

图像处理中数据复用及存储层次设计的研究

魏武, 杨靓

(西安微电子技术研究所, 陕西 西安 710065)

摘要: 图像处理过程中数据传输量大, 通过数据复用可以减少数据传输量提高执行效率。同时图像分块的大小在很大程度上决定了存储器大小, 而存储器大小对系统的功耗影响巨大。文中通过对分块图像平滑算法的研究, 提出了四种数据复用层次。再以 TS101 为仿真平台, 在不同的分块大小下编写汇编代码模拟数据流在不同存储器间的流动, 然后得出其执行效率、存储器大小和数据复用层次、块大小等的关系。为硬件设计中存储器体系结构的优化设计提供了参考。

关键词: 复用层次; 块大小; 存储层次; 汇编代码

中图分类号: TP31

文献标识码: A

文章编号: 1673-629X(2012)12-0043-04

Data Reuse and Memory Hierarchy Design in Image Processing

WEI Wu, YANG Liang

(Xi'an Micro Electronics Institute, Xi'an 710065, China)

Abstract: There was large amount data transfer in image processing. It could reduce data transmission quantity and improve the efficiency by using data reuse. And block size largely determined the memory size which had a great influence on system consumption. The four data reuse levels were presented based on research of the block image smoothing algorithm and various block methods in this paper. Finally simulations were done on the platform of TS101 in assembly language and different block sizes. Then the relationship between efficiency and memory size was obtained as well as the relationship between data reuse level and block size. It provided a reference design for the optimization of the memory architecture in hardware design.

Key words: data-reuse level; block sizes; memory hierarchy; assembly language

0 引言

随着 IC 技术的发展及其制造成本的降低, SOC (System on Chip: 片上系统) 以体积小、功耗低、速度快等优点被广泛应用于图像处理领域。但图像处理过程中计算复杂、吞吐量大, 因此对图像处理算法进行优化, 发掘算法内部的可复用性, 减少数据的读取和存储次数, 进而减少总的访存数量, 对于提高系统的运行速度至关重要。然而随着数据复用性的增加, 所需存储器大小也要增加。而存储器功耗主要由其访问频率和存储器大小决定^[1], 因此如果在片上仅仅采用单一的大容量存储器来存储数据, 则存储效率低、读写缓慢、资源浪费大而且功耗大。

因此建立存储层次, 采用多层存储结构, 最大程度地使可复用的数据从小容量存储器来访问^[2]不仅可以加快访问速度、节省资源, 而且可以降低功耗。但从另

一个方面来讲, 由于引入了小容量的存储器, 增加了以前不存在数据传输, 从而增加总的传输量。

这样需要进一步研究算法复用性来减小数据传输量, 但复用性的提高又会增大所需存储容量。因此, 各级存储器容量和数据传输量之间互相矛盾、对立, 两者为此消彼长的关系, 从原因上来讲也就是数据复用层次和存储器大小之间互相约束。所以在进行硬件设计的时候, 必须在它们之间做一个折中。

文中以分块的图像平滑算法为例子, 详细地分析计算过程中涉及到数据传输的两个步骤, 提出了可应用于各个步骤的复用层次。并计算出在不同的块大小各个等级所需最小存储器的大小。最后在 TS101 平台下进行仿真, 为了可以更加清楚地了解各条指令的执行情况, 全部仿真程序均用汇编代码写成, 得出执行周期和数据复用及块大小之间的关系, 对于相关硬件的设计提供了一些参考。

收稿日期: 2012-02-15; 修回日期: 2012-05-18

基金项目: 中国航天集团课题预研项目 (417010204)

作者简介: 魏武 (1986-), 男, 硕士研究生, 主要研究方向为计算机系统结构、数字电路设计; 杨靓, 研究员, 硕士生导师, 主要研究方向为嵌入式高速计算机系统结构、ASIC 设计、数字图像处理。

1 算法研究及层次划分

1.1 相关概念及研究现状

数据复用主要是指通过对数据进行缓存来减少对

图 2 下方表示的为 B 复用。点 3 和点 4 为同一列相邻行的两点,它们的运算矩阵公用浅色部分,大小同样为 $N \times (N - 1)$ 。因此在计算点 4 时,只需要读入下边深色部分的 $1 \times N$ 个像素点。

同样的,复用 C 和 D 是针对不同 Block 之间的,如图 3 所示。

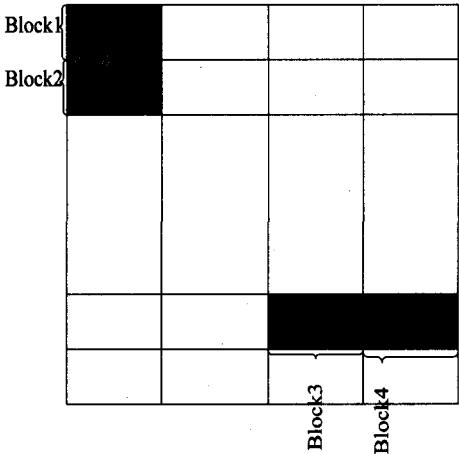


图 3 块间复用

图中左上角所示的为 C 复用。Block1 和 Block2 为所占相同列但行相邻的块。当计算 Block1 中最后两行时,均需要读取 Block2 中前两行的像素点;而当计算 Block2 中第一行时,需要读取 Block1 中最后一行的像素点,所以这三行像素点在两个 Block 的计算中均要用到。因此在 Block1 的计算阶段就直接将这三行读入并缓存,当两个块在行方向相邻称之为 C 复用;同样的,在两个列方向相邻称之为 D 复用。

2 仿 真

整个仿真过程采用 512×512 大小的图像。对原始图像的加噪声、变换为 8 位灰度图像和输出图像矩阵文件均在 Matlab^[9,10] 下进行。然后利用已经生成的图像矩阵进行仿真^[11]。

由于 Visual Dsp++ 可以统计出各个指令的执行周期数,因此用传输周期数来表征传输量大小。由于第一级存储器为 SDRAM,主要用于放置源图像和处理之后的图像,大小固定为 2MB,因此仿真结果不再列出。

此测试仿真程序由四个函数构成:

 _Main(): 主函数,负责外部循环和其他三个函数的调用;

 _Compute(): 主要负责乘累加计算及二级到三级的传输;

 _Dma0(): 读入图像数据,通过 DMA 方式进行一级到二级的传输;

 _Dma1(): 输出图像数据,通过 DMA 方式进行二级到一级的传输,总传输量固定。

以上程序均由 TS101 汇编编写^[12],这样可以详细地了解各条指令执行耗费的周期数,进而研究不同的分块大小、数据复用层次对整个程序运行的影响。另外,由于_Compute()和_Dma0()、_Dma1()中的寄存器有重叠,因此 DMA 均采用查询方式返回主程序。且如果读缓冲采用乒乓缓冲^[13],考虑到数据复用,地址可能在两个缓冲之间来回变换,地址运算代价太大,所以只采用单缓冲。分为五种方案来模拟特定的硬件条件及算法要求下程序的运行情况,如表 1 所示。

表 1 仿真结果

项目\类型	方案一	方案二	方案三	方案四	方案五
块大小	8×512	8×512	8×512	8×16	8×7
复用层次	无	A	A,C	A,C	A,B,C
二级存储器大小	304KB	304KB	416KB	15KB	7552B
三级存储器大小	1KB	1408B	1408B	1408B	1792B
执行总周期	9963669	4974764	5884124	7608210	7100427
总读取周期	5024540	1042635	1241805	1684386	1065841
读取占执行总比例	50.43%	20.96%	21.10%	22.13%	5.01%
Compute 执行周期	9334314	4345413	5350721	6153386	5698594
二级到三级传送周期	4664476	682571	979147	1351904	632253
读取方式	四字	四字	单字	单字	单字
占 Compute 的比例	49.97%	15.71%	18.30%	21.97%	11.09%
Dma0 执行周期	362884	362884	266226	475204	759170
一级到二级传送周期	360064	360064	262658	332482	433588
占 Dma0 的比例	99.20%	99.20%	98.66%	69.97%	57.11%
数据冗余率 R_{a_2}	1.37	1.37	1	1.18	1.42
数据冗余率 R_{a_3}	15.81	3.98	3.98	4.97	2.28

从以上仿真结果可以看出:

(1) 总执行周期由复用层次和分块大小共同决定;复用层次越高,分块越大,则执行总周期越小,但所需内部存储空间越大。

(2) 如图 4 所示,通过对比 Dma0 的执行周期、一级到二级的传输周期和占 Dma0 百分率可以看出:

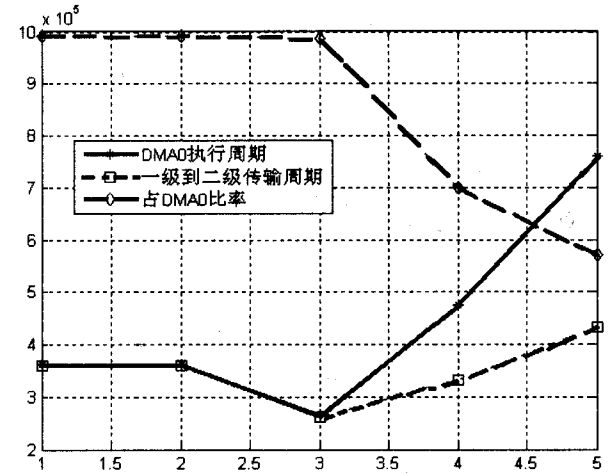


图 4 Dma0 执行情况

①一级到二级的传输周期:一和二均没有数据复用,三采用了数据复用 C 后传输周期迅速下降至最低点,四由于分块减小 8×16,只是同列相邻的两个块之

间 C 复用,所以传输周期上升,方案五类似;

②执行周期:同传输周期的变化基本一致,只是随着块的减小,执行周期迅速增加,且增速远大于传输周期的增速。这是因为随着分块的减小,Dma0 执行的次数越来越多,内部的循环、跳转和地址运算等指令的执行总次数大大的增加。

③占 Dma0 的比例:此比例反映了 Dma0 程序的执行效率。可以看出一、二和三传输周期占 Dma0 的比例均在 98% 以上,但当块变为 8×16 后比例迅速下降至 70% 左右,随着块继续减小至 8×7 ,比例下降至 50% 左右,其原因同对执行周期的分析。从对比可以看出随着分块减小,所需存储器大小减小很快,但同时程序的执行效率会变得很低。

④数据冗余率 Ra_2 :表征了理想的一级到二级的传输周期,同数据复用层次 C、D 和块大小有关。一和二相同,均为 1.37,三达到最小为 1,实现了一次读取,四和五由于只是同列相邻的块间 C 复用,所以数据冗余率开始上升。

(3)可以看出 Dma0 程序的执行时间和执行效率等由复用层次和分块大小共同决定。分块越大、复用层次越多,程序执行时间越短、效率越高,但需要的二级存储器即 SRAM 大小也越大;另外分块大小对程序的执行效率起主要作用,随着分块减小,效率降低很快。因此,在硬件设计时需要在存储器大小、分块大小和复用程度上做一个合理的分配,使程序不仅运行速度快而且资源消耗小。

(4)如图 5 所示,通过对比二级到三级的传输周期、Compute 的执行周期和占 Compute 百分率可以看出:

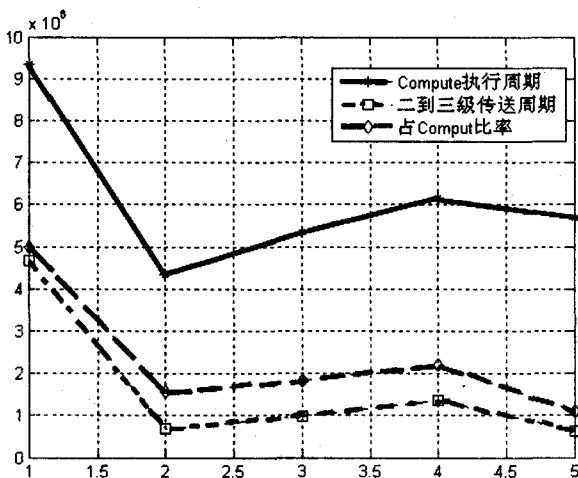


图 5 Compute 执行情况

①二级到三级的传输周期:一时为最大,采用复用层次 A 后,二时传输周期迅速下降;三时虽然块大小没变,且仍然采用复用层次 A,但传输周期有所上升。这是由于 TS101 的数据总线为 128 位,如果存储均四

字对齐,一次可以读取四字。二时存储器刚好满足,但三时为不规则存储,一次只能读取一字,导致传输周期上升。四时由于分块减小,每个 Block 内一行点数减少,A 复用效果下降,传输周期上升。五时虽然块大小进一步减小,即使 A 复用效果继续下降,但增加了复用层次 B,却促使传输周期下降至最低水平,达到了最小传输。

②Compute 的执行周期:同二级到三级的传输周期变化基本一致,只是块减小后执行周期上升比较剧烈,远大于传输周期的增加量。其原因同 Dma0 程序类似,因为块减小,其内循环、跳转以及地址运算等指令执行总次数大大增加。

③占 Compute 比例:其变化趋势同二级到三级的传输周期近似,且在五时达到最小,为 11.09%。此时传输只占 Compute 执行周期的很少部分。

④数据冗余率 Ra_3 :表征了理想的二级到三级数据传输周期,同数据复用层次 A、B 和块大小有关。五时由于两种复用均有,虽然块大小较小,仍然达到了最小冗余率 2.28,接近了一次读取。

3 结束语

从以上仿真可以看出,分块大小和复用层次影响程序的执行效率、传输数据量和所需内部存储器大小。复用层次越多,分块越大,执行效率越高,总传输量越小,但所需存储器越大。因此在进行硬件设计时,必须综合考虑数据传输量和存储器大小,在两者之间做一个折中优化的选择,才能在节省资源的同时又兼顾到效率。

参考文献:

- [1] Wuytack S, Cathoor F, Franssen F, et al. Global communication and memory optimizing transformations for low power systems[J]. VLSI Signal Processing, 1994(10): 178-187.
- [2] van Achtern T, Lauwereins R, Cathoor F. Systematic data reuse exploration methodology for irregular access patterns [C]//Proceedings of the 13th International Symposium on System Synthesis. Washington: IEEE Computer Society, 2000: 115-121.
- [3] Diguett J P, Wuytack S, Cathoor F, et al. Formalized methodology for data reuse exploration in hierarchical memory mappings[J]. Low Power Electronics and Design, 1997(8): 30-35.
- [4] van Achtern T, Cathoor F. Data Reuse Exploration Techniques for Loop-dominated Applications [C]//5th ACM/IEEE Design Test Europe Conf. [s. l.]: [s. n.], 2002: 428-435.
- [5] Tuan Jen-Chieh, Chang Tian-Sheuan, Jen Chein-Wei. On the

(下转第 50 页)

模块,主要有人脸特征训练、人脸图像存储、人脸识别、数据传输和空闲处理等。如果接收到 stop 命令,主程序清除工作环境后退出,否则返回人脸检测模块重复执行。

4 系统结果及分析

本系统的空间复杂度如表 1,可以看出系统需求的总内存大小为 $(0x1eafcc + 0x8ee8 * N)$,其中, N 为系统所支持的最大人脸数,受限于系统的 flash 大小。

表 1 系统空间复杂度

Content	Memory (Bytes)	Content	Memory (Bytes)
Program	0x16e20	Haar lib	0x440e8
Stack	0x2000	EHMM	$0x1b68 * N$
Heap	0x140400	Face lib	$0x7380 * N$
Table	0x1dc4	I/O buffer	0x4b00

在设定人脸区域为 $320 * 240$ 像素大小的情况下,系统执行一次人脸检测需要 60M cycles,在 DM3730 工作在 800MHz 的主频下,系统每秒可以对 13 帧图像进行人脸检测。人脸识别的时间复杂度为 30M cycles/face,对于包含有 100 个人脸的库,本系统在 4 秒内就可以识别结束,满足实时性检测的要求。

5 结束语

文中设计了基于 DM3730 的实时人脸识别系统,系统包含了视频的采集、显示、人脸检测、EHMM 模型训练和人脸识别,同时支持人机交互、有线和无线数据传输等功能。系统支持 NTSC、PAL 和 SECAM 制式的视频信号,构造了包含 7 个超级状态、36 个子状态的人脸 EHMM 模型,这种模型更能有效地表征人脸。在正常光照条件下,采集正面人脸时,系统的人脸识别准

确率达到 95%,对于有 100 张人脸的库,识别时间小于 4 秒,满足实时性应用的要求。

参考文献:

- [1] 赵秀萍. 生物特征识别技术发展综述[J]. 刑事技术, 2011(6): 44-48.
- [2] 董琳, 赵怀勋. 人脸识别技术的研究现状与展望[J]. 安防科技, 2011(10): 22-26.
- [3] 祝秀萍, 吴学毅, 刘文峰. 人脸识别综述与展望[J]. 计算机与信息技术, 2008(4): 53-56.
- [4] 王映辉. 人脸识别: 原理、方法与技术[M]. 北京: 科学出版社, 2010.
- [5] Freund Y, Schapire R E. A Decision-theoretic Generalization of On-line Learning and an Application to Boosting[J]. Journal of Computer and System Sciences, 1997, 55(1): 119-139.
- [6] 陈伟琦, 梁一川, 易虎, 等. 基于肤色和 AdaBoost 算法的人脸检测研究[J]. 计算机技术与发展, 2008, 18(12): 44-46.
- [7] 朱伟冬, 胡剑凌. 基于马氏距离的稀疏表示分类算法[J]. 计算机技术与发展, 2011, 21(11): 27-30.
- [8] Nefian A, Hayes M H. Maximum Likelihood Training of the Embedded HMM For Face Detection and Recognition[C]// Proc. of Int'l Conf. on Image Processing. [s. l.]: [s. n.], 2000.
- [9] 严严, 章毓晋. 基于视频的人脸识别研究进展[J]. 计算机学报, 2009, 32(5): 878-886.
- [10] Viola P, Jones M J. Robust Real-time Face Detection[J]. International Journal of Computer Vision, 2004, 57(2): 137-154.
- [11] Texas Instruments. TPS65930/20 and OMAP3530 Hardware Connectivity[M]. [s. l.]: Texas Instruments, 2008.
- [12] Texas Instruments. TPS65930/TPS65920 Integrated Power Management\Audio Codec (TPS65930 Only)[M]. [s. l.]: Texas, 2008.

(上接第 46 页)

- Data Reuse and Memory Bandwidth Analysis for Full-search Block-matching VLSI Architecture[J]. IEEE Transaction on Circuits and Systems for Video Technology, 2002(1): 61-72.
- [6] Hennessy J L, Patterson D A. 计算机系统结构-量化研究方法[M]. 北京: 电子工业出版社, 2007.
- [7] 刘书明. ADI DSP 应用技术集锦[M]. 北京: 电子工业出版社, 2009.
- [8] Panda P R, Dutt N D, Nicolau A. Efficient Utilization of Scratch-pad Memory in Embedded Processor Applications[C]//EDTC '97 Proceedings of the 1997 European Conference on Design and Test. [s. l.]: [s. n.], 1997.

- [9] 冈萨雷斯. 数字图像处理(MATLAB 版)[M]. 北京: 电子工业出版社, 2005.
- [10] 于方波. 基于 MATLAB 的图像处理[M]. 第 2 版. 北京: 清华大学出版社, 2011.
- [11] 宋淑娜. 一种自适应模糊阈值区间的图像分割方法[J]. 计算机技术与发展, 2010, 20(5): 65-67.
- [12] 陈峰. Blackfin 系列 DSP 原理与系统设计[M]. 第 2 版. 北京: 电子工业出版社, 2010.
- [13] 雷娟琳. 一种基于 H264 的改进的运动估计三步搜索算法[J]. 计算机技术与发展, 2010, 20(12): 110-113.