

一种改进的指纹图像分割算法^①

An Improved Segmentation Algorithm for Fingerprint Images

赵磊 魏书堤 (衡阳师范学院 计算机系 湖南 衡阳 421008)

摘要: 指纹图像分割是指纹识别技术中的重要步骤之一。本文在分析了已有分割算法存在的不足的基础上, 提出了一种改进的指纹图像分割算法, 该算法引入像素聚类的概念, 将灰度特性和像素聚类相结合, 形成以灰度特性为主、像素聚类为辅的分割算法, 该算法克服了前面算法由单一阈值分割的缺陷。实验表明对于背景条件较复杂的指纹图像, 本算法也能很准确地实现指纹图像的分割, 具有较好的鲁棒性。

关键词: 指纹分割 归一化 灰度均值 灰度方差 像素聚类

1 引言

图像分割是从一幅图像中, 按一定规则将一些物体或区域加以分离, 划分出感兴趣的部分或区域。经过分割后的图像更容易进行下一步的分析和识别处理。近年来, 图像分割技术被广泛地应用在自动指纹识别技术中^[1], 成为自动指纹识别领域的一个研究重点。指纹图像分割在归一化后进行, 其目的是将指纹图像中的有效区域与无效区域分割开, 所谓的有效区域指的是清晰的指纹图像, 而无效区域是指采集区域中无纹线部分及纹线非常模糊的部分。采集的指纹图像, 总是不可避免地存在无效区域, 而且有时候无效区域往往要占到整个采集区域面积的三分之一, 甚至是一半。如果不将指纹的无效区域分离出来, 不仅需要处理整个采集区域, 造成处理速度的大幅下降, 而且会严重影响图像处理算法的处理效果。所以指纹图像分割使后续的处理能够集中于有效区域, 从而提高特征提取的精确度, 减少指纹预处理的时间。

迄今为止, 人们已经对指纹图像分割算法进行了许多研究工作^[2-5]。现有的指纹图像分割算法主要是建立在对方向特性或者灰度特性的研究上, 对于一些质量差的指纹图像, 这些方法效果并不理想, 不能很好地适应各种质量的指纹图像。本文将聚类的方法引入到指纹图像分割, 将原有的灰度特性与像素聚类结合起来, 实现对各种质量的指纹图像更为有效的分割。

2 已有的指纹图像分割算法

一幅质量较好的指纹图像, 在指纹纹线分布区域由于脊和谷的交错分布一般灰度方差会比较大, 而对于没有指纹纹线的区域灰度方差会较小。常规的指纹图像分割算法^[6]就是通过对指纹图像的灰度方差进行统计, 方差大于某个预先设定的阈值 T 为前景区域, 反之为背景区域。

常规的指纹图像分割算法的流程如下:

① 将归一化后的指纹图像划分为 $W \times W$ 互不重叠的子块, 分割尺寸 W 一般选择为 8 或者 16;

② 计算每个子块的平均灰度值 $Mean$ 和方差 Var , 其中指纹图像中坐标为 (i, j) 的像素原灰度值为 $Gray(i, j)$:

$$Mean = \frac{1}{N \times N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} Gray(i, j) \quad (1)$$

$$Var = \frac{1}{N \times N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (Gray(i, j) - Mean)^2 \quad (2)$$

③ 对于每一个子块, 若其方差小于预先设定的阈值 T , 此子块为背景块, 否则为前景块。

文献[2]认为常规算法中灰度均值的求取方法没有突出指纹图像纹理的基本特征, 灰度均值应能将某个区域内的像素准确地划分为两个近似相等的像素点集的灰度阶。为此提出一种自适应灰度均值提取算法, 算法如下:

① 针对待求指纹图像块, 采用公式(1)求取灰度

① 基金项目: 湖南衡阳师范学院科学基金启动项目(07B32)

收稿时间: 2008-11-01

均值 A ;

② 分别计算 $N_h =$ 灰度值大于等于 A 的像素点个数; $N_l =$ 灰度值小于 A 的像素点个数;

③ A 如果有 $|N_h - N_l| \leq \alpha (\alpha = w \times w \times 10\%)$, 则即 A 为该区域的灰度均值, 程序结束; 否则采用如下公式 (3) 修改灰度均值 A 的值:

$$A = \begin{cases} A + 1, & \text{if } (N_h > N_l) \\ A - 1, & \text{if } (N_h \leq N_l) \end{cases} \quad (3)$$

④ 返回②, 重新开始。

文献[3]认为在指纹区域, 沿脊线方向的灰度变化较小, 体现了纹线的走向, 在脊线垂直方向上的灰度变化则较大; 而在背景区域和噪声区, 各个方向的灰度变化都较小, 该算法利用了灰度的变化结合方向对指纹图像进行分割。

经过大量的实验, 我们发现对于纹线较为清晰的指纹图像文献[2]和文献[3]算法分割效果很好, 但是对于质量较差, 纹线并不清晰的指纹图像, 上述算法的效果并不理想。对于图 1(a)的指纹图像, 在图像的下面存在

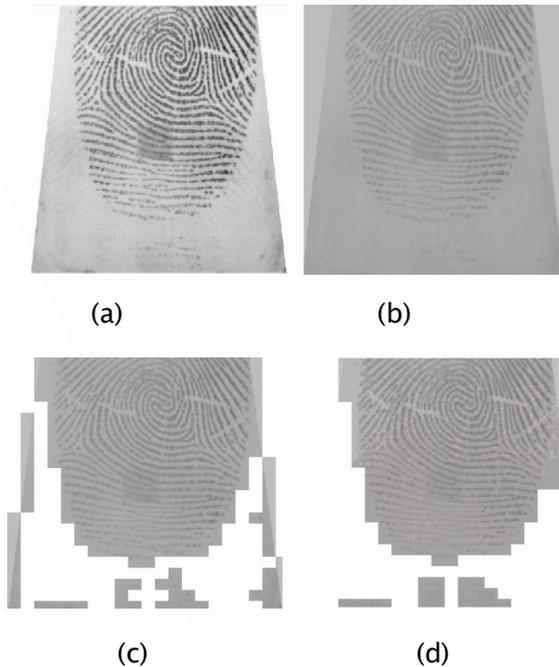


图 1 指纹图像分割示意图

少量的残留指纹, 这些残留指纹是不能作为前景区域的。并且该图像经过归一化后(如图 1(b))两侧出现灰度的梯度, 对指纹分割也造成影响。图 1(c)和(d)分别是文献[2]算法和文献[3]算法对图 1(b)进行分割的结果。

从图 1(c)和(d)可以看出, 文献[2]算法不但保留了残留指纹, 而且对于图像两侧的灰度梯度也保留了下来, 文献[3]算法虽然没有保留两侧的灰度梯度, 但是保留了残留指纹。因为上述两种算法在判断图像子块是否为前景块或背景块时都采取的是单一的灰度变化阈值比较, 虽然阈值的选取是经过大量的实验得到的, 但是指纹图像间的不同对比度和不同指纹采集仪采集的指纹图像会对阈值的选取造成很大的影响, 所以单一的阈值选择不能满足指纹分割的实际需要。

另外, 在实验中有一些指纹图像是偏指纹的情况如图 2 所示。这些指纹图像是在采集指纹时手指没有放在指纹仪的正中而产生的。在这种情况下, 前面的算法将指纹图像分为 $W \times W$ 子块时不一定能被图像的长和宽整除, 从而产生剩余块。 W 取得越大, 剩余块可能就越大指纹的一些特征数据可能在剩余块中而被忽略掉了, 不利于指纹特征数据的提取。



图 2 偏指纹图像

3 改进的指纹图像分割算法

针对上述算法存在的不足, 本文引入像素聚类的方法, 结合上述算法的优点, 提出了一种灰度特性和像素聚类的相结合分割算法。像素聚类属于特征空间聚类方法的一种, 特征空间聚类方法将图像空间中的元素用对应的特征空间点表示, 通过对特征空间的点聚集成团, 然后映射回原图像空间以得到分割的结果。在指纹图像中, 一般以像素的灰度为特征, 用灰度直方图代表特征空间, 用阈值将特征空间划分开, 从而完成指纹图像的分割。像素聚类方法与前面的方法相比最大的特点在于它是从单个像素这一方面出发进行分割, 而不是从图像子块这个整体出发进行分割。

灰度特性和像素聚类相结合的分割算法采用以灰度特性分割算法为主, 像素聚类为辅的分割策略。该算法在以灰度特性分割算法的基础上, 根据有效指纹

脊线与背景区域的差异，结合指纹脊线的相关特征，将指纹图像上的像素点分为前景像素点和背景像素点，然后根据图像分块后各个子块中存在前景点和背景点的相对比例关系来决定该子块为前景块或背景块，从而弥补由单一阈值分割图像存在的不足。同时算法在指纹图像分割时，注意对分割后的剩余块进行处理，避免丢失存在于剩余块中的特征数据。

整个算法的关键部分是得到脊线的中心灰度 μ_1 和谷线的中心灰度 μ_2 。图 3 显示了指纹图像的灰度直方图，从图上可以看出灰度直方图有两个明显的峰值，灰度值较小的峰值为脊线的中心灰度 μ_1 ，灰度值较大的峰值为谷线的中心灰度 μ_2 。但是对于一些图像质量不好的指纹图像，其灰度直方图上没有明显的峰值如图 4 所示，这时要得到脊线的中心灰度和谷线的中心灰度就要困难一些。本文采用下面算法得到脊线和谷线的中心灰度。

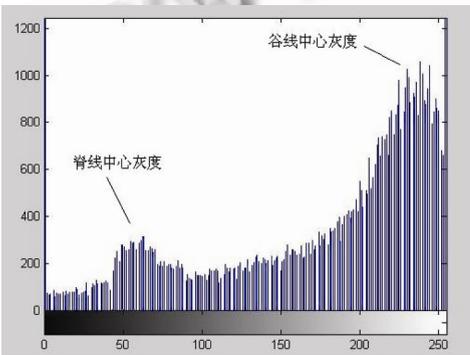
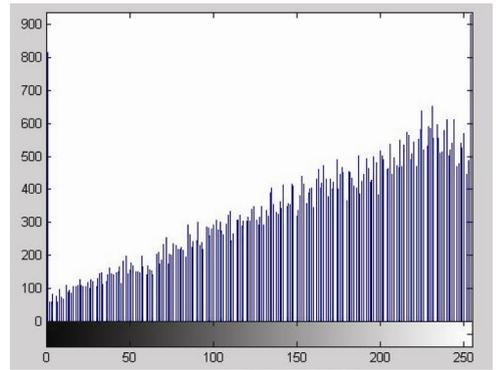


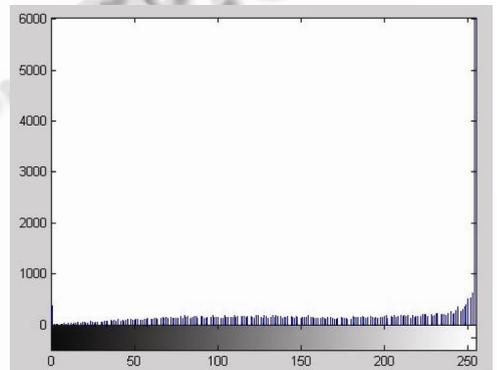
图 3 指纹图像灰度直方图

- ① 用灰度值 T_1 将指纹图像的灰度值分为两个区间，一个区间为 $[0, T_1]$ ，另一个为 $(T_1, 255)$ ，分别计算两个区间的像素点的总数 (T_1 为设定的阈值)；
- ② 对于区间 $[0, T_1]$ ，找到该区间中像素个数最多的灰度值 G_1 ，如果该灰度值的像素个数超过该区间总像素个数的 60%，则该灰度值就为脊线的中心灰度，否则计算区间 $[G_1 - 1, G_1 + 1]$ 的像素个数，如果该区间像素个数超过该区间总像素个数的 60%，则将该区间的加权平均灰度作为脊线的中心灰度，否则计算区间 $[G_1 - 2, G_1 + 2]$ 的像素个数，再进行判断，如果还不满足条件再将区间增加，直到满足条件为止，此时该区间上的加权平均灰度为脊线的中心灰度 μ_1 ；
- ③ 对于区间 $(T_1, 255)$ ，用步骤(2)同样的方法求得谷线的中心灰度 μ_2 。

上述算法通过渐进区间选择和加权平均方法求取



(a)



(b)

图 4 质量不好的指纹图像灰度直方图

指纹图像的脊线和谷线的中心灰度，该算法特别对于质量不好的指纹图像能得到合适的 μ_1 和 μ_2 ，同时具有很好的稳定性。

整个改进的指纹图像分割算法的流程如下：

- ① 将指纹图像划分为互不重叠的 $W \times W$ 子块和剩余子块；
- ② 针对每一个子块指纹图像 (i, j) ，用公式(1)求该区域灰度平均值 $\bar{A}(i, j)$ ；
- ③ 用公式(4)求取各个子块图像的灰度值方差 $Var(i, j)$ ；

$$Var(i, j) = \frac{1}{M \times N} \sqrt{\sum_{m=1}^M \sum_{n=1}^N (g(m, n) - \bar{A}(i, j))^2} \quad (4)$$

其中， M, N 为图像子块的长和宽， $g(m, n)$ 表示像素 (m, n) 的灰度阶。

- ④ 使用前面介绍的算法，得到脊线的中心灰度 μ_1 和谷线的中心灰度 μ_2 ；
- ⑤ 将每个处于区间 $[\mu_2 - T, 255]$ 的像素点直接定义为背景像素点，(T 为设定的阈值)；

⑥ 位于区间 $[\mu_1 + T, \mu_2 - T]$ 的像素点, 如果该像素点与所属子块的灰度均值 $\bar{A}(i, j)$ 的方差小于该子块的方差 $Var(i, j)$, 则该像素点为背景像素点;

⑦ 对于每一个子块和剩余子块, 若其方差大于阈值 T 且子块中背景像素点的数目占整个子块像素点总数百分比小于 70% 则该子块为前景块, 如果方差小于阈值 T 但是该子块中的背景像素点的数目占整个子块像素点总数百分比小于 30% 则该子块也为前景块, 其余不符合条件的子块为背景块;

⑧ 进行基于子块和剩余子块的平滑, 消除孤立的前景块和背景块。

4 实验结果与分析

为了验证新算法的优点, 本文在赛扬 2.8G、1G 内存的计算机上用 VC 编程实现了文献[2]、文献[3]和本文的灰度特性和像素聚类的相结合分割算法, 选取了 FVC2004 指纹数据库和 Secugen 指纹采集器采集的多幅指纹图像进行实验(其中 T_1 取 120, T 取 15), 图 5 显示了上述算法分割实际结果的比较。

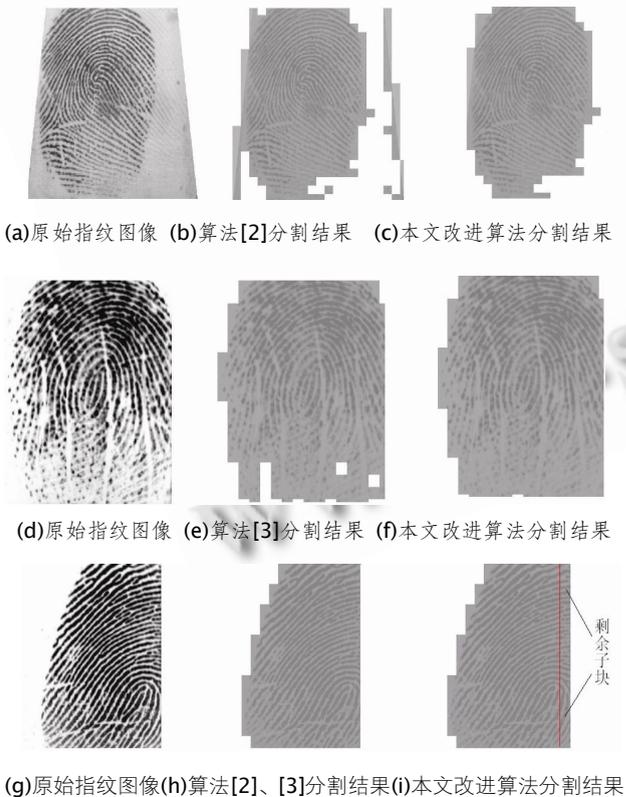


图 5 几种算法实际分割结果比较

在图 5 中, 图像(a) 是来自 FVC2004DB-2 数据库

的原始图像, 图像(d)和(g)是 secugen 指纹采集仪采集的原始图像。图(b)用文献[2]的方法分割图像(a), 图(e)用文献[3]的方法分割图像(d), 由于采用单一阈值作为分割条件, 分割效果并不理想, 图像(b)的两侧和下方存在本应为背景块的前景块, 图像(e)的下方存在本应为前景块的背景块, 图(c)和(f)采用本文改进的算法对图(a)和(c)进行分割, 使用灰度特性和像素聚类相结合的方法, 将图像子块中前景点和背景点的相对比例关系作为单一阈值的补充, 很好地消除了本应为背景块的前景块, 增加了本应为前景块的背景块, 弥补了单一阈值分割时存在的不足。对于偏指纹图像(g), 图(h)用文献[2],[3]的方法将指纹分块后剩余的指纹图像子块作为无效区域处理, 造成剩余子块中特征点的丢失, 而图(i)用本文的改进方法将指纹图像的剩余子块与其它子块一样处理, 从而保留了剩余子块中的特征点。

5 总结

通过实验可以看出, 文献[2, 3]的算法将指纹图像子块当作整体来处理, 采用单一的阈值对图像进行分割, 对于背景条件较复杂的指纹图像, 这两种算法效果并不理想。本文引入像素聚类的方法, 与灰度特性相结合形成改进的分割算法。该算法采用灰度特性为主、像素聚类为辅的思想, 对于图像子块即从整体, 又从各个像素点的局部进行处理, 改变原来单一阈值对于图像处理的局限性。实验证明, 该算法能准确地背景区域从指纹图像中分离出来, 对噪声有很强的抵抗力, 同时具有较好的鲁棒性。

参考文献

- 1 田捷, 陈新建, 张阳阳, 等. 指纹识别技术的新进展. 自然科学进展, 2006, 16(4): 400 - 408.
- 2 詹小四, 尹义龙, 等. 一种改进的指纹图像分割算法. 广西师范大学学报, 2006, 24(4): 207 - 210.
- 3 杨小冬, 宁新宝, 等. 自动指纹识别系统指纹图像分割算法的研究. 南京大学学报, 2004, 40(4): 424 - 431.
- 4 Mehtre BM, Chatterjee B. Segmentation of fingerprint images a composite method. Pattern Recognition, 1995, 28(11): 1657 - 1672.
- 5 Mehtre BM, Murthy N, Kapoor. Segmentation of fingerprint images using the directional images. Pattern Recognition, 1987, 20(4): 429 - 435.
- 6 尹义龙. 自动指纹识别系统研究[博士学位论文]. 长春: 吉林工业大学, 2000.