

基于 DSP 的低码率实时视频编码器设计与实现

刘少华 熊志辉 包卫东 张茂军

(国防科学技术大学信息系统与管理学院 长沙 410073)

摘要: 该文以 TI 公司 TMS320DM642 DSP 为核心处理器设计实现了一个符合 MPEG 标准的低码率实时视频编码器。主要特色是: 提出并实现了中心三步搜索和菱形搜索相结合的方法进行快速运动搜索; 提出并实现了一种新的全零块预先判别方法; 针对 DSP 系统结构以及指令特点对编码过程中的运算密集部分进行专门优化。实验表明, 该文提出的快速运动搜索算法性能优于中心三步搜索算法。全零块预先判别机制在保证图像质量的同时能有效减小运算量并降低码率。

关键词: 视频编码器; 数字信号处理; 运动估计; 全零块判别

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2008)04-0945-04

Design and Implementation of Low Bit-rate Real-time Video Encoder Based on DSP

Liu Shao-hua Xiong Zhi-hui Bao Wei-dong Zhang Mao-jun

(School of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract: A low bit-rate real time video encoding system on MPEG using DSP TMS320DM642 as core processor is designed. The main features are: Diamond Search is integrated with Center-based Three-step Search (CTSS); A new method for pre-determination of all-zero blocks is proposed; The encoder is specially optimized for DSP. Experimental results show that the search method in this paper is better than CTSS. The pre-determination of all-zero blocks reduces operation and bit rate efficiently with little image quality decrease.

Key words: Video encoder; DSP; Motion estimation; All-zero block detection

1 引言

嵌入式视频编码在战场侦察、消费电子(如手机视频编码、PDA 视频编码)等方面的应用需求越来越多, 但视频编码的码率和实时性一直是该编码技术应用的瓶颈。为解决该问题, 本文设计了一个适用于 DSP 并符合 MPEG 标准的低码率实时视频编码器。该编码器提出并综合运用了 3 种视频编码技术, 即适合对运动剧烈视频进行运动估计的搜索算法、全零块预先判别以及针对 DSP 嵌入式架构的专门优化。

首先, 对 MPEG 编码标准来说, 运动估计技术直接影响到编码的效率和质量。全搜索(Full Search, FS)算法可获得最优结果, 但计算量非常大, 必须简化。使用广泛的 3 步搜索法^[1](Three-Step Search, TSS)初始跨度较大, 易陷入局部最优, 对微小的运动估计精度较低。而一些改进的 TSS 算法都与原算法性能相当。基于中心三步法^[2](Center-based Three-Step Search, CTSS)比较适合 DSP 的实现, 但其搜索过程没有跳跃, 适合小图像格式或是运动微弱的图像, 否则会增加许多冗余搜索。本文提出并使用的运动搜索算法吸取 CTSS 算法的优势, 结合菱形算法对大范围的运动进行定位, 对于不同范围的运动都有较好的搜索结果。

其次, 为了提高编码效率, 减少运算量, 在编码过程中

使用了全零块预先判别技术。对于视频中的背景区域运动估计具有很高的精度。如果其残差系数经过 DCT 变换、量化后全部为零, 称这样的块为全零块。显然对于全零块而言, DCT 变换、量化、反变换和反量化都是冗余操作, 因此通过全零块的预先判别, 可以有效地减少计算量。针对这种情况, 本文提出并采用了一种新的全零块预先判别算法, 有效减少了误判和漏判, 实际使用中有较好的结果。

再次, DSP 数字信号处理器是一种哈佛体系结构, 与普通计算机系统的冯·诺依曼体系结构存在较大区别, 并且支持指令的并发执行。因此, 本文对视频编码器中一些运算密集的部分进行针对 DSP 的专门优化, 从而进一步提高编码效率。

该编码器在以 TI 公司 TMS320DM642 数字信号处理器(DSP)为核心的开发板上实现。实验结果表明, 本文设计实现的视频编码器所使用的快速运动搜索算法性能优于中心三步搜索算法, 全零块预先判别机制能有效减小编码运算量并降低码率。

2 运动估计搜索

进行块匹配搜索时, 当前图像帧被划分成若干个互不重叠的宏块, 当前宏块到参考帧中按块匹配准则搜索匹配误差最小的位移坐标作为运动矢量。分析表明各种块匹配函数的

性能差别不显著^[3]。为便于 DSP 实现, 本文采用运算量较小的绝对差之和(Sum of Absolute Difference, SAD)作为该搜索算法的匹配准则, 计算公式为

$$\text{SAD} = \sum_0^{m-1} \sum_0^{n-1} |f(i, j) - g(i + u, j + v)|$$

其中 m, n 为宏块的大小, f, g 分别为当前宏块和参考宏块, u, v 为参考宏块在 X 和 Y 方向上的移动矢量。

为了减少运算量和提高精确度, 本文采用了起始点估计。使用相邻已搜索块的运动矢量均值作为当前搜索块的起始点, 在一定范围内搜索运动增量。由于运动物体的整体性和视频运动的相关性, 相邻块往往具有相似的运动趋势, 这样起始点能尽量接近最终的运动矢量, 同时可防止遗漏微小运动矢量的情况。由于搜索顺序是从左至右、从上至下, 本文选取的相邻块是当前宏块左边和上边的宏块, 它们的运动和当前块有很高的相似性, 且均值计算简单。

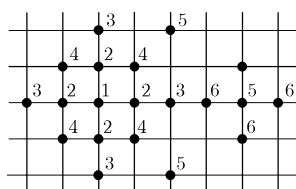


图1 本文使用的运动估计搜索算法原理

图1表示了具体的搜索步骤。图中黑点表示搜索点, 黑点右上方的数字表示该点在第几步中被搜索。

(1) 以相邻块运动矢量均值作为搜索起始点。

(2) 以步长为1进行上下左右4个点的菱形搜索。

(3) 以步长为2进行菱形搜索, 如果目前为止 SAD 最小的点出现在第(1)步和第(2)步的搜索中, 则继续第(4)步, 否则跳过第(4)步。

(4) 以步长为1进行对角线上4个点的矩形搜索, 取第(1), (2)步和当前步搜索中 SAD 最小的点为搜索结果, 搜索过程结束。

(5) 以第(3)步中 SAD 最小的点为中心进行步长为2的菱形搜索。

(6) 以第(5)步中 SAD 最小的点为中心进行步长为1的菱形搜索, 取 SAD 最小的点为搜索结果, 搜索过程结束。

由于视频中的运动以水平和垂直方向居多, 在搜索过程中在这两个方向上的搜索点也相对较多。在整个搜索过程中, 一旦发现某个搜索点的 SAD 值小于预先设置的门限值, 则说明该块已经符合搜索要求, 可直接退出整个搜索过程。

据文献[2]统计, 对普通图像序列, 70%以上的宏块搜索增量为0。因此, 多数情况下, 大部分宏块的搜索都可在第(1)步和第(2)步完成, 少部分的宏块会进入第(3), 第(4)步的搜索。而对于变化剧烈的图像部分, 则通过第(3), 第(5)步的菱形搜索进行大范围的定位, 最后利用第(6)步进行精确的定位。因此第(3)步的搜索实际上起到了一种分类作用, 即将运

动微弱和运动剧烈的宏块分开, 分别采取不同的搜索策略, 从而增强搜索算法对不同图像类型的适应程度。与 CTSS 算法相比, 该搜索算法能覆盖更大的搜索范围, 而且搜索次数也较少, 最多只需要16次计算就可完成整个搜索, 而且该算法对运动剧烈的大格式图像有更好的适应性。

3 全零块预先判别

全零块判别技术由 A Yu 提出^[4], 对于 16×16 的宏块, Yu 算法判据为 $\left| \sum_0^{15} \sum_0^{15} f(x, y) \right| < 16Q$ 。另一种使用较广的算法是 Zhou 算法^[5], 其判据是 $\text{SAD} < 8Q$ 。Yu 算法经常会出现误判, 对图像的损伤非常严重^[6]。另外, Yu 算法必须计算每个被判别块的运动补偿数据和的绝对值, 无疑又增加了计算量。Zhou 算法中使用的 SAD 值在运动估计时已经计算过了, 只需将结果保存下来, 判断时就可以直接引用。Zhou 算法在理论上不会出现误判, 但在实际过程却出现了较大程度的漏判^[6], 没有有效地减少 DCT 和量化的计算时间。

本文采用的全零块判别准则如下: 对于 $N \times N$ 的 DCT 变换, 公式可表达为

$$F(u, v) = \left(\frac{2}{N} \right) k_u k_v \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left(\frac{(2x+1)u\pi}{2N} \right) \cos \left(\frac{(2y+1)v\pi}{2N} \right) \quad (1)$$

c 为量化前的 DCT 变换系数, 则量化后 DCT 系数全为零的充分必要条件为: $|c| < Q$ 。

可等价的表示为

$$\max(|F(u, v)|) < Q \quad (2)$$

根据式(1)和柯西-许瓦兹公式可得:

$$F(u, v) \leq \left[\sum_{x=0}^{15} \sum_{y=0}^{15} f^2(x, y) \right]^{1/2} \quad (3)$$

在低比特率视频编码中, 像素亮度的最小值 Black=16, 最大值 White=235, 有不等式

$$\left[\sum_{x=0}^{15} \sum_{y=0}^{15} f^2(x, y) \right]^{1/2} < \frac{1}{8\sqrt{2}} \times \left[\sum_{x=0}^{15} \sum_{y=0}^{15} |f(x, y)| \right] \quad (4)$$

成立。由式(2)可得:

$$\sum_{x=0}^{15} \sum_{y=0}^{15} |f(x, y)| < 12Q \quad (5)$$

即, 本文认为满足式(5)的宏块即为全零块。

在引入全零块预先判别策略后, 对于 P 帧和 B 帧的编码过程为: 对原始的数据块首先进行运动估计搜索, 获得运动矢量和 SAD 值。对于 SAD 不满足式(5)的非全零块, 按照正常的编码步骤: 计算当前块与参考块的残差、对残差进行 DCT 变换、量化、可变长编码, 同时还要进行反量化和 IDCT 变换, 生成新的参考块, 供下一帧编码时使用。而对于全零块, 过程就简单得多: 只需对运动矢量进行编码即可。由于该块是全零块, 所以新的参考块与当前参考块是相同的, 直接将当前参考块拷贝即可生成新的参考块。图2所示为帧间

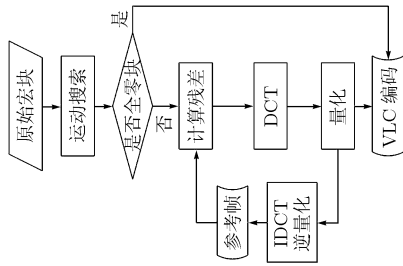


图 2 加入全零块预先判别后的编码过程

编码的完整过程。可以看出，对于全零块的编码过程大大简化，有效地减少了编码的计算量。

4 系统实现

4.1 编码器开发平台

本文设计实现的视频编码器已在自行研制的 DSP 开发平台上完成。该平台结构如图 3 所示。硬件板主要包括视频采集模块、视频输出模块、网络通信模块和核心处理模块。系统的核心处理器采用 TI 公司的 TMS320DM642 DSP 芯片，通过 EIMF 总线连接一块 4M×8 位的 FLASH 闪存，和 8M×64 位的 SDRAM。视频输入为标准 PAL。A/D 转换芯片采用 TVP5150，通过 I²C 总线控制该芯片完成视频的采集。网络通信模块采用 BCM5221。目标板与计算机通过 JTAG 接口连接，完成程序的写入和调试工作。TMS320DM642 DSP 工作主频为 600MHz，峰值处理性能为 4800MIPS。

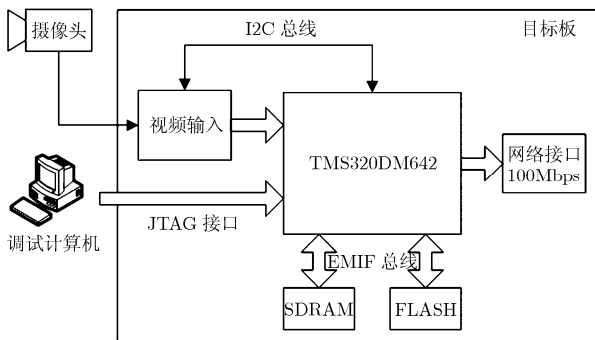


图 3 编码器嵌入式开发平台硬件结构

4.2 编码器的 DSP 实现

软件的实现通过 TI 公司提供的 CCS(Code Composer Studio)开发环境完成。编码主要包括 3 个部分：数据输入模块、编码器模块和数据输出模块。编码器的设计符合 MPEG 标准，主要有 4 个主函数，分别是：MotionEstimation()，进行运动搜索，确定运动矢量和宏块的编码类型；Predict()，根据运动搜索结果进行图像预测；Transform()/iTransform，完成 DCT/IDCT 变换；PutPict()，完成量化和 VLC 编码。编码后的数据由输出模块发送给远程的网络接收地址。为保证同步，数据的输入和输出都采用了双缓冲机制。

4.3 编码器的性能优化

本文分 3 个层次进行编码器的优化：全局优化，算法级优化，指令级优化。优化过程中采用的视频序列为 400 帧 CIF 格式的 foreman 序列。

全局优化是对程序的整体优化，主要采取了以下措施：精简程序，保留核心功能，去除冗余代码；充分利用 DSP 的高速缓存，将核心代码和常用变量放入其中，其他数据放在片外内存；调整数据结构，将需要大规模访问的数据放在一起，采用 DMA 机制访问；另外，TI 公司为 C6000 系列的芯片提供了一些专用于图像处理的库函数，对于编码来说有全搜索函数、SAD 计算函数、DCT 变换函数和量化函数，将 C 编写的函数替换为这些库函数；通过采用以上优化措施，编码速度有了极大提高，平均达到 42 毫秒/帧，具体的时间分布见表 1。表中的其它部分包含了 IDCT 和反量化以及额外耗时。

表 1 全局优化后编码器各部分编码时间占用比例(%)

运动搜索	图像预测	DCT 变换	量化+VLC 编码	其它	总耗时
31	12	17	22	18	42ms

算法级优化指采用更加快速高效的编码算法，从而提高运行速度。在视频编码中，运动搜索耗时最多，使用本文中的搜索算法可进一步减少搜索时间。通过加入全零块预先检测机制，又可进一步减少其它部分的运算时间。通过算法优化，编码速度提高到 30 毫秒/帧。具体的时间分布见表 2。

表 2 算法级优化后编码器各部分编码时间占用比例(%)

运动搜索	图像预测	DCT 变换	量化+VLC 编码	其它	总耗时
29	14	16	21	20	30ms

指令级优化主要是针对耗时的函数采用汇编语言重新编写，编写时应充分考虑 DSP 的指令特点，提高并行性。根据编码中各部分的计算量分布情况，本文对图像预测、VLC 编码等耗时相对较多的函数(如预测函数 Predict()、ZigZag 扫描函数 ZigZag()、写位流函数 PutBits()等)采用 C6000 系列 DSP 的汇编语言重新编写，最终编码速度提高到 21 毫秒/帧。具体的时间分布见表 3。

表 3 指令级优化后编码器各部分编码时间占用比例(%)

运动搜索	图像预测	DCT 变换	量化+VLC 编码	其它	总耗时
40	10	18	22	10	21ms

5 实验结果及分析

本实验使用的测试视频为 CIF 格式的 Foreman，共 400

帧, 宏块大小为 16×16 。首先测试本文提出的搜索算法。测试结果见表 4。进行此项测试时, 全零块预先判别暂时关闭。PSNR 值的计算采用如下公式:

$$\text{PSNR} = \frac{1}{\text{frame}} \cdot \sum_{i=0}^{\text{frame}-1} 10 \lg \left\{ 255 \times 255 \times N_p / \left[\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} (\text{orig}_i(x,y) - \text{deg}_i(x,y))^2 \right] \right\}$$

其中 Frame 为整个序列的总帧数; orig, deg 分别为编码前和编码后各像素点的灰度值; N_p 为每帧图像所含像素数。表中分别给出了 Y, U, V 分量的编码后的 PSNR 值。

由表 4 可看出, 相对于 CTSS 算法而言, 本文算法的平均搜索的次数减少了 4 次, 而编码后图像的 PSNR 值略有提高, 码率也稍有降低。

表 4 搜索算法测试结果(foreman)

搜索算法	平均搜索次数	PSNR 值(dB)			码率增加 (%) (相对于 FS)
		Y	U	V	
FS	181	36.33	41.10	42.70	
TSS	21	36.28	41.10	42.71	4.4
CTSS	14	35.99	41.06	42.61	13.8
本文算法	10	36.02	41.05	42.62	12.3

接下来测试本文的全零块预先判别方法。为方便对比, 测试序列仍为 foreman。采用本文的搜索算法, 按照式(5)进行全零块预先判别。表 5 显示了在不同量化步长条件下全零块所占全部宏块的比率, 以及使用全零块预先判别后带来的码率减少和 PSNR 值的变化。需要指出的是, 对于 foreman 这类运动剧烈的视频, 全零块预先判别的优势并不明显。对于图像背景变化微弱的监控类视频, 使用本文的全零块判别方法, 全零块比率一般可达到 60%, 码率减少近 20%, PSNR 值下降小于 0.01dB。

表 5 使用全零块预先判别后的编码结果(foreman)

量化步长	全零块比率(%)	PSNR 值下降(Y)	码率减少(%)
8	4.6	<0.001	0.5
16	6	0.001	0.6
32	7.2	0.001	0.7

6 结束语

本文基于 MPEG 标准设计并实现了一个适用于 DSP 的

低码率实时视频编码器。实验表明, 文中所提出的搜索方法有着比 CTSS 算法更加优秀的效果, 相对于全搜索算法, 其编码后的图像 PSNR 只有 0.3dB 的下降, 平均搜索次数降低到了 10 次, 大大减少了搜索时间。新的全零块预先判别方法在图像质量降低很小的情况下, 对于视频监控图像可检测出近 60% 的全零块, 可将码率降低近 20%。通过新算法的引入和针对 DSP 系统的优化, 本编码器对 CIF 视频的编码速度达到近 50 帧/秒, 并具有相对较低的码率。

参考文献

- [1] Koga T, Iinuma K, and Hirano A, *et al.* Motion compensated interframe coding for video conferencing. Proc Nat Tele Conf, New Orleans, LA, 1981, G5.3.5-G5.3.10.
 - [2] 胡琳蓉, 朱秀昌. 一种适用于 H.263 的运动估计搜索算法. 通信学报, 2000, 21(6): 65-69.
Hu Lin-rong and Zhu Xiuc-hang. A search algorithm for motion estimation in accordance with H.263. *Journal of China Institute of Communications*, 2000, 21(6): 65-69.
 - [3] 王亮, 徐佩霞. 基于 DSP 实时实现的 H.264 运动估计快速算法. 小型微型计算机系统, 2005, 26(11): 2025-2028.
Wang Liang and Xu Pei-xia. Fast motion estimation algorithm based on DSP according to H.264. *Mini-micro-Computer Systems*, 2005, 26(11):2025-2028.
 - [4] Yu A, Lee R, and Flynn M. Early detection of all-zero coefficients in H.263. Proceedings of the picture coding Symposium, Berlin, Germany, 1997: 159-164.
 - [5] 周璇, 谭径微, 余松煜.H.263中预先判别全零系数的新方法. 上海交通大学学报, 1998, 32(9): 107-109.
Zhou Xuan, Tan Jing-wei, and Yu Song-yu. New early detection method of all-zero coefficients in H.263. *Journal of Shanghai Jiaotong University*, 1998, 32(9): 107-109.
 - [6] 钟伟才, 刘静等, 焦李成, 等. H.263 中全零块预测的新方法. 电子与信息学报, 2003, 25(4): 573-576.
Zhong Wei-cai, Liu Jing, and Jiao Lic-heng, *et al.* A novel early prediction method of all-zero coefficients in H.263. *Journal of Electronics & Information Technology*, 2003, 25(4): 573-576.
- 刘少华: 男, 1982 年生, 博士生, 研究方向为多媒体信息系统与虚拟现实技术。
熊志辉: 男, 1976 年生, 博士, 研究方向为数字图像处理与嵌入式系统。
包卫东: 男, 1971 年生, 博士, 副教授, 研究方向为虚拟现实与系统仿真技术。