

Unsupervised Fingerprint Segmentation Based on Multiscale Directional Information

Raoni F.S. Teixeira and Neucimar J. Leite

Institute of Computing, University of Campinas
Av. Albert Einstein, 1251 - Barão Geraldo - Campinas - SP CEP: 13083-852

Abstract. The segmentation task is an important step in automatic fingerprint classification and recognition. In this context, the term refers to splitting the image into two regions, namely, *foreground* and *background*. In this paper, we introduce a novel segmentation approach designed to deal with fingerprint images originated from different sensors. The method considers a multiscale directional operator and a scale-space toggle mapping used to estimate the image background information. We evaluate our approach on images of different databases, and show its improvements when compared against other well-known state-of-the-art segmentation methods discussed in literature.

Keywords: fingerprint segmentation, biometrics, mathematical morphology, scale-space image simplification.

1 Introduction

In fingerprint context, the term *segmentation* usually indicates the separation of the fingerprint area (also known as *foreground*) from the image *background* [9]. This is illustrated in Fig. 1 which shows a fingerprint image (Fig. 1a), and a boundary separating the *foreground* from the *background* (Fig. 1b) in the original image.



Fig. 1. Segmentation example of a fingerprint image

This task is a very important step in automatic fingerprint classification and recognition, since many methods for extractions features depend naturally on it. For example, in Ref. [3] the segmentation is used to dismiss *singular points* detected in the *background* and in Ref. [12] it is used to reduce the search space in the *minutiae* detection.

In this paper, we present an unsupervised fingerprint segmentation algorithm which explores the simplification properties of a scale-space toggle operator [4] and a multi-scale directional operator [11]. As we will see elsewhere, unlike the well-known approach in [2], which considers a supervised technique based on a learning of specific parameters for each fingerprint sensor, our unsupervised method is quite simple, general and leads to segmentation results comparable with the accurate segmentations obtained in the aforementioned work.

The rest of this paper is organized as follows. Section 2 briefly reviews some fingerprint segmentation approaches found in literature. Section 3 and 4 introduce, respectively, the morphological transformations and the directional field estimation considered in this work. Section 5 describes the proposed segmentation method and Section 6 presents some experimental results. Finally, some conclusions are drawn in Section 7.

2 Related Work

Due to its importance, several approaches for fingerprint image segmentation can be found in the literature (for a review, see, for example, [9]). Generally, these approaches can be broadly classified as *supervised* and *unsupervised*, depending on the training or execution parameters specification.

The work in Ref. [12] introduces an *unsupervised* approach where each block of an image is classified as *background* and *foreground*, according to the variance of the image gray levels, in a direction orthogonal to the ridge orientation computed by a gradient-based method. Another *unsupervised* approach is proposed in Ref. [13], which considers the foreground segmentation through a convolution of each image block with a set of eight Gabor filters [7]. In [15], an *unsupervised* algorithm for rolled fingerprint is presented. The algorithm first binarizes the fingerprint image and then computes three iterations of morphological erosion to preserve only the connected components with the largest number of pixels.

The *supervised* approach described in [16] considers 11 image features and uses a neural network to learn the correctness of the estimated gradient-based orientation of the different blocks in a fingerprint image.

Finally, the method defined in [2] computes three pixel features (coherence, mean and variance) and uses a linear classifier to label the corresponding pixels as *background* or *foreground*. A *supervised* approach is used to train the linear classifier and a final morphological post-processing is performed to eliminate holes and regularize the external silhouette of the fingerprint area. Due to its specificity with respect to the considered database (sensor), this approach yields very accurate segmentation results and is the basis of many techniques, discussed in the literature, for fingerprint image enhancement and analysis [9,5].

3 Mathematical Morphology

The segmentation introduced here is based mainly on mathematical morphological transformations, briefly discussed in this section, and on a directional field

estimation approach presented in Section 4. The morphological transformations try to extract relevant structures of an image, by taking into account a small signal named structuring function [14].

Let $f : \mathcal{D} \subset \mathbf{Z}^2 \rightarrow \mathbf{Z}$ be an image function and $g : \mathcal{G} \subset \mathbf{Z}^2 \rightarrow \mathbf{Z}$ be this structuring function. The two fundamental morphological transformations, erosion and dilation, are:

Dilation : The dilation of a function $f(x, y)$ by a structuring function $g(a, b)$, $[\delta_g(f)](x, y)$, is given by: $[\delta_g(f)](x, y) = \max_{a, b \in \mathcal{G}} \{f(x + a, y + b) + g(a, b)\}$.
 Erosion : The erosion of the function $f(x, y)$ by a structuring function $g(a, b)$, $[\epsilon_g(f)](x, y)$, is given by: $[\epsilon_g(f)](x, y) = \min_{a, b \in \mathcal{G}} \{f(x + a, y + b) - g(a, b)\}$.

3.1 Multi-scale Morphological Transformations

The notion of scale (size) is related to the way we observe the physical world, where different features can be made explicit at different scales. In multiscale morphology, the notion of scale is achieved by scaling the structuring function $g_\sigma : \mathcal{G} \subset \mathbf{Z}^2 \rightarrow \mathbf{Z}$, such that [6]

$$g_\sigma(a, b) = |\sigma| g(\sigma^{-1}a, \sigma^{-1}b) \quad a, b \in \mathcal{G}, \forall \sigma \neq 0, \quad (1)$$

where σ conveys the notion of scale.

In this work, we consider the pyramidal structuring function, given by $g(a, b) = \max\{|a|, |b|\}$, whose scaled version is $g_\sigma(a, b) = -\sigma^{-1} \max\{|a|, |b|\}$.

Specifically, here we use a small 3×3 structuring function, where g_σ is zero at position $(0, 0)$ and $-\sigma^{-1}$ otherwise. As explored in [6] and [1], this non-flat structuring function possesses interesting image simplification properties including, for example, a monotonic filtering of the image extrema (its regional maxima and minima).

4 Directional Field Estimation

The directional field [9] is related to the global shape of a fingerprint and describes the local directions of the segment lines represented by a *ridge-valley* pattern. In this work, we use a multi-scale directional operator for estimating these patterns' orientation.

4.1 Multi-scale Directional Operator

The multi-scale directional operator [11] can be regarded as a generalization of the method presented in [10]. It is based on the observation that, in fingerprint images, the contrast between the direction following a ridge-valley pattern and its orthogonal orientation is greater than the contrast achieved for any other pair of directions.

Shortly, the multiscale operator estimates the orientation of each pixel (x, y) by dividing the semicircle in D discrete directions and computing the standard deviation (*std*) of the gray values for the set of line segments along each direction.

The coordinates (x, y) of the points in a discrete line segment with length n and direction α are computed by considering a sliding window Γ of size $n \times n$, such that

$$\begin{aligned} x &= x_{center} + p \times \cos(\alpha) \\ y &= y_{center} + p \times \sin(\alpha), \end{aligned} \quad (2)$$

for all p such that $-n/2 \leq p \leq n/2$. x_{center} and y_{center} are the coordinates of the point containing the sliding window Γ centered in this location.

The set s_i^n of D discrete lines with length n and discrete direction i is computed by repeating this procedure for all D directions ($i \in \{0, 1, \dots, D-1\}$), by respectively changing the value of α accordingly ($\alpha = 0, 1 \times 180/D, 2 \times 180/D, \dots, (D-1) \times 180/D$). The directional image d' at a finer scale is then computed as follows:

$$d'(x, y) = \begin{cases} i, & \text{if } std(s_i^n) < std(s_{\perp(i)}^n) \\ \perp(i), & \text{if } std(s_i^n) > std(s_{\perp(i)}^n) \\ v, & \text{otherwise} \end{cases} \quad (3)$$

where i and $\perp(i)$ corresponds to the pair of orthogonal directions exhibiting the highest contrast (e.g, $max_{j \in \{0, \dots, D/2-1\}} \{std(j) - std(\perp(j))\}$) and v is a special differentiable value representing the result of the function in a homogeneous region, i.e., region which does not have dominant direction. In our experiments, we consider $D = 8$ and $n = 35$ in the definition of the image given by Eq. 3.

Finally, the directional field image d is obtained by considering a window $\Omega : \mathcal{D} \subset \mathbf{Z}^2 \rightarrow \mathbf{Z}$ (also known as *smoothing window*) centered at each pixel on the d' image, according to the following equation.

$$d(x, y) = mode_{a,b \in \mathcal{D}} \{d'(x+a, y+b)\}, \quad (4)$$

where \mathcal{D} corresponds to the domain of the *smoothing window* and *mode* stands for the statistical mode which, in this case, computes the most frequent direction in Ω .

Note that the size of this smoothing window constitutes a scale factor in the sense that a small window yields a finer representation of the corresponding directional field, while a large one defines a coarser representation.

5 Proposed Approach

The segmentation method proposed in this paper consists of the following steps, as shows the flowchart in Fig. 2.

The *Finer directional field detection* block computes the orientation of the input image by considering the multi-scale directional operator (Section 4) with a small smoothing window, Ω_s , of size 5×5 . This smoothing operation defines an orientation image w representing a fine scale of the directional field. This fine representation preserves important orientation details used in the following algorithm step, namely *Background subtraction*, which performs a subtraction

of the background from the original fingerprint image. This operation separates the ridge-valley foreground regions from the rest of the image, as follows

$$f'(x, y) = f(x, y) - b(x, y), \quad (5)$$

where f' stands for the foreground image conveying the papillary information, f indicates the original input image, and b , an estimate of the background region.

To obtain this background estimation, we define a mapping that explores some simplification properties of a scale-space toggle operation. These properties include, for example, suppression of the image extrema (regional maxima and minima) in a monotonic way, i.e, without creation of new extrema [4]. The toggle transformation corresponding to the estimation b is given by

$$b(x, y) = \begin{cases} f(x, y), & \text{if } \phi_1^k(x, y) - f(x, y) \leq \phi_2^k(x, y) - f(x, y), \\ \phi_3(x, y), & \text{otherwise,} \end{cases} \quad (6)$$

where, again, f corresponds to the input image, $\phi_1^k = [\delta_{g_\sigma}(f)]^k$ and $\phi_2^k = [\epsilon_{g_\sigma}(f)]^k$ are, respectively, the dilation and erosion of f with the scaled structuring function g_σ , k times, and $\phi_3 = [\delta_{g_{\perp w}}(f)]$ corresponds to the linear dilation of f in the orthogonal direction of the orientation indicated by the finer directional field image w , at each location (x, y) .

Informally, the toggle mapping in Eq. 6 defines as estimate for b the pixels converging to the regional maxima in a uniform or quasi-uniform region, based on the proximity of $f(x, y)$ with the dilation $\phi_1^k(x, y)$, and in the regions conveying papillary information, represented by the directional dilation $\phi_3(x, y)$ which takes into account the orientation of the ridges in the original image f . Fig. 3 illustrates the above transformations for a noisy image of the FCV2000 database. This figure corresponds to the background subtraction given by Eq. 5. It is worth noting that regions corresponding to the image background and valleys (without papillary information) converged to the regional maxima of these regions represented here in white.

Finally, the *Coarser directional field detection* block in Fig. 2 considers a large smoothing window, Ω_l , of size 45×45 , in order to regionally define the dominant direction of the ridges in the foreground image f' . The outermost lines of the regions, containing the same directional field, constitute the fingerprint segmentation result depicted by the silhouettes in Fig. 4.

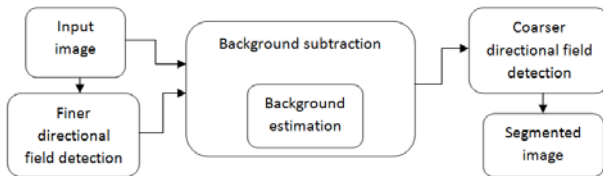


Fig. 2. Flowchart of the proposed segmentation method



Fig. 3. Segmentation example for an FCV2000 database image



Fig. 4. Segmentation examples of different FVC database images

6 Experimental Results

In order to demonstrate the effectiveness of our approach, we compare it against two state-of-the-art methods, namely, the Bazen and Gerez’s segmentation algorithm, and the one proposed by the NIST (National Institute of Standard and Technology) biometric system [15]. To do this, we consider two experiments. In the first, we take into account a set of images, typical of different types of sensors (optical, capacitive, thermal, etc), and obtained from the FVC (Fingerprint Verification Competition) databases [8]. In the second experiment, we work with the rolled fingerprint images provided by NIST and scanned from the FBI database. In the following results, the ground truth was obtained by a manual extraction of the fingerprint regions and the accuracy of the segmentation was established based on the F-measure defined as $\frac{2 \times Precision \times Recall}{Precision + Recall}$, where $Precision = \frac{tp}{tp + fp}$ and $Recall = \frac{tp}{tp + fn}$, fp , fn and tp correspond to false positives, false negatives and true positives, respectively.

6.1 FVC Databases

In this first experiment, we take into account 778 images originated from different sensors and belonging to the FVC2000 (DB1 and DB4) and FVC2002 and FVC2004 (DB1, DB2, DB3 and DB4) databases. We compare our segmentation results against the ones given by the Bazen and Gerez’s algorithm, one of the most accurate method described in literature [2]. The values in Table 1 show the equivalence of our results with those of Bazen and Gerez’s algorithm. Note that our approach does not suffer from the interoperability problem in the sense that it does not need any particular training related to different sensors or databases. Indeed, our method does not include any specific training and was designed to be robust enough to deal with images obtained from a variety of sensors, which means with data of different quality, resolution or gray-level.

Table 1. F-measure values for the segmentation of the FVC database images

Databases	Bazen and Gerez [2]	Our Approach
2000-DB3	0.93048	0.96165
2000-DB4	0.97909	0.95933
2002-DB1	0.96828	0.98211
2002-DB2	0.93827	0.95757
2002-DB3	0.97411	0.96044
2002-DB4	0.98215	0.97939
2004-DB1	0.98347	0.98812
2004-DB2	0.92180	0.90778
2004-DB3	0.96060	0.97837
2004-DB4	0.97727	0.96519

6.2 NIST Database

In the second experiment, we consider a set of images from the NIST database and compare the results obtained by considering the segmentation approach of the NIST system, Bazen and Gerez’s algorithm, and our proposed method. These results are indicated in Table 2, which shows a better performance of our approach with respect to the baseline algorithms.

Table 2. F-measure values for the methods considered with the NIST Database

Bazen and Gerez [2]	NIST	Our Approach
0.94193	0.95565	0.95602

Finally, note that for each database we trained the linear classifier used in [2] by considering 20 images and taking into account the following parameters: epochs = 10^4 , $\eta_0 = 10^{-4}$, $\tau = 10^4$ and Gaussian window with $\sigma = 9$. The fixed parameters obtained through experimental tests, concerned with Eq. 6, and related to our approach are as follows: number of iterations $k = 15$, scale $\sigma = 25$ of the morphological transformations (dilation and erosion), and length $|g| = 17$ of the linear structuring element of the orthogonal dilation given by Φ_3 . Fig. 5 shows a segmentation example of an image of the NIST database.

**Fig. 5.** Segmentation examples of NIST, Bazen and Gerez and Our Approach

7 Conclusions and Future Works

In this paper, we introduced a novel segmentation method designed to deal with fingerprint images captured from different sensors. To assess the method, we have performed experiments with more than 800 images of different databases. These experiments show that the proposed method yields accurate and robust segmentation results with the great advantage of being, unlike other approaches, independent from the type of the different available sensors. Indeed, it is worth noting that most existing fingerprint segmentation algorithms are based on a prior classification of the images foreground and background, according to a certain knowledge. Further, thresholding or training techniques are used in the segmentation process, which algorithmically imply a lack of sensor interoperability. In our case, we avoid this aspect by taking into account global characteristics of a fingerprint image explored here by means of its multiscale representation and the directional field inherent to its basic structure.

References

1. Dorini, L.B., Leite, N.J.: A scale-space toggle operator for morphological segmentation. In: Proceedings of 8th Int. Symp. on Mathematical Morphology, vol. 1, pp. 101–112 (2007)
2. Bazen, A.M., Gerez, S.H.: Segmentation of fingerprint images. In: Proc. of the Workshop on Circuits Sys. and Signal Processing, pp. 276–280 (2001)
3. Bazen, A.M., Gerez, S.H.: 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)
4. Baldo Dorini, L., Jerônimo Leite, N.: Multiscale morphological image simplification. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 413–420. Springer, Heidelberg (2008)
5. Yang, G., Zhou, G.-T., Yin, Y., Yang, X.: K-means based fingerprint segmentation with sensor interoperability. *EURASIP Journal on Advances in Signal Processing*, 1–12 (2010)
6. Jackway, P.T., Deriche, M.: Scale-space properties of the multiscale morphological dilation-erosion. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(1), 38–51 (1996)
7. Jain, A.K., Farrokhnia, F.: Unsupervised texture segmentation using gabor filters. *Pattern Recogn.* 24(12), 1167–1186 (1991)
8. Maio, D., Maltoni, D., Cappelli, R., Wayman, J., Jain, A.: Fvc2000: Fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 402–412 (2002)
9. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: *Handbook of Fingerprint Recognition*, 2nd edn. Springer, Heidelberg (2009)
10. Mehtre, B.: Fingerprint image analysis for automatic identification. *Machine Vision and Applications* 6, 124–139 (1993), doi:10.1007/BF01211936
11. Oliveira, M.A., Leite, N.J.: A multiscale directional operator and morphological tools for reconnecting broken ridges in fingerprint images. *Pattern Recognition* 41(1), 367–377 (2008)

12. Ratha, N.K.: Adaptive flow orientation based feature extraction in fingerprint images. *Pattern Recognition* 28, 1657–1672 (1995)
13. Shen, L., Kot, A.C., Koo, W.M.: Quality measures of fingerprint images. In: Bigun, J., Smeraldi, F. (eds.) *AVBPA 2001*. LNCS, vol. 2091, pp. 266–271. Springer, Heidelberg (2001)
14. Soille, P.: *Morphological Image Analysis*. Springer, Heidelberg (1999)
15. Watson, C.I., Garris, M.I.D., Tabassi, E., Wilson, C.L., McCabe, R.M., Janet, S., Ko, K.: User's guide to nist biometric image software (nbis). Tech. rep., NIST Research Report (NISTIR) - 7392 (2007)
16. Zhu, E., Yin, J., Hu, C., Zhang, G.: A systematic method for fingerprint ridge orientation estimation and image segmentation. *Pattern Recogn.* 39(8), 1452–1472 (2006)