

指纹图像自动识别系统预处理算法的研究

兰州中科院近代物理所(730000) 洪华军 乔为民 朱立新

摘要: 系统地阐述了指纹图像自动识别系统预处理的理论、算法,并对指纹图像预处理中的各种技术进行了分析及实验。在比较了各种预处理方法之后,提出了一种比较好的指纹图像预处理算法,并在实验中得到了验证。

关键词: 指纹识别 预处理 图像增强 二值化 细化

由于指纹具有终身的稳定性和惊人的特殊性,很早以来在身份鉴别方面就得到了应用,且被尊为“物证之首”。但人工对比指纹,效率低、速度慢,已愈来愈不能满足现代社会的需要。70年代在美国已开始用图像处理和模式识别方法进行计算机指纹分析以代替人工对比,并在70年代末80年代初研制成有关设备,投入运行。

我国在近10年来已陆续开展这方面的工作,现已进入实用阶段。本学科的技术用于指纹分析,有两方面的工作:

(1) 指纹特征提取与识别。这是比较成熟的工作,因为每个人都有固定的指纹类型,而指纹纹线总的类型有限。因此,有可能把指纹经过预处理以后,提取特征,再用类聚分析方法进行判决分类。

(2) 指纹存储与归档。这是大容量存储的问题,试想每人有10个指纹,而全国十几亿人要存储上百亿个指纹,这需要多么巨大的存储容量。除了用激光存储等高密度的记录和存储手段外,还急切需要开展指纹的无失真、可恢复的压缩工作。

近几年来,国外发展了用计算机进行指纹对比的自动识别系统。指纹图像信息量大,处理及时,这些系统是基于中、小型计算机,甚至是专用高速并行处理机。

为了迅速展开指纹自动识别的研究,我们在微型机上进行了微型机指纹自动识别系统的尝试。该系统的输入部分把实际捺印指纹或现场指纹转换成数字化图像,经总线存入图像存储器,微机对输入的图像进行各种必要的处理和分类,其处理效果可由彩色显示器立即评价,或由打印机输出判别结果。

一般指纹图像自动识别系统的原理如图1所示。

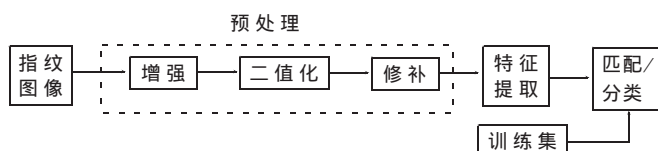


图1 指纹特性识别原理图 (文法推理)

指纹图像预处理的目的在于使指纹图像画面清晰,边缘明显,以便提取特征进行识别。预处理技术的主要目的是对一个给定的指纹,突出指纹图像中的某些信息,消弱或除去某些不需要的信息,使它的结果对后面的识别来说比原始指纹图像更合适。其预处理流程图如图2所示。

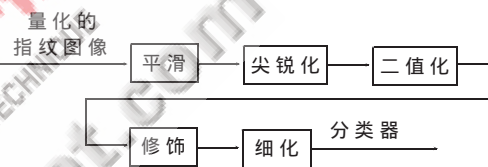


图2 指纹图像预处理流程图

只有经过预处理后的指纹图像,才能进入到下一步(指纹的识别预分类)去处理。预处理结果的好坏直接关系到指纹识别率的高低。

1 图像增强

从处理手段来讲,图像增强处理可分为空域法和频率域法两种。空域法指在图像所在的空间域中直接处理;而频域法指先把图像作傅立叶变换,在频率域中处理后,作傅立叶反变换。

我们采用了先平滑化,然后进行尖锐化两部处理。该方法可以进行自动处理指纹图像,且效果较好。

1.1 平滑处理

指纹图像是通过扫描仪获得模拟信号,并经采样、量化后,以矩阵的形式存入计算机。由于图像的采集为纵列式方式,量化后的指纹图像有许多噪声。它们因其时间的不相干性,含有较高的空间频谱,且多呈点结构。

平滑处理的任務就是去除这些干扰噪声,而又不使图像失真。图像平滑包括空域法和频域法两大类。由空域处理公式知,输出函数为:

$$g(m,n) = \sum_{i=m}^{m+L-1} \sum_{j=n}^{n+L-1} f(i,j)h(m-i+L,n-j+L) \quad (1)$$

其中 $m=1,2,\dots,H'$; $n=1,2,\dots,N'$; $M'=M-L$; $N'=N-L$ 。若令输入图像阵 F 为 $N \times N$,卷积阵 H 为 $L \times L$,输出阵 G 为 $M \times M$,则关系式:

$$G(m_1, m_2) = \sum_{n_1} \sum_{n_2} F(n_1, n_2) H(m_1 - n_1 + 1, m_2 - n_2 + 1) \quad (2)$$

为了平滑噪声,卷积阵 H 需呈低通型。当 H 为 3×3 阵列时,选用:

$$H = (\sum_{i=1}^8 a_i + 2X) / 10 \quad (3)$$

其中 X 为要处理的像素值, $a_i (i=1, 2, \dots, 8)$ 为其八邻域的数值。

采用多图像平均法(即多遍输入叠加取均值的方法),可去除时间噪声。它以噪声干扰的统计学特征为基础。即如果一幅图像包含有噪声,可以假定这些噪声相对于每一坐标点 (x, y) 是不相关的。其数学期望为零。设 $g(x, y)$ 是有噪声 $\eta(x, y)$ 和原始图像 $f(x, y)$ 叠加而成的。即:

$$g(x, y) = f(x, y) + \eta(x, y) \quad (4)$$

若噪音 $\eta(x, y)$ 满足上述的假定,则可以通过将给定的一系列噪声图像 $\{g_i(x, y)\}$ 叠加后取其平均值,以达到平滑图像的目的。

因此,在获得一幅图像时,可用采集 M 次的方法去取平均值,即可达到要求。

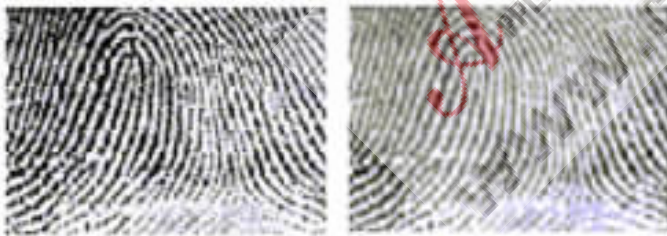
$$g(x, y) = \frac{1}{M} \sum g_i(x, y) \quad (5)$$

因为:

$$E\{g(x, y)\} = f(x, y) \quad (6)$$

一般情况下,当 $n=4$ 时,图像趋于稳定。

实践表明,用这些方法去除噪声,可获得满意的效果。图 3 为去噪声后的指纹图像。



(a) 原指纹 (b) 去噪声后的指纹

图 3 指纹图像增强后结果图

1.2 锐化处理

为强化指纹纹线间的界线,突出边缘信息,以利于二值化,锐化处理是必要的。锐化处理对于增强反差和检测边缘是很有用的。图像锐化的作用就是补偿图像的轮廓,使图像较清晰。图像锐化可分为空间域图像锐化法和空间频率域图像锐化法两大类型。

图像的模糊,是由于高的空间频率成分比低的空间频率成分弱这一原因造成的。这一影响表现在均匀灰度区域间的边界部分(边缘)。因此,为了消除模糊,可以增强高的空间频率成分。

作为在图像空间简单的高频增强滤波有:

$$g(i, j) = f(i, j) - \nabla^2 f(i, j) = 5f(i, j) - [f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)] \quad (7)$$

这里 $\nabla^2 f(i, j)$ 表示输入图像 $f(i, j)$ 的二次微分的数字拉普拉斯算子,被定义为:

$$\nabla^2 f(i, j) = f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1) - 4f(i, j) \quad (8)$$

由于模糊的图像是感光过程中的扩散现象所致,因此,它必然满足线形扩散方程:

$$\frac{\partial f(x, y, t)}{\partial t} = k \nabla^2 f(x, y, t) = k \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right) \quad (9)$$

其中 $f(x, y, t)$ 是时间与空间的函数, k 是扩散系数; $\nabla^2 f$ 是 f 的拉普拉斯算子。如令 $t=0$,且忽略其展开的泰勒级数的高次项,则输出函数:

$$g(x, y) = f(x, y, t) - \tau k \nabla^2 f(x, y, t) \quad (10)$$

其中 τ 为扩散时间间隔的长度。选取不同的 τk 值,可得到不同的拉普拉斯算子的不同加权系数。经过多次实验知, 7×7 方阵的加权系数 p 为:

$$p = 8f(x, y) - f(x-3, y) - f(x-2, y+2) - f(x, y+3) - f(x, y-3) - f(x+2, y+2) - f(x+3, y) - f(x+2, y-2) - f(x-2, y-2) \quad (11)$$

时能更好地反映出泰勒级数展开时函数的连续性,用于指纹图像的锐化处理是适宜的。图 4 是锐化后的指纹图像。



图 4 指纹图像锐化结果图

2 二值化处理

二值图像是指整幅图像画面内仅黑(灰度值为 0)白(灰度值为 1)二值的图像,在它们上面不呈现出灰度的变化。在数字图像处理中,二值图像占有非常重要的地位。这是因为在实用的图像处理系统中,要求处理的速度高、成本低,信息量大的浓淡图像处理开销太大,不是上策。而且二值化后的图像能够用几何学中的概念进行分析和特征描述,比起灰度图像来说方便得多。因而二值图像处理目前已成为图像处理中的一个独立的、重要分支而获得广泛的应用。对于指纹的识别来说,有用的信息是包含在脊线和谷线的二值描述中。因而必须根据原始的灰度图像来确定图像上的每一点应属于客体区域还是背景区域,从而产生对应的二值图像。它不仅大大减少储存量,而且可使得后面的判别过程少受干扰,大大简化其后的处理方法。二值化后的图像是以后处理的基础,它的算

法对后面的处理有直接的影响。一个好的算法可以得到一个高质量的二值图像。反之,如果该阶段引入噪声,就会直接降低图像质量,影响识别精度。

位于 (x, y) 处的像素 $f(x, y)$ 阈值 $V(x, y)$ 是由以 (x, y) 为中心的指纹图像窗口 $(2m+1) \times (2m+1)$ 中诸点的灰度值来确定的。即:

$$V(x, y) = \frac{1}{(2m+1)^2} \sum_{i=-m}^m \sum_{j=-m}^m f(x, y) + \beta \quad (12)$$

当 $(2m+1) \times (2m+1)$ 窗口落在两种不同区域将会引起误判:

- (1) 当窗口较多地落在谷线区时,将会使得一部分乃至大部分的像素点被判为脊线;
- (2) 当窗口较多地落在脊线区时,将会使得一部分乃至大部分的像素点被判为谷线。

借助于参数 β 和固定阈值可以解决这个问题。

令 ε 为一个通过试验获得的正整数,即 $\varepsilon > 0$, 故有:

$$\text{令: } V(x, y) = \frac{1}{(2m+1)^2} \sum_{i=-m}^m \sum_{j=-m}^m f(x, y) \quad (13)$$

$$\text{则: } V(x, y) = \begin{cases} V_1(x, y) + \varepsilon & V_1(x, y) < V_t \\ V_1(x, y) - \varepsilon & V_1(x, y) \geq V_t \end{cases} \quad (14)$$

其中 V_t 为参考阈值。

令二值化后的图像为 $g(x, y)$, 则:

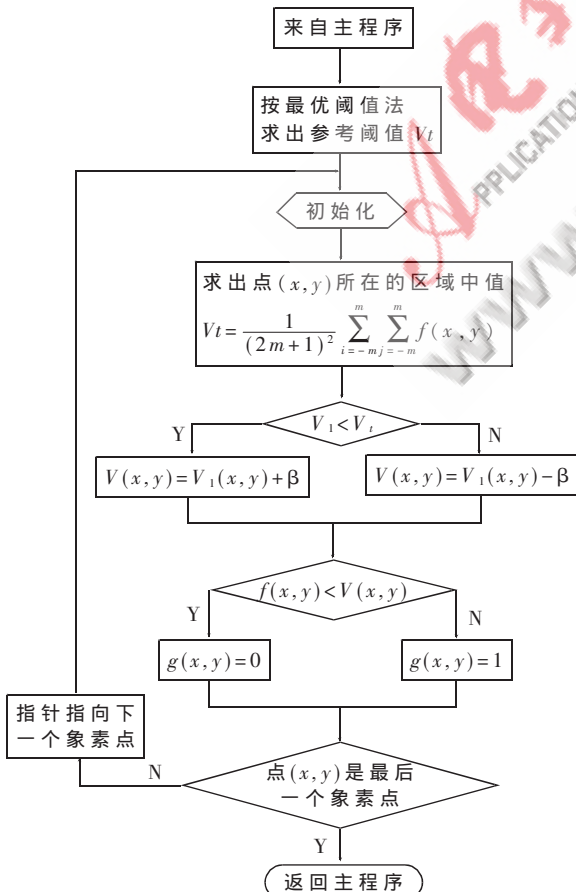


图 5 二值化流程图

$$g(x, y) = \begin{cases} 1 & f(x, y) \geq V(x, y) \\ 0 & f(x, y) < V(x, y) \end{cases} \quad (15)$$

以上过程可用流程图 5 来表示。

3 修饰处理

指纹图像经过二值化后,由于量化等原因,纹线边缘凹凸不齐,受锐化的影响,画面出现离散点。为使图像整洁,边缘圆滑,需要进行修饰处理。

用适宜的模板可去除纹沟中的离散黑点和填补纹线中的空缺白点。如令 3×3 模板中待处理的像素为 x , 其邻域 $\forall a_i \in \text{白} (i=1, 2, \dots, 8)$, 则 $x = \text{白}$; $\forall a_i \in \text{黑} (i=2, 4, 6, 8)$, 则 $x = \text{黑}$ 。

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_8 & x & a_4 \\ a_7 & a_6 & a_5 \end{bmatrix} \quad (16)$$

修饰处理包括去除孤立点及填补空白点等操作。

它们各自的模板表征如下:

去除孤立点:

$$\begin{matrix} * & * & * & * & * & * & * & * & * & 0 & * \\ 0 & 1 & * & * & 1 & 0 & * & 1 & * & * & 1 & * \\ * & * & * & * & * & * & * & 0 & * & * & * & * \end{matrix} \quad (17)$$

(a) (b) (c) (d)

去除空白点:

$$\begin{matrix} * & 1 & * & * & * & * & 1 & * & * & * & * & 1 \\ * & 0 & * & 1 & 0 & 1 & * & 0 & * & * & 0 & * \\ * & 1 & * & * & * & * & * & * & 1 & 1 & * & * \end{matrix} \quad (18)$$

(a) (b) (c) (d)

其中“*”表示为 0 或 1 的任意值。

4 细化处理

由于我们所关心的不是纹线的粗细,而是纹线的有无。因此,在破坏图像连通性的情况下必须去掉多余的信息。在二值图像处理领域,细化是很重要的一个处理环节。如干涉条纹图像,由于条纹粗、宽二边缘弥散,不细化成线状就没有办法去精确地取数计算它。另一方面,一个连接成分如果能用线状结构去值图像的细化问题成为图像处理的一大热门,有不少人研究它的算法。在进行指纹识别时我们主要着眼于指纹脊线的轮廓和有关的端点和交叉点。但是,二值化后的脊线有一定的宽度,这种表征它的特性,无论在图像识别还是在数据压缩方面都有重要意义。所以,二宽度会给后面的识别增添很多麻烦。因而应先将指纹脊线的宽度采用逐渐剥离的方法,使得脊线成为只有一个像素宽的细线,这将非常有利于下一步的分析,这个过程叫细化。其目的是用一组细线来刻画一个连接成分,这不仅能达到压缩数据量的目的,而且易于对连接成分的特征提取。因而,这一操作过程必然需要删除某些像素。

利用两种类型的模板进行细化处理的算法如下:

A 型模板

$$\begin{array}{c|c|c|c} 0 & 0 & * & * & 0 & 0 & * & 1 & * & * & 1 & * \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ * & 1 & * & * & 1 & * & * & 0 & 0 & 0 & 0 & * \\ \hline A1 & A2 & A3 & A4 & & & & & & & & \end{array} \quad (19)$$

B 型模板

$$\begin{array}{c|c|c|c} 0 & 0 & 0 & 1 & * & 0 & * & 0 & 1 & 0 & 0 & 0 & * \\ * & 1 & * & 1 & 1 & 0 & * & 1 & * & 0 & 1 & 1 & \\ 1 & 1 & * & * & * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 1 \\ \hline B1 & B2 & B3 & B4 & & & & & & & & & & \end{array} \quad (20)$$

图 6 即为通过算法 (19) 处理后得到的三种不同类型的指纹。

指纹图像细化后,还可以根据指纹得到一般纹理规律。进行后加工处理,如去掉无效的纹线搭桥及连接豁口等。还可以采用人机交互方式,对残缺指纹进行人工修补。

本文给出了一套完整的指纹图像预处理算法,并在实验中获得了比较满意的结果。

参考文献

- 1 张大鹏著.模式识别与图像处理并行计算机系统设计(第一版).哈尔滨:哈尔滨工业大学出版社,1998:272~318
- 2 [日]田村秀行著,郝荣威等译.计算机图像处理技术(第一版).北京:北京师范大学出版社,1986:8~133
- 3 马颂德,张正友著.计算机视觉——计算理论与算法基



图 6 指纹图像细化结果图

础(第一版).北京:科学出版社,1998:1~34。

- 4 程存学,朱晓昆著.计算机视觉——低层处理技术(第一版).北京:电子工业出版社,1992:85~148
- 5 Kenneth R. Castleman. Digital Image Processing (1). Prentice-Hall International, Inc. 1997:1~115
- 6 张远鹏,董海,周文灵著.计算机图像处理技术基础(第一版).北京:北京大学出版社,1996:1~149

(收稿日期:2000-07-20)