

传像光纤束成像算法在 DM642 上的实现与优化

潘胜达 安博文*

(上海海事大学, 上海 200135)

摘要 针对异型传像光纤束成像的算法结构, 提出一种基于 DM642 的异型传像光纤束成像算法的实现与优化方法。该方法结合 DM642 的硬件结构和汇编指令执行流程, 并针对这两个方面提出了相应的优化策略。实验结果表明, 经过优化的系统在速度和效率方面均得到了明显的提高, 可以满足成像算法的实际需求。

关键词 传像光纤束 成像算法 DM642 优化

中图分类号 TN919.8; **文献标志码** A

随着传像光纤束成像技术的快速发展及其相应的硬件成像平台性能的不不断提高, 在嵌入式系统中实现基于传像光纤束的全实时超分辨率成像已经成为现实。采用高性能的数字信号处理平台, 不仅可实现基本的图像采集和图像输出, 更可以通过软件算法提高输出图像的质量。DM64x 是 TI 公司开发的高性能 DSP, 其强大的数据处理能力和通道传送能力非常适合处理数据量大、传输速度高的传像光纤束成像算法处理^[1]。本文针对 DM642 的硬件结构特点及其指令执行流程并结合光纤图像处理中的具体问题, 提出了算法的优化设计方案。

1 像算法的实现

整个光纤束成像算法的主体思想是: 先采集经过异形传像光纤束耦合传感器实现线阵至面阵转换而得的面阵形式的光纤束图像数据, 然后在 DSP 中将采集的面阵图像转换成线阵, 之后通过连续的水平扫描, 将各幅线阵图像拼接成一幅完整的扫描图像^[2]。

由于现实中一根光纤只代表实际图像中的一个像素, 而在转换而来的面阵图像中一根光纤是由一个大约为 9×9 大小的图像块来表示的(图像的总分辨率为 2592×1944), 因此需要将光纤所包含的实际图像信息抽取出来, 即提取出每根光纤图像块中真正包含图像信息的点, 这个过程成为光纤中心位置的定标。DM642 主要完成的任务就是该定标过程^[3]。整个定标算法的流程如下:

首先, 采用双模板极值滤波器对光纤中心进行粗略定位。所谓双模板滤波器就是指该滤波器由两级模板构成, 大模板中的元素不是单个像素, 是由多个像素构成的小模板。依照光纤的物理特性, 本系统中滤波器的大模板和小模板皆为 $[3 \times 3]$ ^[4]。其各自结构图如图 1 及图 2 所示。

$$\begin{bmatrix} [M_0] & [M_1] & [M_2] \\ [M_3] & [M_4] & [M_5] \\ [M_6] & [M_7] & [M_8] \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

图 1 大模板结构图

图 2 小模板结构图

通过先对小模板内的各个像素进行求和, 求得各个小模板的亮度和, 然后通过大模板对各个小模板的亮度和进行比较, 如果当前小模板的亮度和为 9 个小模板中最大者, 且 9 个小模板各自的亮度和的最小值大于某一阈值, 则认定该小模板位于光纤中心。经过滤波运算后可将每根光纤的中心确定于大约为 5×5 的图像块范围中。光纤出端部分结构局部放大图及其经过双模板极值滤波器处理后

2010 年 3 月 24 日收到

国家高科技研究(863 计划)
(2007AA12Z152)资助

第一作者简介: 潘胜达(1986—), 男, 上海人, 上海海事大学硕士研究生, 研究方向: 基于嵌入式的图像处理。

* 通信作者简介: 安博文(1969—), 男, 湖南黄冈人, 上海海事大学副教授, 博士, 研究方向: 遥感图像处理。

的图像如图3及图4图所示。

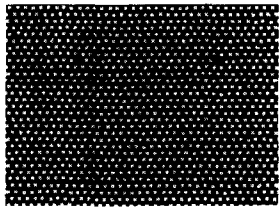


图3 光纤束面阵出端
结构图

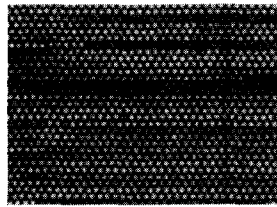


图4 双模板滤波器
处理结果图

接着对上一步中得到的图像进行腐蚀和裁边技术处理。

之后,对亮块进行Z字型搜索法,即采用斜45°顺序搜索第一个根光纤的坐标。其算法示意图如图5所示。搜索到第一个光纤亮块后,用该亮块的几何中心位置代表光纤的中心位置,由此得到第一根光纤的坐标,之后利用光纤的先验知识,即光纤之间的距离为某一个阈值(9个像素左右)进行步进搜索,由此可以得到整个面阵光纤图像中所有光线的位置坐标表。

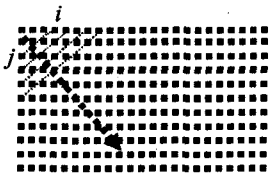


图5 Z字型搜索法示意图



图6 光纤中心亮点图

最后,按照位置坐标表对每一幅扫描的面阵光纤图像进行图像信息抽取,将光纤束出端的一帧图像还原成原始目标图像中的一列并进行拼接,实现空间变换。

2 硬件结构级优化

DM642的片内采用两级Cache结构,第一级由L1P和L1D组成,数据访问只需一个时钟周期,其中L1P大小为16kB,L1D大小16kB,第二级L2是一个统一的程序/数据空间,容量较大有256kB,可以按一定比例配置为L2Cache和SRAM。访问速度较慢,根据L2配置为Cache或SRAM的不同需8个或6个时钟周期^[5,6]。

由于Cache在获取数据的时候会按照空间局部性原则将该数据周围的一部分数据一并读取到Cache中,因此在所处理的数据的空间相关性比较大的情况下,增大Cache可以很大程度上提高CPU的命中率,性能可得到明显地提升,但同时也会减小片内实际的可用空间^[7]。

系统中由于算法运行时,处理一行图像时需要相邻9行即 9×2592 bytes大小的数据参与运算,而且由于算法的需要还需在L2SRAM中开辟一个约为 3×2592 bytes的缓冲区用于存放已处理完并将要回写至外部存储器的数据。除此之外为了提高L1P的命中率,减少CPU等待时间还将常用的核心代码常驻于L2SRAM中。但综合另一方面考虑到算法中数据相关性比较大,最终决定将L2分为128kB SRAM和128kB L2Cache。

由于上述Cache分配策略决定每次读取一帧图像的9行,因此算法处理中会产生大量的数据块的搬移操作。

EDMA与QDMA都是C6000中搬移数据效率最高的手段之一。QDMA申请一般经过5个周期后就可以被真正地发出。与此相比EDMA的申请需要在36个周期后才能被发出。但EDMA比QDMA多一个外部事件触发,因此QDMA更适合完成CPU提出单独的突发性传输控制命令,以便快速移动数据,而EDMA则适合执行外部设备的周期性或重复性数据传输,如Vport口的数据传输。

结合Cache的分配策略和EDMA与QDMA的传输特性,本系统中采用一种基于QDMA的Ping-Pong缓冲处理方式。

其主要流程为:在算法处理之前先提交一个将处理的数据块拷贝至Ping输入缓冲区的QDMA请求,等到数据拷贝完成之后再提交一个将下一部分需处理的数据块拷贝至Pong输入缓冲区的QDMA请求,同时DSP核开始对Ping输入缓冲区的图像数据进行处理,由于QDMA传输数据的速度较CPU处理数据的速度快得多,因此等到DSP核处理完Ping输入缓冲区的图像数据并将处理完的图像输出于Ping输出缓冲区时,Pong输入缓冲区的图像数据就已经准备

完毕, DSP 核就可以接着处理 Pong 输入缓冲区的数据, 同时再提交先将处理完的 Ping 输出缓冲拷贝至外部 SDRAM 再将下一次所需图像数据拷贝至 Ping 输入缓冲区的 QDMA 请求, 如此循环完成 Ping_Pong 缓冲图像处理模式。其结构图如图 7 所示:



图 7 Ping_Pong 缓冲结构示意图

通过这种方式, CPU 每次调用所需图像数据时, 图像数据都已经存放在片内 RAM 的指定缓冲区中, 可以有效提高 Cache 的命中率, 减少 CPU 堵塞时间并节省了从片外 SDRAM 读取图像数据的时间开销。

3 代码级优化

在整个算法过程中, 由于双模板极值滤波器算法每计算一个像素点就需要对该像素相邻的 9×9 的像素矩阵进行计算加上图像分辨率较大, 从而导致其运算量极为庞大, 占整个算法的 60% 以上, 因而将其作为主要的优化对象。

虽然 DM64X 特有的 LDNDW 指令支持双字节读取, 但由于该算法运行一次需要 9×9 个像素参与, 且 DM64X 只支持单字节运算, 因此每对一个像素进行算法处理就需要进行 2×9 次以上双字节读操作, 这将严重的降低算法的运行效率较低, 因此考虑将两个相邻 9×9 矩阵的运算合并在一起进行以提高读操作的利用率^[8]。具体流程如下:

首先, 对相邻的九行数据分别使用两次 LDNDW 指令分别读取每一行中 $d_{i \times j}$ 到 $d_{i \times (j+7)}$ 和 $d_{i \times (j+2)}$ 到 $d_{i \times (j+9)}$ (i 为行, j 为列) 两个双字大小的图像数据, 以下分别简称 DW_{i1}, DW_{i2} 。该操作结构如下所示:

$$\left\{ \begin{array}{l} \{ \{ DW_{01} d_{00}, d_{01}, \{ DW_{02} d_{02}, d_{03}, d_{04}, d_{05}, d_{06}, d_{07}, \} \}^{DW_{01}}, \}^{DW_{02}} \\ \{ d_{08}, d_{09} \\ \vdots \\ \{ DW_{81} d_{80}, d_{81}, \{ DW_{82} d_{82}, d_{83}, d_{84}, d_{85}, d_{86}, d_{87}, \} \}^{DW_{81}} \\ d_{88}, \}^{DW_{82}} \\ d_{89} \}^{DW_{82}} \end{array} \right.$$

这样的话就可以通过对 DW_{i1}, DW_{i2} 的各个单字进行操作得到这两 9×9 模板中任意一个小模板的和。例如左边的 9×9 矩阵中的 M_0 和 M_1 模板可表示如下:

$$M_0 = \begin{pmatrix} d_{00} & d_{01} & d_{02} & d_{03} \\ d_{10} & d_{11} & d_{12} & d_{13} \\ d_{20} & d_{21} & d_{22} & d_{23} \\ d_{30} & d_{31} & d_{32} & d_{33} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \quad (2)$$

$$M_1 = \begin{pmatrix} d_{02} & d_{03} & d_{04} & d_{05} \\ d_{12} & d_{13} & d_{14} & d_{15} \\ d_{22} & d_{23} & d_{24} & d_{25} \\ d_{32} & d_{33} & d_{34} & d_{35} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} \quad (3)$$

以上这两个矩阵操作都可以通过对 DW_{i1}, DW_{i2} 的低位字使用 DOTPSU4 点积指令来完成。

接着, 由于一个 3×3 单像素为 8 bit 大小的图像矩阵的像素和可由一个 16 bit 数来表示, 因此可以将两个 9×9 图像矩阵的相同位置的模板的像素和通过 DM642 的打包指令 PACK2 组合成一个 32 位数, 并由一个 32 位寄存器来存放。

最后, 通过对该 32 位寄存器进行算数指令操作就可以同时对两个模板进行算数处理。

3 试验结果

由于实际图像分辨率较大, 处理十分缓慢, 不利于优化效果的测试。因此在测试优化效果时, 每次测试运行 9×2592 行, 即每次处理一行数据, 并通过多次测量并取测量结果的平均值。优化性能结果如表 1 中所示(已对源程序进行文件级优化 -O3)。

表 1 优化效果对比(单位: Cycles)

优化方法	C 语言未优化	线性汇编
平均消耗时间	1 109 647	38 612

试验证明, 由于线性汇编代码剔除了 C 语言代码中函数调用所消耗的 Cycles, 并针对算法作出了优化, 因而在性能上取得了较大的改进。由此可见以上优化方法是可行的。

4 结论

本文通过对在 DM642 上实现的异形传像光纤束成像算法进行硬件结构级和代码级的优化,使之能更好地适合成像算法的实现,提高了系统的运行效率,解决了算法处理中存储器与 CPU 之间数据传送的速度问题和指令效率较低等问题,实现了实时成像。

参 考 文 献

- 1 李方慧,王 飞,何佩琨. TMS320C6000 系列 DSPs 原理与应用. 北京:电子工业出版社,2003
- 2 朱 翔,方中华,孙胜利. 光纤传像系统中的耦合技术研究. 红外技术,2006;28(5):257—260
- 3 安博文. 超高分辨率异型传像光纤束中的空间变换. 光电工程,2006;33(11):83—87
- 4 安博文. 基于双模板极值滤波器的传像光纤束中的盲元消除. 中国图象图形学报,2007;12(9):1555—1557
- 5 Texas Instruments. TMS320C6000 DSP Cache User's Guide. <http://www.ti.com>,2003
- 6 Texas Instruments. TMS320C64x DSP Two-Level Internal Memory Reference Guide. <http://www.ti.com>,2004
- 7 张永光,张晓蕾,徐健健. DM642 图像数据传输的实现和优化. 科学技术与工程,2007;7(17):1671—1819
- 8 TMS320C6000 CPU and Instruction Set Reference Guide. Texas Instruments Incorporated,2000

Implementation of Imaging Algorithm of Image-carrying Fiber Bundles on DM642

PAN Sheng-da, AN Bo-wen*

(Shanghai Maritime University, Shanghai 200135, P. R. China)

[Abstract] An implemented and optimized method of the imaging with non-conventional image-carrying fiber bundles based on DM642 is presented for the special structure of the imaging algorithm. The method combined with the hardware structure and the instruction process flow of the DM642 and present the corresponding optimizing strategies from these two aspects. Experimental results show that the speed and efficiency of the system have been remarkably improved, which could meet the real-time requirement of the imaging algorithm.

[Key Words] image-carrying fiber bundles imaging algorithm DM642 optimization

(上接第 4174 页)

A New Method to Improve the Performance of Voice Spectrum Transformation Based on GMM

ZHAO Yi-zheng

(702 Studio of Hefei Electronic Engineering Institute, Hefei 230037 P. R. China)

[Abstract] GMM is one of the most successful algorithms in the applications of voice conversion. However GMM causes the overly smoothing of voice spectrum which affects the transformed voice's quality. The overly smoothing problem is analyzed and proposed a new method named mean value substitution. Experiment results show that this method could reduce the affect of overly smoothing phenomena, and improves the converted voice's quality and hearing perception characteristic.

[Key word] GMM spectrum transformation overly smoothing