

流计算和视频编码

张春元 文 梅 苏华友 著
伍 楠 任 巨



科学出版社

流计算和视频编码

张春元 文 梅 苏华友 著
伍 楠 任 巨

科 学 出 版 社

北 京

内 容 简 介

本书主要阐述一种众核并行编程思想——流计算思想,并以典型视频编码标准 H. 264 为例,详细论述具有普适性的流化方法。全书共分 7 章。第 1 章主要介绍视频编码的基本内容、视频编码并行化的潜力和挑战。第 2 章介绍流计算的基本原理,包括流计算思想、流体系结构、编程模型和流应用等。第 3 章详细论述面向媒体处理的流化方法并提出程序结构的优化方法。第 4 章以 H. 264 编码器为实例,阐述利用流化方法将串行编码器转换为并行流化编码器框架 S264。第 5 章系统地阐述 S264 框架在 SIMD 并行处理器上的映射过程和全高清实时的 H. 264 编码器 S264/S 在 STORM 流处理器上的实现。第 6 章从多线程执行的角度描述了 S264 框架在 GPU 上的实现。第 7 章从视频编码应用的角度对未来视频编码的发展与基于流计算的并行化研究进行讨论。

本书是作者及其研究团队十余年科研经验和成果的总结,可供并行处理器体系结构、并行编程、视频编码等方向的研究人员参考,也可作为计算机科学与技术等专业高年级本科生和研究生的教材。

图书在版编目(CIP)数据

流计算和视频编码/张春元等著. —北京:科学出版社, 2013. 1
ISBN 978-7-03-036000-7

I. ①流… II. ①张… III. ①视频编码—研究 IV. ①TN762

中国版本图书馆 CIP 数据核字(2012)第 268687 号

责任编辑:张 濮 陈 静 / 责任校对:张小霞

责任印制:张 倩 / 封面设计:迷底书装

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂印刷

科学出版社发行 各地新华书店经销

*

2012 年 11 月第 一 版 开本:B5(720×1000)

2012 年 11 月第一次印刷 印张:17 1/4

字数:345 000

定价:72.00 元

(如有印装质量问题,我社负责调换)

前 言

• 写作背景

强大、真实的应用需求永远是计算机体系结构的第一推动力。作为体系结构的研究人员,我们一直对各式有着高性能计算需求的应用怀有高度的敏感,特别是从 2003 年 MASA 课题组开始流体系结构研究之后。我们希望能够像斯坦福大学的 Dally 教授领导的研究小组提出流体系结构那样,有相当的应用研究背景。而除了一些核心(Kernel)级的测试 Benchmark,如快速傅里叶变换(Fast Fourier Transformation,FFT)、矩阵乘、离散余弦变换(Discrete Cosine Transform,DCT)等之外,真实的应用是很难获得的,只能自己去研究开发。从哪个应用,甚至从哪个应用领域开始研究,成为困扰课题组的一个很大的难题。

而这时,H.264 压缩编码标准进入了我们的视线。从各个角度,它都符合我们的要求:

(1) H.264 编码属于媒体处理领域,是流体系结构的典型处理领域。随着沉浸通信(immersive communication)的兴起,视频压缩与解压缩(video compression & decompression)扮演越发重要的应用角色。

(2) H.264 是 ITU-T 的视频编码专家组(Video Coding Experts Group,VCEG)和 ISO/IEC 的运动图像专家组(Moving Pictures Experts Group,MPEG)的联合视频组(Joint Video Team,JVT)于 2003 年正式发布的最新的视频编码标准,具有足够的权威性和相当广阔的应用前景,并有公开的 x264 代码作为实现参考。

(3) H.264 编码是一个完整的应用,其计算量大、复杂度高,涵盖了运动搜索、熵编码、量化与反量化、滤波等经典媒体处理的模块。掌握这个应用对其他媒体处理的应用具有一定程度的借鉴意义。

(4) 当时通用 CPU 的计算能力无法满足甚至是较低分辨率图像的实时编码处理,更不用说全高清(1080P)图像的实时处理,它只能是用 ASIC 实现。因此,软件并行处理具有很高的学术前沿研究价值。

因此,从 2005 年中开始,结合前沿体系结构平台研究,我们投入了多名研究生对 H.264 编码的软件并行实现进行了持续的研究,取得了丰硕的成果。我们在 ACM Multimedia 等重要学术会议上发表了一系列学术论文,并获得专家同行的认可。2009 年底,在世界上第一款流 DSP——STORM-I 上开发出了全高清

H. 264实时编码程序。2010 年底,又在图形处理单元(Graphic Processing Unit, GPU)上用 CUDA 开发出了全高清 H. 264 实时编码程序,领先于 NVIDIA 公司在 2011 年公开的 H. 264 实时编码库。

六年的研究过程中,我们目睹了通用图形处理(GPGPU)、异构编程、流式并行处理、显式存储层次管理等新概念从初生到兴盛,从被怀疑到广为接受的过程。可以预见,将来必是众核处理器的天下,而众核并行编程,仍是目前的一个研究热点。我们有理由相信,在将来的并行编程中,必然会看到流处理的身影。希望本书能给国内同行起到抛砖引玉的作用。

• 内容安排

在处理器平台拥有数百个核的今天,如何发挥其计算效率,是一个重大挑战。本书的主要宗旨就是介绍流计算的主要思想,并以典型视频编码标准 H. 264 为例,详细论述流化方法。所谓流计算,其实就是一种以流模型为基础的并行计算。流计算的产生和推广,一方面,源于其本身起源自媒体处理应用,迎合了应用本身的特征,如生产者-消费者局域性、大量的数据并行性等;另一方面,源于新兴体系结构,如流体系结构等,着力于数据级并行的开发。

本书还结合课题研究,比较全面地反映了我们在流计算领域,包括流计算原理、视频编码应用、流化方法、基于单指令多数据(Single Instruction Multiple Data, SIMD)和 GPU 的流化实现等方面取得的研究成果。

全书共分 7 章,内容安排如下。

第 1 章主要阐述了视频编码的基本内容,同时介绍了视频编码并行化的潜力及其在未来发展中面临的挑战。并以 H. 264 编码为例,重点介绍了视频编码的基本原理。

第 2 章介绍了流计算的基本原理。从流计算思想、流体系结构、编程模型和流应用等几个方面进行了阐述,让读者对流计算有全面的认识。

第 3 章详细介绍了面向媒体处理的流化方法,从基本流化方法到高级流化方法,向读者介绍了面对不同问题可以采取的技术手段。在此基础上,本章还介绍了程序结构的优化方法。

第 4 章以 H. 264 编码器的实现 x264 为实例,向读者阐述了如何利用第 3 章提出的流化方法将一个串行的编码器转换为一个并行流化的编码器框架 S264。

第 5 章系统地阐述了流化的编码器框架 S264 在 SIMD 并行处理器上的映射过程,并在 STORM 流处理器上实现了全高清实时的 H. 264 编码器 S264/S。

第 6 章从多线程执行的角度描述了 S264 编码器框架在 GPU 上的实现考虑,针对不同的编码器模块,给出了不同的设计考虑,包括平衡线程与并行粒度、多级并行等。

第7章从视频编码应用的角度出发,对未来视频编码的发展及基于流计算的并行化研究进行了讨论。

本书由 MASA 课题组成员合作完成,由张春元教授、文梅副教授策划和统筹,并与苏华友、伍楠、任巨三位博士共同执笔完成。第1章由张春元撰写,第2章由文梅撰写,第3章由伍楠撰写,第4章和第5章由任巨和苏华友撰写,第6章由苏华友和文梅撰写,第7章由苏华友和伍楠撰写。李海燕、荀长庆、吴伟为本书提供了丰富的素材。管茂林、杨乾明、荀长庆、柴俊、全巍、黄达飞、乔寓然、蓝强和薛云刚等博士(硕士)研究生收集和整理了大量的资料,提供了良好的素材。本书在写作过程中,参阅了国内外许多作者的论文和著作,吸取了其中部分材料,在此向这些文献的作者深表谢意。

感谢国家自然科学基金项目、863 项目、国家教育部博士点基金项目的支持。

感谢吴伟为 H. 264 编码的流化研究做出的卓越贡献。

本书试图在前人的研究基础上,结合我们多年科研工作的体会,向同行介绍我们的研究成果。由于作者的能力和知识面有限,疏漏和错误难免,恳请读者批评指正。

有关 MASA 课题组的情况,读者可参见课题组的主页 <http://masa.nudt.edu.cn>。

MASA 课题组 张春元

于湖南长沙国防科技大学

2012 年 5 月

目 录

前言

第 1 章 视频压缩编码	1
1.1 概述	1
1.2 视频压缩编码面临的新挑战	5
1.3 视频编码并行化潜力	7
1.4 H.264 编码标准	9
1.4.1 基本概况	9
1.4.2 编码器结构	11
1.4.3 特点和关键技术	13
1.4.4 帧间预测	16
1.4.5 帧内预测	20
1.4.6 变换量化	22
1.4.7 熵编码	29
1.4.8 去块滤波	31
第 2 章 流计算原理	34
2.1 流计算思想	34
2.2 流体系结构模型	36
2.2.1 解耦合计算和访存	36
2.2.2 多级存储层次	38
2.2.3 典型流处理器	42
2.3 流程序设计模型	49
2.3.1 流级程序	50
2.3.2 核心级程序	52
2.3.3 流计算模型与并行处理器的结合	53
2.4 流应用	54
第 3 章 流化方法	57
3.1 流化	57
3.1.1 流化定义	57
3.1.2 流化特征	60
3.1.3 特征指标有效性验证	63

3.2 基本流化方法	74
3.2.1 循环划分法	75
3.2.2 核心合并技术	76
3.2.3 流分割技术	78
3.2.4 常规流化步骤	80
3.3 高级流化方法	82
3.3.1 流重组	83
3.3.2 短流加载	85
3.3.3 核心分裂	87
3.3.4 并行粒度调整	88
3.4 面向流化的程序结构优化	90
3.4.1 重负载循环	90
3.4.2 计算窗口重叠	92
3.4.3 循环内分支	94
3.4.4 循环间数据相关	98
第4章 H.264 编码器的流化	100
4.1 x264 编码器概述	100
4.2 x264 编码器结构剖析	102
4.2.1 profile 工具:PAPI	102
4.2.2 函数层次的局限性	104
4.2.3 变量和数据结构层次的局限性	108
4.2.4 控制密集型组件层次的局限性	108
4.3 x264 编码程序的流化	109
4.3.1 函数到 Kernel 的转化	110
4.3.2 结构变量到 stream 的转化	118
4.3.3 控制密集型组件到计算密集型组件的转化	126
4.4 S264:流化的 H.264 框架	127
第5章 S264/S:基于 SIMD 并行的 H.264 流化编码器	129
5.1 SIMD 流处理器	129
5.2 帧间预测的流式实现	132
5.2.1 全搜索:流重用	132
5.2.2 UMHexagonS:非规则流	133
5.2.3 多分辨率多窗口帧间预测	142
5.2.4 分析与讨论	162
5.3 帧内编码:多模式和强数据相关	165

5.3.1	相关性分析	165
5.3.2	相关性问题:可扩展块并行	167
5.3.3	多模式问题:Kernel 参数化	172
5.3.4	分析与讨论	173
5.4	CAVLC 的流式实现:可变长编码	175
5.4.1	相关性分析	175
5.4.2	解耦合分段并行	177
5.4.3	分析与讨论	184
5.5	去块滤波的流式实现:控制密集型	185
5.5.1	相关性分析	185
5.5.2	去块滤波的流式实现:分组并行	187
5.5.3	分析与讨论	190
5.6	基于 STORM 的 SIMD H.264 编码器	190
5.6.1	S264/S 编码器	190
5.6.2	评测与分析	200
第 6 章	S264/G:基于 GPU 的并行 H.264 编码器	208
6.1	GPU 和 CUDA	208
6.1.1	GPU 概述	208
6.1.2	CUDA 硬件模型	210
6.1.3	CUDA 编程模型	211
6.1.4	CUDA 存储模型	212
6.2	基于 CUDA 的 H.264 编码器框架	214
6.3	帧间预测:平衡线程数与并行粒度	216
6.3.1	SAD 计算	217
6.3.2	最佳运动向量的选择	220
6.3.3	小结	222
6.4	帧内编码:开发多级并行	224
6.4.1	开发多级并行	224
6.4.2	存储模型	227
6.4.3	优化	228
6.4.4	小结	229
6.5	分量优先的 CAVLC	229
6.5.1	分量优先的 CAVLC 体系结构	229
6.5.2	基于 CUDA 的并行 CAVLC 实现	231
6.5.3	小结	235

6.6	方向优先的去块滤波	236
6.6.1	方向优先去块滤波结构	236
6.6.2	滤波顺序的调度	238
6.6.3	小结	238
6.7	实验结果与性能分析	241
6.7.1	并行编码器的性能	241
6.7.2	编码器特征分析	243
6.7.3	小结	246
第7章	展望与未来的研究方向	247
7.1	视频编码发展趋势	247
7.1.1	基于内容的视频编码	248
7.1.2	多视角视频编码	248
7.1.3	分布式视频编码	249
7.2	三维视频编码流化并行研究	250
7.3	面向媒体的并行计算	252
参考文献		255

第 1 章 视频压缩编码

随着计算机技术与互联网技术的飞速发展,多媒体内容急剧增长,多媒体应用日益广泛。媒体处理应用已经成为主流计算应用,其中视频压缩编码是媒体处理的重要应用之一,视频编码技术已经成为数字视频广播、数字媒体存储和多媒体通信等应用的基础性、核心共性技术。

1.1 概 述

在当前的信息时代中,对信息准确、及时地获取是至关重要的。人们每天获取的信息大部分是视觉信息。统计数据表明,人们每天通过视觉获取的信息占外界信息总量的 70%左右,而视频信息以其直观性、高效性成为信息时代中获取视觉信息的重要途径。随着计算机和互联网技术的发展,视频信息的处理和传输技术不断进步,使视频信息在人们工作、学习和生活的各个方面迅速普及,目前已广泛应用于视频会议、远程医疗、远程教学、网络视频点播、高清数字电视广播、数字电影、数字视频存储等领域,涉及政府工作、医疗、教育、媒体传播、娱乐、消费电子等各行各业中。上述领域的视频应用迅猛发展,未来前景十分广阔。例如,视频会议正在全球普及,2010 年后视频会议市场仍保持 20%的年增长率;数字电视正在取代模拟信号电视,并由标清(Standard Definition, SD)向高清(High Definition, HD)、全高清(Full High Definition, FHD)甚至超高清晰视频(Super Hi-Vision, SHV)发展;2007 年,平均每天有长达 8600 小时的视频信息通过互联网被上传到 YouTube,2010 年 3 月这一数字上升到 35000 小时;至 2009 年初,我国网络视频用户超过 2 亿户;在无线通信领域,使用移动终端来传输视频是 3G 以及未来 4G 时代的主体业务之一;数码相机和数码摄像机正逐步走入每个家庭。

视频应用在高速发展的同时,也面临着一些必须解决的问题,这主要是原始视频信息量太大造成的。例如,高清晰度电视(High Definition Television, HDTV)业务中一帧原始的全高清视频图像(1920×1080 像素)的数据量约为 3MB,以每秒 30 帧计算,一秒钟的视频数据量约为 90MB,一小时视频的数据量超过 300GB。即使对于低分辨率视频会议采用的 CIF 格式(分辨率仅为 352×288 像素),其原始视频的码率约为 36Mbit/s,已经是我国普及宽带网络带宽的十倍左右。可见,原始视频巨大的数据容量对网络带宽和数据存储都提出了极高的需求,而这些要求显然无法在短期内获得实际应用。为了解决这一问题,视频压缩编码和解码应

运而生。视频压缩编解码是指对视频先压缩,然后传输或存储至目的地再解压缩的过程。

视频压缩就是采用特定的编码方案,去除视频信息中的某些冗余内容,在不影响人们对视频的感观效果的约束下将数字视频压缩到几十分之一、几百分之一甚至几千分之一,大幅度节省存储空间、传输带宽和处理能力。人类视觉感知系统是非线性敏感的,自然视频序列具有时空相关性,压缩过程就是依据这两个特性,去除原始视频数据中的冗余,包括空域冗余、时域冗余、编码冗余和视觉冗余。

空域冗余是指在同一幅图像中相邻像素点之间往往存在着空间连贯性,而基于离散像素采样来表示像素点的方式通常没有利用这种连贯性。空域冗余表现为像素点的颜色值或亮色度分量接近,各像素点的数值可以通过其相邻像素预测出来。

时域冗余是指一组连续的图像之间往往存在着时间连贯性,而基于离散时间采样来表示运动图像的方式通常没有利用这种连贯性。对于静止不动的场景,当前帧与前一帧的图像内容是完全相同的;对于场景中的运动物体,如果知道它的运动规律,那么也可以很容易地从前一帧图像推算出它在当前帧中的大致位置。

编码冗余是指图像数据编码的平均码长不能达到或接近信息熵而形成的冗余,是一种由编码表示方法引起的符号相关性。其冗余大小与编码信息中每个符号的出现概率有关。

视觉冗余是由于人类视觉系统对图像的敏感性是非均匀和非线性而造成的。例如,对亮度变化敏感,而对色度的变化相对不敏感;在高亮度区,人眼对亮度变化敏感度下降;对物体边缘敏感,而对内部区域相对不敏感;对整体结构敏感,而对内部细节相对不敏感等。可以根据这些视觉特性对图像信息进行取舍。

视频编码方法与所采用的信源模型有关^[1]。如果采用“一幅图像由若干像素构成”的信源模型,其模型参数为每个像素的亮度与色度的幅度值,那么对应的压缩编码技术称为基于波形的编码;如果采用“一个图像区域由几个物体构成”的信源模型,其模型参数为各个物体的形状、纹理和运动,那么对应的压缩编码技术称为基于内容的编码。视频压缩编解码系统的基本结构如图 1.1 所示。编码过程由编码器完成,原始视频信号经过信源编码(分析过程)得到量化前的参数;然后使用有损的量化编码,用二进制码表示其量化值;再经过无损的熵编码进一步压缩码率,生成码流。解码器对码流的解码过程是上述编码过程的逆过程。一般情况下,编码过程比解码过程复杂,而且基于重建参考帧的编码技术包含解码过程,因此大部分视频编解码的研究工作都是关于视频编码的。

表 1.1 列出了面向不同应用的原始视频及压缩后的码率估计值,涉及电影、高清电视、视频电话等多种分辨率与帧率需求的视频应用。表中数据表明,视频压缩编码能够使码率降低几十倍甚至几百倍,具有显著的压缩效果。

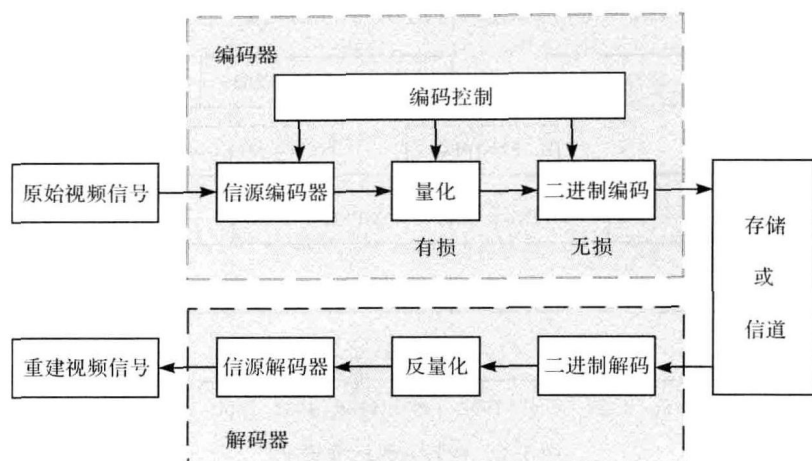


图 1.1 视频压缩编解码系统的基本结构

表 1.1 视频压缩前和压缩后的码率估计值

视 频 应 用	分辨率(像素)与帧率(fps)	压缩前码率	压缩后码率
电影(美国与日本)	480×480@24	66Mbit/s	3~6 Mbit/s
CCIR 数字电视	720×576@30	150Mbit/s	4~15Mbit/s
HDTV 视频	1920×1080@30	747Mbit/s	18~30Mbit/s
HDTV 视频	1280×720@60	664Mbit/s	18~30Mbit/s
ISDN 视频电话	352×288@29.97	36Mbit/s	64~1920kbit/s
PSTN 视频电话	176×144@29.97	9Mbit/s	10~30kbit/s

国际标准化组织(International Organization for Standardization, ISO)的运动图像专家组先后提出了 MPEG 系列标准,而国际电信联盟标准化部门(International Telecommunication Union Telecommunication Standardization Sector, ITU-T)的视频编码专家组(Video Coding Experts Group, VCEG)先后提出了 H.26x 系列标准。这两个主要的视频编码标准化组织联手组成了联合视频组(Joint Video Team, JVT)。除此之外,中国自主研发的数字音视频编解码技术标准(Audio Video coding Standard of China, AVS)^[2-4]以及由美国电影和电视工程师协会(The Society of Motion Picture and Television Engineers, SMPTE)颁布的 VC-1^[5-6]视频编码标准也在近几年涌现出来。图 1.2 是现行视频压缩标准的发展过程,其中 MPEG-4^[7]是独特地采用基于内容的编码方法,其余各标准多是采用基于波形的编码,即预测编码与变换编码相结合的基于块的混合编码方法。具体的视频压缩技术是采用预测编码方法去除时域冗余,用分块变换结合量化的方法去除空域冗余,用变长编码方法去除编码冗余。由预测编码、变换编码和熵编码

共同构成了当前视频编码标准化组织普遍认可的编码器经典框架。

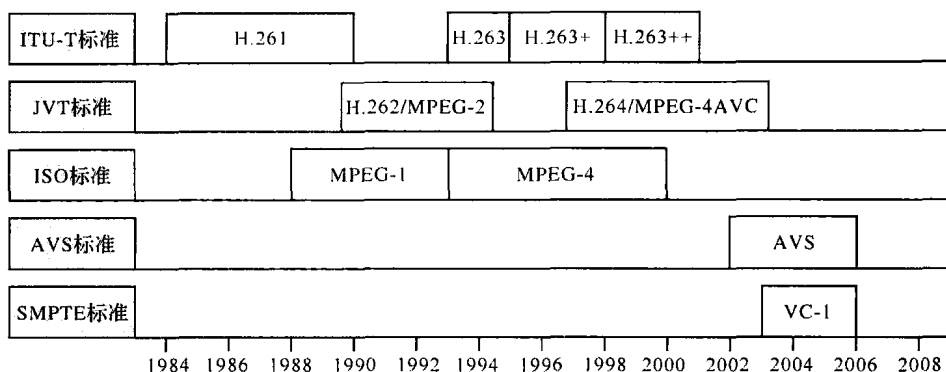


图 1.2 视频编码标准概况

视频系统的发展,依赖于超大规模集成电路(Very Large Scale Integrated circuit, VLSI)技术、通信技术和计算机技术的不断进步。视频采集的分辨率越来越高,压缩算法的效率也越来越高。在 1993 年,飞利浦公司曾报道他们的 CD-I 视频系统使用 MPEG-1 标准来压缩 1min 运动视频需要 80min 计算机时间^[8]。而现在无论计算机硬件还是视频编码算法本身都有了很大的改变。继以 MPEG-2 为核心的第一代编码标准之后,第二代基于分割、分形、模型等原理的新型视频编码标准,包括 H.264/AVC^[9-11](本书简称 H.264)、AVS、VC-1 和 MPEG-4 等,它们提供了更高的分辨率和压缩比,能够在低码率(low bit rate)的情况下提供更好的图像质量,带来更加精细的主观视频体验。实际上,传输码率下降和算法的计算复杂性上升是完全相关的。据统计,H.264 的编码器比 MPEG-2 约复杂 9 倍,AVS 的编码器复杂度约为 MPEG-2 的 6 倍^[12]。

通常,视频编码器的数据层次是基于视频元素的,如图像帧、宏块等,所以有必要介绍视频编码器的几个重要的视频元素定义。

(1) 帧(frame)、场(field)。视频的一帧或一场可用来产生一个编码图像。场多数用于电视信号,为减少电视中大面积闪烁现象,将一帧分成奇数行与偶数行组成的两个隔行的场交替显示。简单地说,视频帧可以是逐行或者隔行的视频,而场一定是隔行的视频。

(2) GOP(Group Of Pictures)、图像帧(picture frame)、片(slice)。GOP 是一组连续的编码视频图像帧的集合,而一个编码视频图像帧通常可以划分为一个或几个片,每个片包含着若干个宏块。

(3) 宏块(Macro Block, MB)、块(block)。通常,宏块是视频编码的基本单位。每个宏块是由一个 16×16 亮度 Y 像素块以及对应的一个 8×8 色度 U 像素块和一个 8×8 色度 V 像素块组成。同时,一个宏块又可以将亮度分量进一步细

分,如 H. 264 标准包含 16×16 至 4×4 范围内的不同子块尺寸(对应色度分量的尺寸为 8×8 至 2×2),这些子块简称为块。

(4) I 帧、P 帧、B 帧。图像帧可以分为 I、P、B 三种,I 帧是内部编码帧,P 帧是前向预测帧,B 帧是双向预测帧。简单地讲,I 帧是一个完整的图像或者是根据自身压缩的视频数据,而 P 帧和 B 帧记录的是相对于前面帧或者前后帧的运动变化。

(5) 参考图像(reference picture)。为了提高预测精度,视频编码器可从一组前面或者后面已编码图像中选出一个或两个与当前最匹配的图像作为帧间编码的参考图像。H. 264 标准支持最多 16 个参考图像的帧间编码。

1.2 视频压缩编码面临的新挑战

视频压缩编码是媒体处理的重要应用之一,视频编码技术已经成为数字视频广播、数字媒体存储和多媒体通信等应用的基础性、核心共性技术^[13-14]。为了确保在一定图像质量的前提下获得高压缩比,视频编码采用了多种高计算复杂度的编码技术,对处理器性能需求很高。然而,现有通用处理器无法满足较高清晰度视频编码的性能要求;专用硬件设计虽然满足了视频压缩编码的性能需求,但是设计周期长,缺乏灵活性,不能适应编码算法的快速发展。另外,人们对质量和性能的追求永无止境。在多领域的数字视频应用中,视频图像的清晰度需求持续提高,正在从普通分辨率向标清、高清甚至超高清发展,随之而来的是视频压缩编码算法复杂度和计算性能需求的全面提升。因此,迫切需要为视频编码加速寻求新的解决方案。

视频压缩编码获得的高压缩比是以编码算法的高复杂度为代价的。视频压缩编码的高计算复杂度对处理器的处理能力提出挑战,通用可编程处理器通常无法满足高分辨率视频实时压缩编码的性能需求。而高清视频的实时编码对处理器性能提出了更苛刻的需求。例如,对于 HDTV 720P(1280×720 像素)@30fps 规格的 H. 264 编码器,需求达到 3600GIPS(Giga Instructions Per Second)的处理速度和 5570GB/s 的访存带宽^[15-16],如此巨大的计算负载远超过当前通用处理器的计算能力。而且,人们对视频清晰度的追求使视频分辨率越来越高,以 HDTV 业务为例,日本多家著名厂商已经宣布支持清晰度为 3840×2160 像素的视频,并计划开发清晰度高达 7680×4320 像素的超高清视频业务,如此高分辨率的视频使压缩编码计算量大的问题雪上加霜。以目前处理器的处理能力,采用传统串行处理方式远远达不到实时编码的性能。因此,必须采用各种方法对视频编码器进行加速。目前对于视频压缩编码加速的研究主要集中在算法加速、专用硬件加速和软件并行加速三个层次。

算法加速的目的在于从算法设计角度减小计算量,提高编码速度。视频编码中的多个模块具有计算复杂的特征,因此针对各个模块的快速算法伴随着视频编码标准的发展被不断开发出来,它们包括钻石搜索算法^[17]、非对称十字型多层次六边形格点算法^[18]等针对运动估计的快速算法,自适应模式选择算法^[19]等针对帧内预测的快速算法。算法层加速的效果是比较好的,尤其是针对某些单个模块的快速算法,在稍微牺牲视频质量的前提下有效降低了编码时间。例如,针对运动估计的非对称十字型多层次六边形格点算法与全搜索算法相比,可以降低运动估计中 80% 的计算量。但是,算法加速往往针对的是视频编码中的部分模块(如熵编码),而不是从整体考虑的加速,受摩尔定律影响,基于模块的算法加速无法有效地提高整体编码器的加速比。

专用硬件加速是指为了达到高速视频压缩的性能需求,通常设计专门的硬件负责完成视频编码处理过程。研究者通常首先分析视频编码的特征,然后根据这些特征为编码器量身定制专用的硬件处理器,包括针对运动估计^[20-21]、帧内预测^[22-23]、离散余弦变换^[24]、小波变换^[25]、变长编码^[26]等模块或核心算法的硬件加速,以及整体编码^[16]的硬件加速。由于编码标准普遍采用基于块的编码方式,大多数情况下对各个块的处理模式是相同的,因此这些专用硬件处理器一般采用宏块或块的流水执行方式,通过保持高吞吐率来获得加速。针对编码模块的专用硬件设计,通常关注编码模块中最细微的运行时特征,从底层的设计上对编码性能进行优化,这样做的优势是能够以较小的硬件开销获得极大的性能提升。但是,硬件设计的精巧性也降低了其灵活性,编码算法或视频分辨率的变化经常导致原有的硬件设计无法奏效,必须重新进行设计。另外,硬件设计的研发周期也比较长,不能适应各类编码算法的快速发展。

由于以上两种方式都难以获得令人满意的结果,因而通过软件并行的方式对视频编码技术进行加速就显得十分迫切。已有基于并行方式的视频编码加速的研究已经取得了很好的效果。然而,面对视频编码的高速发展,现行的并行加速实现方式面临严峻挑战。

首先,当前视频编码器结构和组成各异、图像规模不同、硬件处理平台也有区别,能否灵活面对这些问题是并行编码器设计的关键。其次,受限于压缩编码理论原理,视频编码具有种种约束,与并行计算模型不适配,难以有效地发挥并行计算的高性能。第三,实时视频编码的性能需求随着分辨率的升高而增大,超高分辨率视频实时编码的性能需求过高,超过并行处理器的计算能力。同时,部分并行处理器的执行效率低,也造成了计算性能的下降。这些都是基于并行计算模型的视频编码器所必须面对的问题。

综上所述,基于并行计算模型的视频编码面临的新问题可以归结为以下三类。

(1) 扩展性问题。当前的视频编码领域存在图像规模不同、硬件执行平台各异、编码器结构和组成也有区别的现象,视频编码器必须面对这些方面的扩展。基于并行计算模型的视频编码面临的扩展性问题包括并行粒度扩展性、并行度扩展性、图像规模扩展性和编码模块扩展性等四个方面。

(2) 相关性问题。相关性问题的存在源于视频压缩的理论原理,即利用图像内部、图像之间或其他方面的相关性对视频图像进行压缩。因此,相关性问题广泛存在于视频编码的多个模块中,包括帧间预测、帧内预测、熵编码和去块滤波在内的核心编码模块都有各种不同的相关性问题。这些相关性问题在处理顺序上形成了优先约束,将视频编码限制在串行执行模式内,阻碍了视频编码的并行化进程。

(3) 计算性能问题。高分辨率视频编码对性能的极致需求迫使计算性能成为基于并行计算模型的视频编码器所必须关注的关键问题。例如,HDTV 720P@30fps 规格的 H. 264 编码器需要 3.6TIPS(Tera Instructions Per Second)的计算性能,而 7680×4320 像素@60fps 的超高清视频实时编码所需的计算性能将达到 720P@30fps 的 40 倍以上。然而,现有的并行处理器的性能仍然不够。例如,AMD 公司于 2009 年发布的高性能 GPU 芯片 HD5870 标称的峰值性能为 544GFLOPS(Giga Floating Point Operations Per Second),而 NVIDIA 公司发布的 Tesla C2050 的峰值性能也不过 1TOPS(Tera Operations Per Second)。如果仅采用软件并行方法而不使用专门硬件加速,则远远不能满足高清实时编码的性能需求。

1.3 视频编码并行化潜力

国际上通行的视频编码标准包括 H. 26x 系列、MPEG 系列等,它们的推出都顺应了时代的需求,但在编码模块的构成和编码模块的具体实现上都有差别。例如,H. 264 和 AVS 变换块的大小分别是 4×4 像素和 8×8 像素;MPEG-4 则支持面向图像内容的编码方法和离散小波变换方法;H. 264 采用了可变块大小的运动补偿,多参考帧等预测技术。但是,总体而言这些视频编码标准都采用了混合编码框架,如图 1.3 所示。该框架包括预测编码模块、变换量化编码模块、熵编码模块和去块滤波模块等,是视频编码器的一般框架,也是经典框架。

下面以 H. 264 编码为例介绍视频编码流程。为了利用时间上和空间上的相关性,编码流程分为前向通路和后向通路。前向通路负责根据参考帧对当前帧进行预测、变换、量化、熵编码,然后生成当前帧的码流。后向通路负责生成重构帧,作为后续编码过程的参考帧。

(1) 前向通路编码过程。当前待编码的图像帧 F_n 首先进入预测编码模块,