

基于改进基准点定位的指纹匹配算法

李俊伟, 周立俭, 崔学梅

(青岛理工大学通信与电子工程学院, 山东 青岛 266033)

摘要: 针对指纹匹配过程中基准点定位不准确与耗时长的缺陷, 提出一种基于改进基准点定位的指纹匹配算法。该算法借助指纹图像的中心点构造局部细节结构, 并在该结构上利用全等三角形原则求取基准点, 将所有细节特征转化到极坐标中, 利用可变界限盒的方法进行匹配。实验结果表明, 该算法所确定的基准点比较准确, 且耗时缩短, 可提高识别率与执行效率。

关键词: 指纹匹配; 基准点; 中心点; 局部结构; 极坐标

Fingerprint Match Algorithm Based on Improved Fiducial Point Location

LI Jun-wei, ZHOU Li-jian, CUI Xue-mei

(College of Communication and Electronic Engineering, Qingdao Technological University, Qingdao 266033, China)

【Abstract】 Because the fiducial point location is inaccuracy and time-consuming in fingerprint match, a new match algorithm based on improved fiducial point location is proposed. The algorithm constructs a local structure with the help of the core point, and figures out a fiducial point in this structure by using the principle of congruent triangles. All the minutiae features are expressed in polar coordinates, and the final decision can be made with a changeable boundary box method. Experimental results prove that the fiducial point location is more accuracy and the time is saved, thereby the recognition rate and efficiency are enhanced.

【Key words】 fingerprint match; fiducial point; core point; local structure; polar coordinates

DOI: 10.3969/j.issn.1000-3428.2011.14.054

1 概述

指纹匹配是自动指纹识别技术的关键环节, 它直接决定了整个指纹识别系统的优劣, 如何提高识别率并减少算法复杂度是指纹匹配所要研究的核心问题。目前, 最常用的指纹匹配算法是由美国 FBI 提出的点模式匹配法^[1], 它利用脊线端点与分叉点这 2 种类型的细节点鉴定指纹, 通过将细节点表示为点模式的细节特征, 从而将指纹匹配问题转化为一个点模式匹配问题。

关于基于基准点定位的点模式匹配算法, 前人已经有了较多的研究和探讨, 文献[2]提出的基于中心点的匹配算法能够有效地解决指纹匹配速度较慢的缺陷, 但该算法中心点定位不稳定, 进而影响了识别率。文献[3]提出的基于 Poincare 奇异点检测的细节点局部结构匹配算法, 提高了执行效率, 但所得奇异点在数目和位置上偏差较大, 该算法对指纹类型有很强的依赖性, 匹配精度不高。文献[4]提出基于 Delaunay 三角化的指纹匹配方法, 能够准确地选取参考基准点, 识别率较高, 但算法复杂度较高, 执行效率较低。这些点模式匹配算法大致分为 2 种: (1)参考指纹的奇异点属性信息, 以奇异点为坐标原点, 在极坐标中匹配或者二次匹配; (2)不需要奇异点, 只依靠两细节点集之间的关系, 选取对应的细节点对, 以其为中心在极坐标中进行比对。分析所有基于基准点定位的点模式匹配算法, 发现算法的性能依赖于基准点定位的精度, 从而使基准点的定位成为了匹配算法中最难而又亟待解决的问题。

本文在提高基准点定位的精度方面做了研究, 提出一种改进的基准点求取方法。该方法借助指纹图像的中心点, 通过选取与中心点距离在 (R_{\min}, R_{\max}) 内的细节点构造局部结

构, 然后在此结构上利用全等三角形原则求取基准点和旋转因子, 再以所取得的基准点为极坐标原点, 将所有的细节特征通过坐标变换映射到极坐标中去, 最后利用可变界限盒的方法对 2 幅指纹进行匹配, 得到匹配点对数和匹配分数。对匹配点对数和匹配分数进行拟合, 从而得出最终的匹配结果。

2 基于基准点定位的匹配算法

指纹图像匹配主要解决平移、旋转和形变等问题, 而如果找到一对基准点对, 则只需要解决旋转和形变问题。文献[5]指出在极坐标系中能更好地描述非线性形变; 不需要考虑输入图像与模板图像的对应点之间的平移; 而且, 在极坐标系中很明显比在直角坐标系中更加便于处理 2 幅图像间的旋转。因此, 在指纹匹配的算法中, 一般要先找到一对或几对匹配的基准点, 然后将所有的细节特征点进行极坐标变换, 并旋转图像进行姿势校正, 最后进行比对, 计算出匹配点对数和匹配分数进而判断 2 幅图像是否匹配。

根据上述思想, 文献[6]提出了一种三角形基准点定位的匹配算法。

2.1 基准点定位及旋转因子求取

假定 2 个含有不同数量的点的集合 P 和 Q 分别为模板和输入图像细节点集。首先在模板点集 P 中任意取一点 p_i , 然后在输入图像的细节点集 Q 中寻找与其类型相同的点 q_j , 然后判断与这 2 个点距离最近的点 p_1 、 q_1 或次近点 p_2 、 q_2 的类型是否相同; 如果相同, 则继续判断以这 3 个点为顶点的三

作者简介: 李俊伟(1987—), 男, 硕士研究生, 主研方向: 图像处理, 模式识别; 周立俭, 副教授、博士; 崔学梅, 硕士研究生

收稿日期: 2010-12-29 E-mail: 031xljw@163.com

角形 $\Delta p_i p_1 p_2$ 与 $\Delta q_j q_1 q_2$ 是否近似全等以及对对应边的角度差是否近似相等, 即 $\Delta\theta_{i1}^p \approx \Delta\theta_{j1}^q$, $\Delta\theta_{i2}^p \approx \Delta\theta_{j2}^q$, $\Delta\theta_{i3}^p \approx \Delta\theta_{j3}^q$; 如果相等, 则认定 p_i 和 q_j 为基准点对, 计算旋转因子 $\Delta\theta = 1/3(\Delta\theta_{i1} + \Delta\theta_{i2} + \Delta\theta_{i3})$; 如果上述任何判断出错, 则变换 p_i 并重新寻找 q_j , 直到找到基准点对或查遍所有的细节点为止。

2.2 极坐标变换及姿势校正

对于 2 个特征点集 P 和 Q , 分别以它们对应的基准点 p_i 和 q_j 为极坐标原点, 进行坐标变换, 具体的变换公式如下:

$$\begin{pmatrix} r_i \\ \theta_i \end{pmatrix} = \begin{cases} \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \\ \arctan\left(\frac{y_i - y_c}{x_i - x_c}\right) \end{cases} \quad (1)$$

其中, (x_i, y_i) 为待转换的细节特征点直角坐标; (r_i, θ_i) 为转换后的极坐标; (x_c, y_c) 为基准点直角坐标。

对于输入图像 Q 的极角 θ_j , 应该以模板图像 P 为基准进行校正, 校正公式为:

$$\theta'_j = \theta_j - \Delta\theta \quad (2)$$

在点集校正后, 将 2 个点集中的特征点按照极角递增的顺序排列, 以利于下一步的匹配。

2.3 界限盒匹配及结果判决

对于非线性形变问题, 引入可变界限盒的概念, 如图 1 所示。对靠近基准点的细节特征点, 较小的形变会对极角造成较大的影响, 而对极半径却影响相对较小; 反之, 远离基准点的细节特征点发生形变时, 对极半径影响较大, 而对极角影响相对较小。于是半径为 r 的细节点处界限盒所取参数计算如下:

$$radius_size = \begin{cases} r_small & \text{if } r_size < r_small \\ r_size & \text{if } r_small < r_size < r_large \\ r_large & \text{if } r_size > r_large \end{cases} \quad (3)$$

其中, $r_size = \frac{r}{a}$ 。

$$angle_size = \begin{cases} a_small & \text{if } a_size < a_small \\ a_size & \text{if } a_small < a_size < a_large \\ a_large & \text{if } a_size > a_large \end{cases} \quad (4)$$

其中, $a_size = \frac{b}{r}$ 。

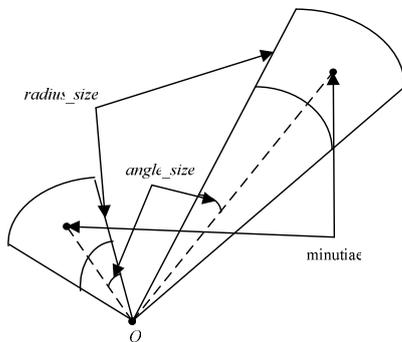


图1 可变界限盒示意图

文献[7]经过大量实验后指出, 如果 2 枚指纹有 12 个以上的匹配点对, 则可断定两者来自于同一个手指。但是对于质量较低的指纹图像, 所提取出来的细节点数目较少, 很可能出现不满足上述条件的匹配图像, 于是又引入了匹配分数概念:

$$match_score = \frac{2 \times mach_num}{p_num + q_num} \quad (5)$$

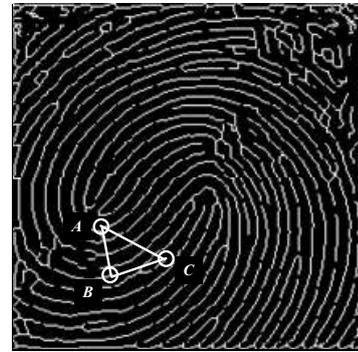
其中, $match_score$ 为输出的匹配分数; $match_num$ 为匹配点对数; p_num 为模板的细节点数目; q_num 为输入图像的细节点数目。于是判决条件可表示为:

$$(match_num > 12) \text{ OR } (match_score > T) \quad (6)$$

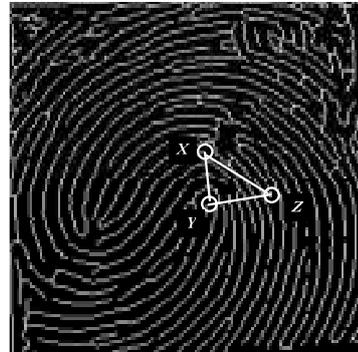
其中, T 为预先设置的阈值。只要式(6)的值为 1, 就认为 2 个指纹匹配成功; 否则, 匹配失败。

3 改进的三角基准点定位算法

由于采集到的指纹图像受到噪声等外界原因影响造成边缘部分提取的细节点中含有大量的伪细节点, 或者图像中模糊部分中含有大量的伪细节点, 造成了上述三角形定位算法的运行时间较长, 算法的执行效率较低; 另外, 指纹中含有伪细节点的特征点集可能形成符合上述算法但又不是对应基准点的全等三角形。如图 2 所示, 模板图像中 ΔABC 与输入图像中 ΔXYZ 近似全等, 而且对应边的角度差也近似相等, 但是 A 点与 X 点却不是对应的基准点, 这样就造成了误判, 从而增加了虚警概率, 降低了识别的精度。



(a)模板图像



(b)输入图像

图2 三角形全等的错误情况

鉴于上述算法的一些缺陷, 为了进一步提高识别精度、减少运算复杂度, 本文借助中心点的选取构造局部结构, 并在局部结构上进行基准点对的选取。

3.1 中心点的选取

中心点定义为图像的曲率场中最大值所对应的像素点, 也就是图像中纹线方向变化最大的点, 位于指纹纹路的渐进中心, 周围纹线大致呈半圆趋势。目前, 最常用的求取中心点的方法是 Poincare 索引法, 在通常情况下, 由于该方法求取的中心点不够准确, 因此不能将其作为基准点直接用于指纹匹配, 而是用作指纹分类的依据^[8]。然而, 细节点的相对位置较为固定, 适宜作为基准点。在本文中, 中心点只起参考作用, 因而中心点的位置不必太精确, 上述算法可以直接应用。具体的计算过程如下:

$$Poincare(i, j) = \frac{1}{2\pi} \sum \Delta k \quad (7)$$

其中:

$$\Delta k = \begin{cases} \delta(k) & \text{if } |\delta(k)| < \frac{\pi}{2} \\ \pi + \delta(k) & \text{if } \delta(k) \leq -\frac{\pi}{2} \\ \pi - \delta(k) & \text{if } \delta(k) \geq \frac{\pi}{2} \end{cases}$$

$$\delta(k) = O(\Psi_x(i'), \Psi_y(i')) - O(\Psi_x(i), \Psi_y(i))$$

$$i' = (i+1) \bmod N$$

3.2 局部结构的构造

以所得得的中心点为圆心, 取与圆心距离为 $R_{\min} \sim R_{\max}$ 之间的细节点组成局部特征点集, 即如图 3 所示的环形区域。以这个局部结构为对象, 求取三角基准点, 这样一方面简化了基准点求取的计算, 提高了程序的执行效率; 另一方面, 也在一定程度减小了上文所述的三角形全等但不是基准点对情况的概率, 在一定意义上提高了算法的识别率。

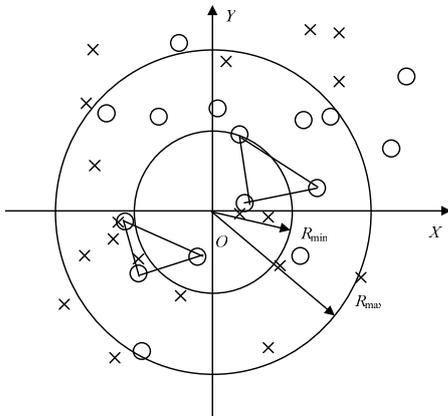


图3 局部区域的构造示意图

3.3 基准点定位

在局部结构上求取基准点的具体过程如下:

(1) 假定 2 个含有不同数量的集合 $P\{p_1, p_2, \dots, p_m\}$ 和 $Q\{q_1, q_2, \dots, q_n\}$ 为分别由模板和输入图像所产生的局部细节点集。任意取 2 个局部结构中的点 p_i 和 q_j , 判断它们的类型是否相同, 如果类型相同, 则 p_i 和 q_j 可能为基准点对, 执行步骤(2)。

(2) 判断距离 p_i 和 q_j 最近点 p_1, q_1 和次近点 p_2, q_2 的类型是否相同, 如果类型也相同, 则 p_i 和 q_j 可能为基准点对, 执行步骤(3), 否则返回步骤(1)。

(3) 判断以上述 3 点为顶点的三角形的边长是否大致相等, 即 $|p_i p_1| \approx |q_j q_1|, |p_i p_2| \approx |q_j q_2|, |p_1 p_2| \approx |q_1 q_2|$, 如果符合条件, 则执行步骤(4), 否则返回步骤(1)。

(4) 判断以上述 3 点为顶点的三角形对应边的角度差是否大致相等, 即 $\Delta\theta_{i1}^p \approx \Delta\theta_{j1}^q, \Delta\theta_{i2}^p \approx \Delta\theta_{j2}^q, \Delta\theta_{12}^p \approx \Delta\theta_{12}^q$, 如果符合条件, 则认定 p_i 和 q_j 为基准点对, 计算旋转因子 $\Delta\theta = 1/3(\Delta\theta_{i1} + \Delta\theta_{i2} + \Delta\theta_{12})$, 否则返回步骤(1)。

4 实验结果与分析

本文根据改进基准点定位算法, 结合 2.2 节、2.3 节所述的极坐标变换和可变量限盒匹配方法, 在双核 2.93 GHz、2 GB 内存的计算机上对国际指纹识别竞赛公布的数据库 FVC2004 DB3 进行了匹配实验, 并与文献[6]算法作了比较。该数据库共有来自 100 个手指的 800 幅指纹图像, 各算法消耗的时间均值如表 1 所示, 其 ROC 曲线如图 4 所示。

表 1 各算法消耗的时间均值 ms

算法	基准点定位	校正匹配	总消耗时间
本文算法	189.2	3.3	192.5
文献[6]算法	231.1	3.1	234.2

从表 1 可以看出, 本文算法在基准点定位的时间上比文献[6]算法少, 从而节省了总的匹配时间, 这是由于本文算法是在局部结构中寻找基准点, 而不是所有的细节特征点集。

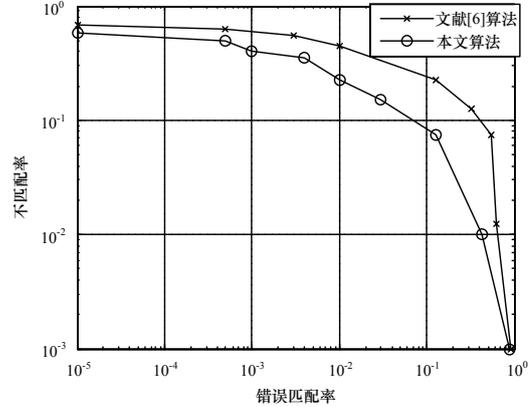


图 4 各算法的 ROC 曲线

评价指纹匹配算法的性能的主要指标有 ROC 曲线线下面积和等错率 EER。从图 4 中可以看出, 本文算法比文献[6]算法的 ROC 曲线更接近原点, 即线下面积更小, 而且当错误匹配率等于不匹配率时所得的 EER 更低, 说明本文算法所能达到的识别率较高, 这得益于局部特征点集的选取在某些情况下破坏了错误的全等三角形结构。

5 结束语

本文总结和分析了前人在指纹点模式匹配方法的研究成果基础上, 提出了一种在局部结构上借助中心点进行基准点定位的改进算法, 并以此基准点为基础进行了极坐标下的可变量限盒的匹配实验。实验结果表明, 该算法不但节省了匹配时间, 提高了算法的执行效率, 而且在某些情况下提高了算法的识别精度。但是对于一些质量较低的指纹, 由于所求得的方向图并不可靠, 因此影响了中心点的求取以及局部结构的构造, 在此情况下的匹配算法有待进一步研究。

参考文献

- [1] 廖 阔, 杨万麟. 点模式指纹匹配算法研究与实现[J]. 电子科技大学学报, 2004, 33(2): 154-156.
- [2] 谭台哲, 宁新宝, 尹义龙, 等. 一种基于指纹中心点的匹配算法[J]. 南京大学学报: 自然科学版, 2003, 39(4): 483-489.
- [3] 袁华强, 叶阳东, 邓见光, 等. 一种基于基准点的指纹匹配算法[J]. 计算机工程, 2008, 34(3): 241-243.
- [4] 尹义龙, 张宏伟, 刘 宁. 基于 Delaunay 三角化的指纹匹配方法[J]. 计算机研究与发展, 2005, 42(9): 1622-1627.
- [5] Jain A K, Hong Lin, Pankanti S, et al. An Identity Fingerprint Recognition[Z]. California Institute of Technology, 1997.
- [6] 王 磊. 自动指纹识别系统的算法研究及实现[D]. 哈尔滨: 哈尔滨工程大学, 2008.
- [7] Hrechak A K, Mchugh J A. Automated Fingerprint Recognition Using Structural Matching[J]. Pattern Recognition, 1990, 23(8): 893-904.
- [8] 项 明, 吴小培, 刘明生. 基于独立分量分析的二级指纹分类算法[J]. 计算机工程, 2010, 36(10): 16-18.