

基于复杂度的 H.264 帧级目标比特分配算法

周志立¹, 郭淑琴¹, 阮秀凯²

(1. 浙江工业大学信息工程学院, 杭州 310014; 2. 南京邮电大学通信与信息工程学院, 南京 210003)

摘要: 针对现有 H.264 码率控制在剧烈运动场景和场景切换时, 目标比特数和量化参数计算存在的缺陷, 提出一种基于帧复杂度的改进策略, 根据图像序列的运动性和帧间差异显著的特点, 利用帧复杂度对目标比特进行重新分配, 同时利用帧的复杂度检测场景切换来调整量化参数。仿真结果表明, 该方法能使目标比特数更接近实际编码的比特数, 提高峰值信噪比, 减少峰值信噪比的波动。

关键词: 码率控制; 比特分配; 复杂度; 量化参数

H.264 Frame-level Target Bits Allocation Algorithm Based on Complexity

ZHOU Zhi-li¹, GUO Shu-qin¹, RUAN Xiu-kai²

(1. College of Information Engineering, Zhejiang University of Technology, Hangzhou 310014;

2. College of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003)

【Abstract】 Aiming at the shortcoming which the computation of target bits and Quantization Parameters(QP) in the situations of high motions and scene change of the existing H.264 rate control, this paper proposes a method of frame-level target bits allocation which based on the frame complexity to improve the above-mentioned two aspects. Due to the characteristic of moving image, which the differences in difference image is obvious, it uses frame complexity to re-allocation the target bits, at the same time, uses the frame complexity to detecting scene change to adjust the quantization parameters. Experimental results show that the improved algorithm can make target bits more closer to the actual bits, increase peak signal to noise ratio, decrease fluctuation of peak signal to noise ratio.

【Key words】 bit rate control; bits allocation; complexity; Quantization Parameter(QP)

1 概述

对于视频通信而言, 通信系统的带宽是受限的, 因此, 有效节省带宽开支显得尤为重要。为保证编码码流的顺利传输和带宽的充分利用, 须对视频编码的码率进行实时控制。现有的码率控制策略主要有 MPEG-2 的 TM5、H.263 的 TMN8^[1]、MPEG4 的 VM8 以及基于 ρ 域模型的码率控制算法^[2]等。但由于 H.264 引入了率失真优化(Rate Distortion Optimization, RDO)进行模式选择, 量化参数(Quantization Parameter, QP)同时使用于率失真优化和码率控制之中, 由此引出了著名的“鸡蛋”悖论, 从而使以往的码率控制策略难以应用到 H.264 中。

在现有的 H.264 码率控制方案中, 在帧层的码率分配所采用的策略是将剩余的比特数平均分配给未编码的帧, 然后结合缓冲区状态给各帧分配目标比特, 并未考虑到实际帧的复杂度, 这样就可能产生以下 2 类情况: (1) 由于实际帧的复杂度的不一致性, 复杂度高的帧若无法得到足够的比特数, 会使该帧的编码质量下降; 而复杂度低的帧若得到过多的比特数, 就会使其编码质量过高。从而导致视觉的不连贯性。

(2) 目标比特完全由缓存状态所决定, 当缓存接近上限时就分配给当前帧较少的比特数; 反之, 当缓存接近下限时, 则当前帧就能得到较多的比特数, 这样同样会导致视觉的不连贯性。上述 2 种情况在剧烈运动场景时尤为明显。

考虑到现有的 H.264 码率控制方案并没有很好地解决场景切换时的比特数分配问题。而在实际应用中, 当场景切换

时, 该帧的复杂度急剧增大, 这样就可能给该帧分配过多的比特数, 从而导致其他的帧只分配到较少的比特数, 信噪比降低, 最终导致视觉的不连贯性。

针对上述问题, 本文在现有 H.264 码率控制方案的基础上结合帧的复杂度调整各帧目标比特数, 利用帧的复杂度调整量化参数, 提高剧烈运动场景及场景切换时的编码质量。

2 G012 目标比特估计和量化参数计算

2.1 目标比特的估计

现有 H.264 码率控制算法可分为 4 个阶段^[3]: 目标比特分配 预测 MAD 用 2 次 R-D 模型计算 QP 执行 RDO 进行模式选择。这里帧层目标比特分配采用了缓存流传输模型: 假设一个 GOP 的第一帧为 I 帧, 其余的为 P 帧; N 表示该 GOP 的帧数; $n_j (j=1, 2, \dots)$ 表示该 GOP 的第 j 帧; $B_c(n_j)$ 表示编完第 j 帧后缓存的占有度, 则缓存流传输模型表示如下:

$$\begin{cases} B_c(n_{j+1}) = \min \left\{ \max \left\{ 0, B_c(n_j) + A(n_j) - \frac{u(n_j)}{F_r} \right\}, B_s \right\} \\ B_c(n_1) = \frac{B_s}{8} \end{cases} \quad (1)$$

其中, $A(n_j)$ 表示该 GOP 第 j 帧编码产生的比特数; $u(n_j)$ 为信道带宽; F_r 为帧率; B_s 为缓存大小。

作者简介: 周志立(1979 -), 男, 硕士研究生, 主研方向: 信号处理, 视频通信; 郭淑琴, 教授; 阮秀凯, 博士研究生

收稿日期: 2009-03-04 **E-mail:** cxzzl@163.com

对 P 帧分配比特数可划分为以下 2 个步骤：

Step1 初始化目标缓存

由于第 1 个 P 帧的量化参数是在 GOP 层中初始化中得到，因此只需对其他的 P 帧定义初始目标缓存：

$$Tbl(n_2) = B_c(n_2) \quad (2)$$

其中， $Tbl(n_2)$ 和 $B_c(n_2)$ 分别是编完第 1 个 P 帧之后的目标缓存和实际缓存占有度。

其他目标缓存定义如下：

$$Tbl(n_{j+1}) = Tbl(n_j) - \frac{B_s}{N_p - 1} \quad (3)$$

其中， $Tbl(n_j)$ 为目标缓存； N_p 为一个 GOP 中 P 帧的数量。

Step2 计算目标比特

第 j 帧的目标比特 \hat{T}_r 由目标缓存、帧率、信道带宽和实际缓存占有度共同决定：

$$\hat{T}_r = \frac{u(n_j)}{F_r} + \gamma(Tbl(n_j) - B_c(n_j)) \quad (4)$$

其中， γ 为常量，通常设为 0.75。

在计算目标比特时，同时需要考虑剩余比特数 \hat{T}_r ：

$$\hat{T}_r = \frac{R_r}{N_r} \quad (5)$$

其中， R_r 为当前 GOP 中剩余的比特数； N_r 为当前 GOP 中未编的 P 帧的数量。

最后得到目标比特 T_r ：

$$T_r = \beta \times \hat{T}_r + (1 - \beta) \times \tilde{T}_r \quad (6)$$

其中， β 为常数，通常设为 0.5。

2.2 量化参数的计算

在得到目标比特数后，通过 MPEG-4 的 2 次 R-D 模型^[4] 计算量化参数 \hat{Q}_{pc} ：

$$T_r = \frac{X_1 \times MAD}{\hat{Q}_{pc}} + \frac{X_2 \times MAD}{\hat{Q}_{pc}^2} \quad (7)$$

其中， X_1, X_2 为率失真模型参数； MAD 为预测帧与当前帧的平均绝对误差(Mean Absolute Different, MAD)，表示编码帧的复杂度。

为保证视觉的连贯性继续对量化参数作调整，假设 Q_{pp} 为当前编码帧的前一帧的量化参数，则有：

$$\tilde{Q}_{pc} = \min\{Q_{pp} + 2, \max\{Q_{pp} - 2, \hat{Q}_{pc}\}\} \quad (8)$$

最后的量化参数还必须保持在 1~51 之间：

$$Q_{pc} = \min\{51, \max\{\tilde{Q}_{pc}, 1\}\} \quad (9)$$

3 码率控制算法的改进

3.1 目标比特分配算法的改进

由式(5)可见，在对剩余比特的分配上是采用了平均分配的原则，这样目标比特率的分配仅由缓存占有度所决定，若缓存占有度较高时，新帧就会分配到较少的比特数，这样很可能引起跳帧和信噪比下降，从而可能使一个视频序列的不一致性失真，最后引起视频的不连贯性；这种情况在急剧变化或场景切换的情况下特别容易发生。而实际上由于序列的运动性，帧间的差异有时是很显著的，因此，在进行帧级目标比特率估计时，须考虑帧内帧间的复杂度，使码率控制能更精准。对帧复杂度的衡量，采用文献[5-6]提出的基于参数 $PSNR_{drop-ratio}$ 和 MAD_{ratio} 的方法。用 MAD_{pi} 表示当前帧的 MAD

预测值，用 MAD_{aj} 表示已编码的 j 帧的实际 MAD 值，则

$$MAD_{ratio} = \frac{MAD_{pi}}{\frac{1}{i-1} \left(\sum_{j=1}^{i-1} MAD_{aj} \right)} \quad (10)$$

$$PSNR_{drop-ratio} = \frac{PSNR_{dropi}}{\frac{1}{i-1} \left(\sum_{j=1}^{i-1} PSNR_{drop,j} \right)} \quad (11)$$

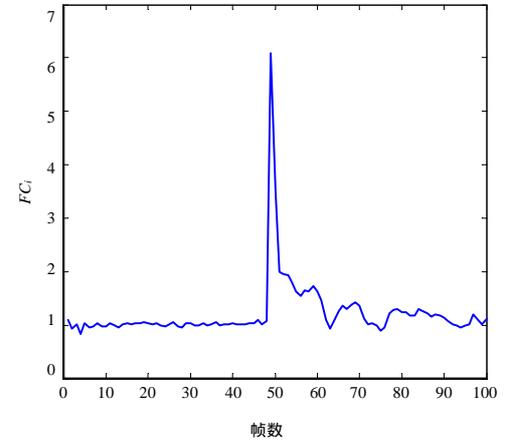
其中， $PSNR_{dropi} = PSNR_{i-1} - PSNR_{skipi}$ ； $PSNR_{skipi}$ 为假设 j 帧被跳过，根据 $j-1$ 帧重建值与 j 帧原始值计算的第 j 帧的 $PSNR$ 值。

由 $PSNR_{drop-ratio}$ 和 MAD_{ratio} 得到当前帧的复杂度 FC_i ：

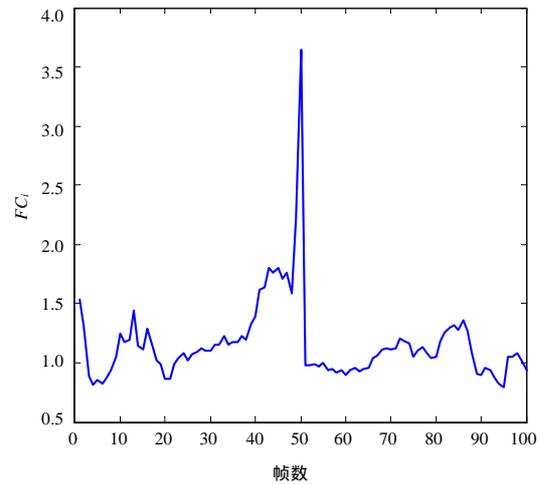
$$FC_i = \lambda \times MAD_{ratio} + (1 - \lambda) \times PSNR_{drop-ratio} \quad (12)$$

其中， λ 为权重因子，通常取 0.7。

为直观表述，本文给出 container-foreman 和 suzie-foreman 序列在第 50 帧时发生场景切换时的 FC_i 值变化图形，如图 1 所示。



(a)container-foreman 序列



(b)suzie-foreman 序列

图 1 图像序列 FC_i 值的变化

通过观察，可以发现 FC_i 能够很好地反映图像的复杂度。在图 1(a)中，后 50 帧的 FC_i 明显大于前 50 帧，这是由于 foreman 序列运动程度高于 container 序列的原因而产生的；由图 1 可见，在发生场景切换时， FC_i 值急剧上升，且 FC_i 值随着运动的剧烈程度发生变化。

基于上述分析，本文对式(5)中的比特分配作如下调整：

$$\hat{T}_r = \begin{cases} 0.8 \times FC_i \times \frac{R_r}{N_r} & FC_i \leq 1.1 \\ (0.6 + 0.3 \times FC_i) \times \frac{R_r}{N_r} & 1.1 < FC_i \leq 2.0 \\ 1.2 \times \frac{R_r}{N_r} & FC_i > 2.0 \end{cases} \quad (13)$$

同时，为了突出帧复杂度对目标比特影响，将式(6)中的 β 值加大为0.7。

3.2 量化参数计算与调整

计算和调整量化参数时分以下2种情况进行讨论：

(1)当目标比特小于0时

由式(4)可知当 $B_c(n_j)$ 大于 $Tbl(n_j)$ 时就有可能导致的目标比特小于0，而 $B_c(n_j)$ 增大的原因是编码时分配了过多的比特，而帧的复杂度越高则分配的比特就越多，基于这种思想，本文利用式(12)中帧的复杂度 FC_i 来调整量化参数，从而得到更精确的目标比特分配：

$$Q_{pc} = \begin{cases} Q_{pp} + 2 & FC_i \leq 1.1 \\ Q_{pp} + 3 & 1.1 < FC_i \leq 2.0 \\ Q_{pp} + 4 & FC_i > 2.0 \end{cases} \quad (14)$$

其中， Q_{pc} 、 Q_{pp} 分别表示当前编码帧和前一帧的量化参数。

(2)当目标比特大于0时

现有的码率控制策略并没有充分考虑到场景切换，而是统一利用得到的目标比特采用式(7)来计算量化参数，从而导致信噪比下降。

当场景切换时，预测得到的复杂度会急剧增大，本文提出用式(12)中帧的复杂度 FC_i 来检测场景切换。由图1可看出，在不发生场景切换时， FC_i 的值均小于2，反之 FC_i 都将急剧增大，同时大量实验结果证实了上述特点。所以将量化参数调整如下：

$$Q_{pc} = \begin{cases} \max(Q_{pp} + 4, Q_0) & FC_i \geq 2.0 \\ Q_{nor} & FC_i < 2.0 \end{cases} \quad (15)$$

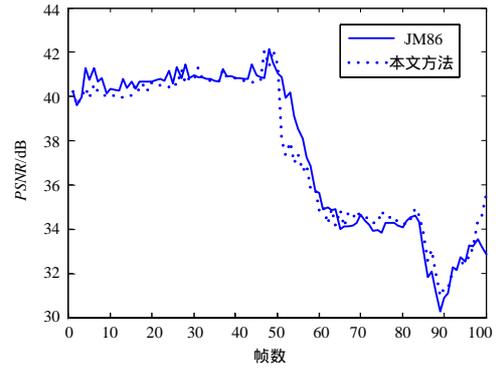
其中， Q_{pc} 、 Q_{pp} 分别表示当前编码帧和前一帧的量化参数， Q_0 表示该GOP的初始量化参数， Q_{nor} 表示利用式(7)计算得到的量化参数。

4 数值仿真及结果分析

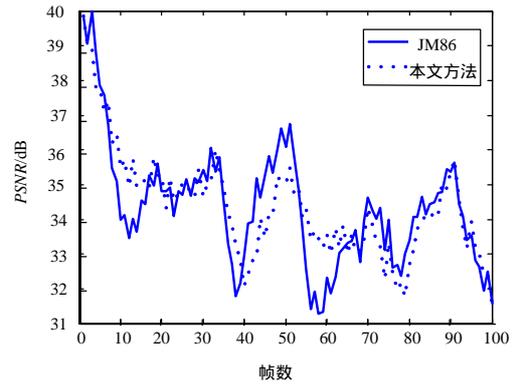
在JVT测试模型JM86上实现本文提出的方法，并与JM86作比较，仿真实验分别采用container-foreman, suzie-foreman, foreman, coastguard这4个QCIF测试序列，其中，container-foreman, suzie-foreman为场景切换序列，目标码率设为128 Kb/s，帧率为30 f/s，第一帧为I帧其余为P帧，共100帧，编码方式为CAVLC，搜索范围为16，参考帧为10帧，采用哈达玛变换。

图2为container-foreman, foreman序列的PSNR值，由图2(a)可见，在50帧处进行场景切换，当信噪比下降时，本方法能得到较好的信噪比。图2(b)为运动剧烈场景的foreman序列，可见本方法信噪比变化缓慢，能得到更好的主观质量。

图3为container-foreman, foreman序列在编码过程中每帧的目标比特与实际编码比特的差异，由图3(a)可见，在50帧处进行场景切换，传统方法在切换时分配了过多的比特给该帧，而本方法则得到了较好的改进，且本方法对于运动剧烈场景(图3(b))也能使目标比特更接近实际编码比特。

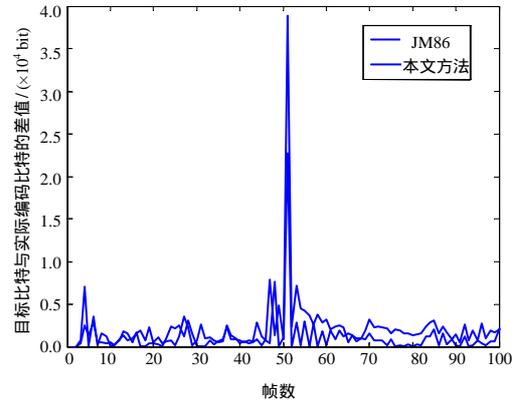


(a)container-foreman 序列

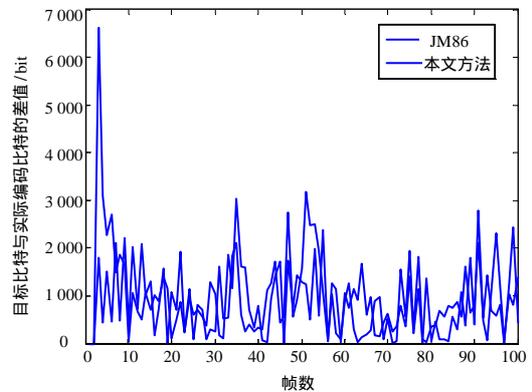


(b)foreman 序列

图2 图像序列的PSNR



(a)container-foreman 序列



(b)foreman 序列

图3 目标比特与实际编码比特的差值

表1给出了改进算法与原算法的性能比较，可见本方法 (下转第227页)