

Gabor 滤波算法在邻域处理器中的实现

丁 亮, 苏光大

(清华大学 电子工程系, 北京 100084)

摘 要: 通过对可编程逻辑门阵列(FPGA)内部乘法器及相应逻辑阵列资源的配置, 将 Gabor 卷积的算法成功嵌入到邻域处理器当中, 并通过高级语言的配置, 使系统能够支持多种尺寸的卷积运算. 实验结果表明, 使用 EP2C70F896C8 型 FPGA, 在外部时钟为 45 MHz、内部时钟为 180 MHz 的工作频率下, 对于一幅 120×90 的灰度图像进行五尺度八方向的卷积运算(系数为 16 位复数)总耗时 47 ms, 相对于使用 Matlab 软件做相同的卷积计算具有 17.6 倍的加速比.

关键词: Gabor; 邻域处理; 卷积; FPGA; 人脸识别

中图分类号: TP33

文献标识码: A

文章编号: 1000-7180(2011)02-0033-04

Implementation of Gabor Filtering Algorithm
in Neighborhood Processors

DING Liang, SU Guang-da

(Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

Abstract: This processor utilizes logic resources of FPGA to build up pipelined parallel convolution matrix, and use high level language to support any scale. Experiment indicates that Gabor filters with five scales and eight orientations (complex kernels) on 120×90 grey image take 47 ms on hardware platform with EP2C70F896C8 FPGA working at 45 MHz external clock, 180 MHz internal clock, gaining a speedup ratio of 17.6 compared to Matlab.

Key words: gabor; O-domain processing; convolution; FPGA; face recognition

1 引言

随着图像处理和图像分析技术的发展,越来越多的图像算法体现出邻域性的特点. Gabor 滤波器作为一种特征提取和图像表征方法,它能够从不同位置、尺度及方向选择性捕捉局部结构信息,很好地提取出表情变化的特征以及指纹纹理的特征^[1]. Gabor 函数很好地描述了哺乳动物初级视觉系统中一对简单视觉神经元的感受特性^[2],是唯一能达到时频测不准关系下界的函数,在合成孔径雷达(SAR)图像预滤波处理、生物特征识别、图像纹理分析等方面都有着广泛的应用.

Gabor 滤波运算本质上是卷积运算,需要做多个尺度多个方向的大邻域二维滤波,卷积核大小都

达到两位数,运算量和数据吞吐率都相当惊人. 倘若使用一个尺寸为 $M \times M$ 的卷积核对一个尺寸为 $N \times N$ 的原始图像做 Gabor 卷积运算,每一次卷积需要进行 $M \times M$ 次乘法^[3-4] 以及 $M \times M - 1$ 次加法运算,所需调用的数据总量为 $2 \times M \times M$ 个数据. 整幅图像需要进行 $M \times M \times N \times N$ 次乘法以及 $(M \times M - 1) \times N \times N$ 次加法运算,数据吞吐量为 $2 \times M \times M \times N \times N$. 对于随机型处理机来说,每周期需要获取 $M \times M$ 个图像像素以及 $M \times M$ 个卷积系数. 如此庞大的数据量及数据吞吐率,目前还没有一种算法可以满足这个条件. 因此,本文采用了行顺序邻域处理机的方法,即每次仅读入一行新的数据,通过流水线并行计算,同样可以实现高速卷积处理. 将 Gabor 滤波器在邻域处理器上实现,不仅可以增强图

收稿日期: 2010-03-13; 修回日期: 2010-06-03

基金项目: 国家“九七三”项目(2005ZDGGQHDX005);公安部重点攻关计划项目(2007CB310600)

像质量,而且能够大幅度提高图像处理速度,保证同步处理的实时性。

2 NIPC-3 邻域并行计算机

2.1 NIPC-3 简介

针对专门的邻域操作,清华大学先后于 1997 年、1999 年、2006 年推出了 NIPC-1、NIPC-2、NIPC-3 型邻域图像并行处理机^[5],分别针对 2×2 、 3×3 与 25×24 的邻域。本文选用了 NIPC-3 型邻域处理机进行邻域处理。

NIPC-3 邻域图像并行计算机是国家自然科学基金支持的研究项目,它包含了视频采集子系统、邻域处理机子系统、DSP 处理器子系统和 PCI 总线接口子系统,其中邻域处理机子系统是硬件系统的核心部分,包括邻域存储器和邻域处理器。邻域存储器包括了邻域帧存和共享内存两个部分。邻域帧存由 4 片位宽为 64 位的 SRAM 组成,用于存储和形成图像邻域。共享内存由两片 32 位宽 2 MB 容量的同步 SRAM 组成,用于存储初始的图像像素数据、Gabor 系数数据以及运算结果数据。邻域处理器由一片 Altera 公司的 EP2C70F896C8 型 FPGA 组成,主要任务包括组织邻域数据,获取邻域数据,处理邻域数据。

它既能处理来自摄像机的视频图像,也能处理计算机通过 PCI 总线送来的数字图像,其系统框图如图 1 所示。

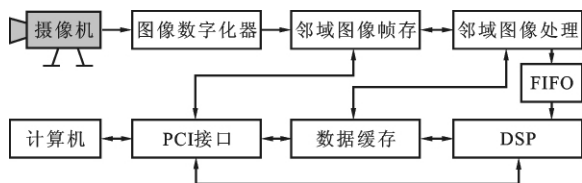


图 1 NIPC-3 系统框图

在运算能力方面,NIPC-3 在处理邻域计算上具有很高的性能。它能在一个系统时钟内形成并处理最大邻域为 25×24 (600 点)的邻域图像,其处理速度远快于用 PC 机软件来进行相同的运算。表 1 给出了部分算法的性能:

表 1 NIPC-3 邻域图像并行计算机部分算法的性能

	图像尺寸	处理时间/ms
Sobel 算子	512×512	0.11
23×23 卷积运算	512×512	3.58
五尺度八方向 Gabor 滤波	90×120	28.57
相关跟踪(25×24 模板)	200×200	0.68

2.2 存储结构及数据流向

本文主要使用 NIPC-3 邻域并行计算机处理由

PC 通过 PCI 总线送来的数字图像。在进行硬件处理之前,首先需要在 PC 上对初始数据进行有效处理,并通过 PCI 接口将其保存在邻域处理器的共享内存(Shram)中。接着,图像数据需要从共享内存倒入邻域帧存(NBram)中进行数据排序,卷积系数需要从共享内存倒入 Fpga 内部的系数存储器(Facram)中进行缓存。之后,再将准备好的图像数据和卷积系数加载到卷积矩阵(Matrix)当中进行卷积运算。其中,对于 31×31 、 41×41 这两种尺寸的卷积,由于其无法在一个时钟周期内计算完成,产生的中间结果需要在缓存器(Accram)中进行缓存。最终的处理结果需要保存在共享内存中并通过 PCI 接口读取出来。整个数据流向如图 2 所示。

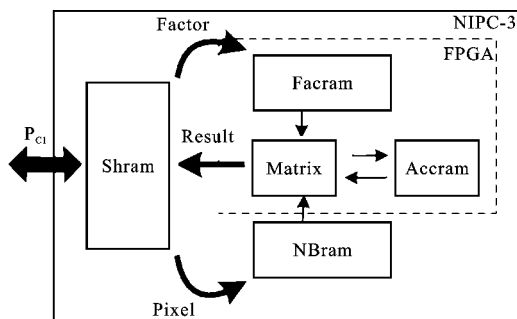


图 2 存储结构及数据流向

3 Gabor 滤波算法的实现

3.1 Gabor 滤波原理

二维 Gabor 函数可看成二维带通滤波器,由二维高斯包络对平面波进行调制而成,分为实部和虚部 2 个分量^[6],即

$$G_k(r) = G_{k,+}(r) + iG_{k,-}(r) \quad (1)$$

$$G_{k,+}(r) = \frac{\kappa^2}{\delta^2} \exp(-\kappa^2 \|r - r_0\|^2 / 2\delta^2)$$

$$\left\{ \cos[k(r - r_0)] - \exp\left(-\frac{\delta^2}{2}\right) \right\} \quad (2)$$

$$G_{k,-}(r) = \frac{\kappa^2}{\delta^2} \exp(-\kappa^2 \|r - r_0\|^2 / 2\delta^2) \sin[k(r - r_0)] \quad (3)$$

式中, $k = \kappa \exp(j\theta_v)$, $\kappa = \kappa_{\max}/\lambda$, $\theta_v = \pi v/n$ 。不同的 u 对应不同的尺度;不同的 v 对应不同的方向。尺度系数 κ^2 保证了不同空间频率的滤波器有大致相等的能量;波矢量 k 控制滤波器的空间分辨率和方向,参数 δ 控制包含函数的震荡次数;由于虚部分量具有零均值而实部分量不具有零均值,所以要从实部分量中减去 $\exp(-\delta^2/2)$,从而使滤波器对图像全局光照变化不敏感。

3.2 卷积矩阵原型

Gabor 滤波在邻域处理器中的实现本质上是进行一系列的卷积运算,即用一个系数模板与其所覆盖的相应位置上的像素作乘加运算,得到的是卷积像素中间位置的卷积结果,将系数模板遍历整个原始图像后,即可得到最终的卷积结果.假设 Gabor 系数的卷积核尺寸为 $M \times M$,原始图像的像素尺寸为 $N \times N$,则用计算公式可简单的表示为

$$r_{x,y} = \sum_l \sum_k f_{l,k} \times p_{x+l-(M-1)/2, y+k-(M-1)/2} \quad (4)$$

式中, $f_{l,k}$ 为模板系数, $p_{x,y}$ 为原始图像像素值, $r_{x,y}$ 为卷积结果.

3.3 Gabor 系数的提取

在提取人脸图像的 Gabor 特征时,为了提取人脸图像在多个尺度多个方向上的特征,通常会采用不同尺度不同方向上的 Gabor 滤波器组成滤波器组,并根据图像特点以及神经生理学的一些结论来选择相应的 Gabor 参数.本文采用了共包含 5 个尺度 ($\kappa_{\max} = \pi/2; \lambda = \sqrt{2}; u = 0, 1, 2, 3, 4$), 8 个方向 ($n = 8; v = 0, 1, 2, 3, 4, 5, 6, 7$) 以及 $\delta = 2\pi$ 的 Gabor 滤波器组^[7].通过对各个尺度的滤波系数按能量集中分布的原则截断为有限长系数后,五个尺度的系数尺寸分别为 11×11 、 15×15 、 21×21 、 31×31 与 41×41 .

由于通过上述方法提取出来的 Gabor 系数的数量级普遍在 $10^{-4} \sim 10^{-3}$ 左右,在应用于硬件实现之前,首先需要对其做相应的扩展,并将扩展结果以二进制的形式保存在硬件存储器上.考虑到之后要进行的移位处理操作,其扩展倍数一般为 2^n ,这就导致无论 n 取多少,扩展结果始终会出现小数,在取整的过程中会被舍去.此外,卷积完成后,又需要将所得结果做移位处理,将低位舍去.如此两次的取舍工作,势必会对其计算精度造成影响.为确保运算精度,本文将 Gabor 系数扩展为 16 位.

3.4 图像像素的存取

本文采用的原始图像是尺寸为 120×90 的灰度图像.根据行顺序邻域图像存取的理论,图像像素在进行卷积处理之前,首先需要保存在位宽足够大的存储器当中,然后再使用 FPGA 进行轮换寻址和数据排序等操作.详细方法请参看相关文献^[8].

NIPC-3 型处理机上嵌入了 4 片位宽为 64 位的邻域存储器,每片存取 8 像素数据,4 片存储体可实现的一列邻域大小为 25×1 ,经过顺序队列可以形成 $25 \times m$ 大小的邻域.此外,该处理机上的 FPGA

包含了 150 个 $18 \text{ bit} \times 18 \text{ bit}$ 的嵌入式乘法器,经 4 倍频后可实现每周处理 600 (即 25×24) 个 $18 \text{ bit} \times 18 \text{ bit}$ 的乘法.因此,系统每周将读入 24 个 8 bit 大小的图像像素数据进行卷积计算.

3.5 卷积单元的配置

鉴于要在 24×25 的卷积计算中完成 11×11 、 15×15 、 21×21 、 31×31 、 41×41 这 5 种尺度的卷积运算,本文将系统中的卷积核拆分成 4 个 12×12 的基本单元进行卷积运算.并通过这些基本单元的组配置,形成如下四种模式,如图 3 所示.

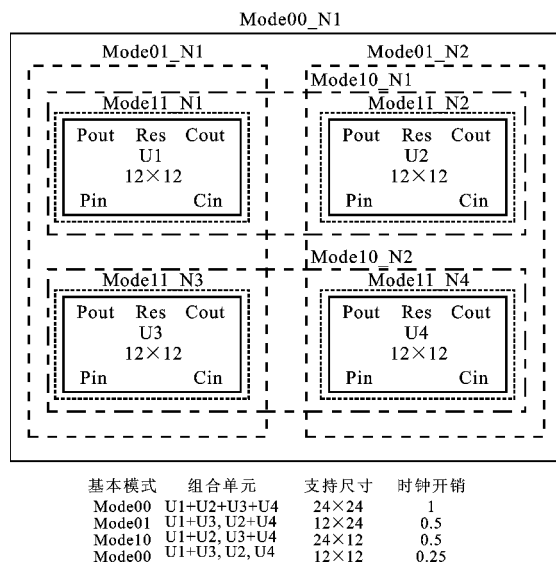


图 3 卷积运算单元

通过如上的配置,加强了组织结构的灵活性,提高了资源的利用率,增加了系统的运行速度,减少了时钟开销.将以上五种卷积尺度进行相应扩展后即可进行卷积运算,其具体的组合方式如图 4 所示.

原始尺寸	扩展尺寸	模式组合	时钟开销
11×11	12×12	11	0.25
15×15	24×24	00	1
21×21	24×24	00	1
31×31	36×36	00+01+10+1	2.25
41×41	48×48	11+11+11+11	4

注:左图所示仅为卷积尺寸为 31×31 扩展成为 36×36 后的卷积大小,其他尺寸的卷积原理相同,此处并未画出

图 4 卷积组合模式

3.6 卷积单元处理

本文采用的是 NIPC-3 型邻域图像并行处理机,其上可最大支持的卷积核尺寸为 25×24 ,而该处理机上所配置的 FPGA 只包含 150 个 $18 \text{ bit} \times 18 \text{ bit}$ 的乘法器,每周只能进行 150 次 16 位 Gabor 系数与 8 位图像像素之间的乘法运算.为对系统资源进行最大化的利用,需要将 FPGA 内部的时钟频率进行 4 倍频,使其每周可以做 600 次乘法运算,以满足卷积核尺寸的最高标准.其具体的实现方

法如图 5 所示. 其中, clk 代指 FPGA 外部时钟, 主要控制了图像像素、Gabor 系数以及卷积结果的数据存取, clkcore 代指 FPGA 内部时钟, 主要控制了输入数据在卷积核内进行的相关运算. 可以看出, 系统在每一个 clk 时钟周期内都会输入一组新的数据, 在经过由 clkcore 控制的多路选择器后, 分四步依次进行卷积运算, 所得结果再依次通过 3 个寄存器顺序缓存. 当 4 个卷积结果全部生成后, 再将这 3 个寄存器的输出值与此时加法器的输出值相加, 即可得到该单元矩阵最终的卷积结果.

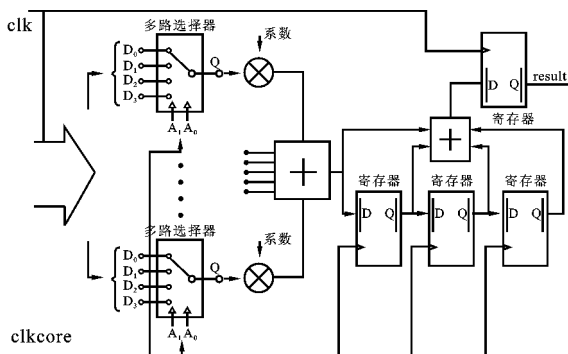


图 5 单元卷积矩阵结构图

3.7 CPU 内核结构

图像处理尤其是 Gabor 卷积的处理拥有着多种多样的应用, 也同时伴随着多种多样的变化. 图像像素大小的变化, Gabor 系数位数的变化, 卷积核尺寸大小的变化, 有效数据来源的变化, 以及系统硬件型号的变化等等. 由于 FPGA 一旦配置完成, 硬件电路将无法变动, 倘若一个系统只能针对某一个特定的应用, 所有的数据传输方式和数据处理结构都已经被写死, 那么这个系统将很难被再次开发使用. 因此, 需要建立一个嵌入式控制单元来管理整个系统.

本文采用了共有三级流水线的 CPU 内核, 分为取指令、指令译码和指令执行, 共包含 32 个 24 位的寄存器, 用来满足不同的应用, 其主要功能如图 6 所示.

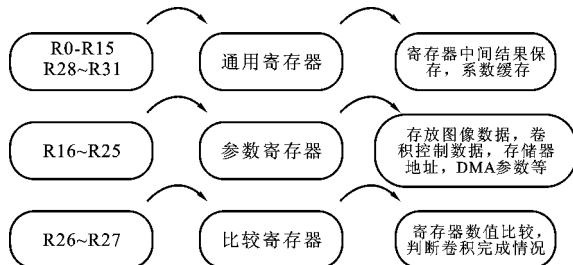


图 6 寄存器机构功能

其中, 寄存器相关数值的修改需要通过使用数学软件 Matlab 来完成. 在 Matlab 下编程完成以后,

可将其编译成一串 24 位大小的控制字, 依次读入相应的寄存器上. 通过该控制单元的配置, 可以完成不同像素大小、不同系数位宽、不同卷积核尺寸的卷积处理.

4 实验结果

本文在 NIPC-3 邻域处理机上实现了 Gabor 滤波算法, 采用了 Altera 公司的 EP2C896C8 型 FPGA 完成主要的 Gabor 卷积运算, 并与在 PC(2.67 GHz Core2 Q9450 CPU, 4 GB 内存) 上使用专业的数学软件 Matlab 计算的结果如表 2 所示进行了比较. 其中, 原始图像采用了 120×90 的灰度图像, 8 位、16 位代表 Gabor 系数扩展的位数, 精确度指硬件处理结果与软件处理结果的偏差.

表 2 测试结果

	时钟频率/ MHz	处理时间/ ms	加速比	精确度/ %
硬件加速(16 位)	45	47	17.6	96.64
Matlab(16 位)	1 970	828.1	/	/
硬件加速(8 位)	60	35.7	10.3	92.04
Matlab(8 位)	1.97	368.6	/	/

5 结束语

本文通过 FPGA 内的卷积处理在邻域处理器上实现了 Gabor 滤波算法的硬件加速. 实验结果表明, 用硬件加速的方法实现 Gabor 滤波算法, 可以大幅度提高 Gabor 特征提取的运算速度, 为 Gabor 小波在图像识别领域的应用起到了积极的作用. 此外, 使用 16 位的 Gabor 系数进行硬件加速, 虽然其在处理速度上比 8 位有所降低, 但是其精确度更大, 加速比更高, 可以实现更为稳定、有效的加速功能. 后续的工作是将本项研究成果应用于人脸识别, 建成一个嵌入式人脸识别系统.

参考文献:

- [1] ZHU Jianxiang, SU Guangda, LI Yingchun. Facial Expression Recognition Based on Gabor Feature and Ada-boost[J]. Journal of Optoelectronics · Laser(光电子·激光), 2006, 17(8):993-998.
- [2] WANG Zhiliang, CHEN Fengjun, XUE Weimin. A survey of facial expression on recognition[J]. Computer Applications and Software(计算机应用与软件), 2003 (12): 63-66.

(下转第 42 页)

- 究[J]. 微电子学与计算机, 2010, 27(4): 66—74.
- [2] 刘颖, 翁健杰, 戎蒙恬. 用改进激活集合法优化 VLSI 互连线[J]. 微电子学与计算机, 2004, 21(12): 203—207.
- [3] Cong J, Pan D Z. Interconnect estimation and planning for deep submicron designs[C]// Proceedings of the 36th ACM/IEEE Design Automation Conference. New Orleans, LA: ACM/IEEE, 1999: 507—510.
- [4] L P P P van Ginneken. Buffer placement in distributed RC-tree networks for minimal elmore delay[C]// IEEE International Symposium on Circuits and Systems. New Orleans; IEEE, 1990: 865—868.
- [5] Alpert C J, Devgan A. Wire segmenting for improved buffer insertion[C]// Proceedings of the 34th IEEE/ACM Design Automation Conference, New York: IEEE, 1997: 588—593.
- [6] Lillis J, Cheng C K, Lin T T Y. Simultaneous routing and buffer insertion for low power and a generalized delay model[J]. IEEE Journal of Solid-State Circuits, 1996, 31(3): 437—447.
- [7] Alpert C J, Gandham R G, Neves J L, et al. Buffer library selection[C]// Proceedings of the 18th IEEE International Conference on Computer Design. Austin: IEEE, 2000: 221—226.
- [8] Brendan J Frey, Delbert Dueck. Clustering by passing messages between data points[J]. Science, 2007, 315(5814): 972—976.

作者简介:

苏晓东 男, (1984—), 硕士研究生. 研究方向为高性能标准单元库设计.

王静秋 女, (1979—), 硕士, 助理研究员. 研究方向为单元库设计.

马 鸿 女, (1981—), 博士. 研究方向为 VLSI 设计自动化.

(上接第 36 页)

- [3] 李楠, 喻明艳. 16×16 快速乘法器的设计与实现[J]. 微电子学与计算机, 2008, 25(4): 156—159.
- [4] 李军强, 李东生, 李奕磊, 等. 32×32 高速乘法器的设计与实现[J]. 微电子学与计算机, 2009, 26(12): 23—26.
- [5] 陈博亚. 大邻域图像处理系统的研制[D]. 北京: 清华大学, 2006.
- [6] Lyons M J, Budynek J, Akamatsu S. Automatic classification of single facial images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21(12): 1357—1362.
- [7] Yan Xiang, Su Guangda. Multi-parts and multifeature fusion in face verification[C]// Proceeding of IEEE CVPR Workshop on Biometrics. Anchorage: IEEE, 2008: 1—6.
- [8] 苏光大. 图像并行处理技术[M]. 北京: 清华大学出版社, 2002.

作者简介:

丁 亮 男, (1985—), 硕士研究生. 研究方向为数字集成电路设计.

苏光大 男, (1948—), 教授. 研究方向为图像处理系统人脸识别.