[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
H04N 7/12 (2006.01)

[12] 发明专利说明书

专利号 ZL 02826248.4

[45] 授权公告日 2008 年 1 月 16 日

[11] 授权公告号 CN 100362863C

[22] 申请日 2002.12.18 [21] 申请号 02826248.4

[30] 优先权

[32] 2001.12.28 [33] US [31] 10/035,017

[86] 国际申请 PCT/IB2002/005487 2002.12.18

[87] 国际公布 WO2003/061265 英 2003.7.24

[85] 进入国家阶段日期 2004.6.28

[73] 专利权人 诺基亚有限公司

地址 芬兰埃斯波

[72] 发明人 L·奥克特姆

[56] 参考文献

CN1072554A 1993.5.26

CN1036303C 1997.10.29

US6205174B1 2001.3.20

US5929916A 1999.7.27

CN1293873A 2001.5.2

CN1241336A 2000.1.12

US6043844A 2000.3.28

审查员 赵向阳

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 李亚非 王 勇

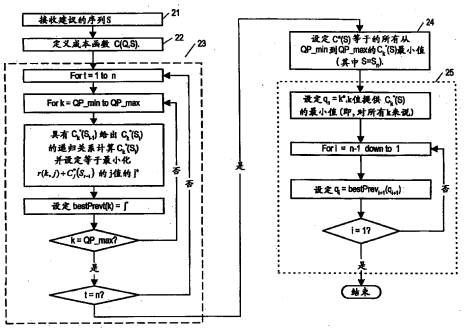
权利要求书 6 页 说明书 17 页 附图 3 页

[54] 发明名称

在视频编码器中选择宏块量化参数的方法和装置

[57] 摘要

一种用来在视频编码器中选择量化参数值序列的方法和相应的装置，所述视频编码器被安排将一个视频帧编码为 n 个宏块的序列，并为所述视频帧的每个宏块分配一个量化参数值。 所述方法包括步骤(23 24 25)，其中分配给所述 n 个宏块序列的至少一个子集的量化参数值以这样一种方式被优化，以便使得与它们编码相关的成本最小化。



1、一种在视频编码器中选择量化参数值序列的方法，所述视频编码器被配置来将一个视频帧编码为n个宏块的序列，并为所述视频帧的每个宏块分配一个量化参数值，所述方法的特征在于它包括如下步骤：选择分配到所述n个宏块序列的至少一个子集的量化参数值，以便使得用一个候选的量化器参数序列代替一个建议的量化器参数序列所产生的成本最少，所述成本具有一个表示根据一个预定标准测量的在所述候选序列和所述建议量化器参数序列之间的差异的分量。

2、如权利要求1所述的方法，进一步特征在于，所述的n个宏块序列的所述子集包括所有所述的n个宏块。

3、如权利要求1所述的方法，进一步特征在于，对分配给所述n个宏块序列的至少一个子集的所述量化参数值的优化是使用一个维特比搜索算法来执行的。

4．如权利要求1所述的方法，其中所述方法是用来对视频编码器的量化器参数QP提供帧级控制，所述视频编码器具有用于一个帧的每个宏块的一个量化器参数，一帧由n个宏块构成，所述方法从一组所有可能的量化器参数序列中提供一个最佳量化器参数序列Q^*，所有的可能量化器参数序列具有一个作为充当一个候选量化器参数序列的任意元素的量化器参数序列Q，所述最佳量化器参数序列Q^*最小化一个指示使用所述候选量化器参数序列Q代替一个建议的量化器参数序列S的成本的成本函数$C(Q,S)$，所述方法的进一步特征在于它包括

a）一个接收建议的量化器参数序列S的步骤；

b)一个定义成本函数$C(Q,S)$的步骤，所述成本函数具有一个表示根据一个预定标准测量的在所述候选序列Q和建议量化器参数序列S之间的差异的分量$D(Q,S)$，以及一个与花费在表示所述候选序列Q的比特数量成比例的分量R(Q)；以及

c)一个确定所述最佳量化器参数序列Q^*作为最小化所述成本函数$C(Q,S)$的序列的步骤；

由此提供一个在比特效率方式上近似于所述建议量化器参数序列S的最

佳量化器参数序列Q^*。

5．如权利要求4所述的方法，进一步特征在于，所述建议量化器参数序列S由一个感兴趣区域分析器来提供。

6．如权利要求4所述的方法，进一步特征在于确定所述最佳量化器参数序列Q^*作为最小化所述成本函数$C(Q,S)$的序列的步骤，包括为落入由所述建议量化器参数序列S的量化器参数跨越的量化器参数值的范围的k值计算一个最佳的约束成本函数$C_k^*(S_t)$的步骤，所述最佳约束成本函数是t个量化器参数的局部建议量化器参数序列S_t的函数，并指示可通过使用任何可能的局部候选量化器参数序列$Q_{t,k}$获得的最低成本，所述序列$Q_{t,k}$具有作为最后一个的第t个元素的一个带有所述k值的量化器参数。

7．如权利要求4所述的方法，进一步特征在于，确定所述最佳量化器参数序列Q^*作为最小化所述成本函数$C(Q,S)$的序列的步骤，包括子步骤：

a1）从所述建议量化器参数序列S中确定一个从一个预定最小值到预定最大值的量化器参数值范围；

b1)设定序列长度t等于1；

c1)设定k值等于所述预定最小值；

d1)计算一个最佳的约束成本函数$C_k^*(S_t)$，所述最佳约束成本函数是t个量化器参数的局部建议量化器参数序列S_t的函数，该最佳约束成本函数当t大于1时的计算是基于根据$C_k^*(S_{t-1})$给出的$C_k^*(S_t)$并涉及一个条件r(k,j)+ $C_j^*(S_{t-1})$的递归关系，其中$r(k,j)$是与花费在表示Q的比特的数量成比例的成本函数分量$R(Q)$的一个元素，所述Q具有在从所述建议量化器参数序列S中确定的范围内的一个值的变量j；

e1)为当前的t和k存储最小化条件$r(k,j)+C_j^*(S_{t-1})$的j值；

f1)对于连续的每个比前一个大一的k值和连续的每个比前一个t值大一的t值来计算所述最佳约束成本函数$C_k^*(S_t)$，并为当前的t和k存储最小化条件$r(k,j)+C_j^*(S_{t-1})$的j值，直到k等于所述预定最大值并且t等于在一个帧中的宏块数量；

g1)根据对于在所述建议序列S中的量化器参数值的范围内的所有k值的$C_k^*(S_{t=n})$的比较，确定所述最佳成本函数$C^*(S_{t=n})$；和

h1)通过这样一个过程建立所述优化序列Q*：首先设定所述量化器参数，

使得所述最后一个宏块等于最小化$C_k^*(S_{t=n})$的k值，然后向后跟踪，将为下一个宏块存储的j值分配到被建立的优化序列中的每个先前的量化器参数。

8．如权利要求4所述的方法，进一步特征在于，所述成本函数分量$D(Q,S)$是由下述方程给出的形式，

$$D(Q,S)=\sum_{i=1}^{n} d(q_i,s_i)\ ,$$

其中 $d(q_i,s_i)$是一个无记忆宏块级非负成本分量，所述分量具有作为自变量的所述候选量化器参数序列的一个元素q_i和一个所述建议量化器参数序列的相应元素S_i。

9. 如权利要求8所述的方法，进一步特征在于，所述成本函数分量$R(Q)$是由下述方程给出的形式，

$$R(Q)=\sum_{i=1}^{n} r(q_i,q_{i-1})\ ,$$

其中$r(q_i,q_{i-1})$是一个无记忆的宏块级的非负成本分量，其表示在为宏块i编码所述量化器参数所使用的比特数量和为宏块i－1编码所述量化器参数所使用的比特数量之间的关系。

10．如权利要求9所述的方法，进一步特征在于，所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$由下述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}0, & \text{其中}|q_i-q_{i-1}|\le 2\\ +\infty, & \text{其它},\end{cases}$$

并且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$由下述方程给出

$$d(q_i,s_i)=(q_i-s_i)^2。$$

11．如权利要求9所述的方法，进一步特征在于，所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$由下述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}\lambda N_{q_i,q_{i-1}}, & \text{如果}q_i\text{是可表示的},\\ +\infty, & \text{其它},\end{cases}$$

其中λ是一个用作速率失真折中参数的预定非负值，并且其中$N_{q_i,q_{i-1}}$是用来表示用q_{i-1}给出 的q_i的比特数；而且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$由下述方程给出

$$d(q_i,s_i)=\|mse(q_i)-mse(s_i)\|\ ,$$

其中$mse(...)$描述了在初始宏块内容和用所指示的量化器参数序列元素q_i

或s_i重建的宏块内容之间的均方差。

12. 一种在视频编码器中选择量化参数值序列的装置，所述视频编码器被配置用于将一个视频帧编码为n个宏块的序列，并为所述视频帧的每个宏块分配一个量化参数值，所述装置的特征在于它包括用于选择分配到所述n个宏块序列的至少一个子集的量化参数值，以便使得用一个候选的量化器参数序列代替一个建议的量化器参数序列所产生的成本最少的装置，所述成本具有一个根据一个预定标准测量的表示在所述候选序列和所述建议量化器参数序列之间的差异的分量。

13. 如权利要求12所述的装置，进一步特征在于，所述的n个宏块序列的所述子集包括所有所述的n个宏块。

14. 如权利要求12所述的装置，进一步特征在于，对分配给所述n个宏块序列的至少一个子集的所述量化参数值的优化是使用一个维特比搜索算法来执行的。

15. 如权利要求12所述的装置，其中所述装置具有对被所述视频编码器使用的一个量化器参数的帧级控制和由所述视频编码器处理的一个帧的每个宏块的一个量化器参数，一帧由n个宏块构成，所述帧级控制基于从一组所有可能的量化器参数序列中提供一个最佳量化器参数序列Q^*，所有的可能量化器参数序列具有一个作为充当一个候选量化器参数序列的任意元素的量化器参数序列Q，所述最佳量化器参数序列Q^*最小化一个指示使用所述候选量化器参数序列Q代替一个建议的量化器参数序列S的成本的成本函数$C(Q,S)$，所述装置的进一步特征在于它进一步包括一个优化器，用于接收建议的量化器参数序列S；用来定义成本函数$C(Q,S)$，所述成本函数具有一个表示根据一个预定标准测量的在所述候选序列Q和建议量化器参数序列S之间的差异的分量$D(Q,S)$，以及一个与花费来表示所述候选序列Q的比特数量成比例的分量R(Q)；以及用来确定所述最佳量化器参数序列Q^*；

所述视频编码器由此而确定一个在比特效率方式上近似于所述建议量化器参数S的最佳量化器参数序列Q^*。

16. 如权利要求15所述的装置，进一步特征在于，所述建议量化器参数序列S由一个感兴趣区域分析器来提供。

17．如权利要求15所述的装置，进一步特征在于，所述优化器用来确定最小化所述成本函数$C(Q,S)$的所述最佳量化器参数序列Q^*，并为落入由所述建议量化器参数序列S的量化器参数跨度的量化器参数值的范围的k值计算一个最佳的约束成本函数$C_k^*(S_t)$的所述最佳量化器参数序列Q^*，所述最佳约束成本函数是t个量化器参数的局部建议量化器参数序列S_t的函数，并指示通过使用任何可能的局部候选量化器参数序列$Q_{t,k}$获得的最低成本，所述序列$Q_{t,k}$具有作为最后一个第t个元素的一个带有所述k值的量化器参数。

18．如权利要求15所述的装置，进一步特征在于，所述优化器用来确定最小化所述成本函数$C(Q,S)$的所述最佳量化器参数序列Q^*，包括：

a)从所述建议量化器参数序列S中确定一个从一个预定最小值到预定最大值的量化器参数值范围的装置；

b)设定序列长度t等于1的装置；

c)设定k值等于所述预定最小值的装置；

d)计算一个最佳的约束成本函数$C_k^*(S_t)$的装置，所述最佳约束成本函数是t个量化器参数的局部建议量化器参数序列S_t的函数，该最佳约束成本函数当t大于1时的计算是基于根据$C_k^*(S_{t-1})$给出的$C_k^*(S_t)$并涉及一个条件r(k,j)+$C_j^*(S_{t-1})$的递归关系，其中$r(k,j)$是与花费在表示Q的比特的数量成比例的成本函数分量$R(Q)$的一个元素，所述Q具有在从所述建议量化器参数序列S中确定的范围内的一个值的变量j；

e)为当前的t和k存储最小化条件$r(k,j)+C_j^*(S_{t-1})$的j值的装置；

f)对于连续的每个比前一个大一的k值和连续的每个比前一个t值大一的t值来计算所述最佳约束成本函数$C_k^*(S_t)$，并为当前的t和k存储最小化条件$r(k,j)+C_j^*(S_{t-1})$的j值，直到k等于所述预定最大值并且t等于在一个帧中的宏块数量的装置；

g)根据对于在所述建议序列S中的量化器参数值范围内的所有k值的$C_k^*(S_{t=n})$的比较，确定所述最佳成本函数$C^*(S_{t=n})$的装置 以及

h)通过这样一个过程建立所述优化序列Q*：首先设定所述量化器参数，使得所述最后一个宏块等于最小化$C_k^*(S_{t=n})$的k值，然后向后跟踪，将为下一个宏块存储的j值分配到被建立的优化序列中的每个先前的量化器参数的装置。

19．如权利要求15所述的装置，进一步特征在于，所述成本函数分量$D(Q,S)$是由下述方程给出的形式，

$$D(Q,S)=\sum_{i=1}^{n}d(q_i,s_i)\ ,$$

其中 $d(q_i s_i)$是一个无记忆宏块级非负成本分量，所述分量具有一个作为自变量的所述候选量化器参数序列的元素q_i和一个所述建议量化器参数序列的相应元素S_i。

20．如权利要求19所述的装置，进一步特征在于，所述的成本函数分量$R(Q)$是由下述方程给出的形式，

$$R(Q)=\sum_{i=1}^{n}r(q_i,q_{i-1})\ ,$$

其中$r(q_i,q_{i-1})$是一个无记忆的宏块级的非负成本分量，其表示在为宏块i编码所述量化器参数所使用的比特数量和为宏块i－1编码所述量化器参数所使用的比特数量之间的关系。

21、如权利要求20所述的装置，进一步特征在于，所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$由下述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}0, & \text{其中}|q_i-q_{i-1}|\le 2\\ +\infty, & \text{其它},\end{cases}$$

并且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$由方程给出

$$d(q_i,s_i)=(q_i-s_i)^2。$$

22、如权利要求20所述的装置，进一步特征在于，所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$由下述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}\lambda N_{q_i,q_{i-1}}, & \text{如果}q_i\text{是可表示的}\\ +\infty, & \text{其它},\end{cases}\ ,$$

其中λ是一个用作速率失真折中参数的预定非负值和其中$N_{q_i,q_{i-1}}$是用来表示用q_{i-1}给出 的q_i的比特数；而且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$由方程给出

$$d(q_i,s_i)=\|mse(q_i)-mse(s_i)\|\ ,$$

其中$mse(...)$描述了在初始块内容和用所指示的量化器参数序列元素qi或si重建的宏块内容之间的均方差。

在视频编码器中选择宏块量化参数的方法和装置

技术领域

本发明涉及一种视频编码系统。尤其涉及使用运动补偿预测：ITU-T H.263和MPEG-4甚低比特率视频（VLBV）来进行视频序列压缩的两种标准系统。

背景技术

使用运动补偿预测来压缩视频序列的两种标准系统是：在1995年2月由甚低比特率视觉技术ITU-T SG 15专家组起草建议的 H.263中描述的ITU-T H.263和在1996年11月由MPEG视频组，Doc. ISO/IEC/JTCI/SC29/WG11，N1469 Maceio在MPEG-4视频检验模型-版本5.0中所描述的MPEG-4 甚低比特率视频（VLBV）。在定义的版本1之后已经对H.263进行了一些扩展。扩展版本经常被称作H.263+。术语H.263在这里被用作指未进行扩展的版本，即版本1。由于在MPEG-4 VLBV和H.263中所使用的算法之间的相似性，在此讨论焦点将集中在H.263的算法上。

H.263是一个以低比特率（<64kbits/sec）来压缩视频序列的ITU-T推荐标准，并且是基于早期的ITU-T推荐标准——H.261。图1描述了H.263编码器的控制元件的框图。（视频编码器未示出。）这样一个编码器的主要元件是一个预测模块11，一对块变换模块13（变换模块T和T'），以及一对量化模块12（量化模块Q和Q'）。此外，还有一个编码控制模块14。

编码控制模块14确定所有编码参数；它充当了所述系统的大脑。从编码控制信号中产生INTRA/INTER判决数据，该编码控制信号用于控制是否使用先前帧的知识来编码当前帧。

编码控制模块提供的量化指示数据流确定每一宏块使用的所谓量化参数。该确定可以位于帧级别或宏块级别。本发明集中于该信号的产生上。

图1中所指的“视频复用编码器”仅仅是一个复用器，并且不使用“视频输入”作为它的其中一个输入。指示为被提供给“视频复用编码器”的信号包括视频信号的压缩表示。

为了利用在连续帧之间的时间相关性，如果先前重建帧有效并且有用，那

么该系统首先执行运动补偿预测，上述术语“有用”表示使用先前帧（INTER 编码）将比不使用它们（INTRA编码）产生更好的压缩性能。如果后续帧不完全相关，INTRA编码可能会产生更好的压缩性能。该系统首先使用运动信息和先前重建的帧来进行运动补偿预测。为了使所述解码器可以进行与所述编码器中所进行的预测相同的预测，（从编码器向解码器）传输所述运动信息（图1中的v）。下一步，称作DCT（即离散余弦变换）的块变换被用于块T的预测误差（或者在无预测的情况下是帧本身），以便利用空间相关性。最后，预测误差的DCT系数在块Q中被量化和熵编码。量化器是用来向视频序列引入损耗以获得较高压缩的主要装置。损耗量（和这样的比特率）由量化器的步长来控制，依次由量化器参数（QP）来确定所述步长的参数，步长参数具有在1到31之间的整数值，并由编码控制模块14提供给块Q。QP必须是解码器所知的，因此它被作为边（辅助）信息来传输（并在图1中指定为qz）。

H.263的基本层是所述的宏块层。宏块（MB）是H.263的基本构建块，在某种意义上说主要元素的编码（预测、块变换和量化）可以通过每次处理一个宏块来执行。一个宏块由一个16×16的亮度块、两个8×8的色度块、以及宏块级编码参数（例如宏块类型等）的所述表示组成；一些宏块级编码参数是可选的。以光栅扫描的顺序传输宏块。因为宏块级编码参数在比特位方面代价很高，所以只有在有必要这样做的时候才在所述宏块级对它们进行编码。（存在诸如所述量化参数QP的帧级编码的编码参数。当没有执行宏块级编码时，就使用这些帧级数值。）

H.263提供了QP的有限宏块级控制；存在一个可选的2比特DQUANT字段，其对当前宏块的QP和先前的编码宏块的QP之间的差值进行编码。（参见H.263+(2/98)的5.3.6部分。）由于在H.263的5.3.6部分中所述比特字段的限制，宏块QP在每次变化时至多可以改变±2。在QP变化被用来进行速率控制的情况下，在QP变化范围上的这种限制非常充分。然而，除了速率控制之外也可以有其它改变QP的原因，例如感兴趣区域的编码，其是一种给一个自动检测到的或用户规定的感兴趣区域，例如人脸分配更多比特（这样引入较少的损失）的技术。

发明解决的问题

考虑为了诸如感兴趣区域（ROI）编码目的而不是速率控制目的所需的宏

块级QP变化的情形。在这样一种情形下，QP的有限宏块级控制造成了严重的限制。由感兴趣区域分析器建议的或由用户输入的任意的QP分布不能完全由H.263（或MPEG-4 VLBV）编码器实现。因此，在某种意义上讲，这种编码器需要选择一个尽可能接近初始建议分布的近似实现。鉴于最优性的定义（或等效的成本量度），本发明提供了一种最佳选择实现的方法，也就是，提供了一种将由约束实现所招致的总成本减小到最少的方法，规定的总成本当所述QP分布越接近于所述建议的分布时越低，但当必须使用的比特越多时则越高。

作为一个相关的问题，在1998年1月由甚低比特率视频技术ITU-T SG 15专家组起草建议的H.263 版本2所规定的 H.263+中，任意分布的精确表示在比特预算上可能非常浪费，因此这不是最理想的方案。量化可以在宏块的基础上进行变化，并且可能没有更精细的变化。任意分布的精确表示指的是多个宏块QP的一个任意选择。考虑通过为每个宏块分配一个独立的QP所形成的矢量QP。表示所述分量矢量其中的一些比表示其它的耗费少得多（在比特上）。如果不考虑相邻宏块的QP值来为每个宏块选择一个QP，那么就可能要花费惊人数量的比特来表示块到块的QP变化。

编码器需要在一方面跟随感兴趣区域分析器的初始建议（对于QP分布来说）和另一方面最小化花费在QP变化上的总比特之间找到一个最好的折中。对于此相关问题，本发明提供了一种当成本函数被定义来表示所述折中时找到最好折中的方法。

早期解决此问题的方法

依据现有技术，对于使用ROI编码的MPEG-4视频编码器来说，通过将每个帧分离成两个视频对象平面（VOP）、一个前景VOP和一个背景VOP来获得QP控制，如在Low Bit-rate Coding of Image Sequence Using Adaptive Regions of Interest ,by N.Doulamis, A.Doulamis,D.Kalogeras, and S. Kollias,IEEE Tran.on CAS for Video Technology,pp.928-934,vol. 8, no.8,Dec.1998中所描述的，并且也在Video Segmentation for Content-Based Coding ,by T.Meier,and N.Ngan,IEEE Tran. On CAS for Video Technology,pp.1190-1203,vol.9,no.8 ,Dec.1999中有描述。这种分离克服了在前景/背景分离边界±2的变化限制。

现有技术方案有两个缺点。第一，存在一个编码VOP分割的比特开销，如果所述的前景不紧密，那么比特开销有可能变得让人望而却步。当决定在所述

前景/背景分离时，很难考虑这些比特。这样，如果并非不可能的话，很难进行优化。第二，现有技术方案不能在每个VOP（即在每个VOP内的变量）内直接提供分级的QP控制。所述QP值必须在所述背景内和/或前景内进一步变化，从而获得每个VOP内的分级控制。在QP如何变化上仍然存在限制，并且优化甚至变得比在包括所述整个帧的单个VOP内进行精密QP控制的最初问题更加不切实际。

在H.263中不支持VOP 结构。因此，即使这样的方案也是无效的。因此，±2的变化限制适用于H.263中的所有宏块。这样在H.263中就没有为进行ROI编码（或任何其它需要进行精密QP控制的情形）来对宏块QP选择进行帧级优化的技术。

H.263+提供了一种用于精确QP控制的比特消耗机制；该机制能够通过花费（使用）6个比特来表示宏块的任意QP。对于甚低比特率应用（其是H.263和H.263+的主要焦点）来说，对任意QP分布的表示在比特预算方面容易变得非常惊人，因此帧级优化尤其有利。然而，如在H.263中那样，在H.263+中没有用来为进行ROI编码（或其它任何需要精密QP控制的情形）而对宏块QP选择进行帧级优化的已知技术。

需要一种以不花费太多比特的方式来提供精确QP控制，即帧级控制的机制，以便其在低比特率应用中有用。

发明内容

因此，本发明的第一方面提供一种用来在视频编码器中选择量化参数值序列的方法，所述视频编码器被安排来将一个视频帧编码为一个n宏块序列，并为所述视频帧的每个宏块分配一个量化参数值，所述方法的特征在于，分配给所述n个宏块序列的至少一个子集的量化参数值以这样一种方式被优化，以便使得与它们编码相关的成本最小化。

根据本发明的第一方面，所述n宏块序列的所述子集包括所有所述n宏块。

也是根据本发明的第一方面，可以使用一个维特比搜索算法来对分配给所述n宏块序列的至少一个子集的所述量化参数值进行量化。

也是根据本发明的第一方面，可以通过将与编码建议的量化参数值序列有关的花费和编码一个候选的量化参数值序列的花费进行比较，来对分配给所述n宏块序列的至少一个子集的所述量化参数值进行量化。

还是根据本发明的第一方面，在用所述方法来对视频编码器的量化器参数QP提供帧级控制的应用中，所述视频编码器具有一个量化器参数，该参数用于帧的每个宏块，所述帧由n个宏块构成，所述方法提供一个优化的量化器参数序列Q*，该序列来自一组具有一个量化器参数序列Q作为充当候选量化器参数序列的任意元素的所有可能量化器参数序列，所述最佳量化器参数序列Q*最小化指示使用所述候选量化器参数序列Q来代替一个建议的量化器参数序列S的花费的成本函数C(Q,S)，所述方法包括：接收建议的量化器参数序列S的步骤；定义具有元素D(Q,S)的成本函数C(Q,S)的步骤，所述元素D(Q,S)表示了在所述候选序列Q和所述依据一个预定标准测量的建议量化器参数序列S之间的差异，并具有一个与用于表示所述候选序列Q的比特数成比例的元素R(Q)；以及将所述最佳量化器参数序列Q*确定为最小化所述成本函数C(Q,S)的序列的步骤；从而提供一个在比特效率方面接近所述建议的量化器参数序列S的最佳量化器参数序列Q*。

进一步，所述建议的量化器参数序列S可以由一个感兴趣区域的分析器提供。

还进一步，所述确定所述最佳量化参数序列Q*作为最小化所述成本函数C(Q,S)的序列的步骤可以包括，用一个落入由所述建议的量化器参数序列S的量化器参数跨越的量化器参数值的范围内的k值来计算一个最佳的约束成本函数$C_k^*(S_t)$的步骤，所述最佳约束成本函数是t个量化器参数的局部建议的量化器参数序列S_t的函数，并指示了通过使用任何可能的局部候选量化器参数序列$Q_{t,k}$可获得的最低成本，所述序列具有作为一个最后第t个元素的带有所述k值的量化器参数。

还进一步，所述确定所述最佳量化参数序列Q*作为最小化所述成本函数C(Q,S)的序列的步骤可以包括子步骤：从所述建议的量化器参数序列S中确定一个从一个预定最小值到一个预定最大值的量化器参数值的范围；设定序列长度t等于1；设定k值等于所述预定最小值；计算最佳约束成本函数$C_k^*(S_t)$，所述最佳约束成本函数是t个量化器参数的局部建议的量化器参数序列S_t的函数，该最佳约束成本函数当t大于1时的计算是基于根据$C_k^*(S_{t-1})$给出的$C_k^*(S_t)$并涉及一个条件$r(k,j)+C_j^*(S_{t-1})$的递归关系，其中r(k,j)是与在表示具有变量j的Q上花费的比特数成比例的所述成本分量R(Q)的一个元素，所述j具有从所建议的量化器参数序列S确定的范围中的一个值；为当前t和k存储最小化条件r(k,j)+

$C_j^*(S_{t-1})$的j 值；计算所述最佳约束成本函数$C_k^*(S_t)$，并为所述当前t和k存储最小化条件r(k,j)+ $C_j^*(S_{t-1})$的j 值，对于连续的k值而言，每一个k值比前一个值大，直到k等于所述预定最大值为止，并且对连续的t值而言，每一个t值比先前t值大，直到t等于在一个帧中的宏块的数量为止；基于在所建议的序列S中的量化器参数值的范围内的所有k值的$C_k^*(S_{t=n})$的比较，来确定所述最佳成本函数$C^*(S_{t=n})$；以及通过这样一个过程建立所述优化序列Q*：首先设定所述量化器参数，使得所述最后一个宏块等于最小化$C_k^*(S_{t=n})$的k值，然后向后跟踪，将为下一个宏块存储的j值分配到被建立的优化序列中的每个先前的量化器参数。

甚至再进一步，所述的成本函数分量D(Q,S)可以是由方程所给出的形式，

$$D(Q,S)=\sum_{i=1}^{n} d(q_i,s_i),$$

其中$d(q_i,s_i)$是一个无记忆的宏块级非负成本分量，其具有作为自变量的一个所述候选量化器参数序列的元素q_i和一个所建议的量化器参数序列的相应元素s_i；于是所述成本函数分量R(Q)可以是由方程所给出的形式，

$$R(Q)=\sum_{i=1}^{n} r(q_i,q_{i-1}),$$

其中$r(q_i,q_{i-1})$是一个无记忆的宏块级的非负成本分量，其表示在为宏块i编码所述量化器参数所使用的比特数量和为宏块i－1编码所述量化器参数所使用的比特数量之间的关系；于是所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$可以由所述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}0, & \text{其中}|q_i-q_{i-1}|\le 2\\ +\infty, & \text{其它},\end{cases}$$

并且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$可以由下述方程给出，

$$d(q_i,s_i)=(q_i-s_i)^2;$$

此外所述成本函数分量R(Q)的一个元素$r(q_i,q_{i-1})$可以由所述方程给出，

$$r(q_i,q_{i-1})=\begin{cases}\lambda N_{q_i,q_{i-1}}, & \text{如果}q_i\text{是可表示的}\\ +\infty, & \text{其它},\end{cases},$$

其中λ是一个用作速率失真折中参数的预定非负值和其中$N_{q_i,q_{i-1}}$是用来表示用q_{i-1}给出 的q_i的比特数；而且所述成本函数分量D(Q,S)的一个元素$d(q_i,s_i)$

可以由方程给出

$$d(q_i,s_i)=\|mse(q_i)-mse(s_i)\|,$$

其中$mse(...)$描述了在初始块内容和用所指示的量化器参数序列元素q_i或s_i重建的宏块内容之间的均方差。

根据本发明的第二方面，提供了一种用来执行依据本发明的第一方面的方法的装置。

因此，本发明基于一个用来选择最佳的宏块QP的帧级算法。所述最佳标准可以被客观地定义，其允许将所述算法定制来适应不同的运算模式。尽管所述算法使用了一个动态编程方法，所述核心算法的计算复杂性是相当低的。

就发明人所知，本发明是用来帧级优化在H.263（+）或MPEG-4框架中选择宏块QP的第一方法。在一个精密QP控制的情形中，本发明的最优化将产生比现有技术更好地利用约束的比特预算，所述现有技术提供了不是最理想的精密QP控制。例如，H.263+编码器提供了完全控制QP变化的能力，但是需要一个高的花费（在比特上）。进一步，H.263编码器只能不是最理想地接近一个由ROI编码引擎建议的QP变化。在甚低比特率视频压缩应用中对于位比特预算的充分利用是尤其有利的。

所述核心算法的计算复杂性是相当低的。如果不考虑计算所述成本函数的元素，例如在某些实施例中的元素$r(q_i,q_{i-1})$和$d(q_i,s_i)$的值的计算成本（因为所述成本函数的元素通常在所述核心算法之外来计算），那么所述核心算法典型地近似执行每个象素的一个加法和一个比较运算（或更少）。（在一个宏块中有256个象素。）在每个象素上这样数量的运算对于高性能的视频编码器来说开销是非常低的。

附图说明

通过考虑结合附图所呈现出的后续的详细描述，将使本发明的上述以及其它目的、特点和优点变得明显，其中：

图1是一个本发明的算法对其有用的类型的H.263编码器的框图，包括作为其主要元素的一个预测模块、块变换模块和量化模块、所有都在编码控制模块控制下；和

图2是本发明的算法的流程图；和

图3是图1的编码控制模块的一些元件的框图/流程图。

执行本发明的最佳模式

在此提供的公开是在H.263标准的背景下（也就是依据H.263标准的编码器），但是由本发明也可以直接应用于MPEG-4 VLBV标准，这是因为在两个标准之间的相似性。

所述最佳算法是称之为维特比算法的一种形式，其中为每一帧寻找所述最佳的QP序列。所述的维特比算法在THE VITERBI ALGORITHM by G.D.Foney,Jr.,Proc.IEEE,pp.268-278,March 1973中有描述。

术语QP序列在这里被用来表示一个具有与在一个帧中的宏块数量相等的长度，并通过以光栅扫描顺序拾取分配给每个宏块的QP建立的数据对象。在进入到更具体的描述之前，有必要定义一些表达式和表示一下在这里所使用的符号。

定义和符号

用s_i表示由感兴趣区域（ROI）编码引擎建议的第i个宏块的QP值，并用$s=\{s_i\}$表示由所述ROI编码引擎建议的正被编码的帧的QP序列。用$Q=\{q_i\}$表示任何由所述优化算法分配作为响应于所述请求S而选择的所述帧的QP序列的候选QP序列。所述序列Q和S的每一个的长度等于在一个帧中的宏块的数量。用n表示该长度。

用s_t表示一个局部建议的QP序列$\{s_i\}_{i=1..t}$（即$\{s_1,s_2,...,s_t\}$），其中$t \le n$。依据该符号，$S_n = S$。类似地，用Q_t来表示一个局部候选QP序列。作为一个Q_t的稍微约束的形式，用$Q_{t,\,k}$表示一个局部序列Q_t，其最后一个元素q_t=k(其中k是在区间[1,31]内的一个整数)。这样，例如，如果t=5(并且小于在一个帧中的宏块的数量n)和k=8，那么Q_t可能是序列{3,5,2,4,8}，但也可以是任何具有五个元素、其中最后一个元素的值是8的序列。

用$C(Q,S)$表示当S是所建议的QP序列时选择Q作为所述QP序列的某一成本。对于一个可以感觉（即与直觉一致）到的成本函数$C(Q,S)$来说，应当至少具有下述的特性。

a) $C(Q,S)$必须是非负的。

b) 如果Q是不可表示的，那么所述相关成本$C(Q,S)$被定义为无穷大。（如

果一个视频编码器可以不用改变 QP序列的值就能对其编码的话，那么可以说QP序列是可以表示的。例如，所述QP序列{5,12,...}不能被一个H.263编码器表示，因为由H.263编码器编码的连续QP值相差不能超过＋－2，但是在本例中12和5相差7。按照约定，我们可以说需要无穷多的比特来表示序列{5,12,...}，以至于序列从来都不被选作最佳近似值。）

c) 必须要有一个与根据一个预定标准测量的Q和S之间的差值直接或间接成比例的分量。这样一个差值或差异分量，在此用$D(Q,S)$表示，并推动所述优化器来选择一个接近于S的QP序列。在大多数实际情况中，$D(Q,S)$可以是所谓的无记忆宏块级非负成本分量的总和，所述分量由下述方程给出，

$$D(Q,S)=\sum_{i=1}^{n} d(q_i,s_i) \qquad (1)$$

这里所用的术语“无记忆”的意思是，如果$i \neq k$，那么所述第i个宏块的成本分量不依赖于q_k和s_k。然而$D(Q,S)$无需具有由方程（1）给出的特定形式。

d)它应该具有一个在此称作R(Q)的分量，其与花费用来表示Q的比特成比例并且不依赖于S。（所述R(Q)值是用来编码一个QP序列Q所需的比特的数量。S是由所述ROI分析器建议的QP序列。解码器不知道S，因此如果R(Q)的确依赖于S，那么Q的表示将是不可解码的。）如果Q是不可表示的，我们使用在此的约定R(Q)＝＋∞。在MPEG-4，H.263和H263+中使用的差分编码表明，R(Q)可以是一个由下述方程给出的宏块级非负成本分量的总和，

$$R(Q)=\sum_{i-1}^{n} r(q_i,q_{i-1}) \qquad (2)$$

其中所述形式$r(q_i,q_{i-1})$是指示了这样的事实，即用来编码第i个宏块的QP的比特数量q_i依赖于前一宏块，也就是q_{i-1}的量化器参数。对于第一宏块（i=1）来说，其中前一个宏块QP是无效的，可以使用在帧层规定的这里称之为PQUANT的QP。同样地，q_0被定义等于PQUANT。

将R(Q)和$D(Q,S)$分量相加来产生所导出的条件，

$$C(Q,S)=D(Q,S)+R(Q)=\sum_{i=1}^{n}\left[d(q_i,s_i)+r(q_i,q_{i-1})\right] \quad (3)$$

现在我们定义一个最低成本函数C^*和一个最佳QP序列Q^*。所述最低成本C^*是S的函数，并且被如下来定义：

$$C^*(S)=\min_{Q}[C(Q,S)] \quad (4)$$

并且Q^*被定义为产生最低成本的QP序列Q。C^*（）的一个约束版，表示为 $C_k^*(\ )$，是一个考虑了在以$q_{last}=k$结束的候选序列上的最小化的成本函数，并在此别以下面给出的方程来定义，

$$C_k^*(S_t)=\min_{Q_{t,K}}\left[C(Q_{i,k},S_t)\right] \quad (5)$$

在下面描述的最佳算法所使用的主要定义涉及所述约束最小化C^*（）。

最后$Q_{t,k}^*$被定义为产生所述最低成本的$Q_{t,k}$。（$Q_{t,k}$表示一个最后元素q_t为k值的局部序列Q_t。）

本发明所使用的优化算法的来源

C^*（）和$C_k^*(\ )$具有如下的相互关系：

$$C^*(S)=\min_{k}\left[C_k^*(S)\right] \quad (6)$$

这里的目标是如由方程（6）给出的那样来计算$C^*(S)$和相关的Q^*。因此，$C_k^*(S)$必须用在建议的QP值的动态范围内k值来计算。为了计算$C_k^*(S)$，使用由方程（5）给出的$C_k^*(S)$定义来达到随后组的递归关系：

$$C_k^*(S_1)=d(s_1,k)+r(k,q_0) \quad (7)$$

$$C_k^*(S_t)=d(s_i,k)+\min_{j}\left[r(k,j)+C_j^*(s_{t-1})\right] \quad (8)$$

其中q_0是在确定了最佳QP序列的宏块之前并紧接着其的宏块的QP值，或其它合适的值，例如帧级QP值（即为整个帧所使用的QP值）。

在递归步骤（8），用j^*表示最小化的j值，即最小化$r(k,j)+C_j^*(s_{t-1})$的值j。然后，$q_{t-1}=j^*$是$q_t=k$的最好的前面状态。（回想所述数量r(k,j)的变量k和j分别是值q_t和q_{t-1}。）换句话说，如果最佳序列具有$q_t=k$，那么也必须有$q_{t-1}=j^*$。通过符号$best\,\mathrm{Pr}ev_t(k)=j^*$来表示该关系。（这样一种关系是维特比算法的基础。）

随后通过根据方程（7）和（8）递归计算$C_k^*(S)$来执行最优化（找到产生所述最低成本的QP序列Q），然后从方程（6）中计算$C^*(S)$。在所述递归计算期间为每个（t,k）对存储最好的先前状态j^*，允许跟踪到与已经计算出的最低成本相关的最佳QP序列。

作为伪码的算法

上面所阐述的算法可以用下面的伪码来表示。

For t=1 to n {其中n是在一帧中宏块的数量 }

For k=QP_{min} to QP_{max} 递增1 { 其中QP_{min} 和QP_{max} 分别是QP 的某个可能的最小值和最大值，典型地是从1到31}

根据等式（7）和8计算并存储 $C_k^*(S)$(S是 S_n)，即根据 i.e. $C_k^*(S_1)=d(s_1,k)+r(k,q_0)$ and $C_k^*(S_t)=d(s_t,k)+\min_j\left[r(k,j)+C_j^*(S_{t-1})\right]$;

将j^*最下化 $r(k,j)+C_j^*(S_{t-1})$的j值;

设置$best\,\mathrm{Pr}ev_t(k)=j^*$;

根据等式(6)计算$C^*(S)$,即根据 $C^*(S)=\min_K\left[C_k^*(S)\right]$, 并确定 k^*,提供全部 ll $C_k^*(S)$中最小量的k值;

设置 q_n=k*;

对于 i=n-1 减到 1

设置 $q_i=best\,\mathrm{Pr}ev_{i+1}(q_{i+1})$;

结束 .

于是所述寻找后的QP序列是$\{q_1,q_2\ldots,q_n\}$，并且是在所建议的QP序列和所述比特预算之间的一个折中，即是最大化所述成本函数C(Q,S)的QP序列。

图2是本发明的QP优化算法的流程图，指示了本发明一般地包括一个接收一个建议的量化器参数序列S的步骤21，一个定义成本函数$C(Q,S)$的下一个步骤22，所述成本函数$C(Q,S)$具有表示一个根据一个预定标准测量的在所述候选序列Q和所建议的量化器参数S之间的差异的元素$D(Q,S)$，并具有一个与花在表示所述候选序列Q的比特数量成比例的分量R(Q)；和一系列通向确定所述最佳量化器参数序列Q^*，即最小化所述成本函数$C(Q,S)$的最佳序列Q^*的步骤23 24 25 。根据上面的描述，所述流程图也表示了通向确定所述最佳量化器参

数序列Q^*一系列步骤23 24 25的细节。

依据本发明的编码控制模块的结构

现在参考图3，图1的编码控制模块14被更具体地展示为包括一个优化器模块14a，其根据本发明的算法确定所述最佳QP序列，所述优化基于具有两个分量$D(Q,S)$和R(Q)中的每一个的特定定义的成本函数$C(Q,S)$。（所述成本函数的定义作为一个输入被提供给所述优化器模块。）所述优化模块将例如由一个ROI引擎31（通常处于所述编码控制的模块14的外部）提供的所述建议序列S用作一个输入。所述最佳序列Q被提供给控制器模块14b，其然后将所述运算参数p，t，qz和q提供给一个视频编码器（没有示出）。

所述编码控制模块14可以包括（如果没有单独示出）所述ROI引擎31，如果这样的话，它应当具有作为一个输入的视频输入。然而，所述ROI引擎也可以作为输入到所述编码控制模块14的所述成本函数信号的源或发生器来示出。所述成本函数可以具有很大差异的形式。

本发明的各种实施例，包括最佳模式

方程（1）和（2）的成本分量$r(q_i,q_{i-1})$和$d(q_i,s_i)$的不同定义导致本发明的不同实施例。任何可感觉的成本函数，即任何具有上述标为（a）-(d)属性的成本函数，都能通向一个很好的实施例。本发明的一个实施例是任何视频编码器，优选是依据H.263，MPEG-4 VLBV或H.263+的视频编码器，其中处理器用上述的QP最佳算法来进行帧级QP控制。

这里描述两个实施例：一个简单模式，其实现起来相对便宜，以及一个高性能模式，其通常是优选的实施例，尽管在某些应用中优选简单模式。

简单模式

所述简单模式主要应用于依据MPEG-4和H.263的视频源编码器。在所述简单模式中，$r(q_i,q_{i-1})$和$d(q_i,s_i)$被如下定义：

$$r(q_i,q_{i-1})=\begin{cases}0, & \text{其中} |q_i-q_{i-1}|\le 2\\ +\infty, & \text{其它,}\end{cases} \qquad (9)$$

和

$$d(q_i,s_i)=(q_i-s_i)^2。 \quad (10)$$

所述优化器的目的是找到“最佳可表示的近似值”。所述简单模式强调所述可表示部分，在某种意义上，其寻找近似值的多数努力花费在寻找可表示的序列。（注意这在H.263＋由于所有序列都是可表示的，因此不是问题，但是一些序列要花费更多的比特。）所述简单模式的目标是用一个可表示的序列尽可能近地接近于所建议的QP序列。（所述ROI引擎总是建议一个序列，其中每个QP在[1,31]的范围内。然而，在[1,31]的范围内的每个QP对于一个序列的可表示性是不够的，例如对于H.263，并且没必要将所述ROI引擎的可表示性建议为一个可表示的序列。使所述ROI引擎免于承担建议一个可表示的序列的责任将使得所述ROI引擎更轻便，因为可表示性是一个依赖于系统的问题。对于H.263+来说，可表示性不是主要问题，因为在区间[1,31]的任何QP（只有整数）可以被表示。因此，所述简单模式对于依据H.263+的源编码器的操作不是特别有用。

一个简单模式的例子

假定ROI引擎已经为在图片内的一个任意位置产生了一个建议的QP序列6、7、3、2、1、3，并且在所述序列中在第一宏块的前面并紧接于其的宏块的QP值是5，因此其是在方程（7）中被用作q_0的值。（如果所述情况是前一宏块QP无效，那么帧级QP的值将被用作q_0。）

在该例子中，在被检查的序列中的宏块的数量是6，因此t=[1,6]并因此在上面的方程中使用的符号中的n=6。注意到建议的QP值的动态范围是从1到7，因为q_0被包括在所述范围内，在上面所用的符号中k=[1,7]。（如果建议的QP值的范围不包括q_0，那么该范围应该被扩展来这样做。）图4展示了下面指示的计算结果。

下一步骤是设定t=1，并为所述约束局部序列$C_k^*(S_1)$计算所述最小成本，其中k从1变化到7。如果在连续的QP值之间的差是2或更少，那么与花费在表示由方程（9）给出的Q的比特成比例的成本函数的分量产生值0，否则产生一个意味着不可接受的无穷大值（∞）。由于前一QP值被假定为5，因此由于由方程（9）给出的成本分量对$C_1^*(S_1)$和$C_2^*(S_1)$中的每个是无穷的，所以为两者的成本也是无穷的。

依据方程（7），对于$C_3^*(S_1)$的成本是

$$C_3^*(S_1) = d(6,3)+r(3,5),$$

其中，通过使用方程（9）和（10），变成

$$C_3^*(S_1) = (3-6)^2+0=9,$$

并因此被记录在图4的表中以$C_k^*(S_1)$开头栏下的k=3和t=1的格子中。

对于t=1和其它k值的成本函数的值被以一种相似的方式来计算，结果是：$C_4^*(S_1)=4$，$C_5^*(S_1)=1$，$C_6^*(S_1)=0$ 和$C_7^*(S_1)=1$。所有这些值都被存储起来以备后来使用（并在图4中的表中示出）。

下一步骤是确定为在序列中的第二宏块选择不同QP值的成本，即对于t=2。从方程（8），k=1和t=2时，

$$C_1^*(S_2)=d(7,1)+\min_j\left[r(1,j)+C_j^*(S_1)\right]。$$

为了求得$C_1^*(S_2)$的值，我们需要选择最佳可能的j值，即最小化$r(1,j)+C_j^*(S_1)$的j值。由于在相邻宏块之间的的QP值中只允许变化2（因为否则依据方程（9）的成本将是无穷大），对于k值为1来说，j的可能选择只有1、2和3。当选择j值时，我们可以向后看并看看对于前一状态（t=t-1）来说选择j所产生的最低总成本，或者我们可以使用方程（8）（也可以是方程（9）和（10））来计算所有可能j值的成本，接着就可以看出选择j所产生的最低成本。不管使用什么方法，结果是相同的。在这种情况下，在j等于1或2时，所述成本是无穷大（因为对于j=1或2来说，$C_j^*(S_1)$是无穷大），因此对于k=1的最低成本是对于j=3。用3来代替j，我们得到

$$C_1^*(S_2) = (1-7)^2 + \left[r(1,3)+C_3^*(S_1)\right]$$

$$= 6^2 + [0+9] = 45$$

并且因此$j^*(k,t) = j^*(1,2) = Best\,\mathrm{Pr}ev_t(k) = Best\,\mathrm{Pr}ev_2(1) = 3$。如图4所示的那样，这两个值（45和3）被存储。

下一步，对于k=2，（并且仍然t=2），

$$C_2^*(S_2)=d(7,2)+\min_j\left[r(2,j)+C_j^*(S_1)\right]$$

再者，不得不作出最佳可能的j值的选择。由于允许± 2的变化（在j和k之间），我们现在有4个可能的j值，即1、2、3和4。在前一状态（t=1），对于这些值

中的任何一个来说，最低成本是$C_4^*(S_1)$=4，因此我们选择j=4，产生，

$$C_2^*(S_2) = (2-7)^2 + \left[r(2,4) + C_4^*(S_1)\right]$$

$$= 5^2 + \left[0+4\right] = 29$$

因此，对于$C_2^*(S_2)$（即k=2和t=2）来说，最好的前面的状态（t=1）具有一个为4的QP值，因此设置$BestPrev_2(2)$=4。如图4所示，对于t=2 的其余值是$C_3^*(S_2)$=17，$C_4^*(S_2)$=9，$C_5^*(S_2)$=4，$C_6^*(S_2)$=1和$C_7^*(S_2)$=0。这些值又被与相应的$j^*(k,2)$值一起存储起来。对于所有的t值来说继续以相似的方式来处理直到计算出$C_7^*(S_6)$。

下一步，比较所有的$C_k^*(S_6)$值，并根据方程（6）选择最小值作为$C^*(S)$。在该例子中，$C^*(S) = \min_k\left[C_k^*(S)\right] = C_3^*(S) = 2$。现在我们通过使用存储的$j^*(k,t) = Best\,\mathrm{Pr}ev_t(k)$值来跟踪产生最低成本的“路线”。为了这样做，接着上面的伪码算法（倒数第三行），首先设定$q_n = k^*$，其中k^*是在t=n处产生最低成本的k值。在本例中，$q_6 = k^* = 3$。因此我们的最佳Q^*序列的最后一个值是3（由于对于k=3的成本是2，对于所有t=6的成本是最少的）。下一步设定i=n-1,并得到，

$$q_i = q_s = Best\,\mathrm{Pr}ev_{i+1}(q_{i+1}) = Best\,\mathrm{Pr}ev_6(3) = j^*(3,6) = 1,$$

因此在我们的Q^*序列中倒数第二个值q_{n-1}是1。对于i=n-2来说，$j^*(k,t)$的值是2，因此在我们的序列中的下一个QP值是2，等等。所选择的序列是这样的：6、6、4、2、1、3，其中除最后一个值以外所有的值都被在图4中用具有这些值的格子的阴影来表示，并且所述最后一个值（QP=3）是相应于t=6 时最低成本为2的k值， 该k值是3。

高性能模式

所述高性能模式可以为MPEG-4，H.263和H.263+中的每一个来使用。这

里，$r(q_i,q_{i-1})$和$d(q_i,s_i)$被定义如下：

$$r(q_i,q_{i-1})=\begin{cases}\lambda N_{q_i,q_{i-1}}, & \text{如果}q_i\text{是可表示的}\\ +\infty, & \text{其它，}\end{cases} \quad (11)$$

$$d(q_i,s_i)=\|mse(q_i)-mse(s_i)\| \quad (12)$$

其中λ是一个用作速率失真折中参数的预定非负值，$N_{q_i,q_{i-1}}$是需要用来表示由q_{i-1}给出的q_i比特数量，以及当q_i被用作所述宏块的QP时，$mse(q_i)$表示在初始宏块内容和重建的宏块内容之间的均方差。（所述均方差是一个在两个矢量、矩阵、图像或一般地在两个具有相同级别的多元素数量之间的测量误差。所述一个（多元素）数量的均方差等于所述指明的数量和某一参考数量之间的元素方面差值的平方的平均数（平均）。）（符号mse被用来表示在宏块的初始象素值与所述宏块的编码的象素值之间的测量误差。所述宏块的编码象素值受各种编码参数所影响，包括为所述宏块使用的QP。然而，所述宏块的初始象素值（所述参考数量）是一个到所述视频编码器的输入，并且不受任何的编码参数以任何方式的影响。这样，为了便于符号化，通常没有对所述参考数量的直接参考。这样，当q被选作用来编码所述宏块的QP时，mse（q）表示在一个宏块的初始象素值和所述块的编码象素值之间的均方差，或用简略的术语来说，当q被选作用来编码所述宏块的QP时，mse（q）表示在初始宏块和编码宏块之间的均方差。）

所述高性能模式强调了比特预算的效率管理，并具有一个在跟随所述建议QP序列和最小化花费在用于QP变化的总比特之间找到最佳折中的目的。依据方程（12）的$d(q_i,s_i)$计算在计算上是代价很高的，但不是令人望而却步的；所需要的计算还是比例如全范围块运动估计所需要的计算少得多，所述全范围块运动估计是视频编码器的主要计算集中的部分。

讨论

由依据本发明的视频编解码器所使用的帧级QP优化，可以通过比较编解码器输出的QP分布与所述视频源来辨别。典型地，一个具有依据本发明的帧级优化器的编解码器将响应于由所述QP的非正常变化所引起的急剧QP变换的需要。换句话说，这样一种编解码器将在产生这样一个需要之前开始增加（或降低）所述宏块的QP。这里，以前与所述宏块的一个光栅扫描顺序有关。所述宏块被以光栅扫描的顺序编码，即从左到右，从上到下。如果有一个可视的重要

特征，例如脸，例如在第六个宏块，并且所有的其它宏块包括不重要的特征，那么所述ROI引擎将建议第6宏块的QP突然降低。接着所建议的QP序列S（由所述ROI引擎建议的）可能看起来象{15,15,15,15,15,10,15,15,...}。术语“以前”和“非正常”指该1－D序列。降低所述QP的需要被认为在等于6的索引（即第六位置）处出现。在H.263编码器中使用的相应的帧级优化QP变化将看起来有点象{15，15，15，15，13，11，13，15，15，...}。所述第一变化在具有等于5的索引的宏块处开始，其小于第一可视重要宏块的索引，即它是在第一可视重要宏块之前。一个不使用帧级优化的近似器将可能只在索引等于6时开始改变QP。

应当明白，本发明除了上述的方式外，还可以包含构成失真测量和成本函数的其它方式。例如，与花费用来表示一个候选的QP序列的比特成比例的成本函数的分量，可以被与花费用来编码实际图像（包括用来表示所述QP序列的比特）而不仅仅是所述QP序列成比例（或其它基于或相关的）的分量所代替。在这样一种表达式中，所述不可能的QP序列应当被再次分配给一个无穷大的成本（即一个相对非常大的值），如在上面所描述的那样，并且所述失真测量（无论它是什么）应当被扩展来包括一个用来说明所述图像的失真（或编码损失）的分量。

应该理解到上述的配置仅仅是本发明原理应用的示例而已。多种其它修改和可选的配置可以被本领域的熟练的技术人员设计而不脱离目前本发明的范围，并且所附的权利要求书目的是覆盖这些修改和配置。

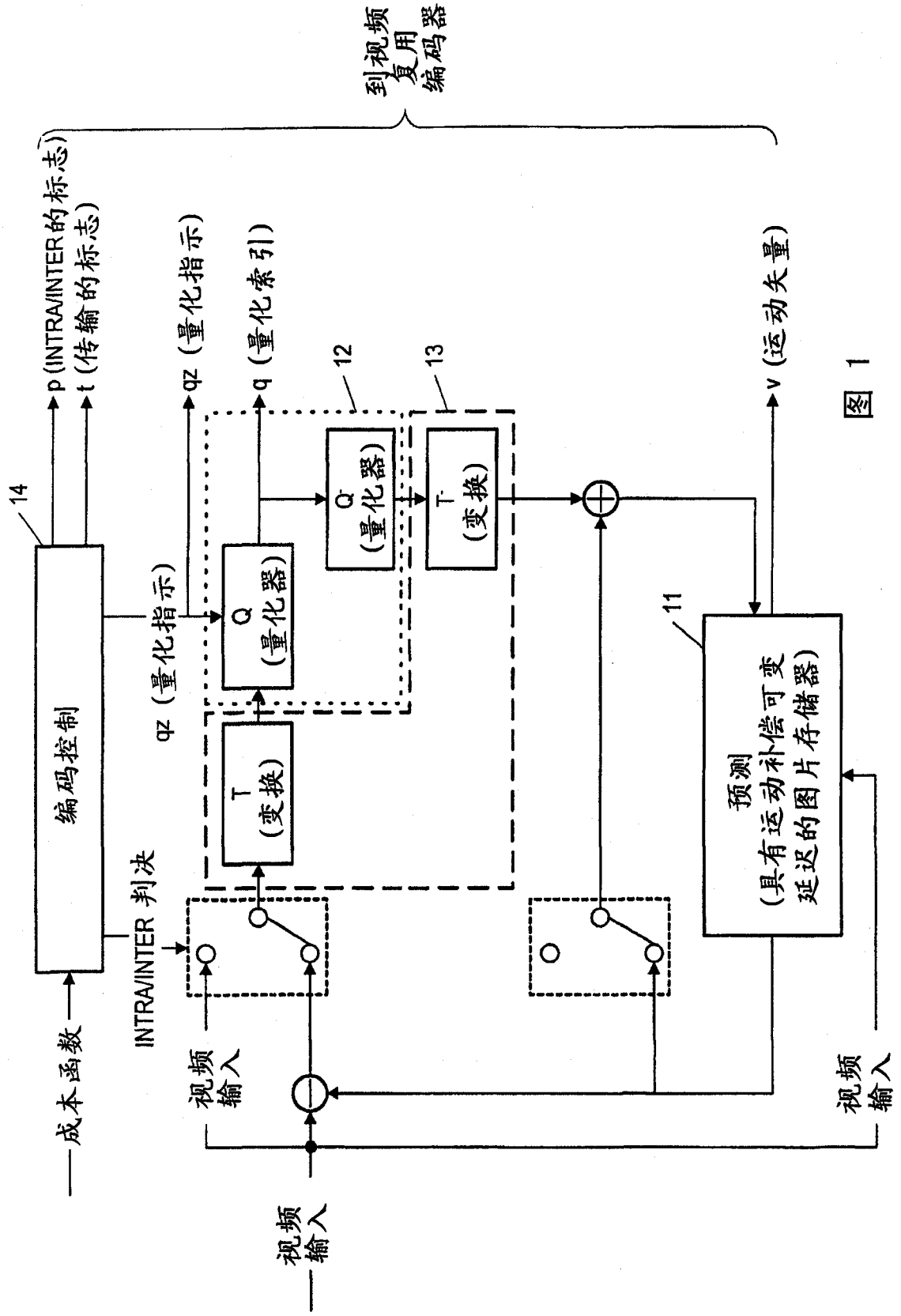

到视频复用编码器
p (INTRA/INTER的标志)
t (传输的标志)
qz (量化指示)
q (量化索引)
v (运动矢量)
12
13
14
11
编码控制
Q (量化器)
T (变换)
Q-1 (量化器)
T-1 (变换)
预测 (具有运动补偿可变延迟的图片存储器)
qz (量化指示)
INTRA/INTER 判决
成本函数
视频输入
视频输入
视频输入

图 1

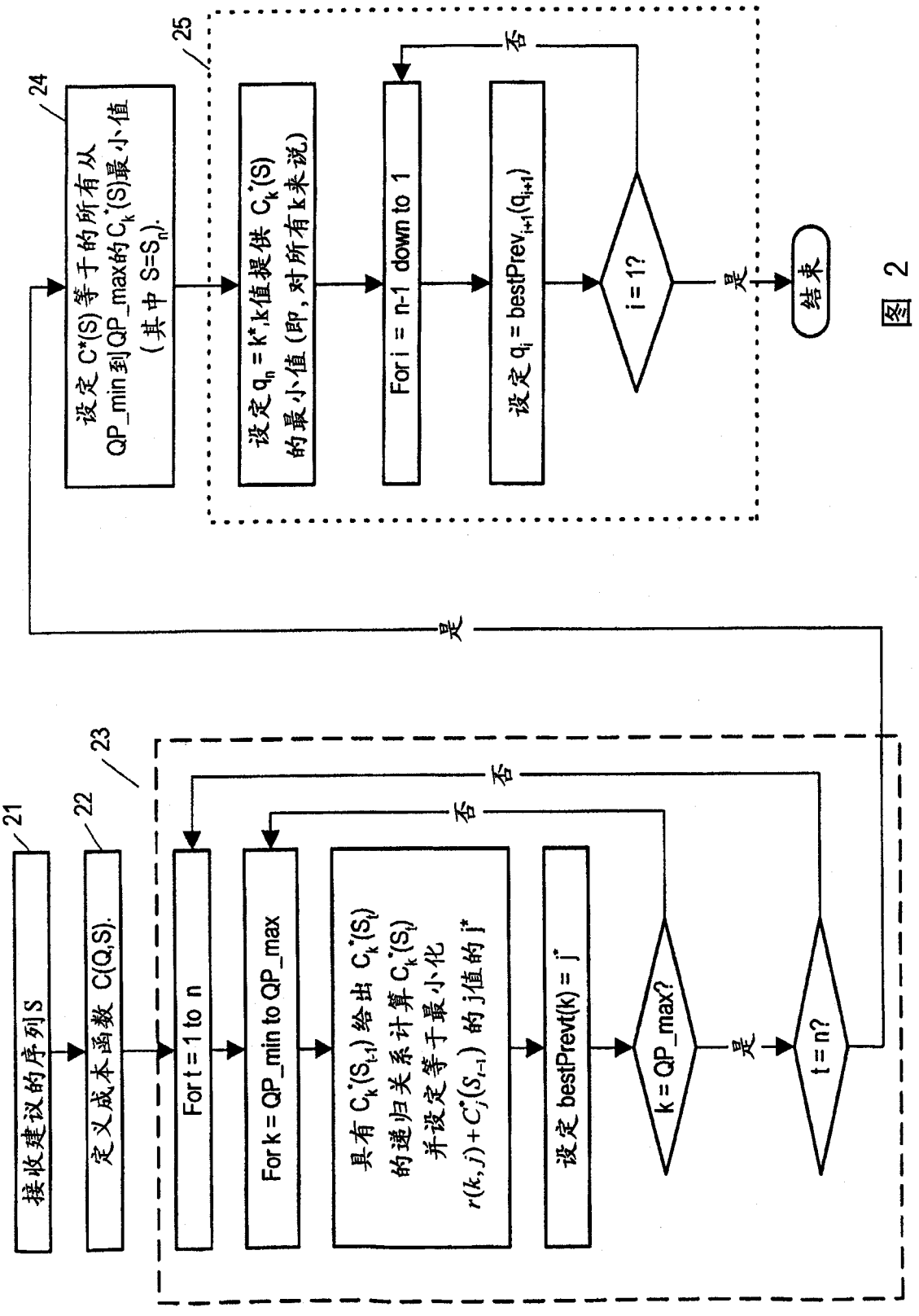

21
接收建议的序列S
22
定义成本函数 C(Q,S).
23
For t = 1 to n
For k = QP_min to QP_max
具有 $C_k^*(S_{t-1})$ 给出 $C_k^*(S_t)$ 的递归关系计算 $C_k^*(S_t)$ 并设定等于最小化 $r(k,j)+C_j^*(S_{t-1})$ 的j值的 j*
设定 bestPrevt(k) = j*
k = QP_max?
否
是
t = n?
否
是
24
设定 C*(S) 等于的所有从 QP_min到QP_max的$C_k^*(S)$最小值 (其中 $S=S_n$).
25
设定 $q_n = k^*$,k值提供 $C_k^*(S)$ 的最小值 (即,对所有k来说)
For i = n-1 down to 1
设定 $q_i = bestPrev_{i+1}(q_{i+1})$
i = 1?
否
是
是
结束

图 2

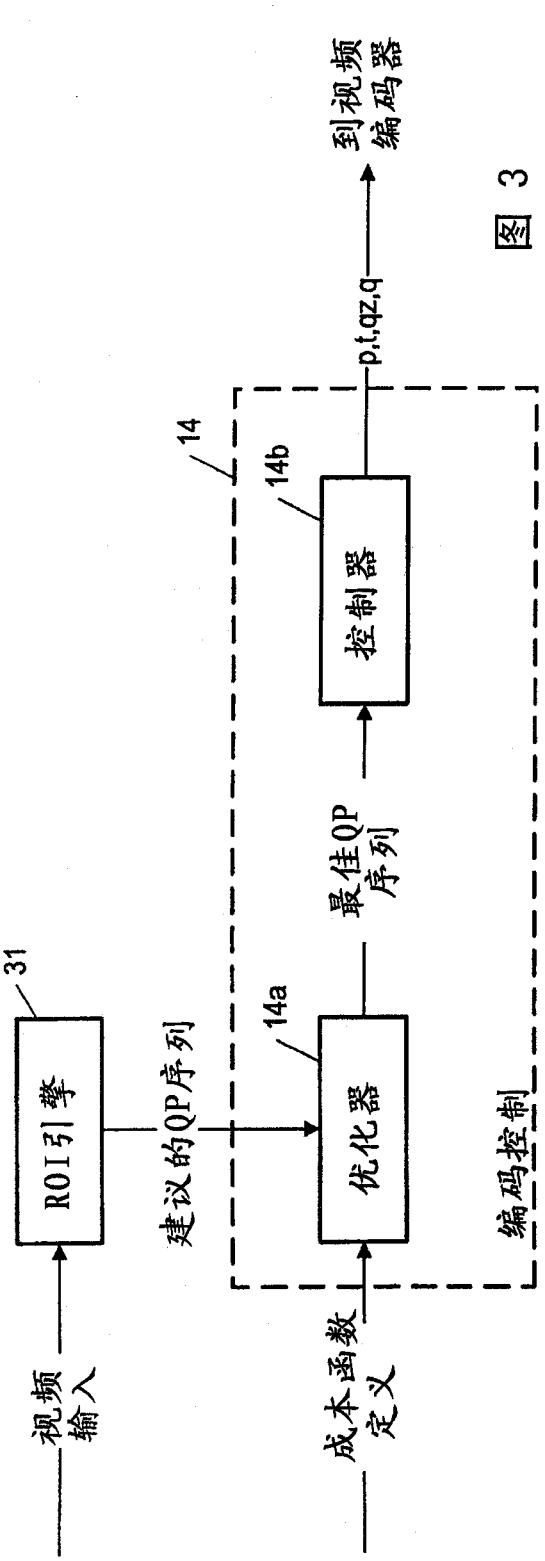


图 3

k \ t	1				2				3				4				5				6			
	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$	$(q_t-s_t)^2$	j^*	$C_t^*(S_t)$		j^*	$C_t^*(S_t)$
1	25	N/A	∞	36	3	45	4	3	21	1	3	5	0	2	2	4	1	6			
2	16	N/A	∞	25	4	29	1	4	10	0	4	2	1	2	3	1	1	3			
3	9	N/A	9	16	5	17	0	5	4	1	4	3	4	2	6	0	1	2			
4	4	N/A	4	9	6	9	1	6	2	4	4	6	9	2	11	1	2	4			
5	1	N/A	1	4	6	4	4	7	4	9	4	11	16	3	19	4	3	10			
6	0	N/A	0	1	6	1	9	7	9	16	4	18	25	4	31	9	4	20			
7	1	N/A	1	0	6	0	16	7	16	25	5	29	36	5	47	16	5	34			
S	6																				

图 4