

# 一种基于捕捉有效域的快速指纹特征提取算法

冯涛<sup>1)</sup> 张士伟<sup>2)</sup> 刘其真<sup>3)</sup>

<sup>1)</sup>(上海第二工业大学,上海 200041) <sup>2)</sup>(上海师范大学中新技术管理学院,上海 200233)

<sup>3)</sup>(复旦大学计算机科学与工程系,上海 200433)

**摘要** 由于指纹识别技术具有很高的实用性和可行性,因此是一种理想的身份认证技术。但目前的指纹特征提取算法却很难保证同时具有可靠性和高效率。为了准确快速地提取指纹特征,在进行有效域捕捉,并结合脊线跟踪的基础上,提出了一种快速的指纹细节提取方法。该方法先通过核心点的检测来确定有效的指纹面积,并且在处理过程中降低了对图像增强的依赖;然后对确定了指纹面积的图像进行脊线跟踪。实验结果表明,该方法不仅有较高的可靠性和效率,而且能较好地平衡它们两者的矛盾。

**关键词** 指纹识别 有效域 脊线追踪

中图法分类号:TP391.41 文献标识码:A 文章编号:1006-8961(2008)10-1857-04

## An Efficient Fingerprint Extract Algorithm Based on Effective Area Capturing

FENG Tao<sup>1)</sup>, ZHANG Shi-wei<sup>2)</sup>, LIU Qi-zhen<sup>3)</sup>

<sup>1)</sup>(Shanghai Second Polytechnic University, Shanghai 200041) <sup>2)</sup>(Shanghai Normal University Science and Technology Management Collage, Shanghai 200233) <sup>3)</sup>(Computer Science and Engineering Department, Fudan University, Shanghai 200433)

**Abstract** Finger print recognition technology is practical and feasible. It is a valuable ID recognition technology. The algorithm we commonly use now cannot provide reliability and high efficient at the same time. Based on effective area capturing and ridges tracing, this article advances an efficient fingerprint extraction algorithm. To capture the effective area and lower down the dependence to the image quality, we first check the core point. Then the tracing ridge is used to extract strange points. Experimental results show that this algorithm can balance the practicality and feasibility at the same time.

**Keywords** fingerprint extract, effective area, tracing ridges

## 1 引言

近年来,随着计算机技术的飞速发展以及高可靠指纹识别算法的实现,使得自动指纹识别技术飞速融入到人们的生活中。由于指纹识别技术具有很高的实用性和可行性,已经被认为是一种理想的身份认证技术,有着十分广泛的应用前景,因此是将来生物特征识别技术的主流之一<sup>[1]</sup>。

现有的指纹特征提取算法大致分为常规的细节特征提取算法和直接灰度指纹图像特征提取算法两类。其中常规的细节特征提取算法大多是在经过细

化和纹线修复的指纹图像上进行。而事实上,由于指纹自身结构的复杂性和目前图像预处理技术的局限性,对图像进行比较理想的细化具有一定的技术难度<sup>[2,3]</sup>,且该方法所涉及的领域在处理上非常繁琐,时间消耗也比较大<sup>[4]</sup>,所以,在实际应用中,常规细节特征提取算法的效果并不总能令人满意。直接灰度指纹图像特征提取算法一般是通过沿纹线方向自适应地追踪指纹脊线,在追踪的过程中,通过局部增强指纹图像来得到一幅细化后的指纹脊线骨架图和附加在其上的细节点信息。由于该算法只在占全图比例很少的点上估算方向,并进行滤波处理,因此计算量相对较少。但由于该算法的独特性,致使

收稿日期:2008-06-20; 改回日期:2008-07-01

第一作者简介:冯涛(1975~),男,讲师。2001年获东华大学工学硕士学位。主要研究方向为电子与通信技术、图像处理与模式识别。E-mail:fentao@adult.sspu.cn

其对低质量指纹图像的适应性不是太理想,并容易在特征点的判断上出现偏差<sup>[5]</sup>。

因此,在对现有的指纹特征提取算法进行研究后,本文提出了一种基于有效域捕捉,并结合脊线跟踪的快速指纹细节提取方法,即先通过梯度矢量来求得指纹中心点位置,再通过向外部扩张来获得包含有足够数量特征点的有效域,最后通过与一种改进的脊线跟踪算法的结合来获得指纹的特征信息。由于此方法综合考虑到程序运行效率及可靠性,因此能较好地平衡它们两者的矛盾。实验结果表明,该方法具备了现有方法所不具有的高效率和可靠性。

## 2 指纹细节提取方法的相关研究

由于目前指纹算法主要面临的问题是:(1)指纹图像质量不能满足要求;(2)指纹面积的不确定性;(3)指纹图像具有旋转与扭曲,因此现行的指纹算法几乎都是针对如何快速,并准确地处理以上问题而进行的优化与提高。但大都无法在计算效率、准确度以及指纹方向容忍度上做到很好的平衡。

在传统算法中,先期的图像增强是很重要的一步。如果不能很好地对图像进行增强,则很容易导致后续步骤出错或者失败。针对图像增强通常使用的基于 Gabor 滤波器的方法<sup>[6]</sup>其时间消耗较大的问题,其核心点检测是关键的一步。文献[7]提出用基于

神经网络的方法检测核心点,文献[8]提出采用基于方向场局部能量的方法检测核心点、文献[9]提出采用 Park 标记方向场中的水平方向来定位核心点的方法检测核心点。在这些算法中,文献[7]需要大量的前期工作,文献[8]将核心点存在,并能被检测视为必备条件,文献[9]的方法更是要求指纹不能任意旋转。这些先决条件都导致了以上算法不能具有很好的兼容性,且对指纹图像的先期要求较高,并可能因为无法确定中心点而导致花费大量时间。在特征点提取方面,传统算法的主要设计思想包括计算方向图、图像滤波、二值化、细化、提取特征和后处理这些过程。虽然上述方法可靠性较高,但步序繁琐。文献[5]提出的基于脊线跟踪的细节点提取方法,虽能够从灰度指纹图像中直接抽取细节点及其相关信息,可是该方法对低质量图像及指纹面积较为敏感,容易发生效率降低及误判。因此本文通过提取用于指纹识别的有效域,在有效域中进行脊线跟踪,可很好地解决低质量指纹图像识别、指纹面积敏感度等问题,并提出了一种基于有效域捕捉,并结合脊线跟踪的快速指纹细节提取方法。

## 3 有效域捕捉和脊线跟踪

基于捕捉有效域结合脊线跟踪的快速指纹细节提取方法过程如图 1 所示:

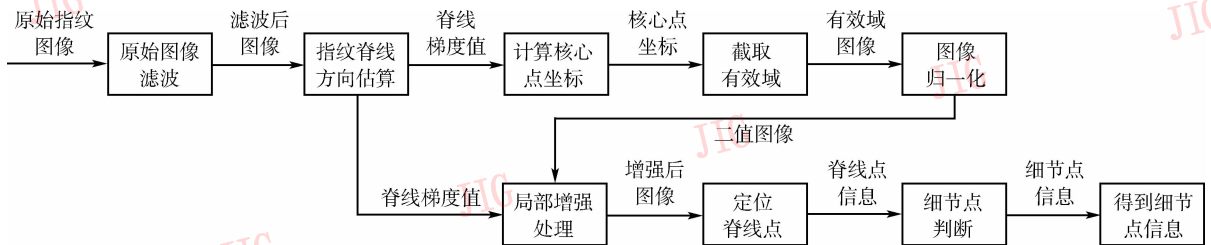


图 1 基于有效域捕捉结合脊线跟踪的快速指纹细节提取方法过程的基本流程

Fig. 1 The basic process of fingerprint distillate algorithmic based on effective area capturing

### 3.1 有效域捕捉

所谓有效域捕捉就是对指纹核心点的捕获。捕获时,首先计算指纹图像在每个像素处的梯度矢量,然后在得出这些梯度数据后,由于局部区域内梯度矢量较大的像素位于图像纹理的边缘,因此局部区域的脊线方向主要由这些边缘像素决定,其梯度矢量将垂直于脊线方向。由于脊线的梯度矢量都会指向中心点,因此在  $\pi < \theta < \frac{3\pi}{2}$  角度内,对其进行跟踪后就可将频率最大的跟踪线末端的坐标视为核心

点的位置。最后即可根据指纹图像的相对面积来截取包含足够细节点信息的有效域。

#### 3.1.1 原始图像滤波

为更好地对指纹图像进行处理,一般需要先对图像进行滤波。通常采用 Gabor 滤波器进行滤波,但其耗时相对较长。本文采用维纳算子可明显降低噪声对于估算脊线方向的梯度算法的影响,且速度较快。对于一幅  $N \times M$  大小的图像(图 2),可先采用式(1)、式(2)计算每个像素  $a$  周围的局部均值

$\mu$ 、方差  $\sigma$  和噪声方差  $v$ ,再根据式(3)来进行处理。

$$\mu = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} a(n_1, n_2) \quad (1)$$

$$\sigma^2 = \frac{1}{NM} \sum_{n_1, n_2 \in \eta} a^2(n_1, n_2) - \mu^2 \quad (2)$$

$$b(n_1, n_2) = \mu + \frac{\sigma^2 - v^2}{\sigma^2} (a(n_1, n_2) - \mu) \quad (3)$$

### 3.1.2 指纹脊线方向估算

为获得指纹走向,需要获得指纹脊线的方向信息。常用的计算脊线方向的方法是 最小方差估计和梯度算法。本文是采用索贝尔梯度算子(式(4)、式(5))来计算某像素点在  $z$  区域的  $x$  和  $y$  方向上的梯度  $G_x$  和  $G_y$ 。

$$G_y(x, y) = \sum_{u=x-z/2}^{x+z/2} \sum_{v=y-z/2}^{y+z/2} 2\partial_x(u, v) \partial_y(u, v) \quad (4)$$

$$G_x(x, y) = \sum_{u=x-w/2}^{x+w/2} \sum_{v=y-z/2}^{y+z/2} [\partial_x^2(u, v) - \partial_y^2(u, v)] \quad (5)$$

不同于其他的算法,由于在之后的脊线跟踪时仍将用到该梯度值,因此这里对其进行单独计算,并加以保存,以避免之后可能发生的重复计算。对图 2 进行处理后得到的脊线方向估计结果如图 3 所示。



图 2 原始指纹图像 Fig.2 Original fingerprint image 图 3 脊线方向估算 Fig.3 Calculate ridges angel

### 3.1.3 计算核心点坐标

在获得了脊线的梯度值后,再采用式(6)计算脊线上梯度的倾角。

$$\theta(x, y) = \frac{1}{2} \cdot \tan^{-1} \left( \frac{V_y(x, y)}{V_x(x, y)} \right) - \frac{\pi}{2} \quad (6)$$

$$\text{当 } \theta \text{ 为, } \begin{cases} \pi < \theta < \frac{3\pi}{2} & \text{继续} \\ \text{其他} & \text{终止} \end{cases} \quad (7)$$

核心点可根据式(7)所描述的条件来进行跟踪。连续搜索  $\theta(x+1, y+1)$ , 仅当倾角处于  $\pi < \theta < \frac{3\pi}{2}$  的范围中时,频率记数增加 1; 否则保

存频率记数,并开始下一轮搜索。最后将频率记数最大的  $\theta(x, y)$  的坐标视为核心点。实验结果如图 4 所示。



图 4 计算核心点坐标 Fig.4 Calculate core point coordinate



图 5 截取有效区域 Fig.5 Effective area capturing

### 3.1.4 截取有效区域

由于指纹图像大小  $S$  的不确定性,因此本文使用相对大小来对有效区域进行截取,即首先对步骤 1 中获得的处理后的图像面积进行计算,其判断标准为非 0 (白色) 像素; 然后由点  $(x - \sqrt{\frac{S}{8}}, x + \sqrt{\frac{S}{8}})$  开始,截取  $(\sqrt{\frac{S}{2}} \times \sqrt{\frac{S}{2}})$  大小的矩阵区域为有效区域(如图 5 所示)。

## 3.2 基于脊线跟踪的细节特征点提取

### 3.2.1 图像归一化

根据相邻像素灰度的期望的均值  $M_0$  和期望的方差  $V_0$ ,采用式(8)来对图像进行归一化处理,以降低灰度值不同所造成的判断偏差。

$$N_i(x, y) = \begin{cases} M_0 + \sqrt{\frac{V_0 \times (I(x, y) - M_i^2)}{V_i}} & I(x, y) > M_i \\ M_0 - \sqrt{\frac{V_0 \times (I(x, y) - M_i^2)}{V_i}} & \text{其他} \end{cases} \quad (8)$$

### 3.2.2 局部增强处理

根据已确定的脊线方向,先将垂直于该方向上的点保存为点集  $\Omega$ ; 然后根据  $\Omega$  上的灰度均值得到平滑后的点集  $\Omega_1$ ; 最后使用高斯低通滤波器(式(10))对点集  $\Omega_1$  进行卷积运算,即  $\Omega_2 = \Omega_1 \otimes G$ , 使图像得到局部增强。实验结果如图 6 所示。

$$G(y) = \frac{1}{\sqrt{2\pi}r} \exp \left[ - \left( \frac{y}{r} \right)^2 \right] \quad (9)$$

### 3.2.3 定位脊线点

定位脊线点时,首先以灰度值为参照,搜寻局部最大值点  $(X_{\max}, Y_{\max})$  和最小值点  $(X_{\min}, Y_{\min})$ 。最

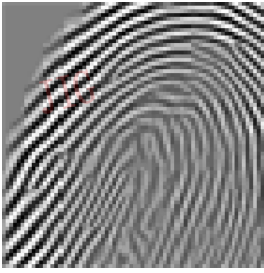


图 6 局部增强

Fig. 6 Local enhanced

大值点就是与脊线骨架最接近的脊线点,即要寻找的脊线点,而最小值点就是离脊线边缘最近的非脊线点;然后计算最大值点与最小值点的灰度差  $d$  和点  $(x_{\max}, y_{\max})$  处的方向估计值  $\theta$ ;最后以一定的步长  $u$  ( $u$  一般取 1 或 2),沿  $\theta$  方向探测点  $(x_i, y_i)$  直至点  $(x_{n+i}, y_{n+i})$  与点  $(x_{\max}, y_{\max})$  的灰度之差大于某个与  $d$  相关的值,即得到延伸的步长  $l = n \times u$ 。

### 3.2.4 细节点判断条件

首先定义  $s_{\max}$  为计算图像  $s$  中单位区域的灰度最大值的函数,然后根据以下所描述的条件

$$\begin{cases} s_{\max}(x, y) = 1 & \text{接触到已标记脊线,判断为细节点} \\ \text{同时 } s > m & \text{(特征点)} \\ s_{\max}(x, y) = +\infty & \text{接触到脊线外部,判断为结束跟踪} \\ l = 0, & \text{接触到末梢点,判断为结束跟踪} \\ \text{其他} & \text{继续跟踪} \end{cases} \quad (10)$$

来对  $(x_n, y_n, l)$  进行判断,如果满足结束探测条件,则结束沿一个方向的脊线跟踪,返回初始点,并沿反方向继续跟踪。当另一方向的探测也结束后,则得到了一条脊线。对脊线上的所有点  $(x, y)$ ,修改  $s[x, y] = m$  ( $m$  为脊线的序号)。最后从下一个块中寻找灰度极值点(最大值与最小值点),如果该点未被访问过,则以该点作为新的出发点,继续跟踪。

## 4 仿真实验结果

为验证本文算法的效果,采用 VeriFinger, BioLAB 及 UareU 指纹数据库作为数据源进行了仿真实验,并与常规的细节特征提取算法的提取结果进行了比较(如表 1 所示)。由表 1 可见本文算法在保证识别率的情况下比传统算法的平均耗时降低了许多。实验环境为基于 PIII-667 的普通 PC 机。

表 1 指纹识别速度及识别率对照表

Tab. 1 Regeneration rate and performance result

指纹库	图像大小	识别时间(s)		识别率(%)	
		传统算法	本文算法	传统算法	本文算法
Veri	504 × 480	5.6	2.0	83	83
BioLAB	256 × 256	2.3	1.0	90	85
UareU	326 × 357	2.8	1.2	82	80

实验结果表明,该方法具备了传统方法所不具有的高效率和可靠性。

## 5 结论

本文提出了一种基于有效域捕捉并结合脊线跟踪的快速指纹细节提取方法。该方法先通过核心点的检测来确定指纹面积,并在处理过程中降低了对图像增强的依赖;之后对确定了指纹面积的图像进行脊线跟踪。从实验结果可以看出,该方法有较高的可靠性和效率,且能较好地平衡它们两者的矛盾,具备了现有方法所不具有的高效率和可靠性。对于如门禁、口令系统等对安全性要求相对较低,而速度要求相对较高的场合,有较高的应用价值。

## 参考文献 (References)

- Jain A K, Lin Hong, Pankanti S, *et al.* An identity authentication system using fingerprints [J]. Proceedings of the IEEE, 1997, **85**(9): 1365 ~ 1388.
- Yua Shiaw-shian, Tsai Wen-hsiang. A new thinning algorithm for gray-level images by the relaxation technique [J]. Pattern Recognition, 1990, **23**(10): 1067 ~ 1076.
- Datta A, Parui S K. A robust parallel thinning algorithm for binary images [J]. Pattern Recognition, 1994, **27**(9): 1181 ~ 1192.
- Xiao Qing-han, Raafat Hazem. Fingerprint image post processing: A combined statistical and structural approach [J]. Pattern Recognition, 1991, **24**(10): 985 ~ 992.
- Jian Bing, Zhuang Zhen-quan, Li Hai-ying, *et al.* A minutiae extraction algorithm based on ridgeline following [J]. Journal of Circuits and Systems, 2001, **9**(6): 1 ~ 5.
- Hong Lin, Wan Yi-fei, Jain Anil. Fingerprint image enhancement: algorithm and performance evaluation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, **20**(8): 777 ~ 789.
- Drets G A, Lijenström H G. Fingerprint sub classification: a neural network approach, in intelligent biometric techniques in fingerprint and face recognition [A]. In: Jain L C, Halici U, Hayashi I, *et al.* Eds: Intelligent Biometric Techniques in Fingerprint and Face Recognition [C], Boca Raton, FL, USA: CRC Press Inc. 1999: 109 ~ 134.
- Perona P. Orientation diffusions [J]. IEEE Transactions on Image Processing, 1998, **7**(3): 457 ~ 467.
- Park Chul-hyun, Oh Sang-keun, Kwak Dong-min, *et al.* A new reference point detection algorithm based on orientation pattern labeling in fingerprint images [EB/OL]. <http://www.springerlink.com/content/kc8bf0yn57kcmeh/>.