



(12) 发明专利申请

(10) 申请公布号 CN 102271248 A

(43) 申请公布日 2011. 12. 07

(21) 申请号 201110232053. 2

(22) 申请日 2011. 08. 15

(71) 申请人 复旦大学

地址 200433 上海市杨浦区邯郸路 220 号

(72) 发明人 范益波 钟慧波 沈沙 赵晋

刘家良 曾晓洋

(74) 专利代理机构 上海正旦专利代理有限公司

31200

代理人 陆飞 盛志范

(51) Int. Cl.

H04N 7/24 (2006. 01)

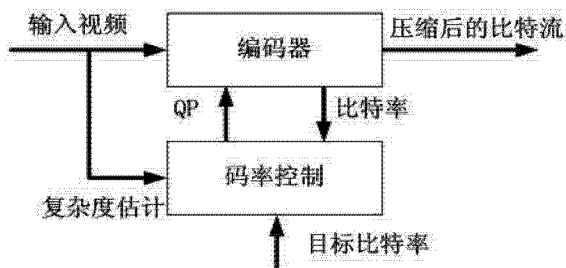
权利要求书 2 页 说明书 8 页 附图 2 页

(54) 发明名称

一种基于权重窗口模型的 H. 264/AVC 宏块级码率控制算法

(57) 摘要

本发明属于视频编码技术领域,具体为一种基于权重窗口模型的 H. 264/AVC 宏块级码率控制算法。本发明利用当前宏块周围宏块组成的一个窗口的空间相关性,以及前一帧的与当前宏块相同位置宏块及其周围宏块组成的一个窗口的时间相关性,提出了一种低复杂度的平均绝对值偏差(MAD)的预测模型和量化参数(QP)计算模型,并基于上面两种模型,提出用于 H. 264/AVC 编码器中的宏块级码率控制算法。实验表明,基于权重窗口模型的宏块级 H. 264/AVC 码率控制算法比 H. 264/AVC 的参考软件的码率控制算法得到更高的峰值信噪比(PSNR)和更小的比特偏差。



1. 一种基于权重窗口模型的QP计算模型和MAD预测模型,其特征在于,由当前宏块和其周围的8个宏块组成一个3x3的窗口;同样,对于前一帧,与当前宏块相同位置的宏块和它周围的8个宏块组成一个3x3的窗口;设MB0~MB8依次分别代表在当前编码帧的当前宏块,以及该当前宏块的左边,右边,上边,下边,左上,右上,左下,右下位置的宏块;

QP计算模型如下式所示:

$$QP[0,i] = \{S_c \times N_p \times \phi - \sum_{k=1,3,5,6} \alpha_k \times QP[k,i] - \sum_{k=2,4,7,8} \alpha_k \times QP[k,i-1]\} / \alpha_0 \quad (3)$$

其中:

$$S_c = \sum_{k=0,1,3,5,6} \alpha_k \times MAD_a[l,i] + \sum_{k=2,4,7,8} \alpha_k \times MAD_a[k,i-1] \quad (4)$$

$$N_p = \sum_{k \in W_p} (\beta_k \times QP[k,i-1]) / \sum_{k \in W_p} (\beta_k \times MAD_a[k,i-1]) \quad (5)$$

$$\phi = \frac{Bit[0,i]}{\sum_{k \in W_p} \beta_k \times Bit[k,i-1]} / \sum_{k \in W_p} \beta_k \quad (6)$$

$QP[0,i]$ 为当前宏块的QP值, α_k 和 β_k 是权重参数,其取值与其所在的位置有关, W_c 和 W_p 分别表示在当前帧和前一帧中的窗口, $0 \sim 8$ 对应的是 MB0~MB8 宏块位置, $QP[k,i]$ 表示当前帧的窗口中相应 k 位置宏块的QP, k 为 $0 \sim 8$ 的整数, $QP[k,i-1]$ 表示前一帧窗口中相应 k 位置宏块的QP; $MAD_a[k,i]$ 表示当前帧的窗口中相应 k 位置宏块的MAD, $MAD_a[k,i-1]$ 表示前一帧窗口中相应 k 位置宏块的MAD; $Bit[0,i]$ 表示当前宏块编码所用比特数, $Bit[k,i-1]$ 表示前一帧窗口中相应 k 位置宏块编码所用比特数;权重参数取值如下:

$$\alpha_k = \begin{cases} 3, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases}, \beta_k = \begin{cases} 4, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases} \quad (7)$$

MAD的预测模型如下式所示:

$$MAD_a[0,i] = \{M_p - (\sum_{l=1,3,5,6} \gamma_l \times MAD_a[l,i] - \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l,i-1]) / \sum_{l \in W_p} \gamma_l\} / \gamma_0 \quad (8)$$

其中 M_p 的计算方法为:

$$M_p = \sum_{k=0}^8 \beta_k \times MAD_a[k,i-1] / \sum_{k \in W_p} \beta_k \quad (9)$$

在上述两式中 $MAD_a[0,i]$ 表示当前宏块的MAD预测值,其中 $MAD_a[l,i]$ 和 $MAD_a[k,i-1]$

代表值和上述一致, γ_l 和 β_k 为 MAD 计算时所用到的权重, 其中 β_k 由(7)给出, γ_l 取值如下所示:

$$\gamma_l = \begin{cases} 3, & \text{if } l = 0 \\ 1, & \text{else} \end{cases} \quad (10)$$

2. 一种基于如权利要求 1 所述的 QP 计算模和 MAD 预测模型的 H. 264/AVC 宏块级码率控制算法, 其特征在于具体步骤为:

对于 GOP (画面组) 中的第一个 I 帧和 P 帧, 采用参考软件中的算法计算得到初始 QP, 设 QP_{ave} 是前一帧的所有宏块 MAD 的平均值, $QP[k-1, i]$ 是当前帧前一个宏块的 QP 值; 根据预测得到的 MAD 值和当前帧所有已编码宏块的平均 MAD 值来得到比特分数, 具体如下式所示:

$$T_c = T_r \times MAD_a[0, i] / (MAD_{ave,c} \times N) \quad (11)$$

其中 T_c 和 T_r 分别表示当前宏块的目标比特数和当前帧的剩余比特数, $MAD_{ave,c}$ 表示当前帧中所有已编码宏块的平均 MAD 值, N 表示当前帧剩余的未编码的宏块数目; T_r 通过减去已编码宏块所用比特数来更新;

如果当前宏块为当前帧的第一个宏块, 那么当前宏块的 QP 设为 QP_{ave} , 对于当前帧的其它宏块, QP 的计算方法为: 如果当前帧的剩余比特数为负值, 则当前 QP 直接设为 $QP[k-1, i] + 1$, 否则, 先给当前宏块分配比特数, 然后通过权利要求 1 中的 QP 计算模型得到 QP, 并且, QP 首先由下式来限定:

$$QP[k, i] = \min \{QP[k-1, i] + 1, \max \{QP[k-1, i] - 1, QP\}\} \quad (12)$$

为了保证图像的平滑性, QP 进一步由下式来限定:

$$QP[k, i] = \min \{QP_{ave} + 2, \max \{QP_{ave} - 2, QP\}\} \quad (13)$$

最后, QP 需要满足 H. 264/AVC 的规定, 即限定在 1 ~ 51 内:

$$QP[k, i] = \min \{51, \max \{1, QP\}\} \quad (14)$$

在完成一个宏块的编码后, 记录编码信息, 包括: 实际 MAD, 实际所用的 QP, 实际编码所用的比特数, 更新剩余比特数。

一种基于权重窗口模型的 H. 264/AVC 宏块级码率控制算法

技术领域

[0001] 本发明属于视频编码技术领域,具体为一种 H. 264/AVC 宏块级码率控制算法,可应用于 H. 264/AVC 编码器中编码码率的控制实现。

背景技术

[0002] 视频编码中的码率控制是指采用一定的方法对一个视频源进行编码,使用经过编码压缩后的比特率可以满足一些特定的限制:比如说比特率,缓存的上溢和下溢的保护等。在一些视频应用中,特别是视频通信类,其信道带宽往往是有限的。为了满足实际应用中的带宽约束,同时追求在此带宽下的最优视频质量,码率控制在视频传输应用中非常重要。这是因为过高的码率会导致比特率超过信道容量,造成数据丢失;另一方面来说,如果码率过低又会造成带宽的浪费。

[0003] H. 264/AVC 是 JVT 组织最新提出的一个视频编码标准。因为 H. 264/AVC 能够比之前一些编码标准得到更高的压缩效率和图像质量,所以它的应用越来越广泛。而码率控制作为编码器的关键技术之一,因此,用于 H. 264/AVC 的码率控制算法一直是最近几年的研究热点。根据码率控制算法中的基本单元的大小,编码控制算法可以分为三个层次:GOP 级别,帧级,宏块级别的码率控制算法。在码率控制算法中,主要需要解决两个关键的问题:一是如何给当前编码单元分配合适的比特数;二是如何根据分配的比特数编码实现最好的图像,即如何决定一个量化参数。和以前的一些视频编码标准的码率控制算法相比,H. 264/AVC 的码率控制算法实现起来会更加困难一些。这是因为在 H. 264/AVC 编码中,量化参数(QP)在码率控制算法和率失真优化(RDO)中都会用到,这会导致下面的蛋鸡悖论:为了给当前编码宏块进行率失真优化(RDO),首先要利用当前编码宏块帧的 MAD 来计算量化参数(QP),而当前编码宏块/帧的 MAD 只有在 RDO 之后才能得到。为了解决这个悖论,以往大多数的算法采用了下面表达式所示的线性模型:

$$MAD_c = a_1 * MAD_p + a_2 \quad (1)$$

上式中的 a_1, a_2 为模型参数, MAD_c 和 MAD_p 分别是指当前宏块或前一帧与当前宏块相同位置宏块的 MAD。这样,借助上述线性模型,我们可以预测当前编码单元的 MAD,通过预测得到的 MAD,采用一定的 R-Q(比特-量化参数)模型来计算编码所要采用的 QP。下面是在 H. 264/AVC 的参考软件 JM 采用的二次 R-Q 模型:

$$R_i = c_1 \frac{MAD_c}{Qstep_i} + c_2 \frac{MAD_c}{Qstep_i^2} \quad (2)$$

R_i 代表分配给第 i 个编码基本单元分配的比特数, c_1, c_2 为模型参数, $Qstep_i$ 代表该编码基本单元的量化步长。式(1)和(2)一起的四个模型参数 a_1, a_2, c_1, c_2 在编码完每个编码单元之后进行更新。

发明内容

[0004] 本发明的目的在于提出一种能够消除宏块级码率控制算法中复杂的线性回归的 H. 264/AVC 宏块级码率控制算法, 以实现良好的码率控制效果和较高的 PSNR 提高。

[0005] 为了解决前面提到的蛋鸡悖论, 所以需要在编码当前宏块之前, 预测当前宏块的 MAD。以前的算法基本都是采用前面所列的线性模型来完成, 但这个线性模型的模型参数 a_1, a_2 的更新非常复杂。而且因为其线性的关系, 所出现的误差可能会较大。为了得到编码所需要的 QP, R-Q 模型的模型参数的更新跟 MAD 的线性模型类似, 复杂度非常高。本发明可以极大的降低宏块级别的码率控制的复杂度。

[0006] 采用权重窗口预测, 充分利当前时间相关性(前一帧相同位置的宏块和其周围的宏块)和空间相关性(当前宏块周围的宏块)。当前帧的窗口和前一帧的窗口的选择如图 1 所示。对于两个窗口中不同位置的宏块的所用权重是不一样的。

[0007] 本发明利用当前宏块和其周围的 8 个宏块组成一个 3x3 的窗口。对于每个窗口, 其对应的权重也是不一样的, 这就是权重窗口这个名字的由来。同样, 对于前一帧与当前宏块相同位置宏块和它周围的 8 个宏块组成一个 3x3 的窗口。与在当前帧的窗口一样, 对于在前一帧的窗口的不同位置, 其相应的权重也是不一样的。权重窗口模型的示意图如 1 所示。设 MB0 ~ MB8 依次分别代表在当前编码帧的当前宏块(记为 MB0), 以及该当前宏块的左边, 右边, 上边, 下边, 左上, 右上, 左下, 右下位置的宏块。其中, 下标 c 表示当前帧各宏块, 下标 p 表示前一帧各宏块。针对上述的两个窗口。对于每个窗口内的每一个宏块的权重乘以量化参数(QP)的累加除以权重乘以平均绝对值偏差(MAD)的相互关系进行分析。可以得到这两者是线性的关系。

[0008] 本发明基于权重窗口模型提出了一种新的 QP 计算模型和 MAD 预测模型。

[0009] 本发明中的 QP 的计算模型如下所示:

$$QP[0, i] = \{S_c \times N_p \times \phi - \sum_{k=1,3,5,6} \alpha_k \times QP[k, i] - \sum_{k=2,4,7,8} \alpha_k \times QP[k, i-1]\} / \alpha_0 \quad (3)$$

其中:

$$S_c = \sum_{k=0,1,3,5,6} \alpha_k \times MAD_a[l, i] + \sum_{k=2,4,7,8} \alpha_k \times MAD_a[l, i-1] \quad (4)$$

$$N_p = \sum_{k \in W_p} (\beta_k \times QP[k, i-1]) / \sum_{k \in W_p} (\beta_k \times MAD_a[k, i-1]) \quad (5)$$

$$\phi = \frac{Bit[0, i]}{\sum_{k \in W_p} \beta_k \times Bit[k, i-1] / \sum_{k \in W_p} \beta_k} \quad (6)$$

$QP[0, i]$ 为当前宏块的 QP 值。 α_k 和 β_k 是权重参数, 其取值与其所在的位置有关, W_c 和 W_p 分别表示在当前帧和前一帧中的窗口, 其具体位置如图 1 所示。0~8 对应的是 MB0~MB8 宏块位置。 $QP[k, i]$ 表示当前帧的窗口中相应 k 位置(k 取 0~8 整数)宏块的 QP, $QP[k, i-1]$ 表示前一帧窗口中相应 k 位置宏块的 QP。 $MAD_a[k, i]$ 表示当前帧的窗口中相应 k 位置宏

块的 MAD, $MAD_a[k, i-1]$ 表示前一帧窗口中相应 k 位置宏块的 MAD。 $Bit[0, i]$ 表示当前宏块编码所用比特数, $Bit[k, i-1]$ 表示前一帧窗口中相应 k 位置宏块编码所用比特数。权重参数的取值如下:

$$\alpha_k = \begin{cases} 3, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases}, \beta_k = \begin{cases} 4, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases} \quad (7)$$

本发明中的 MAD 的预测模型可以由下式表示:

$$MAD_a[0, i] = \{M_p - (\sum_{l=1,3,5,6} \gamma_l \times MAD_a[l, i] - \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l, i-1]) / \sum_{l \in Wp} \gamma_l\} / \gamma_0 \quad (8)$$

其中 M_p 的计算方法为:

$$M_p = \sum_{k=0}^8 \beta_k \times MAD_a[k, i-1] / \sum_{k \in Wp} \beta_k \quad (9)$$

$MAD_a[0, i]$ 表示当前宏块的 MAD 预测值, 其中 $MAD_a[l, i]$ 和 $MAD_a[k, i-1]$ 代表值和上述一致, γ_l 和 β_k 为 MAD 计算时所用到的权重, 其中 β_k 由(7)给出, γ_l 取值如下所示:

$$\gamma_l = \begin{cases} 3, & \text{if } l=0 \\ 1, & \text{else} \end{cases} \quad (10)$$

根据前面的 MAD 和 QP 的计算模型, 本发明提出一种新的宏块级别的码率控制算法。本发明的码率控制算法方法可以分为三层: GOP 级, 帧级, 宏块级。在 GOP 级和帧级中, 比特分配和编码前的一些预调节和 H. 264/AVC 参考软件中使用的方法一样。本发明中主要的宏块级算法流程见图 3 所示, 具体描述如下:

在图 3 中, 对于 GOP 中的第一个 I 帧和 P 帧, 初始 QP 采用参考软件中的算法计算得到。 QP_{ave} 是前一帧的所有宏块 MAD 的平均值。 $QP[k-1, i]$ 是当前帧前一个宏块的 QP 值。图 2 中的比特分数操作是根据预测得到的 MAD 值和当前帧所有已编码宏块的平均 MAD 值来得到, 具体如下式所示:

$$T_c = T_r \times MAD_a[0, i] / (MAD_{ave,c} \times N) \quad (11)$$

其中 T_c 和 T_r 分别表示为当前宏块的目标比特数和当前帧的剩余比特数。 $MAD_{ave,c}$ 表示为当前帧中所有已编码宏块的平均 MAD 值, N 表示当前帧剩余的未编码的宏块数目。 T_r 是通过减去已编码宏块所用比特数来更新的。

[0010] 如果当前宏块为当前帧的第一个宏块, 那么当前宏块的 QP 设为 QP_{ave} 。对于当前帧的其它宏块, QP 的计算方法可以为: 如果当前帧的剩余比特数为负值, 则当前 QP 直接

设为 $QP[k-1, i] + 1$, 否则, 先给当前宏块分配比特数, 其分配方法如前所述。然后通过本发明中的 QP 计算方法得到 QP。为了保证宏块中不会出现方块效应, 所以 QP 首先由下式来限定:

$$QP[k, i] = \min \{ QP[k-1, i] + 1, \max \{ QP[k-1, i] - 1, QP \} \} \quad (12)$$

为了保证图像的平滑性, QP 进一步由下式来限定使其各帧间的 QP 改变不会过大:

$$QP[k, i] = \min \{ QP_{ave} + 2, \max \{ QP_{ave} - 2, QP \} \} \quad (13)$$

最后, QP 需要满足 H. 264/AVC 的规定, 限定在 1~51 内:

$$QP[k, i] = \min \{ 51, \max \{ 1, QP \} \} \quad (14)$$

在完成一个宏块的编码后, 记录编码信息, 如实际 MAD, 实际所用的 QP, 实际编码所用的比特数, 更新剩余比特数。

[0011] 本发明利用两个窗口内的 18 个宏块的相关性, 对当前宏块进行参数预测, 得到 QP 和 MAD, 去除了 JM 参考软件中的参数更新这个复杂的过程。与原始的 JM 码率控制算法相比, 可以取得更高的 PSNR 和更小的目标比特偏差。

附图说明

[0012] 图 1 是权重窗口模型的示意图。

[0013] 图 2 是码率控制示意图。

[0014] 图 3 是基于权重窗口预测的码率控制算法的流程图。

具体实施方式

[0015] 以下结合附图对本发明做进一步的描述。

[0016] 1. QP 计算模型

在实际码率控制算法中, 预测当前编码宏块的信息主要用到的是空间相关信息和时间相关信息: 如图 1 所示, 设 $MB0_c \sim MB8_c$ 分别代表在当前编码帧的当前宏块 ($MB0_c$), 以及该当前宏块的左边, 右边, 上边, 下边, 左上, 右上, 左下, 右下位置的宏块。同样 $MB0_p \sim MB8_p$ 分别代表在前一帧中对应于当前宏块 ($MB0_c$) 位置的宏块 ($MB0_p$), 以及前一帧中该宏块 ($MB0_p$) 的左边, 右边, 上边, 下边, 左上, 右上, 左下, 右下位置的宏块。针对当前帧和前一帧的两个窗口, 首先引入两个变量 N_c 和 N_p , 这两个变量的值可以由下面两个公式得到:

$$N_c = \sum_{k \in W_c} (\alpha_k \times QP[k, i]) / \sum_{k \in W_c} (\alpha_k \times MAD_a[k, i]) \quad (15)$$

$$N_p = \sum_{k \in W_p} (\beta_k \times QP[k, i-1]) / \sum_{k \in W_p} (\beta_k \times MAD_a[k, i-1]) \quad (5)$$

其中 α_k 和 β_k 是权重参数, 其取值与其窗口位置 (0~8) 所在的位置有关, W_c 和 W_p 分别表示在当前帧和前一帧中的窗口。 $QP[k, i]$ 表示当前帧的窗口中相应 k 位置 (k 为 0~8 的整数) 宏块的 QP, $QP[k, i-1]$ 表示前一帧的窗口中相应 k 位置宏块的 QP。 $MAD_a[k, i]$ 表示当

前帧的窗口中相应 k 位置宏块的 MAD, $MAD_a[k, i-1]$ 表示前一帧的窗口中相应 k 位置宏块的 MAD。对于图 1 中的两个窗口, 为了确定窗口大小, 引入 W_s 这个变量来代表窗口大小。为了确定 N_c 和 N_p 在不同的窗口大小 W_s 的情况下的关系, 做了许多测试, 得到当 $W_s \geq 3$ 时, 可以有一个近似关系: $N_c = N_p$, 本发明选择 $W_s = 3$ 来作为权重模型。

[0017] 由图 1 可知道, 在编码当前宏块 $MB0_c$ 时, $MB2_c, MB4_c, MB7_c, MB8_c$, 这几个宏块还没有进行编码, 所以这几个宏块的信息由前一帧的 $MB2_p, MB4_p, MB7_p, MB8_p$ 来代替, 由 N_c 可以修正为下面 $N_{c,2}$ 。

$$[0018] \quad N_{c,2} = \left\{ \sum_{k=0,1,3,5,6} \alpha_k \times QP[k, i] + \sum_{k=2,4,7,8} \alpha_k \times QP[k, i-1] \right\} / \left\{ \sum_{k=0,1,3,5,6} \alpha_k \times MAD_a[l, i] + \sum_{k=2,4,7,8} \alpha_k \times MAD_a[k, i-1] \right\} \quad (16)$$

和 $N_c = N_p$ 一样, 根据实验结果, 在 $W_s = 3$ 时 $N_{c,2} = N_p$ 。在做宏块级别的码率控制时, 编码每个宏块的比特数也是需要考虑的。这里我们联系宏块实际编码的比特数, 由 $N_{c,2} = N_p$ 得到(1)式即:

$$QP[0, i] = \{ S_c \times N_p \times \phi - \sum_{k=1,3,5,6} \alpha_k \times QP[k, i] - \sum_{k=2,4,7,8} \alpha_k \times QP[k, i-1] \} / \alpha_0 \quad (3)$$

其中 S_c 和 N_p 分别由(4)和(5)给出

权重参数由经过实验得到如(7)式所示。0~8 对应的是图 1 中的 $MB0 \sim MB8$ 。 ϕ 为一个调节参数, 其由(6)给出即:

$$\phi = \frac{Bit[0, i]}{\sum_{k \in W_p} \beta_k \times Bit[k, i-1] / \sum_{k \in W_p} \beta_k} \quad (6)$$

其中 $Bit[0, i]$ 是当前帧宏块的编码比特, $Bit[k, i-1]$ 是前一帧中窗口各相应位置宏块的编码所用的比特, β_k 的取值和前面所用的值一样。 ϕ 的作用是根据消耗的比特来调节计算的 QP 值。一般情况下, 在编码当前宏块时, $Bit[0, i]$ 一般是能通过比特分配得到。比特分配一般是要根据预测得到的当前 MAD。当

$Bit[0, i] > \sum_{k \in W_p} \beta_k \times Bit[k, i-1] / \sum_{k \in W_p} \beta_k$ 时, 意味着当前宏块的用到的比特数可

能会大于其前一帧窗口中用到的比特数, 会得到 $\phi > 1$, 这样通过 ϕ 可以得到一个相对大一点

的 QP。反之, 如果 $Bit[0, i] < \sum_{k \in W_p} \beta_k \times Bit[k, i-1] / \sum_{k \in W_p} \beta_k$, 那就是说当前宏块

所用的编码比特数可能会比前一帧窗口平均所用比特数要小, 所以可以得到 $\phi < 1$, 这个值用来调节来得到一个相对小的 QP。通过前面计算得到的 QP 值可能由下面的式进行一个四舍五入取整的操作:

$$QP = \lfloor QP[0, i] + 0.5 \rfloor \quad (17)$$

这里 $\lfloor \cdot \rfloor$ 是一个取下值的操作。进一步限定 QP 的操作可以由后面的来进行。

[0019]

2. MAD 预测模型

为了解决前面所描述的蛋鸡悖论,我们需要在编码当前宏块前预测当前宏块的 MAD 值 $MAD[0, i]$ 。和计算 QP 时所用到的权重窗口模型一样,我们采用相同的权重窗口模型来预测当前宏块的 MAD,这里引入两个变量 M_p 和 M_c 其值为如下所示:

$$M_p = \sum_{k=0}^8 \beta_k \times MAD_a[k, i-1] / \sum_{k \in W_p} \beta_k \quad (9)$$

$$M_c = \left\{ \sum_{l=0,1,3,5,6} \gamma_l \times MAD_a[l, i] + \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l, i-1] \right\} / \sum_{l \in W_p} \gamma_l \quad (18)$$

这里 γ_l 和 β_k 为 MAD 所用的权重,其中 β_k 由(7)给出。通过实验分析 M_p 和 M_c ,在 $W_s \geq 3$ 时,可以有 $M_p = M_c$ 。所以我们可以(8)式计算 $MAD_a[0, i]$,即

$$MAD_a[0, i] = \left\{ M_p - \left(\sum_{l=1,3,5,6} \gamma_l \times MAD_a[l, i] - \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l, i-1] \right) / \sum_{l \in W_p} \gamma_l \right\} / \gamma_0 \quad (8)$$

其中 $MAD_a[0, i]$ 表示当前宏块的 MAD 值, γ_l 的取值根据实验所得如(10)所示。

[0020] 3. 码率控制算法流程

根据前面的 MAD 和 QP 的计算模型,本发明提出一种新的宏块级别的码率控制算法。本发明的码率控制算法方法可以分为三层:GOP 级,帧级,宏块级。在 GOP 级和帧级中,比特分配和编码前的一些预调节和 H. 264/AVC 参考软件中使用的方法一样。本发明中主要的宏块级算法流程见图 3 所示,具体描述如下:

在图 3 中,对于 GOP 中的第一个 I 帧和 P 帧,初始 QP 采用参考软件中的算法计算得到。 QP_{ave} 是前一帧的所有宏块 MAD 的平均值。 $QP[k-1, i]$ 是当前帧前一个宏块的 QP 值。图 3 中的比特分配操作是根据预测得到的 MAD 值和当前帧所有已编码宏块的平均 MAD 值来得到,具体如下式所示:

$$T_c = T_r \times MAD_a[0, i] / (MAD_{ave,c} \times N) \quad (11)$$

其中 T_c 和 T_r 分别表示为当前宏块的目标比特数和当前帧的剩余比特数。 $MAD_{ave,c}$ 表示为当前帧中所有已编码宏块的平均 MAD 值, N 表示当前帧剩余的未编码的宏块数目。 T_r 是通过减去已编码宏块所用比特数来更新的。

[0021] 如果当前宏块为当前帧的第一个宏块,那么当前宏块的 QP 设为 QP_{ave} 。对于当前帧的其它宏块, QP 的计算方法可以为:如果当前帧的剩余比特数为负值,则当前 QP 直接设为 $QP[k-1, i] + 1$, 否则,先给当前宏块分配比特数,其分配方法如前所述。然后通过本发明中的 QP 计算方法得到 QP。为了保证宏块中不会出现方块效应,所以 QP 首先由下式来限定:

$$QP[k, i] = \min \{ QP[k-1, i] + 1, \max \{ QP[k-1, i] - 1, QP \} \} \quad (12)$$

为了保证图像的平滑性, QP 进一步由下式来限定使其各帧间的 QP 改变不会过大:

$$QP[k, i] = \min \{QP_{ave} + 2, \max \{QP_{ave} - 2, QP\}\} \quad (13)$$

最后, QP 需要满足 H. 264/AVC 的规定, 限定在 1~51 内:

$$QP[k, i] = \min \{51, \max \{1, QP\}\} \quad (14)$$

在完成一个宏块的编码后, 记录编码信息, 如实际 MAD, 实际所用的 QP, 实际编码所用的比特数, 更新剩余比特数。

[0022] 4. 测试结果

为了评价本发明的结果。本发明和 H. 264/AVC 的参考软件算法进行了性能比较。其中, 双方的测试条件选择为:

1) 熵编码方式为 CABAC。

[0023] 2) 参考帧的数目为 5。

[0024] 3) 搜索范围为 16。

[0025] 4) 采用全搜索模式。

[0026] 5) RDO 开启, 码率控制算法模式为 0, 基本单元选择 1 个宏块。

[0027] 6) GOP 的结构为 IPP...P。对于每个测试序列, 编码 300 帧。即对每个 GOP 来说, 第一帧为 IDR 帧, 接下来 299 帧都是 P 帧。

[0028] 7) 对于每个测试码流, 其目标比特流分别 48, 64, 96kbps。帧率为 30 帧每秒。

[0029] 从表 1 可知。本发明可以比 H. 264/AVC 的参考软件得到更高的 PSNR, 最高为 1.06dB (Akiyo @48kbps)。而在三种比特率下, 对所有测试码流其平均增加的 PSNR 为 0.80dB @48kbps, 0.72dB @64kbps 和 0.55dB @96kbps。而本发明在控制比特率的准确度上来说, 对于所有码流其比特率偏差为 0.069% @48kbps, 0.055% @64kbps 和 0.054% @96kbps。而这三种情况下 JM15.1 的结果分别为 0.188%, 0.107% 和 0.122%。这就是说本发明可以比 H. 264/AVC 的参考软件 JM15.1 减小 58% 的比特偏差。

[0030] 表 1

Sequence	Target rate (kbps)	JM 15.1		Proposed	
		PSNR	Rate	PSNR(dB)	Rate
akiyo	48	40.64	48.08(+0.08)	41.70(+1.06)	48.05(+0.05)
	64	42.12	64.10(+0.10)	43.03(+0.91)	64.06(+0.06)
	96	44.51	96.19(+0.19)	45.07(+0.56)	96.07(+0.07)
Carphone	48	32.02	48.09(+0.09)	32.67(+0.65)	48.03(+0.03)
	64	33.26	64.09(+0.09)	33.88(+0.66)	64.03(+0.03)
	96	35.30	96.12(+0.12)	35.83(+0.53)	96.01(+0.01)
Container	48	36.43	48.04(+0.04)	37.17(+0.74)	48.02(+0.02)
	64	37.66	64.04(+0.04)	38.25(+0.59)	64.02(+0.02)
	96	39.34	96.06(+0.06)	39.84(+0.50)	96.04(+0.04)
foreman	48	31.10	48.10(+0.10)	31.50(+0.40)	48.03(+0.03)
	64	32.49	64.08(+0.08)	32.89(+0.40)	64.05(+0.05)
	96	34.50	96.17(+0.17)	34.82(+0.32)	96.06(+0.06)
grandma	48	37.25	48.08(+0.08)	38.18(+0.93)	48.05(+0.05)
	64	38.55	64.06(+0.06)	39.46(+0.91)	64.03(+0.03)
	96	40.92	96.10(+0.10)	41.72(+0.80)	96.09(+0.09)
salesman	48	35.21	48.07(+0.07)	36.25(+1.04)	48.02(+0.02)
	64	37.07	64.04(+0.04)	37.94(+0.87)	63.98(-0.02)
	96	39.90	96.06(+0.04)	40.48(+0.58)	96.04(+0.04)

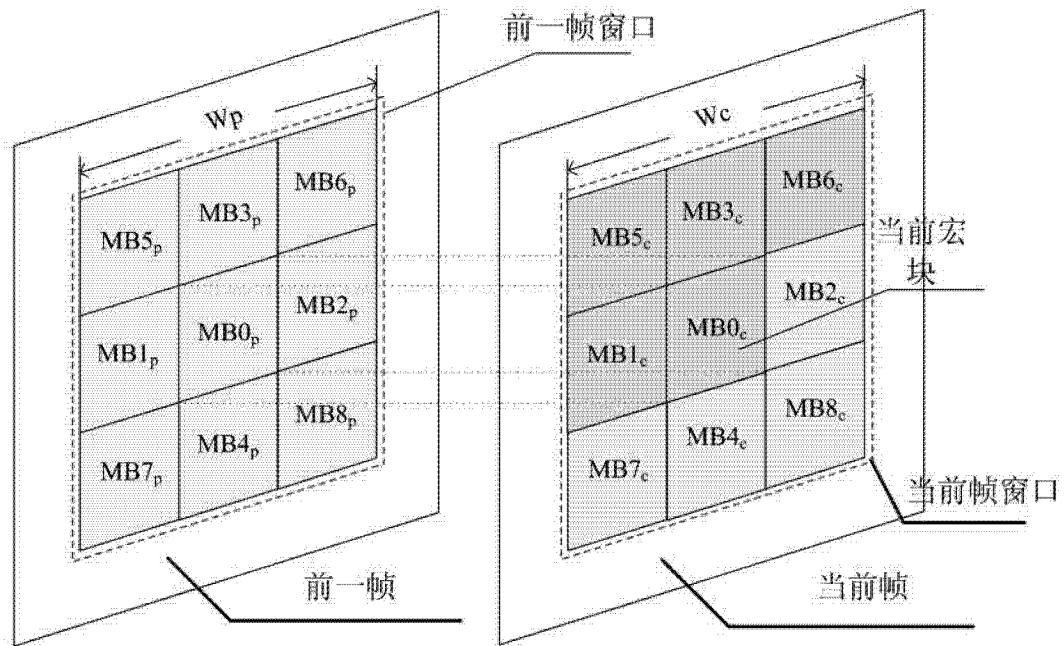


图 1

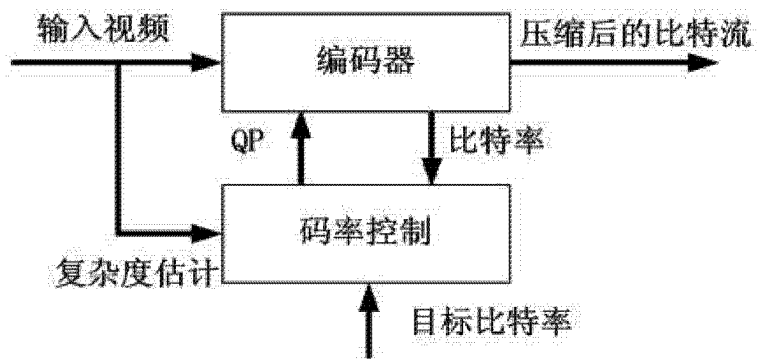


图 2

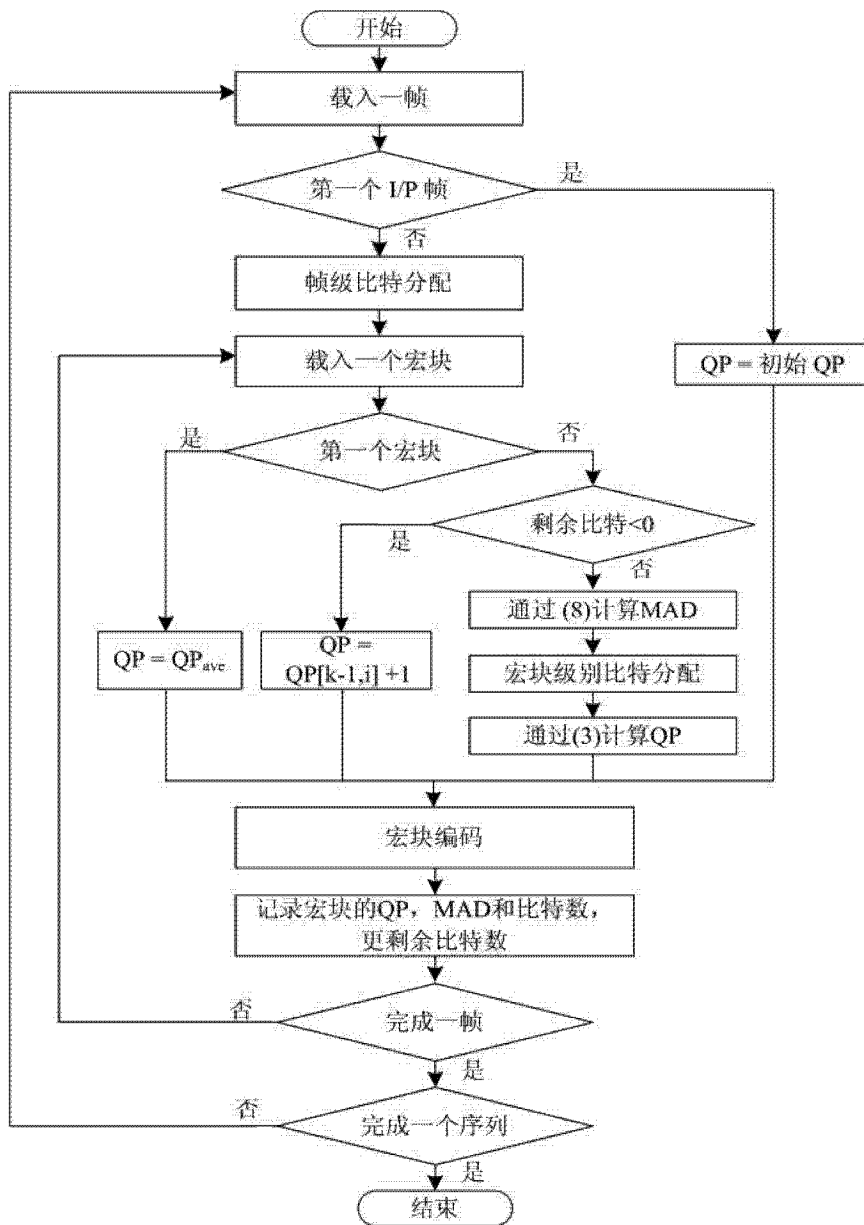


图 3