

# 基于局部细节特征的二次指纹匹配算法的研究

贾珈, 蔡莲红

(普适计算教育部重点实验室(清华大学) 清华大学计算机系 北京 100084)

**摘要:** 为提高指纹匹配的正确率, 综合局部细节匹配算法和全局匹配算法, 提出一种将两者相融合的二次匹配方法。在提取指纹细节特征信息并去除伪特征点后, 首先利用 k-近邻法进行局部细节特征的一次匹配, 获得局部特征之间的匹配分数; 然后根据匹配分数对指纹图像进行旋转校正, 进而对全局特征进行二次匹配, 计算匹配向量, 并利用匹配向量获得匹配率决定最终匹配结果。实验结果表明: 在不同质量的多个指纹数据库上测试, 算法最高正确率达到 FRR=2.5%, FAR=0.22%, 这说明了该方法的有效性。

**关键词:** 指纹识别; 二次匹配; k-近邻法;

**中图分类号:** TP3

手指皮肤皱褶的纹理形成了人的指纹。人类的指纹具有个体间的唯一性和个体发育过程中的不变性, 因此成为个人身份识别最可靠的方法之一。目前, 提高自动指纹识别系统 (AFIS) 的正确性和效率, 已成为指纹识别领域的研究热点。

指纹的唯一性是由整体特征 (纹型、三角点、核心点) 和细节特征 (分叉点和端点) 决定的。细节特征通常用于指纹匹配<sup>[2, 3]</sup>, 但是依靠细节特征的匹配结果往往受限于指纹图像的质量。文[1]尝试将全局检测和局部细节匹配进行融合, 首先在全局检测指纹的核心点, 再在核心点附近寻找一定数量的细节特征点进行局部匹配。这种思路可以大大降低计算量, 提高识别的速度, 但另一方面对于特征提取提出了较高的要求, 需要对核心点进行精确定位, 同时要求核心区域的图像质量不能太差<sup>[4]</sup>。

受采集仪和采集过程的影响, 原始指纹图像往往具有很强的背景噪音或者图像中不存在核心点。鉴于这些困难, 本文综合了局部细节匹配算法和全局匹配算法, 提出了一种二次匹配的方法。首先对局部的细节特征进行一次匹配, 得到局部细节特征点的匹配分数; 然后基于匹配分数对待识别指纹与模板指纹进行旋转校正, 进而进行二次全局匹配。通过全局匹配, 得到一组匹配向量, 并利用匹配向

量获得匹配率决定最终的匹配结果。

## 1 特征提取

### 1.1 细节特征点提取

指纹细节特征点包括指纹的分叉点和端点 (图 1)。



图 1 指纹特征点: 端点和分叉点

1) 特征点坐标。对于图像处理后的指纹细化图像<sup>[5, 6]</sup>, 采用脊线跟踪的方法标定特征点的坐标。

**定义 1** (8 邻域像素和) 令  $N_9(x, y)$  为包括点  $(x, y)$  与其 8 邻域的  $3 \times 3$  点集, 定义  $sum(x, y)$  为计算  $N_9(x, y)$  上所有像素值的和。

则对于脊线上任意点  $(i, j)$ , 有

$$sum(i, j) = \begin{cases} 4, & f(i, j)=1; \\ 2, & d(i, j)=1; \end{cases} \quad (1)$$

公式(1)中,  $f(i, j) = 1$  表示该点为分叉点,  $d(i, j) = 1$  表示该点为端点。 $sum(i, j)$  取其它值时不予考虑。

2) 特征点方向。将预处理后的指纹细化图分成  $w \times w$  的小块。采用 Sobel 算子计算特征点在 x 和 y 方向上的梯度  $G_x$  和  $G_y$ 。则特征点的方向  $\theta$  为:

$$V_x(i, j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2G_x(u, v)G_y(u, v) \quad (2)$$

$$V_y(i, j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} (G_x^2(u, v) - G_y^2(u, v)) \quad (3)$$

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left( \frac{V_y(i, j)}{V_x(i, j)} \right) \quad (4)$$

特征提取完成后得到的特征点集  $M$ ，其中每一个特征点对应唯一的坐标  $(i, j)$  和方向  $\theta$ 。

## 1.2 特征点去伪

由于受图像处理 and 细化算法的精确性限制，提取的特征点中包含了大量的伪特征点（图 2，图 3，图 4）。本文采用跟踪脊线的方法去除伪特征点，即从特征点  $(i, j)$  开始顺着脊线寻找最远非特征点  $(i', j')$ ，通过这两个坐标可以计算这条脊线的切线方向  $ang(i, j)$ ，对于分叉点还需要记录三个分支上是否遇到特征点以及遇到的特征点的坐标。

- 1) 修复断的脊线：首先从一对断点开始跟踪脊线得到脊线方向，如果  $ang$  角之间的差约为  $\pi$ ，则初步确定属于同一脊线，然后计算两点的连线方向，与脊线方向相比较，取得差值最小的两个点相配对，连接断点得到修复后的脊线（图 2）。从特征点集  $M$  中删去端点对。
- 2) 去除小桥：首先从分叉点开始跟踪三条脊线，如果一个分支上跟踪到分叉点且与另外两个分支的夹角约为  $\pi/2$  则认为小桥，去掉这个分支（图 3a）。从特征点集  $M$  中删去分叉点对。
- 3) 去除小孔：从分叉点开始跟踪三条脊线，在阈值内如果两个以上分支上有分叉点且有两个分叉点相同，则判断为小孔，去掉一条脊线（图 3 b）。从特征点集  $M$  中删去分叉点对。
- 4) 从特征点集  $M$  中删去距离小于阈值的点对。可以分为去除两个相近的终结点，相近的终结点和分叉点以及相近的分叉点。其中相近的终结点用来去除孤立的短线，相近的终结点和分叉点用来去除脊线上的毛刺（图 4），相近的分叉点是为了适应算法的需要，因为太近的两个分叉点对算法的误差过大，不宜采用。

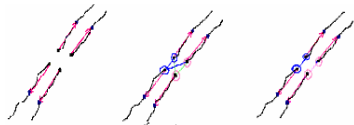


图 2 修复脊线

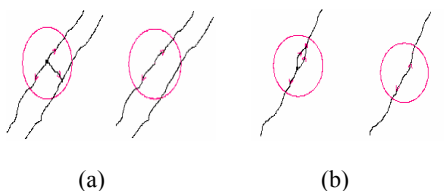


图 3 去除小桥与去除小孔



图 4 去除孤立短线与去除毛刺

去伪之后，特征点集  $M$  中去除了绝大部分的伪特征点，保留了真实的特征点，从而提高了后续匹配算法的准确性和效率。

## 2 匹配模型

本文匹配算法采用二次匹配来提高算法精度。为了减少冗余计算，同时防止离指纹图像中心较远的特征点精度低影响匹配效果，算法只取图像中心区域的 20~30 个细节特征点进行匹配。匹配分为两个阶段进行，第一阶段进行局部细节特征的一次匹配，采用的匹配方法是  $k$ -近邻法，获得局部特征之间的匹配分数；第二阶段进行全局特征的二次匹配，进行匹配之前需要先计算图像旋转角度，进行待匹配图像和模板图像之间的校正，采用的方法是取一次匹配分数最高的 3~4 个点进行加权平均，然后计算一个匹配向量，并利用这个向量获得匹配率决定是否匹配。

### 2.1 一次匹配模型

一次匹配是指待匹配指纹与模板指纹之间的局部细节特征匹配，采用的方法是  $k$ -近邻法。

在中心区域选取 20~30 个细节特征点，作为有效的特征点。每一个特征点的周围确定了一个局部结构，以该特征点作为局部结构的中心点，选取中心点周围的  $k$  个最近的邻点进行匹配，为了减少弹性形变及噪声的影响，采用对应边的相对量  $D$ ， $\alpha$ ， $\beta$ ， $\gamma$  进行匹配（图 5），从而获得局部特征之间的匹配分数，用于第二阶段的匹配。

如图所示，白点代表局部结构的中心点，黑点代表它的一个邻域里的其他特征点，每个中心点及邻近的每个特征点对应一组相对量  $D$ ， $\alpha$ ， $\beta$ ，分别代表两点间的距离，中心点与邻近点脊线切线方向的夹角，中心点与邻近点连线方向与邻近点脊线切线方向的夹角。对于两幅待匹配的指纹图像，若在该局部结构里有一对边的  $D$ ， $\alpha$ ， $\beta$  都比较吻合的话，则进一步求出  $\gamma$ ， $\gamma$  为中心点引出的两条匹配边之间的夹角，若参数  $\gamma$  的数值也比较吻合，就增加相应的匹配分数。

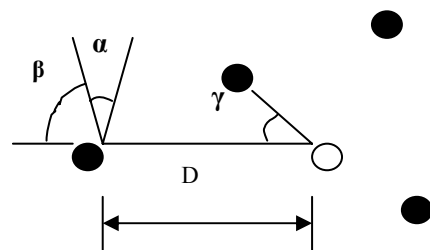


图 5 局部结构特征相对量

## 2.2 局部匹配分数的计算模型

当判定待识别指纹与模板指纹之间的一对局部结构匹配时,需要根据这对局部结构的自身特征计算局部匹配分数。考虑到分叉点的特征比终结点的特征要重一些,算法对每一对边对进行计分,符合阈值约束的边对分数加 2,若计分的点对为分叉点,则相应分数翻倍。按照此方案得到的累计分数即局部结构的匹配分数。

## 2.3 二次匹配模型

二次匹配也就是进行全局特征的匹配,进行匹配之前需要先进行坐标的转换,采用的方法是对匹配分数最高的 3~4 个点进行加权平均,计算出图像的偏转角度。另外选取匹配分数最高的一对点作为基准点,然后计算其余各点的相对特征向量。计算公式如下:

$$\begin{bmatrix} \gamma_i \\ \theta_i \\ o_i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_i - x^r)^2 + (y_i - y^r)^2} \\ \arctan \frac{y_i - y^r}{x_i - x^r} \\ \theta_i - \theta^r \end{bmatrix} \quad (5)$$

在待识别指纹点集与模板指纹点集之间两两计算特征向量距离  $d_{ij}$  为:

$$d_{ij} = |\theta_i - \theta_j| + |\gamma_i - \gamma_j| + |o_i - o_j| \quad (6)$$

选取最佳匹配方案,即匹配点对之间的距离最小。可采用如下方法,构造矩阵记录每组点对的距离。从矩阵中选取距离最小的点对后将矩阵该行该列其余元素置 0,重复操作直至无匹配点对。这样的选取策略可以排除多对多、多对一的匹配。

记选取出来的匹配点对有  $K$  对,则计算匹配率  $R$  为:

$$R = \sqrt{k \times k / m \times n} \quad (7)$$

其中  $m, n$  为待识别指纹点集与模板指纹点集的点数。当匹配率大于设定的阈值时判断匹配成功,否则匹配失败。

## 3 实验指纹库

本文所采用的实验指纹数据包括 3 个指纹数据集(DB1-DB3)。其中, DB1 和 DB2 选自 Fingerprint Verification Competition (FVC) 竞赛提供的具有不同图像质量的两个指纹库; DB3 为采用 veridicom 公司生产的 FPS200 电容式指纹采集仪采集的指纹

库。数据集的具体信息如表 1 所示。

表 1 指纹数据库信息

	数据来源	图像尺寸①	图像数量②
DB1	FVC2002 DB1	388×374	10×8=80
DB2	FVC2000 DB3	448×478	10×8=80
DB3	FPS200 采集仪	256×300	108×3=324

注: DB3 的采集对象为 36 名年龄在 20—22 岁间的男、女学生,每人采集 3 枚不同手指的指纹,每枚指纹采集 3 次。

①指纹图像尺寸以像素点数目为计量单位。

②图像数量/幅=手指个体×每个个体采集的图像数量。

所有数据库中的指纹图像分辨率均为 500 dpi。指纹图像允许  $[-45^\circ, 45^\circ]$  之间(相对于垂直方向)的旋转角度;相同手指的指纹图像有效重叠区域,但不保证指纹图像上有核心点和三角区。

## 4 匹配实验及结果分析

实验中,采用文<sup>[5]</sup>的算法对原始指纹图像进行去噪、增强及二值化处理;采用文<sup>[6]</sup>的算法对二值化图像进行细化处理。

匹配实验采用交叉检验技术。DB1 和 DB2 按不同手指平均分为 5 份,每次将其中 1 份数据作为测试集,其余 4 份作为阈值训练集。匹配实验按不同的测试集共进行 5 次,取 5 次实验的平均值作为识别结果;DB3 按不同手指分为 6 份,每次将其中 1 份数据作为测试集,其余 5 份作为阈值训练集。匹配实验按不同的测试集共进行 6 次,取 6 次实验的平均值作为识别结果。采用交叉检验技术能够降低随机因素的影响,提高匹配结果的可信度。

### 4.1 特征点去伪算法结果分析

图 6 给出本文特征点去伪算法的可视化结果。以 DB1 为例,采用去伪算法前,数据库中指纹图像平均特征点数约为 130 个,去除伪特征点之后的个数约为 20—40 个,且绝大部分为真实特征点,从而有效提高了后续匹配算法的性能。



图 6 特征点坐标、方向信息提取与去伪

从图 6 的比较可以看出：左边图像提取的部分特征点为小桥、小孔等结构造成的伪特征，右边（蓝色线表示特征点方向）采用去伪算法后，保留了真实的分叉点和端点信息，去除了小桥、小孔以及断裂脊线造成的伪特征点，切线方向的提取也基本准确。

## 4.2 匹配算法结果及分析

为了全面衡量算法的性能，匹配实验采用错误接受(False Acceptance Rate - FAR)和错误拒绝率(False Rejection Rate - FRR)作为测试指标，以下是测试结果：

表 2 指纹匹配错误接受率 FAR 与错误拒绝率 FRR %

	FAR①	FRR②
DB1	0.25	3.42
DB2	2.09	4.25
DB3	0.22	2.50

注：①指纹库中每个手指的第一个样本与库中其它手指的第一个样本相匹配计算为 FAR。

②指纹库中每个样本与相同手指的其余样本未能匹配上的计算为 FRR。

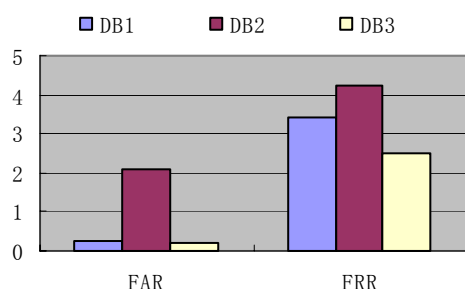


图7 DB1、DB2、DB3上的测试结果

从表 2 和图 7 的结果可以看出，对于不同的指纹数据库，单项评测指标均小于 5%，最高的平均正确率达到了 FRR=2.5%，FAR=0.22%。DB1、DB2 的匹配结果与 FVC2002<sup>1</sup>及 FVC2000<sup>2</sup>公布的正式竞赛结果相比，具有较好的匹配效果。

对于 DB3 获得最高正确率的情况，一方面由于采集个体为年龄介于 20 至 22 岁的学生，指纹发育成熟，无职业造成的指纹磨损或斑记，因此指纹质量规则良好；另一方面也是因为 veridicom FPS200 采集仪采集的图像背景噪音小，因此得到了很好的匹配结果。

DB3 的图像数量较之 DB1 和 DB2 明显增多，但是仍能得到最高的匹配正确率，也说明匹配算法对图像数量具有很好的鲁棒性。

时间性能方面，本文分别在两种配置的计算机

上进行测试，测试结果如下：

表 3 时间性能测试结果

计算机配置			计算次数	总用时	时间效率
CPU/Mhz	MEM/MB	OS	/次	/s	/s/次
1793	256	Windows 2000	4186	1702.468	0.407
1400	256	Windows xp	4186	591.991	0.141

这个时间效率基本上是可以达到实时的要求的。

## 5 结论与展望

本文提出了一种基于局部细节特征的二次指纹匹配算法。为了提供匹配算法需要的特征点坐标及方向信息，同时尽可能的减少伪特征点的干扰，首先进行了特征的提取和去除伪特征的相关工作。在后续匹配算法中，我们采用二次匹配的方法。研究表明，算法在初匹配之后就可以去掉大部分不能匹配的情况，减少了后续计算，提高了时间性能；此外，采用相对量作为匹配依据，可以在不确定旋转和平移参数的情况下就作出局部匹配，优化了性能。实验结果显示，竞赛指纹库最高的平均正确率达到了 FRR=2.5%，FAR=0.22%。这说明了本文所提出匹配方法的有效性。

## 参考文献 (References)

- [1] Zhang W, Wang Y. Core-based structure matching algorithm of fingerprint verification [A]. *Proc of ICPR'02* [C], Aug. 2002, Volume: 1: 11-15.
- [2] Luo X, Tian J, Wu Y. A minutia matching algorithm in fingerprint verification, *The Chinese National Science Foundation*, IEEE, 2000.
- [3] Wang, C, Li J. Fingerprint identification using point pattern matching. *Journal of Chongqing University (Natural Science Edition)*, 2002, Vol. 25, No.6.
- [4] 冯俐俐、李昌禧，指纹中心点的定位和特征匹配方法[J]. 华中科技大学学报(自然科学版), 2002, Vol.30, No.10.
- [5] Lin, H, Yifei W, Jain, A. Fingerprint image enhancement: algorithm and performance evaluation [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, Vol. 20, No.8.
- [6] Ratha, N.K, Chen, S, Jain, Adaptive flow orientation-based feature extraction in fingerprint images [J]. *Pattern Recognition*, 1995, Vol.28, No.1

<sup>1</sup> [http://bias.csr.unibo.it/fvc2002/results/res\\_db1\\_a.asp](http://bias.csr.unibo.it/fvc2002/results/res_db1_a.asp)

<sup>2</sup> [http://bias.csr.unibo.it/fvc2000/results/res\\_db3\\_a.asp](http://bias.csr.unibo.it/fvc2000/results/res_db3_a.asp)

# **A Fingerprint Verification Approach based on Minutiae Re-matching Method**

JIA Jia, CAI Lianhong

(Key Laboratory of Pervasive Computing (Tsinghua University),  
Ministry of Education, Beijing 100084, China)

**Abstract:** To improve the accuracy of fingerprint verification, a minutiae matching approach based on re-matching method is proposed in this paper. In the proposed approach, after basic minutiae extraction and purification task, K-neighbors algorithm was used to get the local matching scores through the first-local-matching step. Then the local matching scores were applied to adjust two images and do a second-global-matching step. The final decision was made from the matching rate of the matching vectors in the second matching step. The approach is supported by experiments on three test collections, including both international fingerprint verification competition databases and the database collected by on-line FPS200 capture system. Experimental results show that our approach is insensitive to noise as well as with effective performance.

**Keywords:** fingerprint verification; re-matching method; K-neighbors algorithm;