



(12)发明专利申请

(10)申请公布号 CN 107958208 A

(43)申请公布日 2018.04.24

(21)申请号 201711131680.0

(22)申请日 2017.11.15

(71)申请人 西安电子科技大学

地址 710071 陕西省西安市太白南路2号西安电子科技大学

(72)发明人 赵恒 秦帅 陈炯 庞辽军

(74)专利代理机构 西安长和专利代理有限公司 61227

代理人 黄伟洪 何畏

(51)Int.Cl.

G06K 9/00(2006.01)

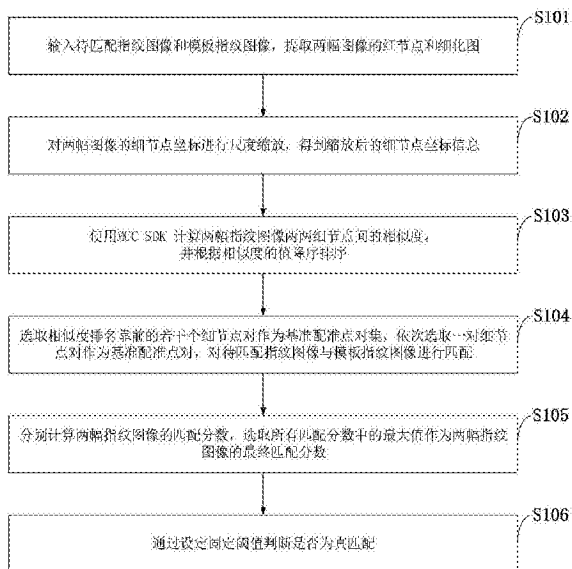
权利要求书3页 说明书8页 附图3页

(54)发明名称

一种基于传播算法的指纹交叉库匹配方法

(57)摘要

本发明属于数字图像处理技术领域,公开了一种基于传播算法的指纹交叉库匹配方法,所述基于传播算法的指纹交叉库匹配方法选择相似程度较高的基准细节点对构建细节点局部结构算子;通过传播算法完成局部区域内细节点间的精确匹配;使用新的基准点对继续进行局部区域内的精确匹配;匹配范围从局部区域逐步传播到整幅图像,进而完成全局层面的匹配工作。本发明实现了较低的EER值,指纹交叉匹配结果更为准确。



1. 一种基于传播算法的指纹交叉库匹配方法,其特征在于,所述基于传播算法的指纹交叉库匹配方法选择相似度较高的基准细节点对构建细节点局部结构算子,完成细节点间的精确匹配;通过传播算法找到满足基准点条件的细节点对,使用新的基准点对继续进行局部区域内的精确匹配;匹配范围从局部区域逐步传播到整幅图像,完成全局层面的匹配。

2. 如权利要求1所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述基于传播算法的指纹交叉库匹配方法包括以下步骤:

(1) 输入待匹配指纹图像 $Q(x, y)$ 和模板指纹图像 $T(x', y')$,使用Verifinger SDK对这两幅图像进行处理,获得两幅图像的细节点集合 $m_Q = (x, y, \theta)_i (i = 1, \dots, N_1)$ 和 $m_T = (x', y', \theta')_i (i = 1, \dots, N_2)$,以及指纹细化图像 $skeleton_Q$ 和 $skeleton_T$,其中 (x, y) 表示待匹配指纹图像 $Q(x, y)$ 的像素点坐标, (x', y') 表示模板指纹图像 $T(x', y')$ 的像素点坐标, θ 与 θ' 表示细节点角度, N_1 代表待匹配指纹图像中细节点总个数, N_2 代表模板指纹图像中细节点总个数,Verifinger SDK是基于PC平台的单机指纹识别算法软件,集成了指纹图像预处理、指纹图像特征提取、指纹图像匹配;

(2) 对两幅指纹图像的细节点坐标进行尺度为 $scale = DPI_T / DPI_Q$ 的缩放,其中 DPI_Q 为采集待匹配指纹图像 $Q(x, y)$ 所使用指纹传感器的分辨率, DPI_T 为采集模板指纹图像 $T(x', y')$ 所使用指纹传感器的分辨率,缩放前待匹配指纹图像中细节点的坐标为 (x_0, y_0) ,缩放后的细节点坐标为 (x, y) ,两者之间的关系如下:

$$(x, y) = (x_0, y_0) \frac{DPI_T}{DPI_Q};$$

(3) 使用MCC SDK计算待匹配指纹细节点集合 $m_Q = (x, y, \theta)_i (i = 1, \dots, N_1)$ 和模板指纹细节点集合 $m_T = (x', y', \theta')_i (i = 1, \dots, N_2)$ 间的MCC相似度,并将细节点MCC相似度降序排序,获得细节点MCC相似度分数集合 $\{S(i, j)\}_{i=1}^{N_1} \}_{j=1}^{N_2}$, $S(i, j)$ 代表待匹配指纹的第 i 个细节点与模板指纹的第 j 个细节点间的匹配分数;

(4) 选取MCC相似度排名前25的细节点对作为基准配准点对集 $M_i (i = 1, \dots, 25)$,利用传播算法根据基准配准点对计算两幅指纹图像的匹配分数;

(5) 将依据25对初始配准细节点对计算的25个匹配分数进行排序,选取最高的值作为两幅指纹图像的最终匹配分数;

(6) 设定固定阈值 TH ,将匹配分数与所设定固定阈值相比,如果匹配分数大于阈值则认为指纹为真匹配,否则为假匹配。

3. 如权利要求2所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述(4)具体包括:

1) 根据MCC相似度从大到小依次选取 $M_i (i = 1, \dots, 25)$ 中的一对细节点对作为基准配准点对;

2) 分别计算两幅图像中其它细节点相对于基准配准点的极坐标 $Q(r, t)$ 和 $T(r', t')$,其中 r, t 和 r' 和 t' 按照以下公式计算如下:

$$\begin{cases} r^2 = x^2 + y^2 \\ \tan t = \frac{y}{x} \end{cases};$$

$$\begin{cases} r'^2 = x'^2 + y'^2 \\ \tan t' = \frac{y'}{x'} \end{cases};$$

r 和 t 分别对应待匹配指纹中其他细节点相对于基准配准点的距离和角度, r' 和 t' 分别对应模板指纹中其他细节点相对于基准配准点的距离和角度;

3) 在待匹配指纹图像和模板指纹图像中,以基准配准点为中心构造细节点局部结构算子;

4) 将位于待匹配指纹细节点局部结构算子内的细节点统一为细节点集 m ,依次遍历所有位于模板指纹细节点局部结构算子内的细节点,寻找能与 m 中细节点匹配的细节点组成集合 m' ,并判断 m 与 m' 中是否存在种子点对 S_d ;

5) 如果存在种子点对 S_d ,则开启传播,以种子点对 S_d 作为新的基准配准点对重复3)与4),没有满足条件的种子点对时,终止传播,即匹配完毕;

6) 所有匹配细节点对的MCC相似度的和为两幅指纹图像之间的匹配分数,计算公式如下:

$$S = \sum_{i=1, j=1}^{num} S_{(i,j)MCC};$$

其中 num 是确定为匹配的细节点对数, $S_{(i,j)MCC}$ 是待匹配指纹中细节点 i 和模板指纹中的细节点 j 之间的MCC相似度分数。

4. 如权利要求3所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述构造细节点局部结构算子包括:

(1) 将以初始配准细节点为圆心,一定半径长度的圆分成 $n = R \times K$ 个区域,其中 R 代表圈数, K 是按照360度角将其分割的数目,取 $R = 5, K = 8, R_{max}$ 是最外圈的半径,为150像素;

(2) 设定待匹配指纹细节点局部结构算子的五个半径的数值分别是:30、60、90、120、150像素,角度区域的范围为: $0^\circ \sim 45^\circ, 45^\circ \sim 90^\circ, 90^\circ \sim 135^\circ, 135^\circ \sim 180^\circ, 180^\circ \sim 225^\circ, 225^\circ \sim 270^\circ, 270^\circ \sim 315^\circ, 315^\circ \sim 360^\circ$;

(3) 五个半径的数值范围是:0~38像素,18~72像素,44~106像素,70~140像素,96~174像素,模板指纹的局部细节点结构算子角度区域的分割数为8,角度区域的范围在原始八个区域的基础上,每个区域的左右边界各扩大 30° ;

(4) 以初始配准细节点方向为基准方向,从内而外的按逆时针方向对细节点结构算子内的扇形区域进行编号。

5. 如权利要求3所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述判断是否存在种子点对 S_d 的流程包括:

1) 比较待匹配的细节点对是否处于同一个扇形区域中,如果处于同一个扇形区域,进行下一步,不满足则直接跳到下一个细节点对;

2) 当判断在同一个局部细节点结构算子中的待匹配指纹与模板指纹中的两个细节点属于同一个扇形区域后,求极坐标的距离差值与角度差值,当距离与角度的差值都小于设定的阈值且二者的MCC相似度高于所设的阈值时,认定这两个细节点匹配;

3) 判断细节点匹配时的两种情况:

是否满足基准配准点条件,满足则可作为下一个传播的种子;

是否满足普通匹配的细节点对的条件,满足基准配准点要求的细节点对必然也是匹配的细节点对。

6. 如权利要求3所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述判定细节点匹配的两种情况包括:

1) 判断待匹配指纹图像与模板指纹图像中的细节点对是否是下一步的基准配准点对,细节点对需满足下式四个条件:

$$\begin{cases} \Delta r < Th_{r1} \\ \Delta t < Th_{t1} \\ \Delta \theta < Th_{\theta 1} \\ MCC > Th_{MCC1} \end{cases};$$

其中 $\Delta r = |r - r'|$, $\Delta t = \min\{|t - t'|, 360 - |t - t'|\}$, $\Delta \theta = \min\{|\theta - \theta'|, 360 - |\theta - \theta'|\}$, MCC表示两个对应细节点的MCC相似度, Th_{r1} 、 Th_{t1} 、 $Th_{\theta 1}$ 和 Th_{MCC1} 为距离、相对角度、细节点角度和相似度等处理的预设阈值,当满足这四个条件的时候,一对细节点被认定为满足基准配准点的条件;

2) 如果这一对细节点不满足基准配准点的条件,继续判断这一对细节点是否满足普通配准的条件,条件也是四个:

$$\begin{cases} \Delta r < Th_{r2} \\ \Delta t < Th_{t2} \\ \Delta \theta < Th_{\theta 2} \\ MCC > Th_{MCC2} \end{cases};$$

如果这一对细节点满足以上条件,则认为是匹配的。

7. 如权利要求3所述的基于传播算法的指纹交叉库匹配方法,其特征在于,所述阈值设定, Th_{r1} 设置为 $3.8 \times R$, Th_{t1} 为 $\pi/22 + R \times \pi/22$, $Th_{\theta 1}$ 为 $\pi/14 + R \times \pi/14$, Th_{MCC1} 为0.57, Th_{r2} 为 $4.5 \times R$, Th_{t2} 为 $\pi/16 + R \times \pi/22$, $Th_{\theta 2}$ 为 $\pi/12 + R \times \pi/12$, Th_{MCC2} 为0.5,其中R表示细节点所在的环数。

一种基于传播算法的指纹交叉库匹配方法

技术领域

[0001] 本发明属于数字图像处理技术领域,尤其涉及一种基于传播算法的指纹交叉库匹配方法。

背景技术

[0002] 当今社会已经进入信息时代,传统的以密码为特征的识别技术,如银行卡、PIN码等越来越难以满足当下的信息安全需求。生物特征识别作为一种颇具应用价值和前景的技术手段能够提高身份认证的安全性,并能以更加便捷的方式满足日常生活中的加解密需求。在对稳定性、识别精度、用户接受度、便捷程度以及低硬件成本等方面综合考量下,指纹识别与其它生物特征识别相比具有天然优势。固有的诸多优点也使得指纹识别技术被广泛应用在社会生活的各个领域,如电子政务、电子商务、考勤管理以及移动终端应用等等。在自动指纹识别系统中,一般把传统的来自于同一采集仪的指纹间匹配称为“常规匹配”,将来自于不同采集仪的指纹间匹配称为“交叉匹配”。由于采集指纹时手指放置角度和按压力度的不同,同一个指纹在同一个采集仪下前后采集的结果会发生一定角度和距离的旋转平移以及指纹脊线结构的形变。由于两幅图像来自于同一采集仪,旋转平移和形变量相对较小,传统的优秀指纹匹配方法可以对其进行修正并取得较高的匹配精度。但目前市面上的指纹采集仪按照成像原理可分为电容采集仪、光学采集仪和超声波采集仪等,按采集模式可分为按压采集仪(滚动按压、平面按压、和整手按压)、刮擦采集仪和非接触采集仪等。指纹采集仪采集原理不同,采集的标准和规格不同,导致来自不同指纹采集仪的指纹图像间存在较大的非线性形变。传统的指纹匹配方法难以修正图像间的非线性形变,导致两张匹配图像之间误识率增大,极大的影响了识别精度。在交叉库匹配中,目前比较主流的非线性形变的解决方法是设计非线性形变模型,修正各个采集仪采集指纹时各种非线性形变,统一划为一个规格的指纹模型。再用传统的指纹预处理及匹配方法进行匹配。然而,这类方法相当于实现了指纹匹配工作之前的指纹配准工作,依然没有针对较大非线性形变进行处理。

[0003] 综上所述,现有技术存在的问题是:现有的指纹匹配方法匹配不同采集仪间指纹时稳定性与鲁棒性较差,识别精度低。

发明内容

[0004] 针对现有技术存在的问题,本发明提供了一种基于传播算法的指纹交叉库匹配方法,可应用于自动指纹识别系统。

[0005] 本发明是这样实现的,一种基于传播算法的指纹交叉库匹配方法,所述基于传播算法的指纹交叉库匹配方法选择相似度较高的基准细节点对构建细节点局部结构算子,完成细节点间的精确匹配;通过传播算法找到满足基准点条件的细节点对,使用新的基准点对继续进行局部区域内的精确匹配;匹配范围从局部区域逐步传播到整幅图像,完成全局层面的匹配。

[0006] 进一步,所述基于传播算法的指纹交叉库匹配方法包括以下步骤:

[0007] (1) 输入待匹配指纹图像 $Q(x, y)$ 和模板指纹图像 $T(x', y')$,使用Verifinger SDK对这两幅图像进行处理,获得两幅图像的细节点集合 $m_Q = (x, y, \theta)_i (i=1, \dots, N_1)$ 和 $m_T = (x', y', \theta')_i (i=1, \dots, N_2)$,以及指纹细化图像 $skeleton_Q$ 和 $skeleton_T$,其中 (x, y) 表示待匹配指纹图像 $Q(x, y)$ 的像素点坐标, (x', y') 表示模板指纹图像 $T(x', y')$ 的像素点坐标, θ 与 θ' 表示细节点角度, N_1 代表待匹配指纹图像中细节点总个数, N_2 代表模板指纹图像中细节点总个数,Verifinger SDK是基于PC平台的单机指纹识别算法软件,集成了指纹图像预处理、指纹图像特征提取、指纹图像匹配等;

[0008] (2) 对两幅指纹图像的细节点坐标进行尺度为 $scale = DPI_T/DPI_Q$ 的缩放,其中 DPI_Q 为采集待匹配指纹图像 $Q(x, y)$ 所使用指纹传感器的分辨率, DPI_T 为采集模板指纹图像 $T(x', y')$ 所使用指纹传感器的分辨率,缩放前待匹配指纹图像中细节点的坐标为 (x_0, y_0) ,缩放后的细节点坐标为 (x, y) ,两者之间的关系如下:

$$[0009] \quad (x, y) = (x_0, y_0) \frac{DPI_T}{DPI_Q};$$

[0010] (3) 使用MCC SDK计算待匹配指纹细节点集合 $m_Q = (x, y, \theta)_i (i=1, \dots, N_1)$ 和模板指纹细节点集合 $m_T = (x', y', \theta')_i (i=1, \dots, N_2)$ 间的MCC相似度,并将细节点MCC相似度降序排序,获得细节点MCC相似度分数集合 $\{S(i, j)\}_{i=1}^{N_1} \}_{j=1}^{N_2}$, $S(i, j)$ 代表待匹配指纹的第 i 个细节点与模板指纹的第 j 个细节点间的匹配分数;

[0011] (4) 选取MCC相似度排名前25的细节点对作为基准配准点对集 $M_i (i=1, \dots, 25)$,利用传播算法根据基准配准点计算两幅指纹图像的匹配分数;

[0012] (5) 将依据25对初始配准细节点对计算的25个匹配分数进行排序,选取最高的值作为两幅指纹图像的最终匹配分数;

[0013] (6) 设定固定阈值 TH ,将匹配分数与所设定固定阈值相比,如果匹配分数大于阈值则认为指纹为真匹配,否则为假匹配。

[0014] 进一步,所述(4)具体包括:

[0015] 1) 根据MCC相似度从大到小依次选取 $M_i (i=1, \dots, 25)$ 中的一对基准配准点对;

[0016] 2) 分别计算两幅图像中其它细节点相对于基准配准点的极坐标 $Q(r, t)$ 和 $T(r', t')$,其中 r, t 和 r', t' 按照以下公式计算如下:

$$[0017] \quad \begin{cases} r^2 = x^2 + y^2 \\ \tan t = \frac{y}{x} \end{cases};$$

$$[0018] \quad \begin{cases} r'^2 = x'^2 + y'^2 \\ \tan t' = \frac{y'}{x'} \end{cases};$$

[0019] r 和 t 分别对应待匹配指纹中其他细节点相对于基准配准点的距离和角度, r' 和 t' 分别对应模板指纹中其他细节点相对于基准配准点的距离和角度;

[0020] 3) 在待匹配指纹图像和模板指纹图像中,以基准配准点为中心构造细节点局部结构算子;

[0021] 4) 将位于待匹配指纹细节点局部结构算子内的细节点统一为细节点集 m ,依次遍历所有位于模板指纹细节点局部结构算子内的细节点,寻找能与 m 中细节点匹配的细节点组成集合 m' ,并判断 m 与 m' 中是否存在种子点对 S_d ;

[0022] 5) 如果存在种子点对 S_d ,则开启传播,以种子点对 S_d 作为新的基准配准点对重复3)与4),没有满足条件的种子点对时,终止传播,即匹配完毕;

[0023] 6) 所有匹配细节点对的MCC相似度的和为两幅指纹图像之间的匹配分数,计算公式如下:

$$[0024] \quad S = \sum_{i=1, j=1}^{num} S_{(i,j)MCC};$$

[0025] 其中 num 是确定为匹配的细节点对数, $S_{(i,j)MCC}$ 是待匹配指纹中细节点 i 和模板指纹中的细节点 j 之间的MCC相似度分数。

[0026] 进一步,所述构造细节点局部结构算子包括:

[0027] (1) 将以初始配准细节点为圆心,一定半径长度的圆分成 $n=R \times K$ 个区域,其中 R 代表圈数, K 是按照360度角将其分割的数目,取 $R=5, R=8, R_{max}$ 是最外圈的半径,为150像素;

[0028] (2) 设定待匹配指纹细节点局部结构算子的五个半径的数值分别是:30、60、90、120、150像素,角度区域的范围为: $0^\circ \sim 45^\circ, 45^\circ \sim 90^\circ, 90^\circ \sim 135^\circ, 135^\circ \sim 180^\circ, 180^\circ \sim 225^\circ, 225^\circ \sim 270^\circ, 270^\circ \sim 315^\circ, 315^\circ \sim 360^\circ$;

[0029] (3) 五个半径的数值范围是:0~38像素,18~72像素,44~106像素,70~140像素,96~174像素,模板指纹的局部细节点结构算子角度区域的分割数为8,角度区域的范围在原始八个区域的基础上,每个区域的左右边界各扩大 30° ;

[0030] (4) 以初始配准细节点方向为基准方向,从内而外的按逆时针方向对细节点结构算子内的扇形区域进行编号。

[0031] 进一步,所述判断是否存在种子点对 S_d 的流程包括:

[0032] 1) 比较待匹配的细节点对是否处于同一个扇形区域中,如果处于同一个扇形区域,进行下一步,不满足则直接跳到下一个细节点对;

[0033] 2) 当判断在同一个局部细节点结构算子中的待匹配指纹与模板指纹中的两个细节点属于同一个扇形区域后,求极坐标的距离差值与角度差值,当距离与角度的差值都小于设定的阈值且二者的MCC相似度高于一所设的阈值时,认定这两个细节点匹配;

[0034] 3) 判断细节点匹配时的两种情况:

[0035] 是否满足基准配准点条件,满足则可作为下一个传播的种子;

[0036] 是否满足普通匹配的细节点对的条件,满足基准配准点要求的细节点对必然也是匹配的细节点对。

[0037] 进一步,所述判定细节点匹配的两种情况包括:

[0038] 1) 判断待匹配指纹图像与模板指纹图像中的细节点对是否是下一步的基准配准点对,细节点对需满足下式四个条件:

$$[0039] \quad \begin{cases} \Delta r < Th_{r1} \\ \Delta t < Th_{t1} \\ \Delta \theta < Th_{\theta 1} \\ MCC > Th_{MCC1} \end{cases} ;$$

[0040] 其中 $\Delta r = |r - r'|$, $\Delta t = \min\{|t - t'|, 360 - |t - t'|\}$, $\Delta \theta = \min\{|\theta - \theta'|, 360 - |\theta - \theta'|\}$, MCC表示两个对应细节节点的MCC相似度, Th_{r1} 、 Th_{t1} 、 $Th_{\theta 1}$ 和 Th_{MCC1} 为距离、相对角度、细节节点角度和相似度等处理的预设阈值,当满足这四个条件的时候,一对细节节点被认定为满足基准配准点的条件;

[0041] 2) 如果这一对细节节点不满足基准配准点的条件,继续判断这一对细节节点是否满足普通配准的条件,条件也是四个:

$$[0042] \quad \begin{cases} \Delta r < Th_{r2} \\ \Delta t < Th_{t2} \\ \Delta \theta < Th_{\theta 2} \\ MCC > Th_{MCC2} \end{cases} ;$$

[0043] 如果这一对细节节点满足以上条件,则认为是匹配的。

[0044] 进一步,所述阈值设定, Th_{r1} 设置为 $3.8 \times R$, Th_{t1} 为 $\pi/22 + R \times \pi/22$, $Th_{\theta 1}$ 为 $\pi/14 + R \times \pi/14$, Th_{MCC1} 为0.57, Th_{r2} 为 $4.5 \times R$, Th_{t2} 为 $\pi/16 + R \times \pi/22$, $Th_{\theta 2}$ 为 $\pi/12 + R \times \pi/12$, Th_{MCC2} 为0.5,其中R表示细节节点所在的环数。

[0045] 本发明利用细节节点的角度和细节节点间距离在局部区域内受非线性形变影响较小的特点,选择相似度较高的基准细节节点对构建细节节点局部结构算子,完成细节节点间的精确匹配。通过传播算法找到满足基准点条件的细节节点对,使用新的基准点对继续进行局部区域内的精确匹配。匹配范围从局部区域逐步传播到整幅图像,进而完成全局层面的匹配工作,实现了较低的EER值,指纹交叉匹配结果更为准确;(EER越低,正确率越高,匹配更为准确)。

[0046] 本发明通过传播算法找到更为精确的匹配细节节点对,能够为指纹图像相似度分数的计算提供进一步的指导。本发明利用局部细节节点结构算子获得中心细节节点邻域内的细节节点分布信息,并且此特征信息具有良好的旋转不变性,使得指纹交叉库匹配实验室识别准确率达到97.44%。用本发明的方法对FingerPass数据库下的AES子数据库和URU子数据库做遍历识别匹配实验,本次仿真的匹配分数EER求解为2.56%。

附图说明

[0047] 图1是本发明实施例提供的基于传播算法的指纹交叉库匹配方法流程图。

[0048] 图2是本发明实施例提供的细节节点局部结构算子示意图。

[0049] 图3是本发明实施例提供的指纹图像匹配结果示意图。

具体实施方式

[0050] 为了使本发明的目的、技术方案及优点更加清楚明白,以下结合实施例,对本发明

进行进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0051] 下面结合附图对本发明的应用原理作详细的描述。

[0052] 如图1所示,本发明实施例提供的基于传播算法的指纹交叉库匹配方法包括以下步骤:

[0053] S101:输入待匹配指纹图像和模板指纹图像,提取两幅图像的细节点和细化图;

[0054] S102:对两幅图像的细节点坐标进行尺度缩放,得到缩放后的细节点坐标信息;

[0055] S103:计算两幅指纹图像两两细节点间的MCC相似度,并根据相似度的值降序排序;

[0056] S104:选取相似度排名靠前的若干个细节点对作为基准配准点对集,依次选取一对细节点对作为基准配准点对,对待匹配指纹图像与模板指纹图像进行匹配;

[0057] S105:分别计算两幅指纹图像的匹配分数,选取所有匹配分数中的最大值作为两幅指纹图像的最终匹配分数;

[0058] S106:通过设定固定阈值判断是否为真匹配。

[0059] 下面结合具体实施例对本发明的应用原理作进一步的描述。

[0060] 本发明实施例提供的基于传播算法的指纹交叉库匹配方法具体包括以下步骤:

[0061] (1)输入待匹配指纹图像 $Q(x, y)$ 和模板指纹图像 $T(x', y')$,使用Verifinger SDK对这两幅图像进行处理,获得两幅图像的细节点集合 $m_Q = (x, y, \theta)_i (i = 1, \dots, N_1)$ 和 $m_T = (x', y', \theta')_i (i = 1, \dots, N_2)$,以及指纹细化图像 $skeleton_Q$ 和 $skeleton_T$,其中 (x, y) 表示待匹配指纹图像 $Q(x, y)$ 的像素点坐标, (x', y') 表示模板指纹图像 $T(x', y')$ 的像素点坐标, θ 与 θ' 表示细节点角度, N_1 代表待匹配指纹图像中细节点总个数, N_2 代表模板指纹图像中细节点总个数,Verifinger SDK是6.3标准版商用指纹预处理算法软件;

[0062] (2)对两幅指纹图像的细节点坐标进行尺度为 $scale = DPI_T/DPI_Q$ 的缩放,其中 DPI_Q 为采集待匹配指纹图像 $Q(x, y)$ 所使用指纹传感器的分辨率, DPI_T 为采集模板指纹图像 $T(x', y')$ 所使用指纹传感器的分辨率,假设缩放前待匹配指纹图像中细节点的坐标为 (x_0, y_0) ,缩放后的细节点坐标为 (x, y) ,两者之间的关系如下:

$$[0063] \quad (x, y) = (x_0, y_0) \frac{DPI_T}{DPI_Q};$$

[0064] (3)使用MCC SDK计算待匹配指纹细节点集合 $m_Q = (x, y, \theta)_i (i = 1, \dots, N_1)$ 和模板指纹细节点集合 $m_T = (x', y', \theta')_i (i = 1, \dots, N_2)$ 间的MCC相似度,并将细节点MCC相似度降序排序,获得细节点MCC相似度分数集合 $\{S(i, j)\}_{i=1}^{N_1} \}_{j=1}^{N_2}$, $S(i, j)$ 代表待匹配指纹的第 i 个细节点与模板指纹的第 j 个细节点间的匹配分数,MCC SDK是一种业界普遍使用求取细节点相似度的算法;

[0065] (4)选取MCC相似度排名前25的细节点对作为基准配准点对集 $M_i (i = 1, \dots, 25)$,利用传播算法根据基准配准点计算两幅指纹图像的匹配分数:

[0066] (4a)根据MCC相似度从大到小依次选取 $M_i (i = 1, \dots, 25)$ 中的一对基准配准点对作为基准配准点对;

[0067] (4b)分别计算两幅图像中其它细节点相对于基准配准点的极坐标 $Q(r, t)$ 和 $T(r', t')$,

t'),其中r、t和r'和t'按照以下公式计算如下:

$$[0068] \quad \begin{cases} r^2 = x^2 + y^2 \\ \tan t = \frac{y}{x} \end{cases};$$

$$[0069] \quad \begin{cases} r'^2 = x'^2 + y'^2 \\ \tan t' = \frac{y'}{x'} \end{cases};$$

[0070] r和t分别对应待匹配指纹中其他细节点相对于基准配准点的距离和角度,r'和t'分别对应模板指纹中其他细节点相对于基准配准点的距离和角度;

[0071] (4c) 在待匹配指纹图像和模板指纹图像中,以基准配准点为中心构造细节点局部结构算子;

[0072] (4d) 将位于待匹配指纹细节点局部结构算子内的细节点统一为细节点集m,依次遍历所有位于模板指纹细节点局部结构算子内的细节点,寻找能与m中细节点匹配的细节点组成集合m',并判断m与m'中是否存在种子点对Sd;

[0073] (4e) 如果存在种子点对Sd,则开启传播,以种子点对Sd作为新的基准配准点对重复步骤(4c)与(4d),没有满足条件的种子点对时,终止传播,即匹配完毕;

[0074] (4f) 所有匹配细节点对的MCC相似度的和为两幅指纹图像之间的匹配分数,计算公式如下:

$$[0075] \quad S = \sum_{i=1, j=1}^{num} S_{(i,j)MCC};$$

[0076] 其中num是确定为匹配的细节点对数, $S_{(i,j)MCC}$ 是待匹配指纹中细节点i和模板指纹中的细节点j之间的MCC相似度分数;

[0077] (5) 将依据25对初始配准细节点对计算的25个匹配分数进行排序,选取最高的值作为两幅指纹图像的最终匹配分数;

[0078] (6) 设定固定阈值TH,将匹配分数与所设定固定阈值相比,如果匹配分数大于阈值则认为指纹为真匹配,否则为假匹配。

[0079] 在本发明的优选实施例中步骤(4)的构造细节点局部结构算子包括:

[0080] 1) 将以初始配准细节点为圆心,一定半径长度的圆分成 $n=R \times K$ 个区域,其中R代表圈数,K是按照360度角将其分割的数目,取 $R=5, R=8, R_{max}$ 是最外圈的半径,为150像素;

[0081] 2) 设定待匹配指纹细节点局部结构算子的五个半径的数值分别是:30、60、90、120、150像素,角度区域的范围为: $0^\circ \sim 45^\circ, 45^\circ \sim 90^\circ, 90^\circ \sim 135^\circ, 135^\circ \sim 180^\circ, 180^\circ \sim 225^\circ, 225^\circ \sim 270^\circ, 270^\circ \sim 315^\circ, 315^\circ \sim 360^\circ$;

[0082] 3) 在待匹配指纹细节点局部结构算子基础上,对于模板指纹细节点局部结构算子,扩大其中每一个环的上下边界和左右边界,使相邻环之间的半径和角度互有重叠部分。设定五个半径的数值范围分别是:0~38像素,18~72像素,44~106像素,70~140像素,96~174像素,模板指纹的局部细节点结构算子角度区域的分割数为8,角度区域的范围在原始八个区域的基础上,每个区域的左右边界各扩大 30° ;

[0083] 4) 以初始配准细节点方向为基准方向,从内而外的按逆时针方向对细节点结构算

子内的扇形区域进行编号；

[0084] 在本发明的优选实施例中步骤(4)中的判断是否存在种子点对Sd的流程包括：

[0085] 1) 比较待匹配的细节点对是否处于同一个扇形区域中，如果处于同一个扇形区域，进行下一步，不满足则直接跳到下一个细节点对；

[0086] 2) 当判断在同一个局部细节点结构算子中的待匹配指纹与模板指纹中的两个细节点属于同一个扇形区域后，求其极坐标的距离差值与角度差值，当距离与角度的差值都小于设定的阈值且二者的MCC相似度高于所设的阈值时，认定这两个细节点匹配；

[0087] 3) 判断细节点匹配时的两种情况：

[0088] (a) 是否满足基准配准点条件，满足则可作为下一个传播的种子；

[0089] (b) 是否满足普通匹配的细节点对的条件，满足基准配准点要求的细节点对必然也是匹配的细节点对；

[0090] 在本发明的优选实施例中步骤(4)中的判定细节点匹配的两种情况包括：

[0091] 1) 判断待匹配指纹图像与模板指纹图像中的细节点对是否是下一步的基准配准点对，需要细节点对满足下式四个条件：

$$[0092] \quad \begin{cases} \Delta r < Th_{r1} \\ \Delta t < Th_{t1} \\ \Delta \theta < Th_{\theta 1} \\ MCC > Th_{MCC1} \end{cases} ;$$

[0093] 其中 $\Delta r = |r - r'|$ ， $\Delta t = \min\{|t - t'|, 360 - |t - t'|\}$ ， $\Delta \theta = \min\{|\theta - \theta'|, 360 - |\theta - \theta'|\}$ ，MCC表示两个对应细节点的MCC相似度， Th_{r1} 、 Th_{t1} 、 $Th_{\theta 1}$ 和 Th_{MCC1} 为距离、相对角度、细节点角度和相似度等处理的预设阈值，当满足这四个条件的时候，这一对细节点被认定满足基准配准点的条件；

[0094] 2) 如果这一对细节点不满足基准配准点的条件，就要继续判断这一对细节点是否满足普通配准的条件。条件也是四个：

$$[0095] \quad \begin{cases} \Delta r < Th_{r2} \\ \Delta t < Th_{t2} \\ \Delta \theta < Th_{\theta 2} \\ MCC > Th_{MCC2} \end{cases} ;$$

[0096] 如果这一对细节点满足以上条件，则认为是匹配的。

[0097] 下面结合仿真对本发明的应用效果作详细的描述。

[0098] 1、仿真条件

[0099] 在PC机的MATLAB.R2015b环境下进行仿真，PC机配置CoreI7处理器，主频为3.4-GHz。仿真指纹图像来自于FingerPasscross-matching数据库中的两个子数据库：AES数据库和URU数据库，实验中采用了两种市场上有代表性的采集仪：URU4000B光学按压式采集仪和AES2501刮擦式采集仪。

[0100] 2、仿真内容与结果分析

[0101] 仿真1，用本发明的方法获得指纹交叉库匹配效果图，如图3所示，其中图3(a)为指

纹图像AES2501_0009_01VSURU_0009_01,图3 (b) 为指纹图像AES2501_0189_01VSURU_0189_01。

[0102] 仿真2,用本发明的方法对FingerPass数据库做遍历识别匹配实验,本次仿真的匹配分数阈值为0.0535,EER求解为2.56%。

[0103] 实验表明,本发明能够准确的进行指纹交叉库图像匹配。

[0104] 以上所述仅为本发明的较佳实施例而已,并不用以限制本发明,凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明的保护范围之内。

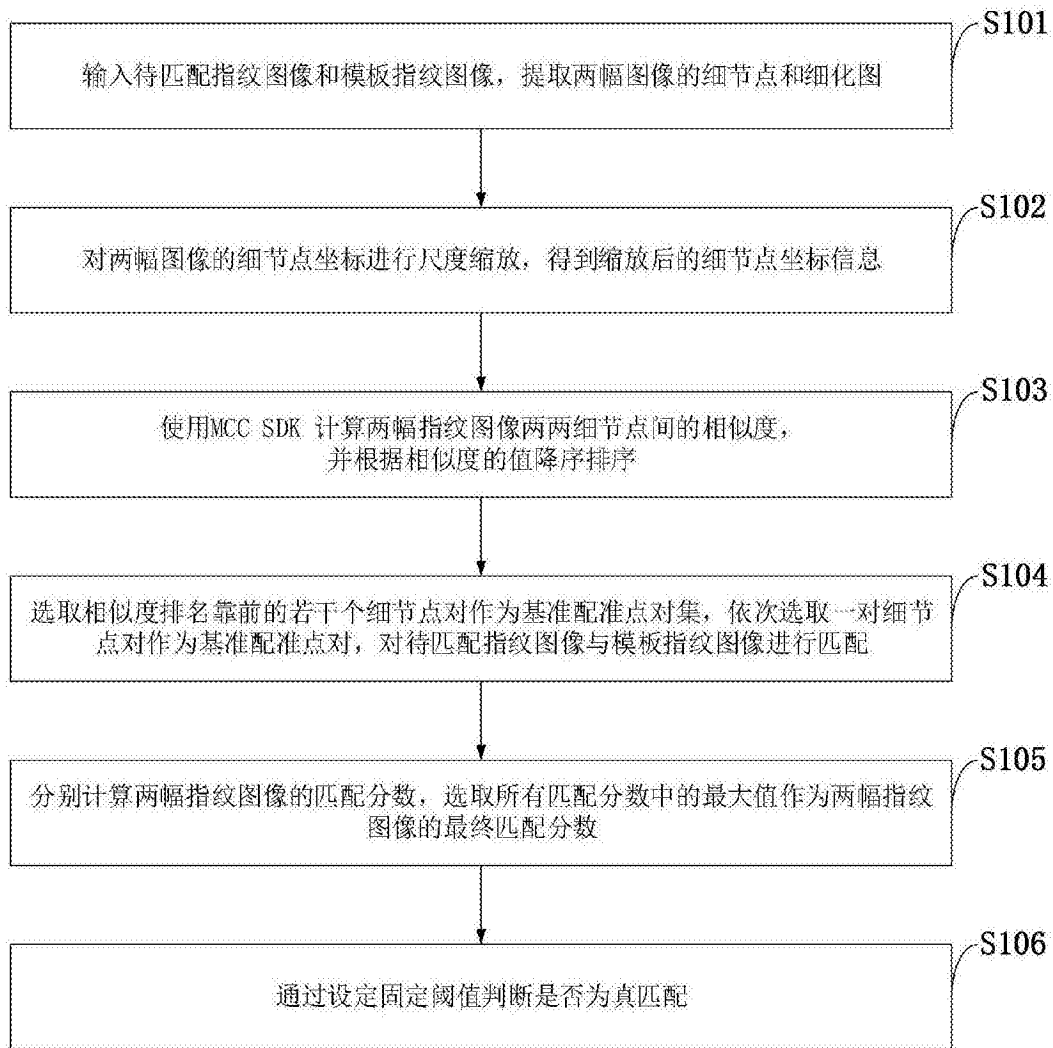


图1

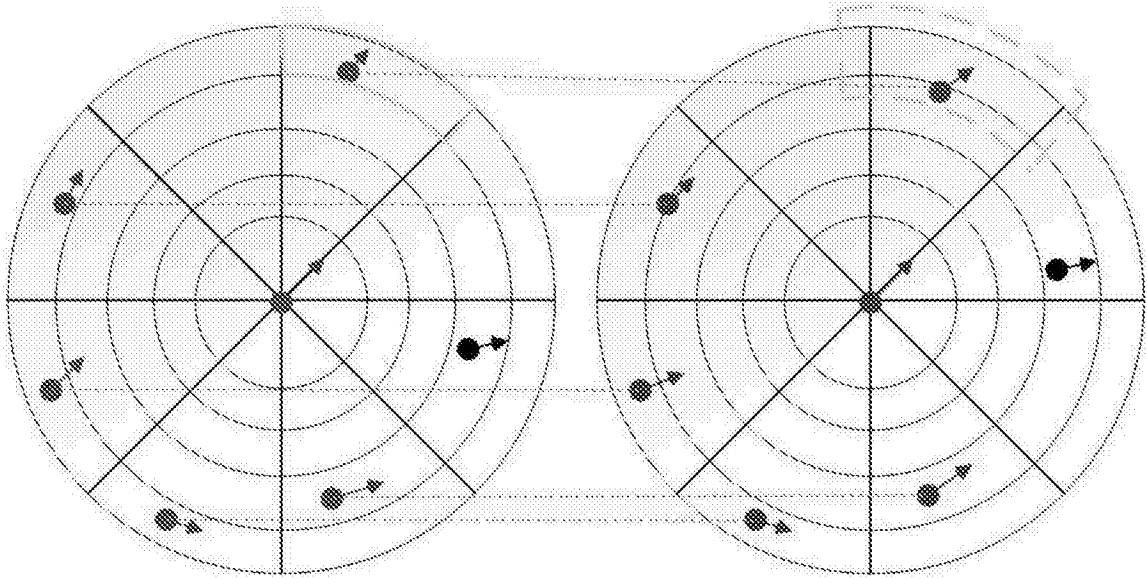


图2



(a)



(b)

图3