

在 VLIW 处理器上实现视频图像的运动估计算法的方法

谷俊丽, 何 虎, 孙义和
(清华大学 微电子研究所, 北京 100084)

摘 要: 利用超长指令字(VLIW)处理器处理单指令多数据(SIMD)的优势, 采用加速 SIMD 指令计算的数据拼接方案和多方向并行搜索方法以及适合图像数据复用的插值图像存储结构, 优化实现了一个高效的 ME 软硬件结合的架构, 分别在 TMS320C64 $\times\times$ 及自行设计的 LILY Processor 上实现了 H.264 的 QCIF 图像数据的搜索及 H.263 的 CIF 图像数据的搜索. 测试实验表明 ME 的搜索速度提高了 3 倍到 15 倍.

关键词: 运动估计; VLIW; SIMD; 数据准备; 图像重组; 图像数据复用

中图分类号: TN919.8

文献标识码: A

文章编号: 1000-7180(2010)06-0005-03

Implementation Methods of Video Motion Estimation Algorithms on VLIW Processors

GU Jun-li, HE Hu, SUN Yi-he

(Institute of Microelectronics, Tsinghua University, Beijing 100084, China)

Abstract: We utilized the advantages of single-instruction-multi-data (SIMD) techniques of very-long-instruction-word (VLIW) processors, adopted data packing methods to speedup SIMD instructions and multi-direction parallel searching technique as well as the storage format of interpolated pictures suitable for picture data reuse and finally proposed an effective implementation of ME algorithm based on HW/SW combined framework. We implemented H.263 and H.264 ME algorithms on TI's TMS320C64 $\times\times$ and LILY Processor developed by Institute of Microelectronics of Tsinghua University. Experimental results show the searching speed of ME rises to 3x~15x.

Key words: ME; VLW; SIMD; data preparation; picture reorganize; picture data reuse

1 引言

运动估计(ME)算法依靠消除图像间的空间冗余提高了视频编码的压缩性能, 已被广泛用在 AVS、MPEG-4、H.264^[1]、H.263^[2]等多种视频编码标准中. H.263 中 ME 占整个计算量的 50%~60%, H.264 中单参考帧 ME 占全部计算量的 60%, 多帧参考 ME 则占了整个计算量的 80%^[3]. ME 导致了编码复杂度的急剧增加, 对处理器也提出了越来越高的性能要求^[4].

为了多媒体应用满足越来越高的计算性能需求, 很多数字信号处理器采用了超长指令字^[5](VLIW)和单指令多数据^[6](SIMD)技术. VLIW 技

术在一个周期内发射并行执行数条指令, 能获得很高的指令并行度; SIMD 技术在一个周期内完成一组数据的并行计算, 能充分利用数据并行性. 现有文献提出的基于 DSP 的实现大多侧重于存储空间、数据调度等系统级的优化, 模块实现则大多采用编译器自动生成的代码, 导致程序本身的执行效率并不高. 或者优化技术仅局限于 H.264 的某个搜索方法进行优化没有普适性^[3]. 文中根据 ME 算法映射到 VLIW 处理器时存在的典型问题, 提出了适用于多种搜索算法实现的优化技术. 结合 ME 算法和 VLIW 处理器的特点, 提出了一些加速 SIMD 指令执行和复用图像数据的优化技术, 使得 ME 的计算获得 3~15 倍的加速比.

2 SIMD 指令加速

本节提出了一种针对 SIMD 指令加速的数据拼接方案,并通过插值图像重组改变了不利于 SIMD 指令执行的据组织存放方式.

2.1 数据拼接

ME 计算中要同时读取源图像数据和参考图像数据,由于参考图像子块的起始地址在搜索过程中可能是任意情况导致了数据地址不对齐,成为影响 ME 计算效率的关键问题.对于不支持非对齐读数的

处理器,只能采用字节读取计算或者通过打包指令进行数据拼接后再利用 SIMD 指令计算,由于数据准备跟不上导致计算性能很低.为了解决这个问题,文中设计了如下的数据拼接方案.以 16×16 块的 SAD 计算为例,每次冗余读取参考图像对应行中的 20 个字节分别放入 5 个寄存器中,移除不需要的 4 个字节,把剩余的 16 个数按照在图像中的自然顺序对齐放入 4 个寄存器中,同时读取源图像的数据分别放在 4 个寄存器中,可以直接采用 SIMD 指令进行计算.如图 1 所示.

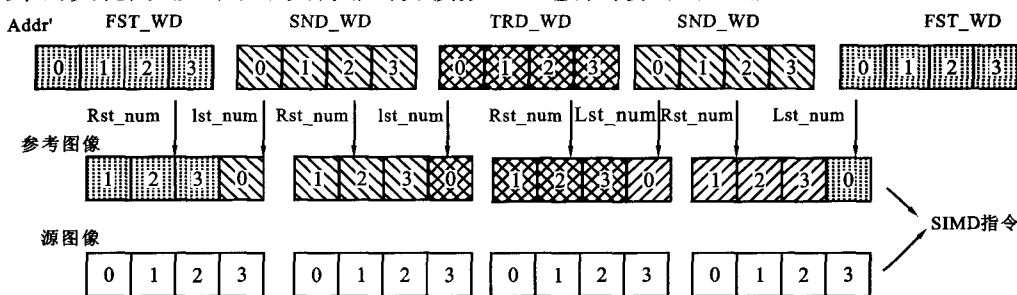


图 1 数据拼接示意图,其中 Lst_num, Rst_num 代表左移和右移的位数

该方案通过冗余读数的方式,解决了数据地址不对齐问题,加速了 SIMD 指令的计算.同样的方式可用于其他尺寸块的搜索,不再赘述.在 LILY Processor 上实现了该方案,计算效率提高为原来的 3

倍,同时读数指令总数也降低为原先的 22%,以 Sad (16×16) 为例给出了的优化结果对比见表 1,其中 L-S 代表循环体大小(cycles).

表 1 数据拼接方案优化结果对比

16 * 1	方案	LDB	LD	LDDW	add/shift	SIMD	L-S	ILP	Sad
16 * 1	字节	32	0	0	48/0	0	6	5.1	304
	拼接	0	5	2	4/8	4	6	6	102

支持非对齐读的处理器如 TMS320C64 $\times\times$,通过硬件自动拼接数据解决了地址和数据的不对齐问题.但是由于硬件资源受限导致一个周期只能执行一条非对齐读指令,且不能同时使用其他的存储器访问指令,限制了存储器带宽.采用数据拼接方案可以尽量使用对齐读提高取数速度,也可以通过复用图像数据来加速计算,详见第 4 节.

2.2 图像重组

本节采用一种拆分插值图像分开存储的方法,改变了传统算法中不利于 SIMD 指令执行的据组织存放方式.亚像素搜索过程是,首先将计算得到的插值图像块放到存储器中,然后再隔行隔列取出一个子块参加计算.因此插值图像的存放格式直接影响着 SIMD 指令的执行效率.

如果整像素最佳匹配位置为 (32, 14), $1/2$ 像素搜索搜索 (31.5, 13.5) 这个位置,那么只有 (0.5, 0.5) 这样的数据组合参加计算,是按照搜索矢量从

插值图像中隔行隔列取出的一个子块.根据这个规律,插值图像可以拆分为 (0, 0.5)、(0.5, 0)、(0.5, 0.5)、(0, 0) 四块分别存放. $1/4$ 像素插值图像也按同样的方法拆分重组,如图 2 所示.图像重组使取数机制由字节读优化为字读,加速了 SIMD 指令数据准备的速度,使 SIMD 计算效率获得 3~4 倍的提高,同时也为下文的数据重利用提供了基础.

3 图像数据复用

本节提出并行搜索技术使得相邻搜索点的共用源图像数据和参考图像的重叠数据,进一步提高 SIMD 指令的计算速度^[7].

整像素搜索中相邻搜索点的数据绝大部分数据都是重叠的.数据重叠分为竖直重叠和水平重叠两种如图 3 所示,竖直重叠表示相邻搜索点只有少数行的数据不同,水平重叠表示只有少数列的数据不同.竖直方向并行搜索相对简单,只需在循环体中采

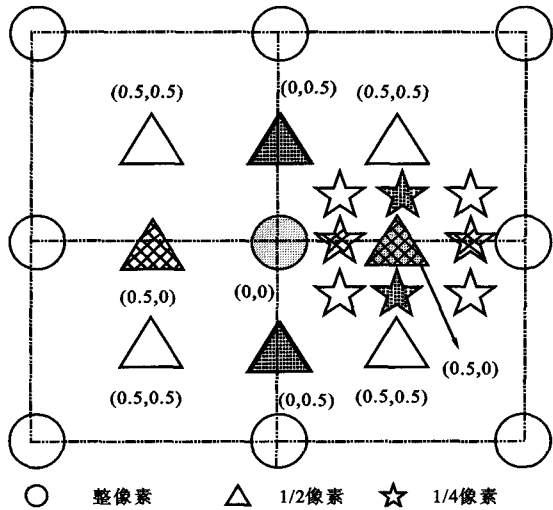


图 2 插值图像重组示意图

用数据暂存和循环流水技术,使得每行源图像数据和参考图像的竖直方向相应的多行数据进行计算.优化后 16×16 、 8×8 和 4×4 块的计算时间平均为 48、17 和 5.5 个周期.水平方向并行搜索要解决不同搜索点的 SIMD 指令的数据对齐问题,在提出的数据拼接方案的基础上,通过控制循环体中的移位变量来实现多个方向的数据对齐.以 TMS320CC64 $\times \times$ 为例进行优化给出了循环体对比数据见表 2.为了避免循环体太小不利于提高程序并行度,文中

对 4×4 块采用循环展开技术同时进行两个 4×4 的并行搜索计算.由表 2 数据可知, 16×16 和 8×8 块计算效率分别提高为 TI 库的 1.4 倍和 1.6 倍.

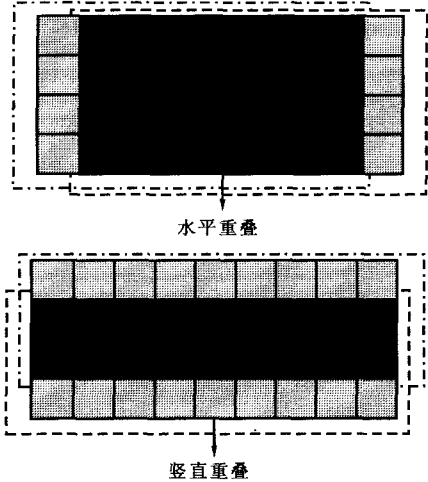


图 3 参考图像子块重叠示意图

亚像素搜索在拆分后的 $(0, 0.5)$ 、 $(0.5, 0)$ 、 $(0.5, 0.5)$ 子块中分别搜索 2、2、4 次,在同一子块中的不同搜索绝大部分数据都是重叠的,也可以用并行搜索技术复用图像数据.优化后对于一个搜索点 16×16 、 8×8 、 4×4 块搜索只需要 496、144、108 个周期即可完成.

表 2 整像素 sad($N \times N$)三种方案对比

$N \times 1/M$	方案	LD W	LD DW	LD NDW	add, sub /shift	Subabs4 /doptu4	Loop Size	ILP	Sad($N \times N$)/ (cycles)
$16 \times 1/2$	复用	0	2	3	13/8	8/4	5	7.8	89/2
	TI 库	0	4	4	8/0	8/8	3	6	61
$8 \times 1/3$	复用	0	1	2	10/8	6/6	5	6.8	46/3
	TI 库	0	3	3	6/0	6/6	3	6	25
$4 \times 1/8$	复用	0	1	1	14/12	8/8	7	6.6	38/8
	TI 库	—	—	—	—	—	—	—	—

注:① $N \times 1/m$ 表示 $N \times 1$ 块 m 个搜索点的并行搜索;②46/3 表示 46 拍完成 3 个搜索点的并行搜索

并行搜索结合了 ME 计算特点和 VLIW 处理器的指令优势,通过复用图像数据提高了计算效率.优化后指令的比例分配更合理,计算指令和读指令比例分别为 90% 和 10% 左右,其中 SIMD 指令比例提高到了 35%,提高 SIMD 指令的比例能大大提高计算能力.另外降低读指令数目能显著降低处理器功耗,因为读指令功耗比较大,尤其是通过硬件拼接数据的非对齐读指令.

4 结束语

提高运动估计(ME)算法的计算效率对于提高视频编码计算效率是至关重要的,而现有的基于

DSP 的实现并没有给出针对 ME 的优化方法.文中在充分挖掘算法特点和 VLIW 处理器指令集优势的基础上,采用加速 SIMD 指令的数据读取和拼接方案和多方向并行搜索方法以及适合图像数据复用的插值图像存储结构,优化实现了一个高效的 ME 软硬件结合的架构.分别在 TMS320C64 $\times \times$ 及自行设计的 LILY Processor 上实现了 H.264 的 QCIF 图像数据的搜索及 H.263 的 CIF 图像数据的搜索,测试实验表明 ME 的搜索速度分别提高了 15 倍和 3 倍以上.

(下转第 12 页)

式(4)的括号中第一项为分支预测技术对性能的提升作用,第二项为分支预测错误带来的性能损失,从表 1 的数据可以看出,第一项约是第二项的 16 倍,优化分支预测器的原则就是尽量提高第一项,减小第二项,提高分支预测技术对性能的贡献 CPI_{impr} .

5 结束语

在体系结构的研究中,CPI 栈通常用来表现程序在运行过程中的行为细节,文中用 CPI 栈组成本来表示分支指令对流水线吞吐率的影响和分支预测技术对处理器性能的提升作用,并在 RTL 级设计了分支预测器的时序精确模型,通过硬件模拟方法对分支指令特性和分支预测器的性能进行了研究.实验发现分支指令在分支预测器命中时跳转的概率很大,而缺失时的不跳转的概率相对更高的统计特性;并验证了分支预测器对 CPI 栈影响的理论推导,为单发射嵌入式处理器中分支预测器的设计与优化提供了精确的实验依据.

参考文献:

- [1] 郑纬民. 计算机系统结构[M]. 2 版. 北京:清华大学出版社,1998:512-514.
- [2] Evers M, Yeh T Y. Understanding branches and designing branch predictors for high-performance microprocessors

[J]. IEEE, 2001, 89(11):1610-1620.

- [3] 陈恺冬,史江一,郝跃. 一种基于双模结构的预测器[J]. 微电子学与计算机,2008,25(5):110-113.
- [4] Stijn Eyerman, Lieven Eeckhout, Tejas Karkhanis, et al. A performance counter architecture for computing accurate CPI components[C]// Proceedings of the 2006 ASPLOS Conference on ACM SIGARCH Computer Architecture News. San Jose, 2006: 175-184.
- [5] 黄海林,范东睿,许彤,等. 嵌入式处理器中访存部件的低功耗设计研究[J]. 计算机学报,2006,29(5):815-821.
- [6] Perleberg C, Smith A. Branch target buffer design and optimization[J]. IEEE Transactions on Computers, 1993, 42(4): 396-412.
- [7] 武萌,沈海斌. 一种 gshare 分支预测器的低功耗设计方法[J]. 微电子学与计算机,2007,24(3):200-206.
- [8] Lee J K L, Smith A J. Branch prediction strategies and branch target buffer design[J]. Computer, 1984, 17(1): 6-22.

作者简介:

喻明艳 男,(1965-),教授.研究方向为 VLSI/SoC 设计方法及其实现技术研究,计算机体系结构.
张祥建 男,(1984-),硕士研究生.研究方向为嵌入式处理器体系结构与 SoC 设计.
王晨旭 男,(1977-),博士研究生.研究方向为嵌入式系统、数字系统设计.

(上接第 7 页)

参考文献:

- [1] JVT of ISO/IEC MPEG, ITU-T VCEG. ITU-T Rec. H.264 ISO/IEC 14496-10 AVC[S]. Document JVT-GO50, 2003.
- [2] ITU-T. ITU-T Rec. H.263(02/98) Video coding for low bit rate communication[S]. ITU-T, Document JVT-GO50, 2000.
- [3] Hyunchang Im, Wonchul Lee, Wonyong Sung. Implementation of an H.264 motion estimation algorithm on a VLIW programmable digital signal processor[C]//IEEE Workshop on Signal Processing Systems Design and Implementation. Athens, 2005:302-306.
- [4] Hasan F Ates, Yucel Altunbasak. Sad reuse in hierarchical motion estimation for H.264 encoder[C]//IEEE International Conference on Acoustics, Speech, and Signal Pro-

cessing. Pennsylvania, 2005:905-908.

- [6] Deepu Talla, Lizy Kurian John, Doug Burger. Bottlenecks in multimedia processing with SIMD style extensions and architectural enhancements[J]. IEEE Transactions on computers, 2003, 52(8):35-46.
- [5] 周启棠,罗嵘. TMS320DM642 上 H.264 视频编码手工汇编优化方法[J]. 微电子学与计算机,2009,26(1):13-17.
- [7] 张彤宇,苏睿,刘宝兰. H.264/AVC 视频编码器在 DM642 平台上的实现与优化[J]. 微电子学与计算机, 2005,22(12):165-168.

作者简介:

谷俊丽 女,(1984-),博士研究生.研究方向为嵌入式多核和并行计算.