

分布式视频编码中的边信息改进算法

张晓星 刘冀伟 胡广大 崔朝辉

(北京科技大学信息工程学院 北京 100083)

摘 要 边信息是影响分布式视频编码系统率失真性能的关键因素之一。针对传统的运动补偿帧内插方案的固定宏块划分法忽略了视频序列中各区域运动强烈程度不均的特性,提出基于图像活跃度的灵活宏块划分算法。在前向运动估计之前,对相邻关键帧的差值图像统计活跃度,根据各区域的运动强烈程度自适应地调整宏块大小。实验结果表明,对于不同运动特性的序列,本算法可以明显提升边信息帧的重建质量,使系统的率失真性能提高 0.3~1.3dB,有效提高了系统性能。

关键词 分布式视频编码,图像活跃度,帧内插,边信息

中图法分类号 TN919.81

文献标识码 A

Improved Side Information Generation Algorithm in Distributed Video Coding

ZHANG Xiao-xing LIU Ji-wei HU Guang-da CUI Zhao-hui

(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China)

Abstract Side information generation is the most critical factor leading to the performance of the distributed video coding(DVC) scheme. An improved macroblock partition technique according to image activity measure(IAM) was proposed for the motion compensation frame interpolation process with respect to its neglect of the motion intensity inequality at different regions. Before the forward motion estimation, IAM of the residual image between adjacent key frames was calculated to estimate motion activity, by which various macroblock sizes are judged for various regions. Experimental result shows that for different video sequences, 0.3~1.3dB gain in the rate-distortion performance is achieved by applying the proposed algorithm in the DVC system, thus it effectively improves the system performance.

Keywords Distributed video coding, Image activity measure, Frame interpolation, Side information

1 引言

在传统的数字视频编码技术中,诸如 ISO/IEC 的 MPEG 和 ITU-T 的 H. 26X 等系列,以基于宏块的变换编码和基于运动估计与运动补偿的帧内预测编码为基础,配合量化、熵编码、预测等技术,获得了很好的压缩性能。在这种编码方案中,编码端主要包括变换、量化、熵编码、对应的解码过程以及运动估计和运动补偿等技术^[1,2],以提取视频序列中的时间、空间和统计冗余。根据有损预测理论,编码端通常都隐含一个解码器,此解码器可以实现解码端大部分功能。显然这是一种不对称的编码方式。据统计,运动估计和运动补偿环节占用了编码端 50%~70% 的资源,使得编码端复杂度是解码端的 5~10 倍。这种不对称的编码方式更多地适用于电视广播等一次编码、多次解码的应用场合。

近年来,随着无线多媒体传感器网络、无线低功耗视频监控网络、无线 PC 相机和移动视频电话等应用的推广,视频编码技术有了新的挑战:1)编码器功耗低,复杂度低,这对无线摄像头的功耗控制尤为重要;2)由于传输速率的限制,要求编码器有较高的压缩效率;3)无线传输的高误码率要求编码系统有较好的鲁棒性。针对这些要求,一种新的视频编码技

术——分布式视频编码(Distributed Video Coding, DVC)开始受到关注,其特点是编码简单、解码较复杂、压缩性能接近传统的编码方式、抗误码能力强。

分布式视频编码的思想最早分别由 Slepian 和 Wolf, Wyner 和 Ziv 在信息编码理论上提出并且加以论证。Slepian-Wolf 定理^[3]论述了相关信源进行无损编码需要满足的条件,指出相关信源“独立编码-联合解码”的压缩效率能够达到相关信源“联合编码-联合解码”的水平。Wyner-Ziv 定理^[4]主要讨论在有损编码的情况下相关信源编码的率失真函数描述,通常 Wyner-Ziv 编码可以等效成信源经过量化后再进行 Slepian-Wolf 编码。这种方法突破了传统的视频编码的束缚,把编码端的运动估计和运动补偿转移到解码端来进行,在不影响重建视频质量的前提下,以增加解码端运算量为代价使编码端的复杂度大大降低,满足了低能耗、实时的特性要求。同时,由于编码端包含信道编码等技术,使得分布式视频编码在鲁棒性方面有着先天的优势,基本满足了上述新的应用领域要求,逐渐成为国内外研究的热点。

DVC 编码器的编码效率很大程度上取决于边信息(Side Information, SI)的重建质量。边信息是解码端对 WZ 帧的近似估计,边信息越精确,信道解码所需要的校验位就越少,码

率越低;同时,因为其直接参与最终的 WZ 帧重建,同样影响着整体重建质量,所以边信息精确化一直是影响分布式视频编码系统的率失真性能的关键问题之一。

传统的获得边信息的方法一般都是基于前后最近邻关键帧之间运动估计的帧内插技术,并没有脱离传统视频编码技术的范畴。但两者对于运动矢量的要求不同:前者由于 WZ 帧对解码端不可见,要求运动矢量尽可能接近物体真实的运动情况;后者一般只要求运动矢量的率失真性能最好。分布式编码这种新的要求使原有的固定宏块大小的分割模式已经不再适用,而传统的宏块分割方案(如 H. 264)仅仅对 16×16 的固定宏块进行分割。对于视频序列中诸如图像背景之类稳定不变或者运动相对平缓的区域,更大的宏块尺寸不仅能降低计算复杂度,还能有效地避免运动矢量噪声;同时对于运动相对剧烈的区域,小尺寸宏块可以使运动矢量更加精细,提高内插质量。本文提出一种基于图像活跃度(Image Activity Measure, IAM)准则的分割方案,其以整个图像为对象进行更加灵活的宏块划分,获得更高质量的边信息,从而提高 DVC 的压缩性能。

2 Wyner-Ziv 视频编码系统

近年来越来越多的学者、组织已经开始对 DVC 进行深入的研究,并建立了相应的 DVC 系统,其中最典型的有损 DVC 系统是 A. Aaron 等提出的 Wyner-Ziv 视频编码方案^[5,6]。

这种分布式视频编码系统,首先将原始视频序列分为 Key 帧和 Wyner-Ziv 帧(GOP=2 时,一般将奇数帧 X_{2i+1} 作为 Key 帧,偶数帧 X_{2i} 作为 Wyner-Ziv 帧)两部分。Key 帧采用传统帧内编码模式编码, Wyner-Ziv 帧采用 Wyner-Ziv 编码器进行编码。编码框架如图 1 所示,首先对 WZ 帧做 4×4 块的 DCT 变换,经过标量量化后得到 DCT 量化系数,按宏块中系数相应位置提取相同位置的系数组成条带,分别对每一个系数带比特面分解,采用低密度奇偶校验码(Low Density Parity Check Code, LDPC)对各个比特面进行独立编码。编码后只将校验位存储在缓存中,根据解码端的反馈信息,传输部分校验位到解码端。

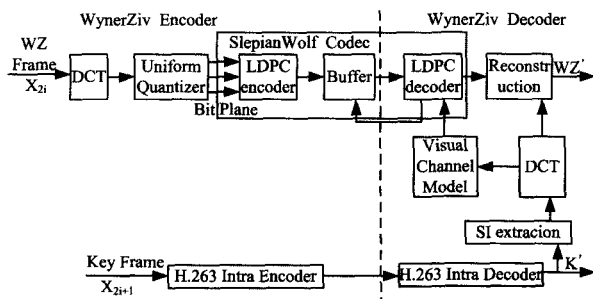


图 1 变换域 Wyner-Ziv 视频编解码框图

解码时,在相邻两个 Key 帧之间用运动估计帧内插方法得到原始 WZ 帧的边信息帧 SI, DCT 变换后得到边信息的变换系数,联合虚拟信道模型计算各个比特面信息位的信道似然值,与接收到的部分校验位共同输入到 LDPC 解码器中联合解码。如果不能正确解码,则向编码端申请更多的校验位,否则利用成功解码后的比特面信息和边信息变换系数联合重建出解码后的 WZ 帧 WZ' 的变换系数,最后经过 IDCT 得到重建图像。

在整个过程中,边信息生成得越准确,正确解码所需要的校验位就越少,编码端所发送的信息量就越少, Wyner-Ziv 系统的压缩率就越高;同时,因为边信息直接参与 WZ 帧的重建,其准确与否影响着 WZ 帧的重建质量。所以,在 DVC 系统中边信息的质量起着非常关键的作用^[7]。

3 改进的 Wyner-Ziv 编码系统

3.1 图像活跃度

图像活跃区域是图像中有较强的边缘和纹理的区域。研究表明,在有损图像压缩中,大部分人为痕迹以及误差恰巧都产生在这些区域,而人类的视觉对此极其敏感。早年文献有提出基于方差、边缘、变换、梯度等的图像活跃度(Image Activity Measure, IAM)定义^[9-12]。本文做如下定义,对文献^[9]进行了修正:

$$IAM(I) = \frac{1}{X \times Y} \left[\sum_{i=1}^{X-1} \sum_{j=1}^Y |I(i, j) - I(i+1, j)| + \sum_{i=1}^X \sum_{j=1}^{Y-1} |I(i, j) - I(i, j+1)| \right] \quad (1)$$

3.2 基于 IAM 的灵活宏块分割方案

本文提出一种基于图像活跃度的灵活的宏块划分算法,如图 2 所示。算法的流程如下:

- (1) 读入原始图像。
- (2) 计算原始图像的 IAM 值。如果小于阈值 T , 则终止程序。否则, 原始图像被标记为需要进行划分, 并作为块列表的第一个元素。
- (3) 从块列表中取出一个需要划分的块。如果没有, 则终止程序。

- (4) 由宏块的边长情况确定划分模式:

- a) 若 $blockHeight > blockWidth$, 则将宏块横向划分为两个子块, 其中一个子块为允许最大化子块, 即

$$UpSubblockHeight \text{ or } DownSubblockHeight = V = 2^{\lceil \log_2 blockHeight \rceil - 1}$$

$$IAM_{up} = IAM(UpSubblock) + IAM(DownSubblock), \text{DownSubblockHeight} = V$$

$$IAM_{down} = IAM(UpSubblock) + IAM(DownSubblock), \text{UpSubblockHeight} = V$$

分割模式

$$TYPE = \begin{cases} \text{HoriUp}, & IAM_{up} \leq IAM_{down} \\ \text{HoriDown}, & IAM_{up} > IAM_{down} \end{cases} \quad (2)$$

- b) 若 $blockHeight < blockWidth$, 则将宏块纵向划分为两个子块, 其中一个子块为允许最大化子块, 即

$$LeftSubblockWidth \text{ or } RightSubblockWidth = V = 2^{\lceil \log_2 blockWidth \rceil - 1}$$

$$IAM_{left} = IAM(LeftSubblock) + IAM(RightSubblock), \text{RightSubblockWidth} = V$$

$$IAM_{right} = IAM(LeftSubblock) + IAM(RightSubblock), \text{LeftSubblockWidth} = V$$

分割模式

$$TYPE = \begin{cases} \text{VertLeft}, & IAM_{left} \leq IAM_{right} \\ \text{VertRight}, & IAM_{left} > IAM_{right} \end{cases} \quad (3)$$

- c) 若 $blockHeight = blockWidth$, 则将宏块平均划分为横向或纵向两个子块:

$$IAM_{Hori} = IAM(UpSubblock) + IAM(DownSubblock)$$

$$IAM_{vert} = IAM(Leftsubblock) + IAM(Rightsubblock)$$

分割模式

$$TYPE = \begin{cases} Hori, & IAM_{Hori} \leq IAM_{vert} \\ Vert, & IAM_{Hori} > IAM_{vert} \end{cases} \quad (4)$$

(5) 选定划分方式并且划分子块之后,从块列表删除原宏块,计算新子块 IAM 值。如果小于阈值 T ,则标记为划分完成;否则标记为需要进行划分,将该子块添加到块列表中。返回(3)。

由于视频序列本身的长宽限制,限定在划分过程中,对于 4×4 宏块不再细分,允许最大宏块尺寸为 64×64 。本分割算法可以将图像划分为 $2m \times 2n$ 大小的一系列宏块,图像中平缓区域具有较大的分块,复杂区域有比较精细的划分,如图 3 所示。

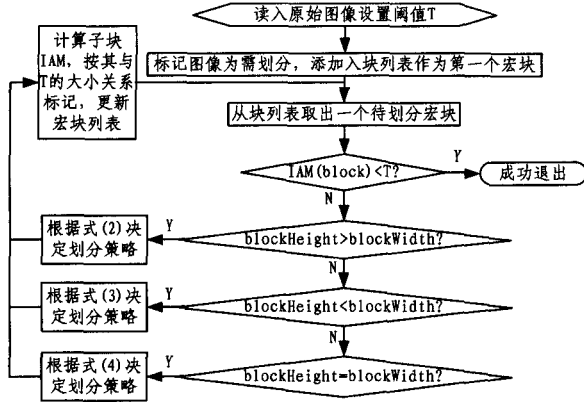


图 2 基于图像活跃度的灵活宏块划分算法

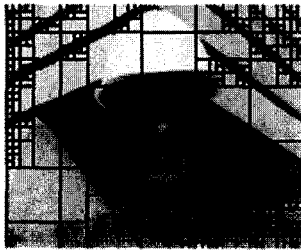


图 3 Threshold=15, foreman 序列 #2

3.3 基于灵活分块的边信息改进算法

在大多数关于 Wyner-Ziv 方案的文献中,边信息的估计基本都是基于运动估计的帧内插技术, J. Ascenso 提出的算法是比较经典的一种。在内插过程中,无法获得目标帧的任何原始信息,这就要求运动矢量的估计要尽可能接近运动目标的真实运动轨迹。对于视频序列中运动相对平缓的区域,更大的宏块尺寸不仅能降低计算复杂度,还能有效地避免运动矢量噪声;同时对于运动相对剧烈的区域,理应采用更小尺寸的宏块使运动矢量更加精细,提高内插质量。本文在文献[8]的基础上对参与帧内插的两幅 Key 帧的差值图像应用自适应宏块尺寸调整算法,以此作为运动估计时的宏块分割方案,提高边信息质量。改进后的流程框图如图 4 所示。

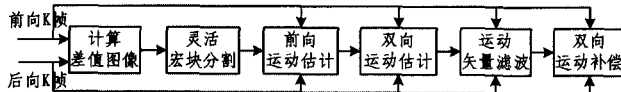


图 4 改进的帧内插方案

在前向运动估计过程中,首先用块匹配算法在 X_F 和 X_B 之间估计运动矢量。对参考帧 X_F 中的块做前向运动估计,

在 X_B 中寻找相应的匹配块。一般来说,要求目标块与匹配块之间的、在某种匹配准则下的代价函数(cost function, CF)最小。本文用改进后的带有边缘信息权值的平均绝对误差(mean absolute difference, MAD)作为匹配准则。 M 代表搜索范围, N 代表块中像素个数, B 代表 X_F 的目标块, Residual 代表目标块和匹配块的差值块。运动矢量在搜索范围内以整数像素位置寻优。

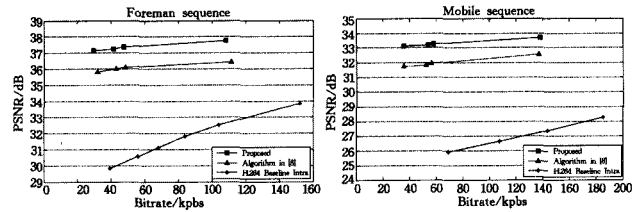
$$(MV_x, MV_y) = \arg \min_{d_x, d_y} CF(d_x, d_y), d_x \in [-M, M], d_y \in [-M, M]$$

$$MAD(d_x, d_y) = \frac{1}{N} \sum_{x,y \in B} |X_F(x, y) - X_B(x + d_x, y + d_y)|$$

$$CF(d_x, d_y) = MAD(d_x, d_y) \times \lfloor 1 + k \times \sqrt{IAM(Residual)} \rfloor$$

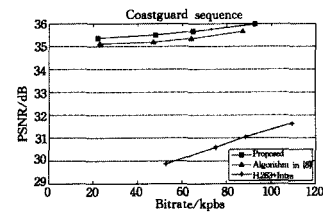
4 实验结果与分析

为了评估本文提出的算法的有效性,以图 1 所述的编码框架为软件实验平台,对典型图像序列 Foreman、Mobile 和 Coastguard(QCIF, 30fps) 的 1~101 帧进行测试。测试过程中采用如下条件:(1)设定 GOP 长度为 2,即奇数帧为 Key 帧,偶数帧为 WZ 帧;(2)采用 4 种量化等级的均匀量化器,产生不同的输出码率,获得不同的率失真点;(3)在测试过程中只计算 WZ 帧(帧率为 15fps)的平均码率和平均 PSNR;(4)在文献[8]算法中,做 8×8 块大小、搜索步长为 2、搜索范围为 8 的运动估计,双向运动估计的参数 $M_x = 2$;本文算法除宏块大小外,各参数均与对比算法相同。分别用上述两种方案对图像序列编码。测试结果如图 5 所示。



(a) Foreman sequence

(b) Mobile sequence



(c) Coastguard sequence

图 5 WZ 系统 RD 性能对比

可以看出,文献[8]的固定宏块尺寸分割方案,虽然对于运动对象大小符合预定尺寸的视频序列有较好率失真性能,但对于运动对象大小多样化的视频序列的性能表现并不完美。考虑到实际应用环境中视频本身的多样性,显然原方案已经不再适用;本文方案考虑这一缺陷,引入了灵活的宏块分割算法,有效地提高了 DVC 系统的对不同序列的适应能力,比原系统的率失真性能提高了 0.3~1.3dB。经测试,算法对其他视频序列也有着不同程度的提升,尤其是运动对象复杂的视频序列。

结束语 本文针对 DVC 系统中视频序列各区域运动强烈程度不均的特性,提出了自适应宏块划分算法。算法在运

(下转第 295 页)

DAG2 任务比重不同时的布线连通率和算法开销时间,将这 3 种任务按表 1 所列比重分为 10 组,分别进行布线连通率和算法开销的对比仿真。

表 1 独立任务、DAG1 和 DAG2 任务比重不同时的分组

任务类型	分组	1	2	3	4	5	6	7	8	9	10
独立任务		75%	72%	69%	66%	63%	60%	57%	54%	51%	49%
DAG1		15%	17%	19%	21%	23%	25%	27%	29%	31%	33%
DAG2		10%	11%	12%	13%	14%	15%	16%	17%	18%	19%

从图 9 中可得,随着 DAG1 和 DAG2 任务比重的加大,DAG 任务的布线连通率逐渐减小,算法的开销时间不断增加,最大时由 96% 减到 74%,即 DAG 任务比重对 μ_1 的影响很大。布线数目增多,导致 μ_1 减小; μ_2 由最大时 91% 减到 58%,随 DAG 任务比重的加大减小,可得出 μ_2 更易受 DAG 任务比重的影响。主要是由于布线过程中,尤其是 DAG2 任务,布线必须全部一次性连通,否则视为无效布线。DRS-TCW 开销时间 t_1 远大于 FF 开销时间 t_2 ,并随分组序号呈增加的态势。这主要是由于 DAG 任务增加,对任务进行相对位置方向的搜索耗时加大。对于在线布局布线算法,需在任务分组、连通率和开销时间之间取一平衡点。第 4 组即 3 种任务分别占 66%、21% 和 13% 时,连通率 μ_1 和 μ_2 较高,算法开销时间较少,且 DAG 任务比重较大,为 34%。

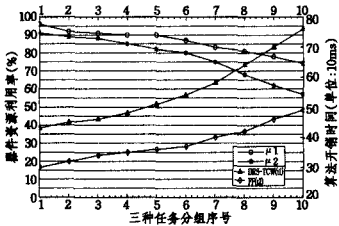


图 9 任务比重不同时的布线连通率和算法开销时间对比

(上接第 277 页)

动估计之前实时地对每一帧进行基于 IAM 的灵活宏块尺寸判别,根据不同的码率和重建质量的要求,可以灵活地调整最大和最小宏块尺寸及 IAM 阈值 T ,以保证解码端能够得到较好的边信息,适用于实际视频编码系统。实验结果表明,针对具有不同运动特性的视频序列,本文提出的两种算法使得 DVC 系统的率失真性能提升了 0.3~1.3dB。进一步的工作将考虑如何使帧内插技术更好地适应大 GOP 序列组的处理,提高 DVC 系统的压缩效率。

参考文献

[1] 向友君,雷娜,余卫宇,等.运动估计算法匹配准则研究[J].计算机科学,2009,36(9):278-280
[2] 陆寄远,朝红阳.一种快捷的高精度运动估计方法[J].计算机科学,2010,37(2):282-285
[3] Slepian S, Wolf J K. Noiseless Coding of Correlated Information Sources[J]. IEEE Transactions on Information Theory, 1973, IT-19:471-480
[4] Wyner A, Ziv J. The Rate-Distortion Function for Source Coding

结束语 本文提出的 DRS-TCW 算法有效解决了多种任务的布局布线问题,取得了较高的器件利用率和连通率。但任务布线难免会碎片化资源,而且任务布局问题是与其调度策略紧密结合的,好的调度算法结合好的布局布线算法,能进一步提高器件的资源利用率。所以未来工作将重点研究任务的调度算法和支持任务布局的碎片量化问题,以进一步提高可重构系统的整体性能。

参考文献

[1] Steiger C, Walder H, Platzer M. Operating systems for reconfigurable embedded platforms: Online scheduling of real-time tasks [J]. IEEE Trans Computer, 2004, 53(11):1393-1407
[2] Tsbero J, Septien J, Mecha H, et al. Task Placement Heuristic Based on 3D-Adjacency and Look-ahead in Reconfigurable Systems[C]//IEEE. 2006:396-401
[3] Redaelli F, Santambrogio M D, Rana V, et al. Scheduling and 2D Placement Heuristics for Partially Reconfigurable Systems[C]//IEEE. 2009:223-230
[4] 龚育昌.部分可重构系统布局的一种新算法[J].中国科技大学学报,2007,37(9)
[5] 李涛.可重构资源及硬件任务布局算法研究[J].计算机研究与发展,2008,45(2):375-382
[6] Hagemeyer J, Kettelhoit B, Koester M, et al. Design Homogeneous Communication Infrastructures for Partially Reconfigurable FPGAs[A]//Proceedings of the International Conference on Reconfigurable Systems and Algorithms [C]. Las Vegas, 2007:238-247
[7] Sharma A, Hauck S, Ebeling C. Architecture-adaptive Routability-driven Placement for Fpgas[C]//IEEE. 2005:0-7803-9362-7
[8] 李德华.异质可重构硬件任务在线布局算法研究[D].2010

with Side Information at the Decoder[J]. IEEE Transactions on Information Theory, 1976, IT-22:1-10
[5] Aaron A, Zhang R, Girod B. Wyner-Ziv coding of motion video [C]//Proceedings of the Asilomar Conference on Signals and Systems. 2002:240-244
[6] Girod B, Aaron A, Rane S, et al. Distributed video coding[J]. IEEE Special Issue on Advances in Video Coding and Delivery, 2005, 93(1):71-83
[7] Ascenso J, Prerira F. Advanced side information creation techniques and framework for Wyner-Ziv video coding[J]. Journal of Visual Communication and Image Representation, 2008, 19(8):600-613
[8] Ascenso J, Brites C, Prerira F. Improving Frame Interpolation with Spatial Motion Smoothing for Pixel Domain Distributed Video Coding[C]//5th EURASIP Conference on Speech Image Processing, Multimedia Communication and Service. 2005
[9] Saha S, Vemuri R. How Do Image Statistics Impact Lossy Coding Performance[C]//Proceedings of the International Conference on Information Technology, Coding and Computing. Las Vegas, NV, USA: IEEE Computer Society, 2000:42-47