文章编号：1672-2892(2010)02-0190-06

基于FPGA的指纹预处理技术

谭 斌，刘平净，李 锋

(重庆大学 通信工程学院，重庆 400030)

摘 要：指纹图像预处理是指纹识别系统中的重要部分，通常包括归一化、方向滤波、二值化和细化等环节，算法复杂，计算量大。针对这种特点，设计了一种新的指纹图像预处理方法，运用SOPC技术，将预处理过程中耗时大的部分用FPGA实现，耗时小但处理繁琐的部分用软件实现，由下载到FPGA的软核Nios II处理器负责软件的运行和软、硬件的协调。实验结果表明该方法在确保图像质量提高的前提下大幅度提高了处理速度，保证了自动指纹识别系统的实时性，为指纹识别系统的研究和实现提供了新的设计思路和方法。

关键词：预处理；现场可编程门阵列；指纹方向；指纹频率；Gabor滤波

中图分类号：TN911.73；TP391.41 **文献标识码：**A

Fingerprint image preprocessing based on FPGA

TAN Bin，LIU Ping-jing，LI Feng

(College of Communication Engineering，Chongqing University，Chongqing 400030，China)

Abstract：The fingerprint image pre-processing is a very important part of Automatic Fingerprint Identification System(AFIS). It includes such processes like normalization，orientation filtering，binarization，thinning，etc. The fingerprint image pre-processing algorithm is complex with plenty of computation. This study designed a new fingerprint image pre-processing method by using System On Programmable Chip(SOPC) technology，realized the most of time-consuming processes by using Field Programmable Gate Array(FPGA)，and implemented the part of less time-consuming but handling complexly with the software. The soft-core Nios II processor which had been downloaded to FPGA was responsible for running the software and coordinating the software with hardware. Experimental results showed that this method could improve the quality of images and processing speed greatly，thus it ensured the real-time of AFIS and provided a new design thought for the study of AFIS.

Key words：pre-processing；Field Programmable Gate Array；fingerprint orientation；fingerprint frequency；Gabor filter

指纹图像预处理是对低质量的指纹图像采用一定的算法进行处理，使其纹线结构清晰、突出，并保留固有的特征信息而避免产生伪特征信息，其目的是确保特征信息的准确性和可靠性。指纹图像预处理过程包括图像归一化、方向提取、频率提取、方向滤波、二值化等环节。传统的嵌入式指纹识别系统中，指纹图像预处理都是由嵌入式处理器独自完成的，但是由于方向提取、频率提取和方向滤波等算法复杂度较高，单独使用嵌入式处理器来完成所需时间太长，整个指纹识别系统很难取得较好的实时性。针对该情况，提出了一种新的指纹图像预处理实现方案，即以下载到FPGA的软核CPU为控制器，运用FPGA逻辑资源实现指纹图像预处理中的各个环节，从而达到提高处理速度的目的。该方法采用可编程逻辑器件实现，具有灵活的设计方式，可裁剪，可扩充，并具备系统软硬件可协同设计的特点，指纹图像预处理中的复杂算法均采用硬件实现，极大地提高了处理速度。

1 指纹图像预处理的基本原理

指纹识别算法主要依赖于一枚指纹中的纹理和细节点(主要是端点和叉点)，然而在指纹图像的采集过程中，

收稿日期：2009-08-10；修回日期：2009-10-26

基金项目：国家自然科学基金资助项目(30570473)

由于手指本身、外部采集条件等因素的影响，从指纹传感器上采集到的指纹图像往往质量较差，表现为：指纹脊线粘连或断开，产生了虚假细节特征点，遗漏了正确的细节特征点，细节特征点位置和方向属性失真等。这些因素都在不同程度上影响了系统的识别效果，所以在对一幅指纹图像提取特征前一定要对该图像做增强处理，以保证所提取的特征是正确可靠的指纹特征信息。

常见的指纹图像预处理流程如图 1 所示，包括指纹图像的归一化、指纹方向提取、指纹频率提取、方向滤波、二值化和细化。原始指纹图像通过预处理后，得到一幅清晰的、单像素的指纹框架。

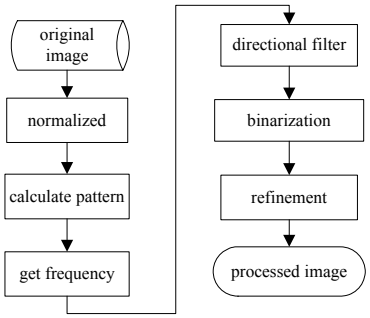

Fig.1 Fingerprint image preprocessing
图 1 指纹图像预处理流程

2　总体设计

系统的总体设计思路是运用 SOPC 技术，将预处理过程中耗时大的部分用 FPGA 实现，耗时小但处理繁琐的部分用软件实现，由下载到 FPGA 的软核 Nios II 处理器负责软件的运行和软、硬件的协调。基于 FPGA 的指纹预处理系统硬件总体设计如图 2 所示。

系统处于工作状态时，由指纹传感器采集指纹图像，Nios II处理器将采集到的图像存储到静态随机存储器(Static Random Access Memory，SRAM)中，之后由Nios II对预处理控制器发出指令实现指纹图像的预处理，预处理控制器和归一化、计算方向、计算频率、方向滤波模块等外设一起构成了预处理系统的硬件实现部分，Nios II发出指令后，预处理控制器负责从SRAM中读取指纹图像并送到相应的处理模块进行处理，最后将处理结果写回SRAM，并发起中断通知Nios II处理完毕。整个系统的程序均运行在SDRAM中，提取到的指纹特征存储在FLASH中，键盘和LCD负责人机交互。

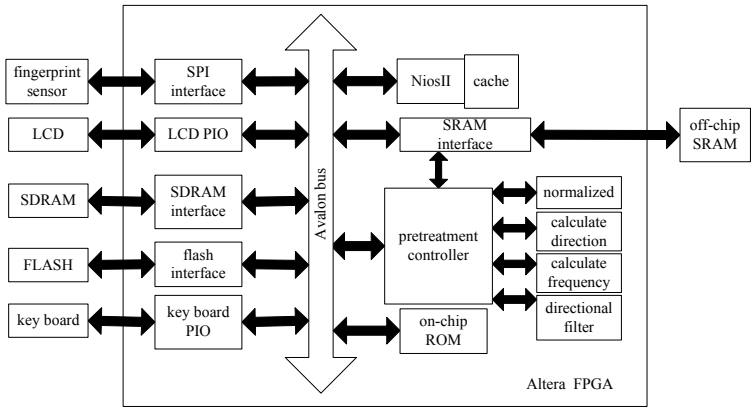

Fig.2 Overall design graph of the system
图 2 系统总体设计图

3　预处理的设计与实现

3.1 指纹图像归一化

指纹图像归一化的主要目的是降低沿脊线和谷线方向的灰度变化程度，使图像具有预定的均值和方差，有利于方向图和指纹频率的提取。一般按照式(1)进行归一化：

$$G(i,j)=\begin{cases} M_0+\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & I(i,j)>M \\ M_0-\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{其它} \end{cases} \tag{1}$$

式中：$I(i,j)$为指纹图像在(i,j)处的灰度值；M,VAR为指纹图像的均值和方差；M_0,VAR_0 则是归一化后期望得到的图像均值和方差。为了便于该算法的FPGA实现，对式(1)做进一步的修改，修改后的公式如下：

$$G(i,j)=\begin{cases} M_0+\lambda\times VAR\times\left|I(i,j)-M\right| & I(i,j)>M \\ M_0-\lambda\times VAR\times\left|I(i,j)-M\right| & \text{其它} \end{cases} \tag{2}$$

式中λ为一常数。为了减小由于数据取整引入的误差，同时又能够得到较为准确的数据，需要将数据进行扩大，而在 FPGA 中实现，采用移位的方式进行。归一化处理后的效果见图 3。

3.2 指纹方向提取

方向提取时以指纹的块方向作为该块区域内点的方向。块方向的提取采用梯度算法[1]，原理如下：设$f(i,j)$表示指纹像素点(i,j)处的灰度值。

a) 将图像分成大小为$W\times W$互不重叠子块，W的大小一般包含一条脊线和一条谷线为宜，这里W取10。

b) 采用Sobel算子计算图像中每个像素点(i,j)在x方向和y方向上的梯度$G_x(u,v)$和$G_y(u,v)$，其中Sobel算子的模板系数如图4所示。

(a) image of pre-normalized (b) normalized image

Fig.3 Fingerprint normalization

图 3 指纹图像归一化效果图

c) 计算以(i,j)为中心的$W\times W$子块的块方向：$\theta(i,j)$，$V_x(i,j)$，$V_y(i,j)$，由于x方向和y方向上的梯度$G_x(u,v)$和$G_y(u,v)$在前面均已计算出来，因此块方向$\theta(i,j)$只需按式(3)便可计算出来。但实际要求该方向要唯一，而通过式(3)计算可获得2个值：$\theta(i,j)$与$\theta(i,j)+\pi$，这便与预先要求有矛盾，因此在实际处理中，通常计算$2\theta(i,j)$的值，因为$\theta(i,j)\,2(+\pi)=2\theta(i,j)+2\pi=2\theta(i,j)$，这样便能保证块方向的唯一性。

1	0	−1
2	0	−2
1	0	−1

x-axis direction

1	0	−1
2	0	−2
1	0	−1

y-axis direction

Fig.4 Sobel template coefficient

图 4 Sobel算子模板系数

$$\theta(i,j)=\cot\left(\frac{G_x(u,v)}{G_y(u,v)}\right) \tag{3}$$

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

$$\theta\left(i,j\right)=\frac{1}{2}\cot\left(\frac{2\tan\theta\left(i,j\right)}{1-\tan^2\theta\left(i,j\right)}\right) \tag{5}$$

把式(3)代入式(5)经简化可得：

$$\theta\left(i,j\right)=\frac{1}{2}\cot\left(\frac{2G_x(u,v)G_y(u,v)}{G_x^2(u,v)-G_y^2(u,v)}\right) \tag{6}$$

在以(i,j)为中心的$W\times W$子块中包含的指纹像素点数目相当多，因此块方向$\theta(i,j)$应该是所有像素点各自方向的平均值，从而以(i,j)为中心的$W\times W$子块的块方向为：

$$\theta(i,j)=\frac{1}{2}\cot\left(\frac{\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}}\sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}}2G_x(u,v)G_y(u,v)}{\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}}\sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}}(G_x^2(u,v)-G_y^2(u,v))}\right) \tag{7}$$

为了简化式(7)，令$V_x(i,j)$，$V_y(i,j)$分别为：

$$V_x(i,j)=\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}}\sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}}2G_x(u,v)G_y(u,v) \tag{8}$$

$$V_y(i,j)=\sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}}\sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}}(G_x^2(u,v)-G_y^2(u,v)) \tag{9}$$

$$\theta(i,j)=\frac{1}{2}\cot\left(\frac{V_x(i,j)}{V_y(i,j)}\right) \tag{10}$$

采用FPGA实现该算法，也相应地分为几步来进行，首先使用图 4 中给出的Sobel算子计算x方向和y方向的梯度，计算过程中，计算方向模块直接从SRAM中读取原始图像的灰度值，然后将得到的x和y方向的梯度矩阵分别存入SRAM中的空闲区域中，最后利用得到的x,y方向梯度矩阵以及式(5)计算方向，反正切函数在FPGA中实现比较困难，这里采用查表法实现。

3.3 指纹频率提取

在指纹图像的局部非奇异区域内，沿着垂直于脊线方向(即纹线的梯度方向)，指纹的脊线和谷线象素点灰度值的变化大致构成了一个二维正弦波[2]，从图 5 可以看出指纹的脊线和谷线具有很好的局部频率特性。

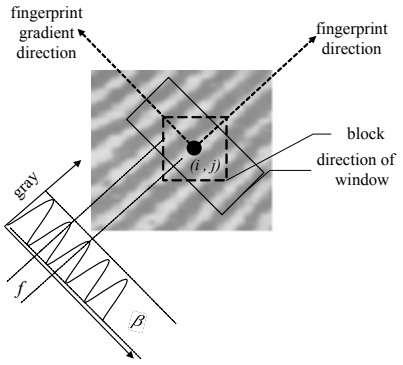

Fig.5 Fingerprint frequency map
图 5 指纹的频率示意图

结合图 5，计算指纹频率的原理如下：

a) 将归一化后的指纹图像平均分成 $W\times W$ 的互不相重叠的子块，W 取 16；

b) 针对每个以 (i,j) 为中心的图像子块，以子块中心点 (i,j) 的指纹方向 $\theta(i,j)$ 为短轴，作一个尺寸为 $l\times W$(32×16)的长方形方向窗口(如图 5)；

c) 对于每个长方形方向窗口，按照下列公式计算沿垂直于指纹图像子块中心点方向 $\theta(i,j)$ (即指纹的梯度方向)的灰度离散信号 $X[0],X[1],X[2],\cdots,X[L-1]$：

$$X[k]=\frac{1}{W}\sum_{d=0}^{w-1}N(u,v)\quad k=0,1,2,\cdots,L-1 \tag{11}$$

$$u=i+(d-\frac{W}{2})\cos\left[\theta(i,j)\right]+(k-\frac{l}{2})\sin\left[\theta(i,j)\right] \tag{12}$$

$$v=j+(d-\frac{W}{2})\sin\left[\theta(i,j)\right]-(k-\frac{l}{2})\cos\left[\theta(i,j)\right] \tag{13}$$

式中：u 和 v 是点(d,k)经过坐标变换后在图像坐标系下的坐标值，其中点(d,k)是长方形窗口中平行于 W 方向上的点；$N(u,v)$表示指纹图像经过归一化后坐标点(u,v)上的像素灰度值。由于 $X[k]$大致组成了一个二维的正弦波，通过计算此正弦波波峰间的距离的平均值 L_0，可以得到正弦波的频率 $f=1/L_0$。

在频率提取的 FPGA 实现中，为了避免出现除法运算，将式(11)修改如下：

$$X'\left[k\right]=W\times X\left[k\right]=\sum_{d=0}^{w-1}N\left(u,v\right) \tag{14}$$

同时式(12)和(13)中出现的三角函数也采用查表的方式实现，这样所得到的结果相对于实际值扩大了 65 536 倍，得到 $X[L]$后，采用邻域比较的方式判断峰值，若 $X[m]$同时大于其两边 $N(N=1$ 或 $2)$点的幅值，则认为 $X[m]$为峰值；因为利用 FPGA 实现除法运算比较耗时，且数据的精度也不能得到保证，所以在系统实现的时候以“频数”(L_0)来代替频率 f。

3.4 方向滤波

为了大幅度地提高指纹图像质量，采用方向滤波对指纹图像进行增强。在去除指纹噪声和保持指纹脊线结构的处理方面，带通滤波器因具有带通特性且与人类视觉模型相吻合，而得到大量应用，并取得了较好的效果。Gabor 滤波器作为一种带通滤波器，其优越性在于满足“不确定性原理”所确定的有效持续时间和有效频率带宽乘积的下限,这就意味着 Gabor 滤波器可以同时在时域和频域两个方面获得最佳的局部化,故本设计中采用 Gabor 滤波器对指纹图像进行增强处理。

Gabor 滤波器的数学表达方式[3]如下：

$$G\left(x,y\right)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right)\exp\left(-2\pi ifx\right) \tag{15}$$

式中：σ_x,σ_y为高斯包络面常数；f为频率。

由 Gabor 滤波器的数学模型可以看出，此种滤波器在 x 方向上带通，y 方向上低通。在对指纹图像进行滤波时，只需对滤波器进行旋转，使其与指纹方向一致，可以实现沿着指纹纹线的信息得以最大程度地增强，而垂直于指纹方向(指纹的梯度方向)的信息则相对减弱。根据欧拉公式，可以对式(15)做进一步的简化，简化后的 Gabor 滤波器如下：

$$g(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^{\ 2}}+\frac{y^2}{\sigma_y^{\ 2}}\right)\right)\cos(2\pi fx) \tag{16}$$

将式(16)旋转到与指纹方向一致，有：

$$g'(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^{\ 2}}+\frac{y'^2}{\sigma_y^{\ 2}}\right)\right)\cos(2\pi fx') \tag{17}$$

式中：

$$x'=y\sin\theta+x\cos\theta\ ;\quad y'=y\cos\theta-x\sin\theta \tag{18}$$

θ 为指纹的纹理方向。

以原指纹图像中的每一个像素点为中心，根据式(17)计算 $W\times W$ 子块内每个像素沿其指纹方向上的 Gabor 增强系数[4]，然后利用式(19)对指纹图像进行滤波增强。

$$G(x,y)=\sum_{u=-w/2}^{w/2}\sum_{v=-w/2}^{w/2}g'(u,v)N(x-u,y-v) \tag{19}$$

$$G'(x,y)=\begin{cases}0 & G(x,y)<0\\ G(x,y) & 0\leqslant G(x,y)\leqslant 255\\ 255 & G(x,y)>255\end{cases} \tag{20}$$

式中：$N(x,y)$为归一化后指纹图像(x,y)处的像素灰度值；$G'(x,y)$为经过 Gabor 滤波增强后的指纹图像(x,y)处的像素灰度值。

用 FPGA 实现方向滤波主要有 2 个难点，即 $\cos(2\pi fx)$ 和 $\exp\left(-\frac{1}{2}(x'^2/\sigma_x^2+y'^2/\sigma_y^2)\right)$的计算，$\cos(2\pi fx)$ 可以利用查表法实现，具体做法为：首先计算 $2\pi fx$，然后将该值转换到 $[0,2\pi]$ 范围内，最后查表得到余弦值；针对第二个难点，先将式(18)代入该式，然后将其简化为 $\exp\left(-(x^2+y^2)/32\right)$ (取 $\sigma_x=\sigma_y=4$)，因为式中 $x,y\in[-W/2,W/2]$ 为有限离散序列，所以该式也可以由查表法实现，在系统实现中只需输入 x^2+y^2 的值即可查表得到 $\exp\left(-(x^2+y^2)/32\right)$，出于对数据精度的考虑，在 FPGA 设计中输出数据均扩大 65 535 倍并取整。

3.5 二值化和细化

指纹图像的二值化和细化分别采用局部自适应二值化算法[5]和快速细化算法[6]实现，由于该处理计算量较小，且用 FPGA 实现比较困难，笔者采用了软件的方式，Nios II 处理器从 SRAM 中读取滤波后的图像到 SDRAM，然后在 SDRAM 中完成指纹图像的二值化和细化。

4 实验结果对比分析

针对图 6(a)的原指纹图像，经过 3.1~3.5 所描述的算法处理后，得到处理后的图像如图 6(b)所示，通过 2 幅图的对比可以发现，处理后的单像素二值图像保留了原图像的有效细节特征点(主要是端点和叉点)，同时去除了原图像中的噪声，并在一定程度上连接了指纹断裂的脊线，为后续指纹识别工作打下了良好基础。

表 1 给出了预处理过程软/硬件实现的性能(主要

(a) original fingerprint image (b) processed fingerprint image

Fig.6 Pre-processing effect

图 6 指纹图像预处理效果

是时间指标)对比，从实验数据可以看出用 FPGA 实现指纹图像预处理过程比同等条件下单独使用软件实现提升几百倍甚至上千倍，采用 FPGA 实现后，整个预处理过程能在 1 s 内实现，提升效果明显。

表 1 软/硬件实现性能对比

Table1 Comparison of different methods

process	achieve by software/s	achieve by hardware/s	improvement
normalize	4.017 5	0.016 0	251
orientate	20.228 0	0.045 0	449
frequency	24.560 0	0.068 5	358
filtering	2 768.50	0.460 0	6 018

5　结论

在自动指纹识别系统中，为了获得稳定而准确的指纹细节特征，对初始指纹图像进行预处理是指纹识别算法中必须执行的一环。在实际指纹识别算法设计中，通常将指纹预处理算法与指纹识别算法都采用软件来实现，但由于预处理算法计算量很大，往往耗费太多处理时间。为此笔者设计了指纹预处理的 FPGA 实现方法，在进行初步实验后表明：该方法不但能有效地保证指纹图像预处理的效果，而且在很大程度上缩短了指纹预处理的时间，从而能够满足自动指纹识别系统实时性的要求。

参考文献：

[1] 臧兰云,刘瑞华. 指纹方向图提取算法研究[J]. 计算机工程, 2005,31(z1):239-243.
[2] 陈沛华,陈晓光. 指纹的脊频率估计[J]. 电路与系统学报, 2007,12(4):109-110.
[3] Weldon TP,Higgins WE. Efficient Gabor Filter Design For Texture Segmentation[J]. Pattern Recognition, 1996,29(12):2005-2015.
[4] 李鹏,杨康. Gabor 滤波器算法在指纹识别中的应用[J]. 沈阳工业学院学报, 2004,23(9):6-8.
[5] 黄贤武,王加俊. 指纹识别预处理组合算法[J]. 计算机应用, 2002,22(10):31-32.
[6] 廖开阳,张学东,章明珠. 一种新的指纹图像快速细化算法[J]. 计算机工程与应用, 2008,44(5):93-95.

作者简介：

谭　斌(1985-)，男，广东韶关人，在读硕士研究生，研究方向为信号与信息处理.email: tb254704629@126.com.

刘平净(1984-)，男，河北邯郸人，在读硕士研究生，主要研究方向为信号与信息处理.

李　锋(1983-)，男，湖北潜江人，在读硕士研究生，主要研究方向为信号与信息处理.