kuosmanen. Fingerprint matching using an orientation-based minutia descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(8):1009–1014, 2003.
48. Pen Li, Xin Yang, Qi Su, Yangyang Zhang, and Jie Tian. A novel fingerprint matching algorithm using ridge curvature feature. In *ICB*, pages 607–616, 2009.
49. Jucheng Yang. Non-minutiae based fingerprint descriptor. *Biometrics*, 2011.
50. Ju Cheng Yang and Dong-Sun Park. A fingerprint verification algorithm using tessellated invariant moment features. *Neurocomputing*, 71(10-12):1939–1946, 2008.
51. Ju Cheng Yang and Dong-Sun Park. Fingerprint verification based on invariant moment features and nonlinear bpnn. *International Journal of Control, Automation, and Systems*, 6(6):800–808, 2008.
52. Haitham Hasan and Sameem Abdul Kareem. Fingerprint image enhancement and recognition algorithms: a survey. *Neural Computing and Applications*, 23(6):1605–1610, 2012.
53. Alessandra A. Paulino, Jianjiang Feng, and Anil K. Jain. Latent fingerprint matching using descriptor-based hough transform. *IEEE Transactions on Information Forensics and Security*, 8(1):31–45, 2013.
54. Nalini K. Ratha, Kalle Karu, Shaoyun Chen, and Anil K. Jain. A real-time matching system for large fingerprint databases. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(8):799–813, 1996.
55. Ram P. Krish, Julian Fierres, Daniel Ramos, Javier Ortega-Garcia, and Josef Bigun. Pre-registration for improved latent fingerprint identification. In *Proceedings of IAPR/IEEE 22nd International Conference on Pattern Recognition, ICPR2014*, pages 696–701, August 2014.
56. Ram P. Krish, Julian Fierres, Daniel Ramos, Javier Ortega-Garcia, and Josef Bigun. Partial fingerprint registration for forensics using minutiae-generated orientation fields. In *2nd International Workshop on Biometrics and Forensics (IWBF2014)*, Valletta, Malta, March 2014.
57. Jianjiang Feng and Anil K. Jain. Fingerprint reconstruction: From minutiae to phase. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2):209–223, 2011. ISSN 0162-8828. doi: http:\-//doi.\-ieeecomputersociety.\-org/\-10.1109/\-TPAMI.2010.77.
58. Ala Balti and Mounir Sayadi. A new fingerprint identification approach based on svd features. In *Proceedings of 1st International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2014*, pages 301–304, Sousse, Tunisia, March 2014.
59. Mario González, David Dominguez, Francisco B. Rodríguez, and Ángel Sánchez. Retrieval of noisy fingerprint patterns using metric attractor networks. *International Journal of Neural Systems*, 24(7), 2014.
60. C.I. Watson and C.L. Wilson. *Fingerprint Database, Special Database 4*, April 1992.
61. Zhiheng Wang, Fuchao Wu, and Zhanyi Hu. Msld: A robust descriptor for line matching. *Pattern Recognition*, 42(5):941–953, 2009.

62. Anush Sankaran, Tejas I. Dhamecha, Mayank Vatsa, and Richa Singh. On matching latent to latent fingerprints. In *IJCB*, pages 1–6, 2011.
63. V. N. Dvornychenko. Evaluation of fusion methods for latent fingerprint matchers. In *ICB*, pages 182–188, 2012.