

基于图像处理的指纹识别图像预处理方法

林香宜 谢敏* 何华光

广西大学计算机与电子信息学院, 南宁 530004

摘要 本文主要从研究背景和意义对指纹识别技术做简单介绍, 简要概述指纹的特性和识别技术的流程, 重点阐述指纹图像预处理流程, 主要包括图像增强、归一化处理、指纹图像分割、二值化处理和细化处理等技术的研究。本文使用 MATLAB 编程实现所有过程, 并给出处理结果。图像处理结果表明, 本文采用的基于图像处理的指纹识别图像预处理方法可以比较有效地对指纹进行预处理, 提高了指纹图像识别的效率。

关键字 图像增强, 归一化处理, 指纹图像分割, 二值化处理, 细化处理

The Preprocessing Process of Image Processing-Based Fingerprint Recognition

Lin Xiangyi, Xie Min, He Huaguang

School of Computer, Electronics & Information of Guangxi University
Nanning 530004, China

{gxxiem@163.com, linxiangyi@stu.jnu.edu.cn}

Abstract—This paper briefly introduces fingerprint recognition technology from the research background and significance, then briefly summarizes the characteristics of fingerprint and the process of identification technology. The preprocessing process of fingerprint image is mainly expounded, including image enhancement, normalization processing, fingerprint image segmentation, binary processing and refinement processing. This article implements all procedures using MATLAB programming and gives the processing results. The image processing results show that the fingerprint recognition image preprocessing method based on image processing adopted in this article can effectively preprocess fingerprints and improve the efficiency of fingerprint image recognition.

keywords—image, enhancement, normalization, processing, fingerprint image, segmentation, binary processing, refinement processing

1 引言

在信息时代, 身份具有数字化和隐蔽性的特征。以前的安全系统采用的是“用户名+密码”的身份认证, 而其它地方的身份认证则是核对证件, 无论是用密码还是证件来认证, 都有被遗忘和盗窃的危险, 已经不能适应现在的发展需要。人们希望身份认证能更可靠、更安全, 并且方便使用。因此, 传统的身份认证就被生物认证技术取代。

生物识别技术是基于人的生理与行为特征来判别人身份的技术, 例如通过人脸、声音、虹膜、指纹、签名等生物特征来进行身份识别, 其广泛应用在手机、银行、门禁、公安等领域^[1]。

个人指纹在持久性、唯一性和采集指纹的方便性等各方面有着相对的优势, 使指纹识别技术成为目前身份鉴定的可靠手段之一^[2]。指纹识别主要是通过对不同个体指纹的局部与整体特征, 进行比较来实现。

2 指纹识别技术

通过指纹识别技术, 将储存在计算机中的指纹与个人指纹进行比对, 从而确定身份。随着计算机各领域迅猛发展, 指纹识别技术得到了广泛的应用, 已经深入到人们生活的各方面, 是目前研究最为深入、应用最为广泛的技术。

2.1 指纹的特征

指纹具有唯一性和永恒性, 决定了每个人的指纹都是独特的。指纹匹配依靠指纹的特征来判断两枚指纹的相似度, 指纹特征有局部特征与全局特征之分^[3]。

(1) 指纹的脊线与谷线

交错在手指上的脊线和谷线, 是指纹最显著的结构特征。大部分的手指外伤, 例如表面灼伤, 擦伤或切割伤, 都不会影响脊的结构。如图1所示。

*通讯作者: 谢敏 gxxiem@163.com

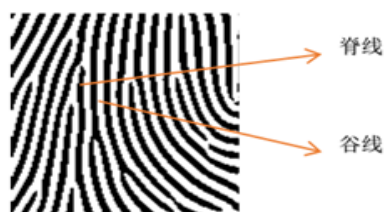


图 1 局部特征中的脊线和谷线

(2) 指纹的局部特征

指纹的局部特征是眼睛难以观察到的特征节点，这些特征节点就是指纹的细节点或者称为特征点。指纹的局部分析通过脊线不连续性的位置（称为细枝末节点）提供有关个体的信息^[3]。在指纹细节特征中有多种类型的特征点，具体分为：端点、分叉点、断裂点等。其中端点和分叉点出现次数最多也最稳定，所以在指纹匹配中选择端点和分叉点作为参考点^[45]。细节特征点提供了指纹唯一性的确认信息。

如图2所示，端点一般指脊线的始端或者结束端，分叉点指的是从一条脊线某个位置分出两条或者多条脊线的点。断点指的是脊线在某处断开形成的一对断点，桥接点指的是两条脊线由一条或多条脊线连接，连接处就是桥接点。

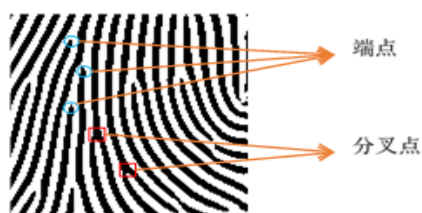


图 2 局部特征中的端点和分叉点

(3) 指纹的总体特征

一般用眼睛直接看到手指指纹的特征是全局特征。它由纹线、模式区、核点、三角点等组成。每个人的总体特征，有的时候会有相似之处，例如指纹的纹形是相同的，但不同人指纹的局部特征并不会完全相同，这也是指纹独一性的体现，是指纹最重要的一个特征。



图 3 指纹的纹形

指纹的纹形是由指纹的全局形态来划分的，具体如图3中所示的三类：环形、弓形和螺旋型。三大类型的不同在于：环形仅有一个中心点，弓形则无中心点，螺旋形有两个中心点。仅仅依靠指纹的纹形并不能将所有的指纹都区别开来，指纹的纹形是对指纹进行一

个大致的分类，在指纹识别的过程中，可以减轻一些工作量，从而提高指纹识别的速度。

2.2 指纹识别的流程

指纹识别系统包括两个部分：离线部分就是进行指纹的注册，在线部分就是指纹的识别匹配。

指纹识别的基本过程：通过指纹传感器或采集器得到指纹后，对指纹图像进行一系列预处理操作后得到二值化图像，再对该图像进行形态学细化，然后从细化图中提取指纹的特征。无论是离线部分还是在线部分都会对指纹进行上述操作，区别在于离线部分下一步操作是将指纹特征存在指纹数据库中，而在线部分则将抽取出来的特征输入到指纹匹配器中。指纹模板数据库是匹配器的另一个输入，匹配器对这两个输入进行比较，判断两枚指纹是否相同，就可以验证是否是同一个人。

2.3 指纹图像的预处理

系统采集到的原始图像，如果没有进行处理，就会产生噪声，使图像质量变差，也会给特征提取带来不便。因此，在提取图像特征之前，需要对其进行预处理^[6]。预处理的目的是在保证指纹的细节特征不变下，去除指纹图像中的噪声，恢复指纹的纹理结构，得到正确的指纹特征，最后将指纹纹线转换为单像素宽度^[44,5]。指纹识别系统中，预处理过程是最重要的，能否有效去除指纹图像中的噪声直接关系到最终指纹识别的准确率，预处理的结果直接决定了指纹识别的效果。

2.4 指纹图像增强

为了去除指纹图像中存在的干扰噪音，并提高后续的指纹识别效果，在分割之前，首先要对指纹图像进行均衡化和滤波处理。这个处理过程就是要增强指纹图像，一般是对质量较差的指纹图像进行的。通过图像增强技术，可以消除指纹部分虚假特征信息，减少指纹存在的噪声，还能很好地修复间断线，有效地凸显了指纹的细节，让指纹的结构和纹路可以更加清晰地呈现出来，保留了对指纹识别至关重要的特征信息，从而提高了指纹识别的准确性和可靠性^[5,7]。一般可以从空间阈或者频率域对图像进行增强^[8]。频域法是在图像的某个变换域内，修改变换后的系数；空间域增强方法是基于直接处理图像像素来改变图像，它属于直接增强的方法。空间域包括基本的灰度变换、直方图处理、消除噪声平滑和增强边缘锐化方法。本文讨论了属于空间域增强方法的直方图增强和中值滤波。

(1) 直方图均衡化

直方图均衡化其基本思想是，在指纹图像中，展

宽像素个数多的灰度级，缩减像素个数少的灰度级，这样就可以达到清晰图像的目的。直方图均衡化是一种广泛应用于数字图像处理的工具，其目的在于扩大图像的像素值分布，从而增强图像的知觉信息^[9]。直方图中的横坐标是灰度级，纵坐标是该灰度级的像素个数，或者是这个灰度级出现的概率。关于指纹图像，脊线部分的灰度值较小，而谷线的灰度值较大，质量好的指纹图像其直方图是一个双峰型的分布，对于质量较差的指纹图像其谷线和脊线的灰度值差别不大，此时的直方图就会集中在一个范围分布。通过直方图均衡，可以对灰度值范围进行很好的展宽，可以更好地区分指纹的脊线和谷线。对数字图像而言，用频率表示概率，即：

$$p(r_k) = \frac{n_k}{N} \quad (1)$$

其中， r_k 表示第 k 个灰度级， n_k 表示第 k 个灰度级的像素个数， N 表示图像的像素总数。 s 是变换后的图像灰度， r 是灰度化后的原始图像的灰度。设变换函数为 $T(r)$ ，则有 $S=T(r)$ ，变换前后像素总数是不变的有式：

$$\int P_s(s)ds = \int P_r(r)dr \quad (2)$$

根据直方图均衡的要求，变换后的概率密度分布为1，即 $P_s(s)=1$ ，将其代入(2)式并去掉积分号，有以下等式成立：

$$ds = dT(r) = P_r(r)dr \quad (3)$$

两边取积分可得下式：

$$S = T(r) = \int_0^r P_r(r)dr \quad (4)$$

式(4)即变换函数，表明变换函数是原图像的累积分布函数。

对于离散图像，类比连续图像的推导，将积分号转换成累加和号，概率密度函数转为频数，可以得出离散图像的变换函数如下式所示。

$$t_k = T(s_k) = \sum_{j=0}^k p_r(s_j) = \sum_{j=0}^k \frac{n_j}{N} \quad (5)$$

其中， N 为总像素，灰度级总数为 L ， s_k 为第 k 个灰度级的值，其像素数目为 n_k 。

直方图均衡化步骤：先列出原始图像灰度级，统计原始图像中每个灰度级出现的次数，从而计算其出现的概率，显示出原始图像直方图；然后再计算累积概率，利用累积分布函数对原图像的统计直方图做变换，得到新的图像灰度。

图4和图5给出了直方图均衡化前后指纹图像和直

方图分布的对比图。



图 4 原始指纹图像

图 5 均衡化的指纹图像

图4是原始指纹图像，图5是经过均衡化后的指纹图像，经过对比可以看出经过直方图均衡化后的指纹纹路更加清晰地呈现出来，但缺点就是有较多的椒盐噪声，接下来需要经过滤波处理来消除这些噪声。

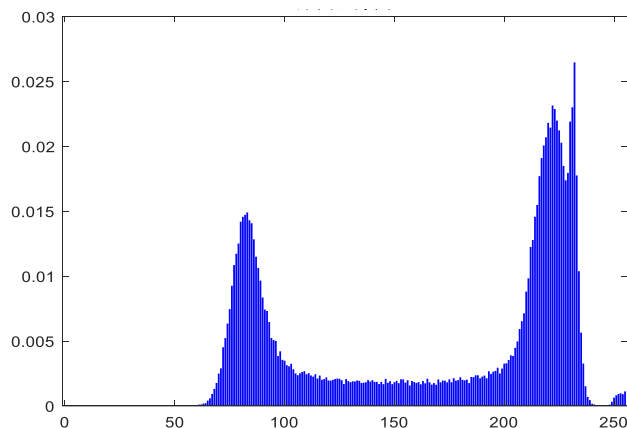


图 6 原指纹图像直方图分布

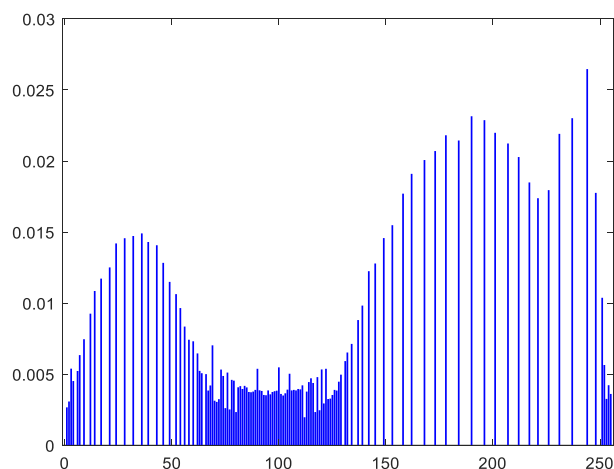


图 7 均衡化后的指纹图像直方图分布

图6看出原始图像灰度分布是双峰形态，像素灰度值集中在一个小区域内，图像对应的动态范围小，对比度低；对其直方图均衡化处理后，图7灰度值范围进行了展宽，像素灰度值分布更均匀，动态范围大，对应图像对比度高，增强了可视化效果。

(2) 中值滤波

中值滤波法就是把每个像素点灰度值都设定在该

点某个邻域窗口中所有像素点灰度中值的一种非线性平滑技术，可以有效去除指纹中存在的椒盐噪声^[10-12]。中值滤波以排序统计理论为基础，它的基本原理是在数字图像或数字序列中一点的值，用它的一个邻域中各点值的中值代替，让周围的像素值更加接近真实值，从而有效地消除孤立噪声点^[12]。

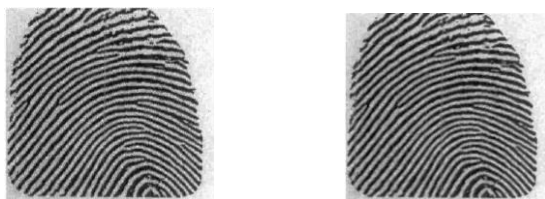
首先需要找到中值，再用其去替代一个位置的像素。步骤如下：①在指纹图像中，取3×3的图形块，在这个图像块中，共有9个像素点，我们找出其中的像素中值。②用像素中值去代替最中间位置的像素值。图8所示：灰度值排序为5, 17, 18, 21, 23, 25, 28, 42，其中值为23，用23代替5。



图8 图像中值滤波原理图

使用中值滤波处理前后的指纹图像如图9所示。图

(a)是经过直方图均衡化而没有经过滤波处理的图像，可以看出含有较多的噪声。图(b)是经过中值滤波处理后的图像，可以看出，中值滤波可以有效减小指纹的椒盐噪声，并使指纹结构、纹理变清晰。



(a) 均衡化的指纹图像 (b) 滤波后的指纹图像

图9 滤波前后指纹图像对比

(3) 指纹图像分割

指纹图像分割是从指纹图像背景中分离出我们感兴趣的区域。前景区域对应于包含山脊和山谷的清晰指纹区域，这是感兴趣的区域。背景对应于指纹区域边界之外的区域，这些区域不包含任何有效的指纹信息。当细节特征提取算法应用于图像的背景区域时，它会导致提取到嘈杂和虚假的细节，因此，采用分割来丢弃这些背景区域，避免在噪声和背景区域提取特征，不用花费代价对背景区域进行处理，降低预处理的计算复杂度，提高指纹识别系统的处理速度^[11,13]。

要判断一个指纹分割算法的好坏，就要看其是否具备以下几个功能：

- ① 检测能力强，可以检测到图像中的污迹和背景区域。
- ② 由于对比度容易改变，对输入图像的对比度

及灰度分布不敏感。

③ 分割得到的结果具有很好的适应性，可以应用到后续过程中。

④ 对不同质量的图像，提供标准一致的结果。本文的指纹分割算法从图像灰度特性方面去研究。

受采集仪器和外部环境等因素的影响，所得到的指纹图像的灰度分布差异较大。为了降低这种差异，也为了提高指纹图像的质量，本文给出了一种基于给定指纹图像局部特性的自适应归一化算法^[14]。该算法将所有样本图像转换成具有相同灰度均值和方差的标准图像，使其归一化到一个闭区间即灰度规格化^[15]。归一化处理只改变指纹图像的灰度值，使得指纹图像变为低频图，对其清晰度不会产生影。归一化步骤描述如下：

假设I代表原指纹图像，大小为m×n；I(i, j)表示第i行第j列像素所指的灰度值，计算原始图像的灰度均值和方差的计算公式如下：

$$M_{can} = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} I(i, j) \quad (6)$$

$$V_{ar} = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i, j) - Mean)^2 \quad (7)$$

令G表示灰度规格化W后的指纹图像，G(i, j)是规格化后像素点(i, j)的灰度值，其数学过程如下：

$$G(i, j) = Mean_0 + \sqrt{\frac{Var_0(I(i, j) - Mean)^2}{Var}}, I(i, j) > Mean \quad (8)$$

$$G(i, j) = Mean_0 - \sqrt{\frac{Var_0(I(i, j) - Mean)^2}{Var}}, \text{其他} \quad (9)$$

其中，Mean(0)和Var(0)分别为期望的均值和方差，分别取值为150和2000。规格化灰度处理指纹图像，其效果如图10所示。与图8(b)对比看出，对指纹图像进行规格化处理，不改变其质量，只改变指纹灰度对比度。

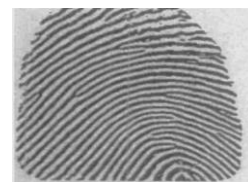


图10 指纹图像归一化

基于灰度方差指纹图像分割方法^[15]的基本原理

是：利用指纹前景区域的方差大均值小，背景区域方差小而灰度均值大的特点，采用区域灰度方差和灰度均值等统计对指纹图像进行分割处理，从而分割出有效指纹图像区域。将指纹图像分割成块，计算每一块的方差，设置阈值，若某块的方差大于等于阈值则为前景区域，否则为背景区域。具体步骤如下：

Step1: 将低频图分割为 $w \times w$ 大小无重叠小块，指纹图像总共分成 $H \times L$ 个子块，本文 $w=3$ 。

Step2: 计算每一指纹块的灰度强度的平均值，计算公式如 (10)

$$M(i, j) = \frac{1}{W \times W} \sum_{u=x-w/2}^{u=x+w/2} \sum_{v=y-w/2}^{v=y+w/2} G(x, y) \quad (10)$$

Step3: 计算指纹块的灰度强度方差，计算公式如下：

$$V_{Ar}(i, j) = \frac{1}{W \times W} \sum_{u=x-w/2}^{u=x+w/2} \sum_{v=y-w/2}^{v=y+w/2} (G(x, y) - M)^2 \quad (11)$$

Step4: 计算所有子块的灰度均值，计算公式如下：

$$Gall = \frac{1}{W \times W} \sum_{x=0}^{H-1} \sum_{y=0}^{L-1} M(x, y) / (H \times L) \quad (12)$$

Step5: 计算所有子块的灰度方差，计算公式如下：

$$Vall = \sum_{x=0}^{H-1} \sum_{y=0}^{L-1} V_{Ar}(x, y) / (H \times L) \quad (13)$$

Step6: 图像清晰区域的灰度均值，计算公式如下：

$$G_1 = \frac{S_{GBK1}}{S_{NGBK2}} \quad (14)$$

其中， S_{GBK1} 为所有的灰度均值大于 $Gall$ 的子块灰度均值之和， S_{NGBK2} 为所有灰度均值大于 $Gall$ 的块数之和。

Step7: 图像清晰区域的灰度方差，计算公式如下：

$$V_1 = \frac{S_{VBK1}}{S_{NVBK1}} \quad (15)$$

其中， S_{VBK1} 为所有的灰度方差大于 $Vall$ 的子块灰度方差之和， S_{NVBK1} 为所有灰度方差大于 $Vall$ 的块数之和。

Step8: 图像模糊区域灰度均值，计算公式如下：

$$G_2 = \frac{S_{GBK2}}{S_{NGBK2}} \quad (16)$$

其中， S_{GBK2} 为所有的灰度均值大于 G_1 的子块灰度均值之和， S_{NGBK2} 为所有灰度均值大于 G_1 的块数之和。

Step9: 图像模糊区域灰度方差，计算公式如下：

$$V_2 = \frac{S_{VBK2}}{S_{NVBK2}} \quad (17)$$

其中， S_{VBK2} 为所有的灰度方差大于 0 且小于 V_1 的子块灰度方差之和， S_{NVBK2} 为所有灰度方差大于 0 且小于 V_2 的块数之和。

Step10: 如果 $M > G_2$ 且 $Var < V_2$ ，则认为背景，如果 $M \leq M_1$ 且 $Var \geq Var_1$ ，则认为前景，否则待定。对所有分块划分完毕后，重新修正背景块，如果在 8 邻域中，背景个数小于等于 4 则修改为前景，否则是背景。

指纹图像分割结果如图11所示。可以看出，指纹图像的前景区域与背景区域被分割出来了，方便后续之前处理前景区域，从而减少一些不必要的处理，加快指纹识别速度。



图 11 指纹图像分割

(4) 指纹图像的二值化

指纹图像二值化的定义：用黑色表示指纹的脊线，用白色表示指纹的谷线和背景，从而凸显出指纹的脊线^[4,9]。这样可以压缩原图像的数据大小，使图像变简单，也使后面指纹特征提取和匹配变得简单。指纹图像没有经过二值化处理前，其灰度值变化范围在 0—255，二值化处理后，灰度值只有 0 和 1，脊线部分灰度值为 0，谷线部分灰度值为 1。二值化的基本思想是：预先设定灰度阈值，通过比较指纹图像中灰度值和阈值的大小，重置指纹图像灰度值。在指纹图像中，当其灰度值高于或等于一个门限时，其值被表示为 1，当其灰度低于门限时，其值被表示为 0。

常用的二值化算法有全局阈值法^[16]、局部动态阈值法^[16]，这两种方法都需设定阈值。

• 全局阈值二值化

计算公式如下：

$$\begin{aligned} f(x, y) &= 0, F(x, y) < T \\ f(x, y) &= 1, F(x, y) \geq T \end{aligned} \quad (18)$$

其中， $F(x,y)$ 为指纹图像中的灰度值， T 为指定的全局阈值， $f(x,y)$ 表示二值化后的像素点的值。经过二值化后，指纹图像中的黑白两种像素的个数差不多相等。计算过程如下：

Step1: 计算图像像素点总个数, $m \times n$ 。

Step2: 统计指纹灰度值范围在 $0-L$ 的总数, 如果满足式 (19), 其中 L_i 是灰度值为 i 的像素个数, 式中 h 就是阈值, $T=h$ 。则有:

$$\sum_{i=0}^h L_i = \frac{1}{2} m \times n \quad (19)$$

Step3: 根据比较图像的灰度值与 T 的大小, 确定图像中每点像素的值。

使用全局阈值法得到的二值化指纹如图 12 所示。对于图 (a) 来说, 二值化效果较好; 对于图 (b) 来说, 很多指纹纹线模糊不清, 二值化效果不好。全局阈值法对质量差的指纹二值化的处理效果不好。



图 12 全局阈值二值化指纹图像

• 局部动态阈值法

因为不同区域的指纹图像的亮度和对比度不同, 所以全局阈值二值化算法不适用。局部动态阈值算法可以很好解决这个问题。经过增强处理的指纹都会有较为清晰的方向场, 而方向场是否可以估计的准确, 直接决定指纹图像二值化算法的效果。为了估计方向场, 把指纹脊线走向分如图12的8个方向。

局部动态阈值法具体步骤如下:

Step1: 对图像进行平均滤波^[17], 采用模板系数对

$$\frac{1}{9} \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix}$$

像素点进行处理。

Step2: 在像素点中心 9×9 的窗口中, 分别计算 8 个方向上, 经过处理后的灰度值。即将图 12 中标了 i (i 取 0 到 7 的值, 分别代表 8 个方向) 位置像素灰度值, 取其中最大的值 $summax$ 和最小的值 $summin$, 计算像素均值 $B=sum/8$, 如满足 $(summax+summin+ 4 * I(x,y)) > (3 * B)$, 则这个像素点的脊线方向为 $sumf=summin$, 否则为 $sumf=summax$ 。

2		3		4		5		6
1		2	3	4	5	6		7
		1				7		
0		0		●		0		0
		7				1		
7		6	5	4	3	2		1
6		5		4		3		2

图 13 在一个像素点的 8 个指纹脊线方向

Step3: 若 $sumf > B$, 指纹图像灰度值设置为 128, 否则将灰度值设置为 255。

Step4: 重新修正 I_{cc} (分割时用来区分背景的, 值为 1 时是背景指纹纹线) 的值, 使图像的灰度值只有 0 和 1, 完成对图像二值化的处理。方法如下: 如果灰度值为 128, 并且 $I_{cc}=1$ 时, 修正为 $I_{cc}=0$, 指纹图像像素值为 0。背景以及灰度值为 255 的纹线, 指纹图像像素值为 1。目的是除去不确定的点, 此时指纹图像背景为白, 纹线为黑。

二值化结果如图14所示, 与图12的实验结果对比, 可以看出局部动态阈值法的二值化效果比全局阈值化效果好。



图 14 指纹图像二值化

指纹图像只有黑和白两种像素, 突显出指纹的脊线。

(5) 指纹图像的细化处理

细化处理是对二值化指纹图像进行的处理, 也是预处理过程中的最后一步。在不影响纹线连接的前提下, 该处理方法将指纹纹线边缘的一些象素剔除, 直至将指纹的纹线的宽度变成单像素的宽度为止, 指纹的细化又被称为指纹纹线骨架化^[18,19]。通过对指纹纹线的侵蚀, 使得纹线达到一个象素宽度, 从而保证指纹图像的精细程度。经过二值化的指纹图象中仍有大量的多余信息。

指纹图像经过二值化后, 会导致空洞和毛刺等伪特征引入到二值指纹图中, 也会造成纹线宽度不均匀, 对指纹特征提取产生影响。因此, 为避免毛刺空洞等伪特征带来的影响, 需要对指纹图像进行去毛刺和空洞的处理。如图15所示, P点8邻点是P1到P8, P的4邻点是P2、P4、P6、P8。

P1	P2	P3
P8	P	P4
P7	P6	P5

图 15 8领域点示意图

• 去毛刺

若 $P(x,y)$ 点的灰度值为 0, 其 4 邻点灰度值为 1 的个数不小于 3, 满足图 16 所示的情况之一, 修改 P 点的灰度值为 1。

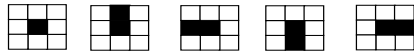


图 16 毛刺点

• 去空洞

若 $P(x,y)$ 点的灰度值为 1, 当其 8 邻点满足下面式 (20) 时, 修改 P 点的灰度值为 0。

$$\sum_{i=1}^8 |P_{i+1} - P_i| \neq 1$$

$$(P2 + P3 + P4) \times (P6 + P7 + P8) + (P1 + P2 + P8) \times (P4 + P5 + P6) = 0$$

(20)



图 17 去毛刺



图 18 去空洞

指纹细化的过程就是在保持原指纹纹线拓扑结构, 连接状态以及细节特征等信息的前提下, 将二值化后的指纹图像脊线腐蚀细化至单像素宽纹线^[18]。指纹图像为单像素宽时, 最大限度地消除了无用信息, 从而降低数据量和后续处理的复杂度。

形态学处理在图像处理中应用最广的技术之一, 在图像简化和物体形状主要特征的保留等方面, 发挥着至关重要的作用^[20]。论文使用MATLAB自带的函数 `imopen` 和 `imclose` 函数对指纹图像进行开闭运算, 再使用 `bwmorph` 函数对指纹图像形态学进行操作, 最后完成预处理的最后一步指纹图像的细化处理。

形态学腐蚀是求局部最小值的运算, 而膨胀是求局部最大值的运算。其腐蚀和膨胀的数学表达式分别为:

$$[\varepsilon_B(I)](x) = \min_{b \in B} (X + b) \quad (21)$$

$$[\delta_B(I)](X) = \max_{b \in B} (X + b) \quad (22)$$

其中, $I(X)$ 表示图像, B 结构元素。

对图像进行开运算处理过程是先腐蚀, 再膨胀。通过这种方法, 可以消除孤立的微小点、毛刺和连接两个区域的小点, 从而保持整体位置和形状的完整性。指纹图像经过开运算后, 能够平滑图像的轮廓, 削弱狭窄的部分, 去除一些孤立的小点。闭运算则与之相反, 先进行膨胀操作, 再进行腐蚀操作, 能够填平小湖(即小孔), 弥合小缝隙, 而整体的位置和形状不变。指纹图像经过闭运算, 可以弥合断裂的地方, 填补轮廓上的缝隙。开、闭运算的表达式如下:

$$I^{\vee}(x) = \delta_B[\varepsilon_B(I(X))] \quad (23)$$

$$I^{\circ}(X) = \varepsilon_B[\delta_B(I(X))] \quad (24)$$

`bwmorph(I, 'thin', n)` 在 $n = \text{Inf}$ 时, 通过从对象边界删除像素, 将对象收缩为线。使不带孔洞的对象收缩为具有最少连通性的线, 带孔洞的对象收缩为每个孔洞和外边缘之间的连通环。

指纹细化图如图19所示, 细化指纹突出了纹线的有效特征。



图 19 指纹细化图

3 结束语

本文在确保指纹的相似性和连接性的前提下, 对指纹进行预处理, 以求获得指纹细化图。预处理过程包括指纹的增强、归一化、分割、二值化以及细化处理。首先研究指纹图像增强技术, 通过直方图均衡化增强指纹图像可视化效果, 中值滤波对指纹图像进行去噪声处理; 然后研究指纹分割技术, 在分割之前需要先对指纹图像进行归一化处理, 再将指纹的分离出指纹的前景区域; 之后研究指纹图像的二值化处理相关技术; 最后研究指纹图像细化过程。所有处理过程均使用MATLAB实现, 并给出处理结果。

参考文献

- [1] 陈静. 数字图像处理中指纹识别技术的分析[J]. 电子测试, 2019, (18):116-117.
- [2] 王霞英. 基于 MATLAB GUI 指纹图像特征提取与对比的研究[D]. 太原: 中北大学, 2013.
- [3] H. M. Ali and S. Corraya, "Line profile-based fingerprint matching,"[C]. 2016 International Workshop on Computational Intelligence (IWCI), Dhaka, Bangladesh, 2016, pp. 115-119.
- [4] 李金铎. 基于图像处理的指纹识别算法研究[D]. 哈尔滨: 哈尔滨工程大学, 2016.

- [5] 任毅. 指纹识别系统的研究和实现[D]. 南京: 南京邮电大学, 2019.
- [6] 王敏, 杨永跃, 洪占勇. 基于 MATLAB 指纹图像处理与识别的研究[J]. 电脑知识与技术, 2014, 10(10):2359-2362.
- [7] 肖大雪. 基于 MATLAB 的指纹识别研究与实现[J]. 科技广场, 2014, (08):30-37.
- [8] Jinwen Yang, Weihe Zhong and Zheng Miao, "On the Image enhancement histogram processing,"[C]. 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCS), Jinzhou, China, 2016, pp. 252-255.
- [9] 陈欣. 指纹识别处理关键技术研究[D]. 南京: 南京理工大学, 2017.
- [10] 王双. 一种改进的中值滤波算法[J]. 电子技术, 2007, (11):120-123.
- [11] S. Vaikole, S. D. Sawarkar, S. Hivrare, et al. Minutiae Feature Extraction from Fingerprint Images[C]. 2009 IEEE International Advance Computing Conference, Patiala, India, 2009, pp. 691-696.
- [12] 黄善斌, 覃勇军. 基于计算机视觉的静态场智能监控系统[J]. 科技信息, 2010, (30):672-673.
- [13] 张天良. 指纹信息提取技术分析的实现[D]. 开封: 河南大学, 2020.
- [14] S. Gayathri and V. Sridhar. ASIC implementation of image enhancement technique for Fingerprint recognition process[C]. 2014 International Conference on Contemporary Computing and Informatics (IC3I), Mysore, India, 2014, pp. 868-873.
- [15] 汤正刚. 自动指纹识别系统中的关键技术研究[D]. 杭州电子科技大学, 2019.
- [16] 叶四民, 陈福祥. 指纹图像预处理中的二值化技术[J]. 自动化与仪器仪表, 2001,(02):32-34+41.
- [17] 薛晨. 基于身份识别技术驾驶员培训信息管理系统[D]. 武汉: 武汉工程大学, 2015.
- [18] 季善东. 指纹识别技术与考勤系统应用研究[D]. 南京: 南京邮电大学, 2017.
- [19] 陈斯宇. 指纹识别图像预处理系统的设计和实现[D]. 吉林: 吉林大学, 2015.
- [20] 张丽华, 孙钢灿, 酒明远. 基于底帽变换与图像融合的图像去雾算法[J]. 软件导刊, 2022, 21(2):177-183.