

Pattern Recognition 34 (2001) 1907–1925

PATTERN RECOGNITION

www.elsevier.com/locate/patcog

# Fingerprint ridge allocation in direct gray-scale domain<sup> $\ddagger$ </sup>

Jeng-Horng Chang, Kuo-Chin Fan\*

Department of Electrical Engineering, Institute of Computer Science and Information Engineering, National Central University, Chung-Li, 32054, Taiwan, ROC

Received 9 September 1999; received in revised form 3 May 2000; accepted 10 August 2000

# Abstract

Ridges and ravines are the main components constituting a fingerprint. Traditional automatic fingerprint identification systems (AFIS) are based on minutiae matching techniques. The minutiae for fingerprint identification are defined by ridge termination and ridge bifurcation. Most AFIS perform ridge line following process to automatically detect minutiae based on binary or skeleton fingerprint image. For low-quality fingerprint images, the preprocessing stage of an AFIS produces redundant minutiae or even destroys real minutiae. The minutiae detection algorithms in direct gray-scale domain have been developed to overcome these problems. The first step of gray-scale minutiae detection algorithm is to determine ridge locations and then perform gray-scale ridge line following algorithm to extract minutiae. However, the existing gray-scale minutiae detection techniques can only work on partial fingerprint image due to the ignorance of image background. Moreover, the gray value variation inside a ridge also generates redundant ridge points. In this paper, we propose a novel method, based on gray-level histogram decomposition, to locate the ridge points in complete fingerprint images. By decomposing the gray-level histogram, redundant ridge points can be eliminated according to some statistical parameters. Experimental results demonstrate that the correct rate can be over 95% even applied to poor-quality fingerprint images. © 2001 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Fingerprint identification; Ridge; Minutiae; Feature extraction; Histogram decomposition

# 1. Introduction

Fingerprints have been used as a personal identification tool for more than 100 years. The major reasons are due to their uniqueness and unchangeable properties. Governments who collect fingerprints for criminal identification and business who collect fingerprints for security purpose store tremendous amount of fingerprint images that continuously increase the importance of automatic fingerprint identification systems. Many automatic fingerprint identification systems (AFIS) have been proposed over the last 30 years [1–3]. The purpose of an AFIS is to find out whether the individual represented by an incoming fingerprint image is the same as an individual represented by one of a large filed fingerprint image database. In a fingerprint image, ridges and ravines are the main constituting components and the minutiae for fingerprint identification are defined by the ridge flow interruption, such as ridge termination and ridge bifurcation. Most automatic fingerprint verification systems verify fingerprints by minutiae matching techniques. Fig. 1 shows the flow diagram of an automatic fingerprint matching system. In this system, the feature extraction stage obtains the minutiae in the fingerprint image by recording their coordinates and tangent directions. The matching process is performed by comparing the minutiae of an incoming fingerprint with the minutiae of fingerprint image files in database until the system find one identical fingerprint image.

The fingerprint acquisition process can be classified into three categories. They are ink technique, optical

<sup>&</sup>lt;sup>\*</sup>This work was supported in part by National Science Council under grant NSC 89-2213-E-008-030.

<sup>\*</sup>Corresponding author. Tel.: + 886-3-4227151, ext. 4453; fax: + 886-3-4222681.

E-mail address: kcfan@csie.ncu.edu.tw (K.-C. Fan).



Fig. 1. An overview of automatic fingerprint identification process.

prisms and holograms. Fingerprint images, which are acquired by ink technique, often produce regions that miss some information due to excessive inkiness or ink deficiency [4]. For acquisition techniques that use optical prisms or holograms, inadequate pressure while pressing finger on optical surface will generate nonuniform illuminated regions in fingerprint images. Furthermore, the prominence of ridge lines from the fingerprints of elder peoples or manual workers can be considerably lower such that the fingerprint pattern might be unreadable. In addition, fingerprint skin diseases, injure, skin moisture or slightly movement while acquiring fingerprint images also produce smudged and noisy regions.

The preprocessing stage plays an important role in automatic fingerprint identification system because the quality of acquired fingerprint image always cannot meet the requirement of most automatic identification systems. From the properties as described above, we know that we cannot improve the identification accuracy if the fingerprint preprocessing technique is not good enough. Traditional fingerprint image preprocessing process usually consists of five stages:

- (1) Filtering fingerprint image to enhance the fingerprint ridges.
- (2) Adaptive segmentation to separate the fingerprint ridges from ravines.
- (3) False minutiae reduction through refinement of abnormal ridge line flow.
- (4) Thinning process that reduces the ridges to one pixel width.
- (5) Noise removal process for eliminating pores and spurs produced by thinning.

The enhancement process increases the contrast between the foreground ridges and the background [5]. A robust segmentation method is required for detecting nonuniform regions and should be insensitive to the contrast of the original images. The composite method [6] that combines segmentation methods based on direction and variance information is promising. Since false minutiae caused by ridge line fragment will appear after the binarization process, a refinement process is necessary to reconstruct some lost information [7]. Fig. 2 illustrates the images after binarization and thinning processes. As we noticed, there exist many redundant minutiae in both binary and skeleton images which do not appear in the original image.

These preprocessing procedures always take over 95% of the identification time [8]. Thus, reducing some of the desired preprocessing stages but keeping the performance means the increase in identification speed. However, there is a trade off between the identification speed and accuracy. Low-quality fingerprint images will decrease the identification accuracy, whereas good-quality images will require much more preprocessing time. Therefore, it is necessary to develop an automatic finger-print identification system which can extract minutiae in gray-scale domain, i.e., no traditional preprocessing is necessary, instead of those systems using binary or skeleton fingerprint images.

Recently, Maio and Malton have developed a minutiae detection method in direct gray-scale domain [4]. By allocating the ridge positions as the start points, a graylevel ridge line following algorithm is proposed by computing the tangent direction of each ridge point sliding the ridges and then the minutiae in the fingerprint image





(c)

Fig. 2. An example illustrating various fingerprint images. (a) Original image. (b) Binary image. (c) Skeleton image.

can be extracted. Their work is equivalent to the minutiae extraction stage in a traditional AFIS but without any preprocessing procedure. However, the major disadvantage of their method is that their algorithm treats the fingerprint image analysis as a bimodal problem. Actually, a fingerprint image is constructed by three portions: ridges, ravines and background. Therefore, fingerprint image analysis should be a trimodal problem [9]. In their approach, only ridges and ravines in fingerprint images are taken into consideration. This means that their method can only work on partial fingerprint images, which exclude the background.

In this paper, we propose a ridge allocation algorithm in gray-scale domain to find out the positions of ridges in a complete fingerprint image. In our approach, the global information about the range of ridges, ravines as well as background in the gray-level histogram are determined by a statistical analysis of this trimodal distribution. The effect of background is also considered in our method. Our ridge allocation algorithm can be considered as the preprocessing stage of an automatic fingerprint identification system in gray-scale domain. We have chosen to allocate ridges in a complete fingerprint image directly from the gray-scale domain without binarization and thinning for the following reasons [4]:

- A lot of information may be lost during the binarization process.
- The binarization technique has been proved to be unsatisfactory when applied to low-quality image, such as broken ridges.
- Binarization and thinning are time-consuming.

The rest of this paper is organized as follows. In Section 2, we analyse fingerprint image patterns in order to understand the intrinsic properties. In Section 3, a histogram decomposition method is introduced to extract the information about the range of ridges, ravines and background in gray-level histogram. In Section 4, we propose a concrete algorithm to solve the problem of allocating ridge position in a gray-scale fingerprint image. The difficulties encountered in the previous works like background elimination and identical ridge points elimination are also discussed in this section. Experimental results are demonstrated in Section 5. Finally, conclusions and future works are given in Section 6.

# 2. Gray-scale fingerprint image analysis

Let I be a  $p \times q$  gray-scale fingerprint image with G gray levels and gray (x, y) be the gray value of pixel I(x, y) with x = 1, ..., p and y = 1, ..., q. Then, the graylevel histogram H of image I is of the form  $H = \{H(j) | j \in [1, G]\}$ . H(1) to H(G) represent the histogram probabilities of the observed gray values from 1 to G, and  $H(g) = \#\{\text{gray}[I(x, y)] = g, g = 1, ..., G, x = 1, ..., p$  and  $y = 1, ..., q\}$ . For representation convenience, let gray level 1 be the brightest pixels and gray level G be the darkest pixels. Fig. 3 depicts the gray-level histograms of a complete fingerprint image with 256 gray levels and the two portions of this image with and without background, respectively. The gray-level histogram of a fingerprint image with background always possesses three distributions in the histogram.

Let  $z = \operatorname{gray}(x, y)$  correspond to image I with  $x = 1, \ldots, p$  and  $y = 1, \ldots, q$ . The discrete surface represents a small area of a fingerprint image, S, as shown in Fig. 4. In this fingerprint surface S, the protruding parts in the surface correspond to the fingerprint ridges, and the concave parts correspond to the fingerprint ravines. The fingerprint ridges can be defined as a set of local maximum points along the one direction in S, and the fingerprint ravines be the local minimum points.



Fig. 3. Three gray-level histograms derived from different parts of a fingerprint image. (a) The sub-image locating at the central part of the image with only two distributions. (b) Considering the whole image, there exist three distributions in the histogram. The cluster on the left belongs to the background. (c) The sub-image includes a small area of the background, which also exist three distributions.



Fig. 4. An example illustrating surface S. (a) Original sub-image of a fingerprint. (b) The corresponding discrete surface.

Let a section set  $\Omega$  be a set of points in a fingerprint image which belong to a line segment lying on the *xy*-plane. Then,  $\Omega$  can be defined as

# $\Omega = \{(x, y, z) | (x, y) \in \text{Line}((x_s, y_s), (x_t, y_t)) \text{ and } z \in [1, G]\},\$

where  $\text{Line}((x_s, y_s), (x_t, y_t))$  represents a line segment with start point  $(x_s, y_s)$  and termination point  $(x_t, y_t)$ . This means that we map the two-dimensional pixels of section set  $\Omega$  to one-dimensional points along the direction of Line. For the points in  $\Omega$ , their gray values g compose a histogram. The ridge allocation algorithm attempts to locate the ridge points by extracting local maximums in a section set  $\Omega$ . Shown in Fig. 5 is an example illustrating a section set and the corresponding ridge locations.

For fingerprint images with only ridges and ravines, i.e., bimodal distribution in the gray-level histogram, Maio and Malton [4] developed a minutiae detection method including the technique of locating ridge positions in direct gray-scale domain. Unfortunately, their method will fail if it is applied to a complete fingerprint image including background in the image. The major reason of the failure is that there still possess 'ripples' while acquiring a section set  $\Omega$  in the background part only. That is, the ridge allocation algorithm will extract local maximum points as ridge locations no matter they are coming from the pattern area, i.e., the area of only ridges and ravines, or not. Moreover, the ripple characteristic also exists if some part of section set  $\Omega$  intersect with the same ridge line. The pores of sweat gland and moisture on the skin make the ridge line a harsh surface on a fingerprint image. Redundant ridge points will be produced if we only employ the phenomenon of height variation in  $\Omega$ . Fig. 6(a) illustrates the section set with the line segment extending to the image background. There exist several redundant ridge points generated from the ripple of background. Fig. 6(b) shows the section set where a part of line segment is lying on the same ridge line. It will also produce some redundant ridge points which belong to the inside of this ridge line.

Obviously, we can locate the genuine ridge points if we understand which pixels are lying on the pattern area, which pixels are lying on the background and which pixels belong to the same ridge line. A possible solution is to interpret the structure of the gray-level histogram to understand pixels in what range of gray levels are tend to be in the pattern area, and what range is for the background.

In order to estimate the gray-level ranges of ridges, ravines, and background in the histogram, we have to decompose the gray-level histogram to understand the information while allocating the ridge positions in a fingerprint image. This task is just like the multi-thresholding of a gray-scale image. For our problem, there will exist three distributions in the gray-level histogram which represent the clusters of ridges, ravines and background, respectively. Our goal is not only to extract the gray values which can separate ridges, ravines, and background in a fingerprint image, but also have to estimate some statistical parameters, i.e., means, variances and probabilities, that can represent these three clusters. There are several methods that can find the threshold values of a gray-level histogram by statistical approach [10-12]. However, the existing statistical histogram decomposition approaches always suffer from their high computational complexity. The parameter estimation procedure employ iterative parameter refinement until convergence. For real-time problems, such as automatic fingerprint identification, it is necessary to develop an efficient method that can decompose the histogram into several nonoverlapping clusters and then estimate the parameters representing each cluster.

In the next section, we will introduce a fast histogram decomposition method. In this method, the gray-level histogram is converted to mixture Gaussian distribution that has been formulated and proven by Zhuang [12].



Fig. 5. An example of the section sets. (a) The section set is acquired by drawing a line segment from lower-left to upper-left. (b) The numbers of the marks represent the appearing sequence of the ridge points from left to right in the section set.



Fig. 6. The section set with background. (a) A section set including some ridge points derived from the ripple of background. (b) A section set with some ridge points derived from the same ridge line.

That is, each object in an image will be a Gaussian-like distribution in this gray-level histogram with different mean and variance values. In our work, we use the statistical approach and some heuristic parameters to decompose this histogram into nonoverlapping distributions without a priori knowledge about the number of objects. The proposed method does not employ conventional iterative parameter refinement. Instead, by estimating the initial nonexact mean and variance values as the cues for determining the initial threshold value, the Skewness of certain interval in this candidate distribution is calculated to quickly locate the deterministic optimal estimation interval. After optimally estimating the mean and variance values of each distribution in the histogram, a maximumlikelihood-based decision criterion is applied to determine the optimal threshold values among distributions. Then, the information about the range of ridges in the gray-level histogram can help us in determining the genuine ridge points among a random selected section set  $\Omega$ .

#### 3. Fast gray-level histogram modeling and decomposition

Generally, there exist a number of 'mountains' in the histogram if it is a multimodal distribution. Each distribution in the histogram will map to an object in the image. For any gray-level histogram with n distributions, the multi-thresholding techniques are to automatically determine n - 1 threshold values that can be used to separate this multimodal histogram into n nonoverlapping distributions.

For nature scene with large samples, we assume that the observation comes from a mixture of n + 1 Gaussian distributions, name f, having respective means and variances  $(m_1, \sigma_1^2), \ldots, (m_{n+1}, \sigma_{n+1}^2)$  with respective proportions  $P_1, \ldots, P_{n+1}$ . Therefore, the mixture distributions reflected in the histogram will be in the form of

$$f(k) = \sum_{i=1}^{n+1} \frac{P_i}{\sqrt{2\pi\sigma_i}} \exp\left\{-\frac{1}{2}\left(\frac{k-m_i}{\sigma_i}\right)^2\right\}.$$

Our objective is to find the parameters, i.e., means, variances, and proportions, to satisfy the minimization

$$\min(|f-H|).$$

In order to decompose a gray-level histogram into several nonoverlapping distributions, we have to find the local minimums first and then perform further parameter estimation tasks. However, the histogram distribution, which was acquired from real-world scene, is always anomalously distributed. Hence, a histogram smoothing process is necessary before performing the decomposition process.

Let Wg be a Gaussian masking window with 2p + 1bins and  $b_k, k = 1, 2, ..., 2p + 1 (p \ge 0, b_k \ge 0, \sum b_k = 1)$ be the elements of Wg. The new gray level in  $\tilde{H}$  is calculated as the convolution of H and Wg:

$$\tilde{H} = H \otimes Wg,$$

where ' $\otimes$ ' denotes the convolution operation. Thus,

$$\tilde{H} = \{\tilde{H}(j)|j \in [1,G]\}$$

forms the smoothed histogram where

$$\tilde{H}(i) = \frac{1}{2p+1} \sum_{u=-p}^{p} b_{p+1+u} H(i+u)$$

for i = p + 1 to G - p.

Fig. 7 is an example of Gaussian masking window with seven bins. For a 2p + 1 bins masking window, each bin can be calculated by

$$b_k = 0.5(1 - \cos(\pi k/p)).$$

Fig. 8 shows the histograms before and after convoluting with a Gaussian masking window of 21 bins (p = 10).

After the smoothed histogram  $\tilde{H}$  has been obtained, the peaks and valleys in the histogram can be determined by the following rule: For any gray value  $i, i \in [1, G], \tilde{H}(i)$ is a peak if  $\tilde{H}(i) > \tilde{H}(i-1)$  and  $\tilde{H}(i) \le \tilde{H}(i+1)$ . On the other hand,  $\tilde{H}(i)$  is a valley if  $\tilde{H}(i) < \tilde{H}(i-1)$  and  $\tilde{H}(i) \ge \tilde{H}(i+1)$ .

Suppose that there exist an *n* distinct Gaussian clusters  $C_i$ , i = 1, ..., n, then  $\tilde{H}$  must have *n* peaks, denoted by R(1), ..., R(n), and n-1 valleys, denoted by V(1), ..., V(n-1). Then, the interval of  $C_i$  in the smoothed histogram  $\tilde{H}$  will be [V(i-1), V(i) - 1], with V(0) = 1 and V(n) = G + 1. We will define an optimal estimation interval within each cluster to estimate the parameters that can represent the distribution of the clusters.

The *Skewness*,  $\beta_1$ , involving the second- and thirdorder central moments can be defined as

$$\beta_1 = \frac{\mu_3}{\sqrt{\mu_2^3}}.$$

Central moments are defined as  $\mu_n = E\{(x - m)^n\} = \int_{-\infty}^{\infty} (x - m)^n f(x) dx$ . If the random variable **x** is discrete type with unknown mean value,  $\mu_n$  can be rewritten by its sample mean,  $\bar{m}$ , as  $\mu_n = \sum_i p_i(x_i - \bar{m})$ , with  $\bar{m} = \sum_i p_i x_i$ , where  $p_i$  is the occurrence probability of  $x_i$ . Here, *Skewness* is a symmetric measurement of distributions.  $\beta_1 > 0$  means that the distributions. For univariate normal distributions N( $m, \sigma^2$ ), since f(-x) = f(x), the odd-order central moments will all be zero. That is,  $\mu_3 = E\{(x - \bar{m})^3\} = 0$ . Thus,  $\beta_1$  will be zero if this distribution is normal distribution.

For mixture Gaussian distributions with clusters  $C_i$ , i = 1, ..., n, the overall *Skewness* for the range of each cluster is meaningless because each Gaussian cluster is contaminated by the neighboring clusters at the margins of both sides. However, the *Skewness* is also close to zero



Fig. 7. Gaussian masking window.



Fig. 8. The histograms before and after convolution. (a) Original histogram distribution. (b) Smoother histogram after convoluting with Gaussian masking window.

at a certain interval which is near the center of each cluster. Therefore, this special interval should be determined first to obtain optimal initial estimation of the cluster centers and then perform other further estimations and decisions.

For each cluster  $C_i$  in Gaussian mixture, we select the interval for initial cluster's mean value estimation by length  $r_i = \frac{1}{2}(V(i) - V(i - 1) - 1)$ . This length,  $r_i$ , possesses the properties of  $r_i \gg \sigma_i$  and  $r_i \propto \sigma_i$ . The length of interval satisfies *Tchebycheff inequality* and can be used as the length of optimal parameter estimation interval.

Now, let us define a searching window w with length  $r_i$  to search the location of optimal estimation interval of cluster  $C_i$ . The searching window w which starts by placing the leftmost point at V(i - 1) slides toward the end of cluster V(i) - 1 by moving one bin at a time. The searching process stops if the rightmost point reaches the end of cluster V(i) - 1. They will have  $V(i) - V(i - 1) - 1 - r_i$  searching windows. Meanwhile, the Skewness  $\beta_1$  is calculated for each searching window  $w_j$ ,  $j = 1, 2, ..., V(i) - V(i - 1) - 1 - r_i$ , denoted by  $\beta_1(w_j)$ . For our problem, the *Skewness* of each searching window can be calculated as

$$\tilde{m}_{w_j} = \frac{\sum_{i \in w_j} iH(i)}{\sum_{i \in w_j} H(i)}, \quad \mu_n(w_j) = \frac{\sum_{i \in w_j} (i - \tilde{m}_{w_j})^n H(i)}{\sum_{i \in w_j} H(i)}$$

and  $\beta_1(w_j) = \mu_3(w_j) / \sqrt{\mu_2^3(w_j)}$ .

Therefore, the optimal interval  $w^*$  for estimating the mean and variance of each cluster is determined by the interval which has minimum absolute skewness value. That is

$$w^* = \min_j |\beta_1(w_j)|.$$

The optimal estimation interval  $w_i^*$  of cluster  $C_i$ will be located at  $[a_i, b_i)$ . Fig. 9 is an example illustrating the position of searching windows w and the location of optimal estimation interval  $w^*$  for one distribution.



Fig. 9. Optimal searching interval.

Then the initial mean, variance and proportion of this cluster can be optimally determined by

$$\hat{m}_{i} = \frac{\sum_{k=a_{i}}^{b_{i}} kH(k)}{\sum_{k=a_{i}}^{b_{i}} H(k)}, \quad \hat{\sigma}_{i}^{2} = \frac{\sum_{k=a_{i}}^{b_{i}} (k - \hat{m}_{i})^{2} H(k)}{\sum_{k=a_{i}}^{b_{i}} H(k)}$$
  
and  $\hat{P}_{i} = \sum_{k=a_{i}}^{b_{i}} H(k) / \sum_{u=1}^{g} H(u).$ 

For the *i*th observation H(i), it is more likely generated by cluster  $C_k$  if

$$\frac{\hat{P}_{k}}{\sqrt{2\pi}\hat{\sigma}_{k}}\exp\left\{-\frac{1}{2}\left(\frac{i-\hat{m}_{k}}{\hat{\sigma}_{k}}\right)^{2}\right\}$$
$$>\frac{\hat{P}_{j}}{\sqrt{2\pi}\hat{\sigma}_{j}}\exp\left\{-\frac{1}{2}\left(\frac{i-\hat{m}_{j}}{\hat{\sigma}_{j}}\right)^{2}\right\}$$

for  $1 \leq j \leq n$ ,  $1 \leq k \leq n$ , and  $j \neq k$ .

If there are *n* clusters, we will obtain n - 1 threshold values  $T_i$ , i = 1, 2, ..., n - 1. Therefore, the *i*th threshold  $T_i$  can be determined as follows:

 $T_i = \max\{k: H(k) \text{ is generated by the ith Gaussian cluster}\}.$ 

Finally, for each cluster  $C_i$ , i = 1, 2, ..., n, the range becomes [T(i-1), T(i) - 1] with T(0) = 1 and T(n) = g. The mean, variance and proportion of the cluster can then be determined by the following equations:

$$m_i = \frac{\sum_{i \in C_i} iH(i)}{\sum_{i \in C_i} H(i)}, \quad \sigma_i^2 = \frac{\sum_{i \in C_i} (i - m_i)^2 H(i)}{\sum_{i \in C_i} H(i)}$$
  
and  $P_i = \sum_{i \in C_i} H(i) / \sum_{i=1}^g H(i).$ 

#### 4. Gray-scale fingerprint ridge allocation algorithm

For the discrete surface S in a fingerprint image I, we acquire a section set  $\Omega$  by intersecting the surface S with an arbitrary length line segment in randomly selected direction. By determining the local maximums in  $\Omega$ , we can extract ridge points which belong to this section set. However, noise is always present in fingerprint images no matter what fingerprint image acquisition method is used (as mentioned in Section 1). Thus, we first smooth the section set  $\Omega$  by convoluting it with a seven bins Gaussian masking window. Then, we determine the local maximums as the candidate ridge points by comparing the gray value variation of three consecutive points in  $\Omega$ . Fig. 10 illustrates the comparison of a section set  $\Omega$  before and after the smoothing process. The method of histogram smoothing and local minimums or local maximums extraction have been described in the previous section.

Suppose there exist *m* ridge points **R** and **n** valley points **V** in a section set  $\Omega$ , namely  $R_1, R_2, \ldots, R_m$  and  $V_1, V_2, \ldots, V_n$ , respectively. These ridge points and valley points are alternately distributed from left to right along the direction of section set  $\Omega$ . That is,  $R_1$  will be the



Fig. 10. Comparison of ridge section before and after smoothing.

leftmost ridge point in  $\Omega$  and  $R_m$  will be the rightmost one. The same condition also exists for the valley points. The ridge points **R** and the valley points **V** in  $\Omega$  compose a feature set, namely the *train* set. Since the location of section set  $\Omega$  is arbitrary selected in a fingerprint image, the initial element (the leftmost element) and the terminal element (the rightmost element) in the *train* set become uncertain. Moreover, some candidate ridge points are probably lying on the same ridge line and the locations in the section set  $\Omega$  are always unknown. These properties make the fingerprint ridge allocation an obscure problem and are difficult to find a straightforward solution.

# 4.1. Estimation of gray-level ranges of components in fingerprint images

First, we decompose the gray-level histogram of a fingerprint image to obtain the information about the global gray-level range of ridge points as well as the ravines and image background by the fast histogram decomposition algorithm as described in the previous section. Generally, as we have already shown in Fig. 3, there always exist three distinctive distributions in the gray-level histogram of a complete fingerprint image. After decomposing the histogram into three nonoverlapping clusters that represent the gray-level range of ridges, ravines and background, we will obtain two threshold values which separate these three clusters and some parameters, i.e., means, variances and probabilities, of the extracted clusters.

In Fig. 11, three clusters  $C_b$ ,  $C_v$ , and  $C_r$ , which correspond to the clusters of background, ravines and ridges in the histogram distribution from left to right, respectively, are separated by two threshold values  $T_B$  and  $T_R$ . The gray values of the pixels in each cluster will map into the gray-level range in the histogram from gray level 1 to  $T_B$  for cluster  $C_b$ ,  $T_B + 1$  to  $T_R$  for cluster  $C_v$  and

 $T_R + 1$  to G for pixels in cluster  $C_r$ . The cluster center m and the standard deviation  $\sigma$  of clusters  $C_b$ ,  $C_v$ , and  $C_r$  are denoted by  $m_b, m_v, m_r$  and  $\sigma_b, \sigma_v, \sigma_r$ , respectively. From the statistical point of view, the population of cluster  $C_r$  will concentrate in the range of  $[m_r - \sigma_r/2, m_r + \sigma_r/2]$ , and the population of  $C_v$  will concentrate in the range of  $[m_v - \sigma_v/2, m_v + \sigma_v/2]$ . Let us define another threshold value  $T_D$ , whose value is the gray-level difference between the ridge cluster  $C_r$  and the ravine cluster  $C_v$  by

$$T_D = \left(m_r - \frac{\sigma_r}{2}\right) - \left(m_v + \frac{\sigma_v}{2}\right),$$

 $T_D$  is the minimum requirement of the gray-level change of two adjacent elements in the *train* set if the candidate element belongs to the genuine ridge but are obtained from a blur or dark part in the fingerprint image. We will discuss the usage of  $T_B$ ,  $T_R$ , and  $T_D$  in the following part of this section.

Then, by analyzing the appearance sequence of elements and the degree of gray-level shift according to  $T_B$ ,  $T_R$ , and  $T_D$  for the adjacent elements in the *train* set, we can determine which ridge points come from the pattern area or which points are derived from the image background or inside the same ridge.

# 4.2. The ridge allocation algorithm

Let the function P represent the position of elements in a section set  $\Omega$ , and  $P(R_i) < P(R_j)$  represent the position of  $R_i$  which is at the left in  $\Omega$  compared to the position of  $R_j$ . That is,  $P(R_1) < P(R_2) < \cdots < P(R_m)$  and  $P(V_1) <$  $P(V_2) < \cdots < P(V_n)$ . From our observation, the section set  $\Omega$  always comes from the four types according to the role of both elements at the initial and the terminal



Fig. 11. Example of the decomposition result.

positions by the following rules:

$$\Omega \in \begin{cases} Type \ 1; \ \text{if } P(V_1) < P(R_1) \text{ and } P(R_m) < P(V_n) \\ \Rightarrow \text{init} = 0 \text{ and end} = 0, \\ Type \ 2; \ \text{if } P(V_1) < P(R_1) \text{ and } P(V_n) < P(R_m) \\ \Rightarrow \text{init} = 0 \text{ and end} = 1, \\ Type \ 3; \ \text{if } P(R_1) < P(V_1) \text{ and } P(R_m) < P(V_n) \\ \Rightarrow \text{ init} = 1 \text{ and end} = 0, \\ Type \ 4; \ \text{if } P(R_1) < P(V_1) \text{ and } P(V_n) < P(R_m) \\ \Rightarrow \text{ init} = 1 \text{ and end} = 1, \end{cases}$$

where 'init' and 'end' denotes the initial element and the terminal element in the *train* set, respectively. For representation convenience, we use zero as the valley points and one for the ridge points just like the representation of digital signal with only low- and high-voltage conditions. Fig. 12 illustrates these four types of section sets.

Since the ridge elements and the valley elements are alternately distributed in the *train* set, we define the reference element of a certain valley point  $V_i$ , i = 1, ..., n, to be the ridge point which is at the left of  $V_i$  and closest to  $V_i$ , named  $R_j$ . For the four different types of section sets as described above, the index j of the reference ridge can be determined by j = i + init - 1. That is, the refer-



Fig. 12. Four types of ridge sections.

ence ridge of the valley point  $V_i$  will be  $R_{i+\text{init}-1}$ . For Types 1 and 2 section sets, the reference ridge of  $V_i$  is  $R_{i-1}$  because init = 0 (starting from a valley point) for these two types. On the other hand, the reference ridge of  $V_i$  is  $R_i$  for Types 3 and 4 because init = 1 (starting from a ridge point). The information which is acquired from the  $(R_{i+\text{init}-1}, V_i)$  pair will give us an explicit solution in allocating the genuine ridge points.

In order to interpret the condition of elements contained in the *train* set, we divide the *train* set into three portions and discuss them individually. These three portions of elements are the initial elements, the intermediate elements, and the terminal elements.

#### 4.2.1. The initial elements

For Types 1 and 2 section sets, the starting element in the train set is a valley point without reference ridge point at its left. Therefore, we discard the initial valley element if this train set comes from Types 1 and 2 section sets. For Types 3 and 4 section sets, the initial element pair that belongs to the train set is  $(R_1, V_1)$  and  $R_1$  is the starting element (see Fig. 12).

#### 4.2.2. The intermediate elements

After eliminating the starting valley point for Types 1 and 2 section sets, the rest of elements in the *train* set are all intermediate elements excepts the terminal ridge points of Types 2 and 4 section sets. There exist three different conditions while observing the gray-level shift for the intermediate element pairs ( $R_{i+init-1}$ ,  $V_i$ ). Fig. 13 illustrate these three conditions. The element pairs with larger drop height are obtained from distinct ridges, and the element pairs with smaller drop height are coming from an identical ridge line. The image background also produces many element pairs.

We can extract the genuine ridge points in the *train* set by the following rules. Let the function gray( $\cdot$ ) denote the gray value of elements in the train set. For any candidate element pair  $(R_j, V_i)$ , j = 1, ..., m and i = 1, ..., n, there exist four possibilities for the belonging of element  $R_j$ .

- (1) R<sub>j</sub> is derived from the ripple of image background if the gray value of R<sub>j</sub> is smaller than T<sub>B</sub>. That is, R<sub>j</sub> ∈ C<sub>B</sub>if gray(R<sub>j</sub>) < T<sub>B</sub>. We will discard this ridge element R<sub>j</sub>:
- (2) R<sub>j</sub> is accepted as genuine ridge point if the element pair (R<sub>j</sub>, V<sub>i</sub>) satisfies the following conditions: The gray value of R<sub>j</sub> is larger than T<sub>R</sub> and the gray value difference of R<sub>j</sub> and V<sub>i</sub> exceeds T<sub>D</sub>. By defining the set of genuine ridge points as ℜ, R<sub>j</sub> ∈ ℜ if and only if gray(R<sub>j</sub>) > T<sub>R</sub> and (gray(R<sub>j</sub>) - gray(V<sub>i</sub>)) > T<sub>D</sub>.
- (3) Both R<sub>j</sub> and V<sub>i</sub> are derived from the same ridge line of a fingerprint image if the gray value of V<sub>i</sub> is larger than T<sub>R</sub> and the gray value difference of R<sub>j</sub> and V<sub>i</sub> is smaller than T<sub>D</sub>. That is, R<sub>j</sub>, V<sub>i</sub> ∈ C<sub>R</sub> if gray(V<sub>i</sub>) > T<sub>R</sub> and (gray(R<sub>j</sub>) gray(V<sub>i</sub>)) < T<sub>D</sub>. We will store this ridge element R<sub>j</sub> as a temporal ridge point R<sub>t</sub>. Several element pairs, which belong to the identical ridge line, appear following the first identical ridge element pair R<sub>t</sub>. We will search the other element pairs still left in the *train* set from left to



Fig. 13. Intermediate conditions.

right. If there exists an element pair  $(R_{k+\text{init}-1}, V_k)$  with i < k < n that satisfies rule (2), we mark the temporal ridge point  $R_t$  as a genuine ridge element and discard all ridge points from  $R_{j+1}$  to  $R_{k+\text{init}-1}$ . On the other hand, if there does not exist any element pair, which satisfies rule (2) while searching to the end of train set, we keep  $R_t$  and make decision according to the terminal elements.

(4) The element pair  $(R_j, V_i)$  that does not satisfy any of the rules (1), (2) or (3) will be discarded. Since there will exist some blur area in a fingerprint image which is produced by slightly movement while acquiring this fingerprint image, we will discard these uncertain ridge elements to avoid locating erroneous ridge points, i.e., the ridge points that are actually derived from background or ravines.

## 4.2.3. The terminal elements

The terminal elements are defined as the rightmost ridge element that has not been referenced by any element. Thus, only Types 2 and 4 section sets possess terminal ridge elements. Figs. 14(a) and (b) illustrate the terminal conditions for the section sets that are terminated inside a ridge. There will exist a temporal ridge  $R_t$  with  $P(R_t) < P(R_m)$ . For this condition, we accept the terminal ridge element  $R_t$  as a genuine ridge point without any limitation. The section sets as illustrated in Fig. 14(c), which possess the terminate ridge  $R_m$ , are acquired from Types 2 and 4 section sets. We accept the terminal ridge element  $R_m$  if the element pair  $(R_m, V_n)$ satisfies the acceptance rule (2).

The gray-scale ridge allocation algorithm is a combination of finding the genuine ridge points by the rules described in the three individual portions of the elements in the *train* set. After reaching the terminate element, the ridge elements  $R_l$ , with  $l \in [1, m]$  and  $R_l \in \Re$ , become the extracted ridge points in the section set  $\Omega$ . The corresponding coordinate of each ridge point in  $\Re$  is the location of a ridge in the fingerprint image **I**.

# 5. Experimental results

Some experiments have been conducted to evaluate the performance of the proposed gray-scale ridge allocation algorithm with *NIST Special Database 4* fingerprint images [13]. The fingerprint images were acquired and quantized into  $512 \times 512$  with 256 gray levels in the test data set. Since the gray-scale minutiae detection algorithm proposed by Maio [4] has already developed an explicit method to allocate all ridges in fingerprint images, our experiments will present only some critical conditions by a selected line segments as supplements. The experiments include the ordinary distinct ridge sections, the ridge section which extends to the image background and the ridge section with a large portion runs through a ridge line. The ridge sections derived from



Fig. 14. Terminal conditions.

a poor-quality fingerprint images with nonuniform illumination conditions or some contaminated areas are also considered to demonstrate the robustness of the proposed ridge allocation algorithm. For representation convenience, the section sets in our experiments were acquired along the direction of the line segment with an arrow. The allocation results presented on the ridge section are then marked back to the corresponding coordinate in the fingerprint image. Finally, a quantitative measure about the correctness of our method is presented.

#### 5.1. Distinctly distributed ridge sections

For the sub-images derived from a part of complete fingerprint images with only ridges and ravines, the ridge section will be distinctly distributed. For this condition, the histogram decomposition process extracts only one threshold value  $T_R$  and will ignore the background effect while locating the ridge points. In Fig. 15, the extracted ridge points in the section set do not locate on the highest bins because the smoothing step of section set prevents the appearance of noise inside the ridges. We can observe that some ridge points located.

# 5.2. Ridge sections extending to image background

Considering a complete fingerprint image with background, there exist fake ridge points due to the ripple of gray-level variation if the section set is extended to image background. These ridge points derived from the image background must be eliminated to prevent the generation of false minutiae while applying gray-scale ridge line following algorithm. Fig. 16 illustrates a section set with a portion lying on the image background and the corresponding ridge allocation results in a fingerprint image. For the extraction result of Maio's method [4], as shown in Fig. 16(a), there exist four fake ridge points due to the ripple caused by the gray-value variation on the points, which are lying on the background along the extraction



Fig. 15. Distinctly distributed ridge sections.



(b)

Fig. 16. Ridge section with background. (a) The extracted ridge points by Maio's method. There exist four fake ridge points on the image background. (b) The generated ripple of the background will be ignored in our method.

line. On the other hand, our method ignores the ripple coming from the background and extracts only five genuine ridge points that are locating at the pattern area (see Fig. 16(b)). As we noticed, the forth ridge points counting from the left of this section set with lower height due to the contrast deficiency can also be located correctly.

# 5.3. Ridge sections within a ridge line

The ridge lines of a fingerprint image are arbitrary distributed with various flow directions. An automatic system should be invariant with the rotation and translation of fingerprint images. For the line segment in extracting the section set with regulated direction in the automatic ridge allocation process, it might not vertically intersect with all ridge lines. A portion of the ridge section set probably runs through a ridge line and generates redundant ridge points due to the appearance of sweat pores on the fingertips. These redundant ridge points, which are locating inside an identical ridge line, should be eliminated in the ridge allocation process to prevent tracking a ridge line repeatedly in the ridge line following process.

Fig. 17 illustrates the ridge section run through a ridge line. For the methods that extract ridge points depending only on local maximums, such as Maio's method, some redundant points will be generated. (see Fig. 17(a)). In Fig. 17(b), the ridge points derived from the same ridge line are successfully eliminated by using our method. Although some identical ridge points which are actually coming from the same ridge line still cannot be eliminated perfectly due to the contrast deficiency of fingerprint images, our method can eliminate almost all redundant ridge points no matter they are derived from the image background or belonging to the same ridge lines.



Fig. 17. Ridge section in a ridge line. (a) For Maio's method, there exist two redundant points for the section set runs through a ridge. (b) The ridge points that belong to the same ridge line are eliminated correctly in our method.

#### 5.4. Nonuniform illumination section sets

Most fingerprint images, which are acquired from biometric systems, always possess the nonuniform illumination property. We will demonstrate the performance of our ridge allocation algorithm while applying to nonuniform illumination areas in fingerprint images. Figs. 18(a) and (b) illustrates the section set derived from an area of fingerprint with nonuniform illumination. Although the gray value of the ridge points with higher illuminance are lower than their neighboring ridge points, our method can still locate them correctly because the gray-level differences are large enough to satisfy the acceptance rule. On the other hand, for the binary image as shown in Fig. 18(c), there are four ridge points missing because some ridge lines are broken into fragments after the binarization process. The extraction results of the corresponding skeleton image cannot find all genuine ridge points. As illustrated in Fig. 18(d), there exist five missing points. This evidence shows that binary or skeleton image-based minutiae detection algorithm will miss the detection of some minutiae in the nonuniform illuminated areas.

## 5.5. Ridge sections derived from contaminated areas

Due to the moisture or slightly movement while acquiring a fingerprint image, ridges will be blended with ravines that result in some blur areas. These critical contaminated areas will cause many false minutiae after the binarization or thinning process. Fortunately, our direct gray-scale method can allocate most ridge points even the section sets are derived from contaminated areas. Fig. 19(a) shows the contaminated sub-image of a fingerprint image. The extracted section and the ridge location after applying our gray-scale ridge allocation algorithm are shown in Fig. 19(b). Although missing some genuine ridge points, our method can still locate most of the ridges in this contaminated area. On the other hand, we could hardly find any possible ridge line on the binary or skeleton image at the same position of the line segment lying on the original image. As illustrated in Figs. 19(c) and (d), we can extract some ridge points intersecting with the extraction line. However, no ridge line following algorithm can find the successive ridge path while applying to the contaminated areas in binary or skeleton images.



Fig. 18. Ridge section for nonuniform area. (a) The original gray-scale image, (b) derived ridge section, (c) The corresponding binary image and the extraction result missed four ridge points. (d) The corresponding skeleton image and five ridge points cannot be extracted.

Finally, we perform a quantitative measurement about the accuracy of the proposed gray-scale ridge allocation algorithm with six fingerprint images as shown in Fig. 20. The size of these test fingerprint images are  $512 \times 512$  by 500 dpi resolution and are quantized into 256 gray levels. In order to demonstrate that the proposed method is not only suitable for some special types of fingerprints, these test patterns are selected from different classes of the well-known *Henry's Classification* [14]. In this experiment, ten ridge sections were acquired by equally spaced straight line segments acrossing each fingerprint image in both vertical and horizontal directions. It might have three kinds of errors generated in this experiment:

1. *Missing points*: The ridge points in section sets, which belong to genuine ridges but have not been allocated, are counted as missing points.

- Erroneous points: Points which belong to valleys but are marked as ridges.
- 3. *Redundant points*: There only exist one ridge point for one ridge line. The extra points are considered as redundant points.

By applying the proposed automatic ridge allocation algorithm to the test images, we verify the allocation results by human eyes. The quantitative measurement results are summarized in Table 1. In this experiment, the best allocation result occurs in fingerprint 2 due to the high contrast and no contaminated area appearing in this fingerprint image. Fingerprints 1 and 6 also have high accuracy due to high contrast. However, some contaminated areas will hide the real ridges which will result in more missing points. The worst allocation result occurs in fingerprint 5 due to low contrast and highly contaminated areas appearing in this fingerprint image.



Fig. 19. Ridge section derived from contaminated area. (a) The original sub-image of the contaminated area in a fingerprint image and the corresponding ridge location after applying the gray-scale ridge allocation algorithm. (b) The extracted ridge points in the section set. (c) Binary image of this contaminated sub-image. (d) Skeleton image of this contaminated sub-image.

The only erroneous point also occurs in this poor-quality image. For the redundant ridge points, fingerprint 4 is the worst image because many parallel ridge lines lying on the bottom of this image. The horizontal lines, which run through sweat pores of these parallel ridge lines, generate additional ridge points. This condition is unavoidable if we fix the direction of line segments while extracting the ridge sections. In this experiment, we do not take the number of redundant points into account while evaluating the allocation accuracy. However, the false rates will still be less than 6% if we recognize the redundant points as error allocations. For poor-quality images, the 96.8% overall average accuracy rate is acceptable.

We also present some comparing results as summarized in Table 2. In this experiment, the performances of our direct gray-scale method are compared with binary and skeleton image based methods. There are three independent image operations including smoothing, binarization and thinning implemented in this experiment. The smoothing operation is accomplished based on a two-dimensional median filter with  $5 \times 5$  mask. The binarization operation is carried out based on Mehtre's [6] fingerprint segmentation method. For the thinning process, the algorithm proposed by Baruch [15] which provides good results on fingerprint image is used. This experiment was conducted on the image presented in Fig. 20 with the same rules of setting the extracting lines as the previous experiment. The performance evaluation results, as tabulated in Table 2, shows that unsmoothed binary image possesses the worst allocation result due to the high missing- and erroneous-point rates caused by the ridge line fragments. For binary images, the



Fig. 20. Experiment images.

Table 1									
Summary of the	performance evaluation	results with	different	fingerprint	images	based of	n the direct	gray-scale m	ethod

Finger print	True ridge points	Correct points	Missing points	Erroneous points	Redundant points	Accuracy (%)
1	288	280	8	0	5	97.2
2	294	290	4	0	7	98.6
3	370	357	13	0	9	96.5
4	336	324	12	0	14	96.4
5	310	292	17	1	11	94.2
6	361	354	7	0	8	98.1
Total	1959	1897	61	1	54	96.8

Table 2

Performance evaluation re	sults for different	approaches
---------------------------	---------------------	------------

Method	True ridge points	Correct points	Missing points	Erroneous points	Redundant points	Accuracy (%)
Direct gray scale	1959	1897	61	1	54	96.8
Unsmoothed binary	1959	1777	139	43	93	90.7
Smoothed binary	1959	1828	115	16	86	93.3
Smoothed skeleton	1959	1846	108	5	22	94.2

generated redundant points that are produced by the section sets running through a ridge line cannot be eliminated. This is the major reason of the high redundantpoint rate for both smoothed and unsmoothed binary images. The smoothed skeleton image possesses the best extraction result on the production of redundant points because the sweat pores inside ridge lines disappear after the thinning process. However, the accuracy rate of the

Method	Average computational time (s)							
	Histogram decomp.	Smoothing	Binarization	Thinning	Ridge extraction	Total time		
Direct gray scale	0.33	_	_	_	0.26	0.59		
Unsmoothed binary	_		0.92	_	0.12	1.04		
Smoothed binary	_	0.63	0.94	_	0.11	1.68		
Smoothed skeleton	—	0.63	0.94	1.86	0.08	3.51		

Table 3 Average computational time on PC Pentium-III 550 MHz machine

skeleton image is still lower than that of the direct grayscale method due to its high missing-point rate.

Table 3 tabulates the average computational times spending in automatic ridge extraction measured on a PC Pentium-III 550 MHz machine. As we noticed, the preprocessing stages, such as the smoothing, binarization and thinning, are time consuming. Although it is necessary to perform the histogram decomposition process in order to acquire some essential parameters in our direct gray-scale approach, the total time for extracting the ridge points in a section set is still less than that of the binary and skeleton images.

# 6. Conclusions and future works

Minutiae are the key features for the automatic fingerprint identification system (AFIS). Fingerprint analysis based on direct gray-scale minutiae detection and verification is a new research topic due to the insufficient capability of binary or skeleton image-based approaches. Locating the ridge position is the first step of gray-scale ridge line following algorithm that extracts minutiae in direct gray-scale domain without conventional preprocessing stage. Allocating a ridge position in fingerprint images is not straightforward due to the appearance of image background and the sweat pores inside the ridge lines. In this paper, we propose a direct gray-scale ridge allocation algorithm based on gray-level histogram decomposition. By this technique, experiments demonstrate that the redundant ridge points derived from background or the sweat pores inside a ridge are eliminated correctly. Moreover, ridge points derived from the contaminated areas of fingerprint images can also be located. The allocation accuracy can reach over 95% even applied to poor-quality fingerprint images with nonuniform illuminanced areas or highly contaminated areas. This direct gray-scale approach requires no preprocessing stages and could be applied to on-line fingerprint identification systems due to its low computational time.

The following works are the goals to the pursued in the future:

• Dynamically adjust the directions of the line segments for extracting the ridge sections to possibly intersect

the ridge lines vertically. This will raise the allocation accuracy if less section sets are acquired from the parallel distributed ridges.

- Develop a complete gray-scale ridge line following algorithm which can work on complete fingerprints, i.e., with image background.
- Develop an automatic fingerprint classification system (AFCS) in direct gray-scale domain. Preprocessing of fingerprint images is time consuming and will produce false minutiae. The gray-scale approach probably can solve these problems.

#### References

- B. Moayer, K.S. Fu, A syntactic approach to fingerprint pattern recognition, Pattern Recognition 7 (1975) 1-23.
- [2] D.K. Isenor, S.G. Zaky, Fingerprint identification using graph matching, Pattern Recognition 19 (2) (1986) 113-122.
- [3] N.K. Ratha, K. Karu, S. Chen, A.K. Jain, A realtime matching system for large fingerprint database, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 799-813.
- [4] D. Miao, D. Maltoni, Direct gray-scale minutiae detection in fingerprints, IEEE. Trans. Pattern Anal. Mach. Intell. 19 (1) 1997.
- [5] L. O'Gorman, J.V. Nickerson, An approach to fingerprint filter design, Pattern Recognition 22 (1) (1989) 29–38.
- [6] B.M. Mehtre, B. Chatterjee, Segmentation of fingerprint image — a composite method, Pattern Recognition 22 (4) (1989) 381–385.
- [7] Q. Xiao, H. Raffat, Fingerprint image postprocessing: a combined statistical and structural approach, Pattern Recognition 24 (10) (1991) 985–992.
- [8] J.L. Blue et al., Evaluation of pattern classifiers for fingerprint and OCR application, Pattern Recognition 27 (4) (1994) 485–501.
- [9] A.P. Fitz, R.J. Green, Fingerprint classification using a hexagonal fast fourier transform, Pattern Recognition 29 (10) (1996) 1587–1597.
- [10] X. Zhuang, T. Wang, P. Zhang, A highly robust estimator through partially likelihood function modeling and its application in computer vision, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1) (1992) 19–35.

- [11] M.J. Carlotto, Histogram analysis using a scale-space approach, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9 (1) (1987) 121–129.
- [12] X. Zhuang, Y. Huang, K. Palaniappan, Y. Zhao, Gaussain mixture density modeling, decomposition, and applications, IEEE Trans. on Image Process 5 (9) (1996) 1293–1302.
- [13] C. I. Watson, C. L. Wilson, Fingerprint Database, National Institute of Standards and Technology, Special Database 4, FPDB, April, 1992.
- [14] E. R. Henry, Classification and use of fingerprint. Routledge, London, 1900.
- [15] O. Baruch, Line thinning by line following, Pattern Recognition Lett. 8 (4) (1988) 271–276.

About the Author—JENG-HORNG CHANG was born in Taipei, Taiwan, in 1965. He received his B.S. degree in electrical engineering from National Taiwan Institute of Technology in 1991 and the M.S. degree in electrical and computer engineering from University of Missouri – Columbia in 1993. He has joined the faculty of St. John's and St. Mary's Institute of Technology since 1993 in Taiwan. He is currently a Ph.D. student of CSIE at National Central University. His research interests include pattern recognition, image processing and artificial intelligence.

About the Author—KUO-CHIN FAN was born in Hsinchu, Taiwan, on 21 June 1959. He received his B.S. degree in Electrical Engineering from National Tsing-Hua University, Taiwan, in 1981. In 1983 he worked for the Electronic Research and Service Organization (ERSO), Taiwan, as a Computer Engineer. He started his graduate studies in Electrical Engineering at the University of Florida in 1984 and received the M.S. and Ph.D. degrees in 1985 and 1989, respectively. From 1984 to 1989 he was a Research Assistant in the Center for Information Research at University of Florida. In 1989, he joined the Institute of Computer Science and Information Engineering at National Central University where he became professor in 1994. He was the chairman of the department during 1994–1997. Currently, he is the director of Software Research Center and Computer Center at National Central University. Professor Fan is a member of IEEE, and a member of SPIE. His current research interests include image analysis, pattern recognition, and computer vision.