

中图法分类号: TP391 文献标志码: A 文章编号: 1006-8961(2011)05-0733-07

论文索引信息: 张庆明, 彭强. 运用 H. 264/AVC 宏块编码特征的低复杂度率失真优化算法 [J]. 中国图象图形学报, 2011, 16(5): 733-739

运用 H. 264/AVC 宏块编码特征的低复杂度率失真优化算法

张庆明, 彭强

(西南交通大学信息科学与技术学院, 成都 610031)

摘要: 分析 H. 264/AVC 各帧间编码模式与最佳模式编码同一宏块 (MB) 时在比特率、参考帧选择及 CBP (coded block pattern) 上具有的相似性, 对 MB 各最佳编码模式的比特率和 CBP 分布进行了统计, 根据统计结果分析出比特率和 CBP 相结合能对编码模式进行比较准确的区分定位。并由此提出一种联合参考帧选择和模式选择的低复杂度 RDO (rate distortion optimization) 算法。算法利用 P 16 × 16 模式编码得到的比特率和 CBP 来确定候选编码模式, 用 P 16 × 16 模式的参考帧来确定其余帧间模式的参考帧。最后将候选模式在候选参考帧中进行运动估计和编码模式的 RDO, 选出最佳编码模式。实验结果表明, 提出算法与 H. 264/AVC 参考软件 JM12.0 相比, 平均比特率增加 1.32%、亮度 PSNR 减小 0.075 2 dB, 编码时间减少 71.20%。

关键词: 视频编码; 率失真优化; 宏块编码特征

A low complexity rate distortion optimization algorithm for H. 264/AVC using coding characteristic of macroblock

Zhang Qingming, Peng Qiang

(School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031 China)

Abstract: In this paper, the similarities of bits, reference frame selection and coded block pattern (CBP) of a coding macroblock (MB) between each inter mode and optimal mode have been investigated. The bits and CBP's distribution characteristics of a MB which encoded by the optimal mode have been analyzed, and the result shows that the coding modes can be differentiated by bits and CBP. Based on these characteristics, a novel low complexity rate distortion optimization (RDO) algorithm which combination reference frame selection with mode selection RDO is proposed. This algorithm utilizes P 16 × 16 mode's reference frame to select the candidate reference frames for other inter coding modes, and the candidate modes are selected by bits and CBP of mode P 16 × 16. The optimal mode will be selected in all candidate reference frames and candidate modes with RDO. The algorithm is evaluated using a wide range of test sequences of different classes. Comparing with RDO in JM12.0, the experiments results demonstrate a reduction coding time 71.20% in average with 1.32% of bits increase and 0.0752dB of Luma PSNR loss.

Keywords: video coding; RDO; MB coding characteristics

0 引言

为了尽可能地提高编码性能, H. 264/AVC 参考

软件 JM 分别对运动搜索和编码模式选择采用了计算复杂度很高的 RDO (rate distortion optimization) 算法^[1]。为降低计算复杂度, 文献[2]针对运动搜索提出了一种有效的快速算法, 在保证较高的匹配精

收稿日期: 2010-01-04; 修回日期: 2010-04-06

基金项目: 国家自然科学基金项目 (60972111)。

第一作者简介: 张庆明 (1975—), 男, 西南交通大学计算机应用技术专业博士研究生, 主要研究方向为视频压缩及传输控制技术、数字图像处理等。E-mail: zh. qingming@163. com。

度的同时,大幅度降低了搜索时间,大量测试结果显示,相比全搜索而言,平均搜索时间减少 40% ~ 70%,而亮度的 PSNR 仅降低 0.03 dB 左右,良好性能使得此算法被 JM 采用,作为一种推荐使用的快速算法。文献[3-4]根据在运动估计阶段获得的运动信息、文献[5]根据视频序列特征分别提出了参考帧选择的快速算法。在编码模式选择方面,帧内模式的快速选择主要依靠方向差值^[6]、利用 MB (microblock) 边缘信息^[7]、利用空域相关性^[8]、灰度分级^[9]等手段来实现;帧间模式快速选择主要依靠时/空域相关性^[10]、MB 内对象的运动特征^[11]等手段来实现。这些方法大多数是利用视频的时/空相关性来实现复杂度的降低,算法性能对视频时空相关性的依赖性很强。实验结果显示,时空相关性强的序列,算法性能较好,反之则性能一般。并且上述方法只是从局部减少了 RDO 的计算复杂度,而将运动搜索和编码模式选择结合起来的低复杂度 RDO 算法较少,文献[12]提出的算法虽然将两种 RDO 结合在了一起,但未对参考帧选择进行研究,算法也仅对运动平缓的序列有较好效果。

以上方法多数未考虑从编码过程中去寻找编码模式之间的相似特性,也没有充分利用编码过程中得到的一些重要中间结果,导致编码速度提高比例在很大程度上依赖于视频序列时空相关性的强弱。文献[13]提出的方法虽然利用了当前编码 MB 在编码过程中的 CBP(coded block pattern)信息,但也只将 CBP 信息用在了帧间编码模式选择上,而且没有利用编码过程中得到的比特率及参考帧等重要信息,所以在编码速度上提高依然不够充分。

本文首先分析了 H. 264/AVC 各帧间编码模式与最佳编码模式之间在比特率、参考帧选择和 CBP 上具有的相似性。对最佳模式的比特率、参考帧和 CBP 分布进行了统计,并分析得出比特率和 CBP 对编码模式具有较准确的区分功能。在此基础上提出一种利用 MB 编码特征的低复杂度 RDO 算法。首先,对编码 MB 采用 skip 模式编码,对满足 skip 模式编码条件的 MB 不再进行其余模式的编码,直接采用 skip 模式编码;对不满足 skip 模式编码条件的 MB,再用 P 16 × 16 模式进行编码,根据编码得到的比特率和 CBP 来确定候选编码模式,并利用 P 16 × 16 模式的参考帧来确定其余帧间模式的候选参考帧选择。最后将候选模式在候选参考帧中进行运动搜索和编码模式选择的 RDO 编码,选出最佳编码模

式。实验结果显示,与 JM12.0 的 RDO 相比,本文算法以比特率平均增加 1.32%,亮度 PSNR 平均减少 0.075 2 dB 的微弱代价,将平均编码时间减少了 71.20%。为便于下文叙述,将本文用到的符号和实验条件分别列于表 1 和表 2 中。需要说明的是,实验条件中的 QP 取值从 26 到 38,表明实验的码率范围较大,实验采用了各种运动剧烈程度和纹理丰富程度的序列,使得实验具有普遍性。

表 1 本文使用的符号及其含义

符 号	含 义
m	MB 编码模式
o	MB 最佳编码模式
$b_5b_4b_3b_2b_1b_0$	CBP 的位组成
$b_3b_2b_1b_0$	CBP 表示亮度的 4 位
$F_{m1}, F_{m2}, F_{m3}, F_{m4}$	采用模式 m 编码时,4 个 8 × 8 块的参考帧
$F_{o1}, F_{o2}, F_{o3}, F_{o4}$	采用模式 o 编码时,4 个 8 × 8 块的参考帧
R_m, R_o	模式 m/o 编码时需要的比特数
C_m, C_o	模式 m/o 编码时, $b_3b_2b_1b_0$ 中“1”的个数
N	编码 MB 数目
N_c	C_m 和 C_o 相等的 MB 数目
N_F	F_{mi} 和 F_{oi} 相同的 8 × 8 块数目, $i = 1, 2, 3, 4$
$N_{Co}(m, i)$	C_o 等于 i 的 MB 数目 ($i = 0, 1, 2, 3, 4$)

表 2 本文实验条件

配置项	参数
硬件环境	CPU: Intel Celeron M 1.6 GHz
软件测试模型	JM12.0
Profile	Baseline profile
GOP 结构	IPPP...
帧率	30 帧/s
编码帧数	100
参考帧数目	5
QP	26, 28, 30, 32, 34, 36, 38
搜索范围	32
搜索精度	1/4 像素精度
RDO	采用文献[1]的高复杂度 RDO
快速运动搜索算法	文献[2]的低复杂度搜索算法

1 编码模式特征分析

1.1 CBP 分析

CBP 是 H. 264/AVC 宏块层的码流语法元素,用来标记一个 8 × 8 的亮度块或色度块是否含有非 0 的变换系数^[7]。其位构成如图 1 所示。

图 1 表明, CBP 由 6 位二进制数 $b_5b_4b_3b_2b_1b_0$ 组

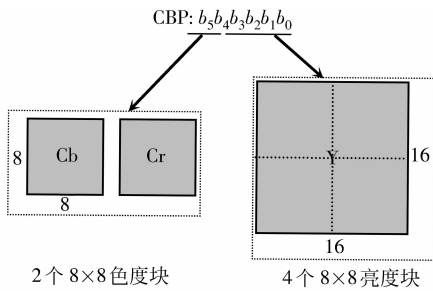


图1 CBP的位构成示意图

Fig. 1 The diagram of CBP's bit composition

成。其中 b_5b_4 用于标记 2 个色度块, $b_3b_2b_1b_0$ 用于标记 4 个亮度块, 各比特位取 1 时表示对应的 8×8 块含有非 0 变换系数, 需要对残差进行编码, 反之, 比特位取 0 时表示对应的 8×8 块不含有非 0 变换系数, 不需要对残差进行编码。显然, 表示亮度块的 $b_3b_2b_1b_0$ 中“1”的个数 C_m 也就是一个 MB 中需要编码的亮度 8×8 块个数。

1.2 各模式与最佳模式编码结果的相似性分析

JM 的 RDO 分两个阶段实现。首先对每种帧间模式在所有参考帧中进行运动搜索的 RDO, 寻找运动搜索 RD 代价 (RD-cost) 最小的块作为该模式编码用的预测块, 预测块所在帧为该模式的最佳参考帧; 然后再进行所有模式选择的 RDO, 选择编码模式 RD-cost 最小的模式作为最佳编码模式。由于每种帧间模式的运动搜索 RDO 都是以 RD-cost 最小的块作为该模式预测块, 直观上来看, 各帧间模式编码的 R_m, F_{mi}, C_m 应该与 R_o, F_{oi}, C_o 有很大的相似性。为准确刻画这种相似性, 分别对 CBP、参考帧和比特率定义相似度

$$K_C = \frac{N_C}{N} \quad (1)$$

$$K_F = \frac{N_F}{4N} \quad (2)$$

$$K_R = 1 - \frac{|R_m - R_o|}{R_o} \quad (3)$$

式(2)分母的因子“4”是因为参考帧数目 N_F 是按照 8×8 块统计的。式(3)中, $m \in \{P16 \times 16, P16 \times 8, P8 \times 16, P8 \times 8\}$ 。为验证这种相似性, 在表 2 条件下对多个 QCIF 序列进行了实验。表 3 统计的是 3 个不同运动剧烈程度的 QCIF 序列 glasgow, carphone, akiyo 的相似度 K_C, K_F, K_R 。可以看出, 各编码模式和最佳模式在编码同一 MB 时具有很强的相似性。以 carphone 序列为例, $P16 \times 16$ 模式编码

下的 K_C, K_F 和 K_R 分别等于 0.74, 0.83 和 0.80, 这说明, 用模式 $P16 \times 16$ 编码和用最佳模式编码的结果相比, 在所有 MB 中, 有 74% 的 MB 的 C_m 和 C_o 值是相同的, 有 83% 的 8×8 块的参考帧是同一参考帧, 比特率的接近程度达到了 80%。对于表 3 中的其他序列分析也有类似结果。以上分析表明, 对大多数 MB 而言, 即使 MB 在经过一种模式 (如 $P16 \times 16$) 编码后, 也能大致获得最终编码的 C_o 值, 参考帧及比特率 R_o 。

表3 3个序列的 K_C, K_F 和 K_R 统计Tab. 3 The K_C, K_F and K_R of 3 sequences

	模式	glasgow	carphone	akiyo	平均值
K_C	$P16 \times 16$	0.66	0.74	0.88	0.76
	$P16 \times 8$	0.69	0.76	0.89	0.78
	$P8 \times 16$	0.68	0.77	0.90	0.78
	$P8 \times 8$	0.80	0.85	0.93	0.85
K_F	$P16 \times 16$	0.83	0.83	0.86	0.84
	$P16 \times 8$	0.84	0.78	0.83	0.82
	$P8 \times 16$	0.84	0.77	0.85	0.82
	$P8 \times 8$	0.90	0.75	0.84	0.83
K_R	$P16 \times 16$	0.82	0.80	0.93	0.85
	$P16 \times 8$	0.82	0.63	0.91	0.79
	$P8 \times 16$	0.81	0.64	0.92	0.79
	$P8 \times 8$	0.82	0.30	0.85	0.65

1.3 各编码模式的 R_o 和 C_o 统计特性分析

H. 264/AVC 的树状结构运动补偿模式划分主要是针对残差的分布而设计的。一般而言, 大的分割尺寸适合于平坦区域, 而小尺寸适合于纹理丰富区域。另一方面, 纹理丰富区域在编码时比平坦区域需要更大的 R_o , 得到的 C_o 也更大。因此, MB 划分模式、 R_o 和 C_o 实际上都能大致反映出 MB 残差纹理的丰富程度。可以推断, 三者之间也应该存在一定的统计特性: 细节越丰富的 MB, 模式划分尺寸越小, 残差编码的 R_o 和 C_o 也越大, 反之亦然。为验证这种统计特性, 在表 2 条件下采用了不同运动剧烈程度的 QCIF 序列 foreman, carphone, suzie, highway 进行了实验。图 2 和表 4 为实验结果, 其中 C_o 分布百分率 $P_{Co}(i)$ 定义为

$$P_{Co}(i) = \frac{N_{Co}(m, i)}{N} \times 100\% \quad (4)$$

从图 2 可以看出, 各模式之间的 R_o 分布呈现较大差异。仔细观察不难发现, 4 种大尺寸模式 ($P16 \times 16, I16 \times 16, P16 \times 8, P8 \times 16$) 和 2 种小尺寸模式 ($P8 \times 8, I4 \times 4$) 编码的比特率分布存在一条比较

明显的分界线(图 2(b)中竖直虚线所示)。这条分界线约处于 30 比特处,具体表现为采用块尺寸较大的模式编码的 MB,绝大多数比特率都低于 30 比特,而采用块尺寸较小的模式编码的 MB,其比特率大部分都高于 30 比特。这说明,MB 的编码比特率对编码模式具有很好的区分作用。

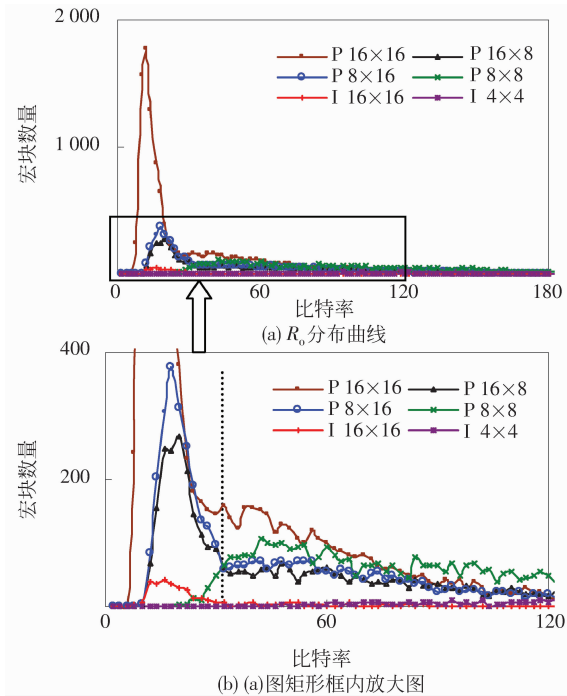


图 2 6 种模式的 R_o 分布曲线图

Fig. 2 The distribution curves of R_o for 6 modes

表 4 结果也显示, C_o 也对编码模式具有很强的区分作用,并且 C_o 的区分作用刚好与 R_o 的区分作用形成互补。一方面, C_o 只能取 5 个值,对编码模式的区分作用有限,从表 4 可以看出,对于模式 P 16x8, P 8x16 和 P 8x8,大约有 70% 至 80% 的 MB 的 C_o 值在 1 至 3 之间,并且分布很均匀,难以区分,但图 2 的结果表明, R_o 能很好地将 P 16x8, P 8x16 与 P 8x8 区分开,从这一点来说, R_o 弥补了 C_o 在模式区分上的不足。另一方面,图 2(b)分界线左侧曲线显示,4 种大尺寸模式都有大量 MB 的比特率集中在 0 至 30 这个区域,不易区分,但 C_o 的分布较好地解决了这个问题:对模式 P 16x16 和 I 16x16,约 90% 的 MB 的 C_o 值小于 2,而模式 P 16x8 和 P 8x16 的 C_o 值大部分分布在 1 至 3 之间,所以 C_o 能将 P 16x16 和 I 16x16 与 P 16x8 和 P 8x16 区分开。对两种小尺寸模式进行分析,也有类似的结果。这表明, C_o 弥补了 R_o 在模式区分上

的不足。上述分析表明,可以结合 R_o 和 C_o 的分布情况对编码模式进行比较准确的区分。

表 4 C_o 分布统计结果

Tab. 4 Statistical results of C_o

	P 16 × 16	I 16 × 16	P 16 × 8	P 8 × 16	P 8 × 8	I 4 × 4
$P_{C_o}(0)$	64.72	72.60	23.47	25.99	3.54	4.89
$P_{C_o}(1)$	17.07	15.47	26.83	27.77	16.23	3.33
$P_{C_o}(2)$	10.70	6.51	25.88	26.41	29.07	21.86
$P_{C_o}(3)$	5.06	3.45	15.02	13.56	29.16	10.93
$P_{C_o}(4)$	2.45	1.97	8.81	6.28	22.00	59.00

2 基于 MB 级编码特征的低复杂度率失真优化算法

考虑将 1.2 节和 1.3 节的分析结果结合起来。首先用一种模式对 MB 进行编码,用编码得到的 R_m 和 C_m 分别替代 R_o 和 C_o 来进行编码模式区分,选出若干适合当前 MB 编码的模式作为候选编码模式,余下的率失真优化编码过程仅在候选编码模式中进行。基于这一思路,提出一种基于 MB 编码特征的低复杂度 RDO 算法。考虑到前面分析是建立在 P 帧上进行的,同时由于 I 帧在整个序列中数量很少,所以提出算法仅针对 P 帧。

2.1 skip 模式优先选择

skip 作为一种特殊编码模式,不需要进行运动搜索,残差和运动向量差也都不需要编码,极大地节约了编码时间和比特率,但会带来一定程度的失真。如果当前编码 MB 满足一定条件^[1], skip 模式编码也会将失真控制在可以接受的程度。因此该算法在编码过程中优先采用 skip 模式编码,一旦满足 skip 模式编码条件,则跳过其余编码模式,将当前编码 MB 的最佳模式确定为 skip 模式,这样可以在一定程度上加快编码速度。

2.2 候选参考帧和候选模式的快速选择

从表 3 可以看出, P 16x16 模式与最佳模式的 K_c , K_f 和 K_r 值都较高,这说明 P 16x16 模式在各方面与最佳模式都比较接近。因此,对一个不满足 skip 模式编码的 MB 而言,可以先采用 P 16x16 对其进行 RDO 编码,获得此模式下的参考帧、 R_m 和 C_m ,用于确定候选参考帧和候选编码模式。如图 3 所示,假设 P 16x16 模式的参考帧为第 $n-k$ 帧,考虑到大多数 MB 的最佳参考帧都是邻近当前编码帧

的,只有少数 MB 的最佳匹配块在与编码帧相隔较远的参考帧中,因此,该算法将第 $n-k$ 到第 $n-1$ 帧 (图 3 中的灰色帧) 的所有帧都选为其余帧间编码模式的候选参考帧。

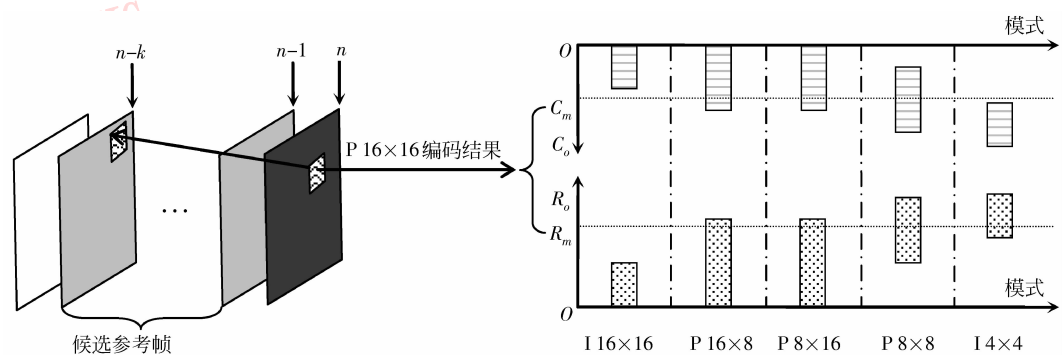


图 3 候选编码模式和候选参考帧选择过程示意图

Fig.3 The diagram of candidate reference frames and candidate modes' selection process

候选编码模式由模式 P 16 × 16 编码的 R_m 、 C_m 和 R_o 、 C_o 分布共同确定。根据 1.3 节讨论结果,不同模式下的 R_o 和 C_o 会比较集中地分布在不同的子区间。因此,考虑为每种最佳模式的 R_o 设计一个子区间,使得 90% 左右的 MB 编码的 R_o 能够落入这一子区间。类似地,也可以为 C_o 设计一个子区间。以 14 × 4 为例,在比特率子区间 $[79, +\infty)$ 就包含了 90% 以上的 MB, C_o 子区间 $[2, 4]$ 也包含了 90% 以上的 MB (见表 4)。除 P 16 × 16 外,为其余模式设计的子区间如表 5 所示。

表 5 本文算法中设置的 R_o 和 C_o 子区间

Tab.5 The subintervals of C_o and R_o in this paper

	I 16 × 16	P 16 × 8	P 8 × 16	P 8 × 8	I 4 × 4
C_o	[0, 1]	[0, 2]	[0, 2]	[2, 4]	[2, 4]
R_o	[0, 51]	[0, 115]	[0, 107]	[32, + ∞)	[79, + ∞)

由于 P 16 × 16 的 R_m 和 C_m 分别与 R_o 和 C_o 比较接近,并且在编码未结束时 R_o 和 C_o 是未知的,因此,考虑用 R_m 和 C_m 分别替代 R_o 和 C_o 进行模式选择。假设 M_R 为所有包含 R_m 的子区间对应的模式的集合, M_C 为所有包含 C_m 的子区间对应的模式的集合。因为 R_m 和 C_m 结合起来能更准确地确定编码模式,所以将 M_R 与 M_C 的交集 M 中所有的模式设定为候选编码模式。以图 3 所示为例进行说明,在这种情况下:

$$M_R = \{P 16 \times 8, P 8 \times 16, P 8 \times 8, I 4 \times 4\} \quad (5)$$

$$M_C = \{P 16 \times 8, P 8 \times 16, P 8 \times 8\} \quad (6)$$

因此,

$$M = \{P 16 \times 8, P 8 \times 16, P 8 \times 8\} \quad (7)$$

待候选参考帧和候选编码模式确定以后,再对候选模式在候选参考帧内进行运动估计的 RDO 和编码模式选择的 RDO,选择最佳的参考帧和最佳编码模式。注意到本文算法中没有涉及 8 × 8 尺寸以下几种小模式 (即 P 8 × 4, P 4 × 8 和 P 4 × 4) 的选择,这是由于 CBP 是用于 8 × 8 块的,能区分最小模式尺寸是 8 × 8,所以本文算法在选择候选模式时不考虑 8 × 8 以下的子模式快速选择,如果在候选模式中有 P 8 × 8,则采用 JM 的方法来为每个 8 × 8 块选择子模式。

综上所述,本文提出的低复杂度 RDO 算法步骤如下:

1) 对编码 MB 首先用 skip 模式进行编码,判断 MB 是否满足 skip 模式编码条件,如果满足,则将编码 MB 的最佳编码模式确定为 skip 模式,并结束当前 MB 编码。否则进入步骤 2);

2) 结合运动估计的 RDO,对 MB 采用 P 16 × 16 模式进行编码,获得此模式下的最佳参考帧 $n-k$, 编码比特率 R_m 和 C_m ;

3) 将其余帧间编码模式的候选参考帧设定为第 $n-k$ 帧到第 $n-1$ 帧;

4) 在表 5 中查找所有包含 R_m 的子区间对应的模式,将这些模式组成集合 M_R ,类似的方法得到集合 M_C 。再计算出 M_R 与 M_C 的交集 M ,将 M 中的所有模式确定为候选编码模式;

5) 如果 M 为非空集合,用 M 中的模式在步骤 3) 得到的候选参考帧中进行 RDO 编码,联合 P 16 × 16 模式编码结果,选出最佳编码模式。如果 M 为空集,则将当前 MB 的最佳编码模式确定为 P 16 × 16。

3 模拟实验和分析

为验证算法的有效性,在表 2 实验条件下,采用不同运动剧烈程度的 9 个 QCIF 序列 mobile, glasgow, coastguard, foreman, carphone, suzie, hall_monitor, container, akiyo 进行实验。将 JM12.0 采用 2 个参考帧、3 个参考帧的 RDO 算法、文献[12]算法、本文算法分别与 JM12.0 采用 5 个参考帧的 RDO 算法进行了对比。算法性能通过比较编码比特率增加率 ΔB 、亮度 PSNR 下降量 ΔY 和编码总时

间加速比例 ΔT 3 个方面进行对比。 ΔB 、 ΔY 、 ΔT 定义为

$$\Delta B = \frac{b_{\text{compare}} - b_{\text{org}}}{b_{\text{org}}} \times 100\% \tag{8}$$

$$\Delta Y = Y_{\text{compare}} - Y_{\text{org}} \tag{9}$$

$$\Delta T = \frac{T_{\text{org}} - T_{\text{compare}}}{T_{\text{org}}} \times 100\% \tag{10}$$

式中,org 表示 JM12.0 采用 5 个参考帧的 RDO 算法,compare 表示拟比较的 4 个算法。 B 、 Y 和 T 分别表示编码一个序列的总比特率,每帧亮度分量的 PSNR 平均值和编码总时间。表 6 为实验结果统计。

表 6 3 个算法在 ΔB 、 ΔY 、 ΔT 方面的性能实验对比
Tab.6 The performance compare of three algorithms on ΔB , ΔY , ΔT

	JM12-2REF			JM12-3REF			JM12-FAST ^[12]			本文算法		
	$\Delta B/\%$	$\Delta Y/\text{dB}$	$\Delta T/\%$	$\Delta B/\%$	$\Delta Y/\text{dB}$	$\Delta T/\%$	$\Delta B/\%$	$\Delta Y/\text{dB}$	$\Delta T/\%$	$\Delta B/\%$	$\Delta Y/\text{dB}$	$\Delta T/\%$
glasgow	2.57	0.001 4	42.09	0.90	0.001 0	29.60	-0.18	-0.008 6	9.84	2.91	-0.025 9	61.54
mobile	5.59	-0.189 4	38.42	2.49	-0.111 3	27.04	-0.03	-0.018 0	3.54	2.26	-0.116 4	60.52
coastguard	0.35	-0.028 5	40.61	0.37	-0.007 9	28.49	-0.18	-0.002 9	10.47	0.90	-0.025 7	66.63
foreman	0.19	-0.113 2	38.40	-0.08	-0.041 5	27.38	-0.49	-0.008 5	12.42	1.49	-0.123 6	72.54
carphone	3.35	-0.248 1	36.09	0.81	-0.092 5	26.11	-0.41	-0.005 4	16.37	1.71	-0.156 1	73.57
suzie	0.24	-0.020 8	39.41	-0.09	-0.002 6	28.37	0.08	-0.004 5	19.62	2.15	-0.079 0	74.36
hall_monitor	-0.35	-0.028 7	32.56	-0.34	0.001 0	23.72	-0.09	-0.017 9	42.83	0.81	-0.038 7	75.51
container	4.52	-0.083 4	32.06	0.26	-0.017 0	23.68	-0.72	-0.010 0	50.16	0.31	-0.049 8	77.22
akiyo	-0.70	-0.065 9	32.87	-1.11	-0.038 6	24.42	-0.66	-0.007 9	47.76	-0.65	-0.061 4	78.91
平均值	1.75	-0.086 3	36.95	0.35	-0.034 4	26.53	-0.30	-0.008 3	23.67	1.32	-0.075 2	71.20

从表 6 的实验结果来看,各算法的性能还是有比较明显的差异。从所有测试序列的平均实验结果来看,文献[12]算法在比特率增加率、亮度 PSNR 下降量上保持了较好性能,平均 PSNR 仅下降 0.008 3 dB,平均码率甚至还下降了 0.3%,但在编码总时间加速比例上则不理想,平均时间减少只有 23.67%。JM12.0 采用 2 个参考帧、3 个参考帧的 RDO 算法,因为参考帧数目固定,所以对各种序列的加速比例保持了较好的稳定性,平均加速比例分别只有 36.95%和 26.53%。本文算法在比特率增加率、亮度 PSNR 下降量上介于 JM12.0 采用 2 个参考帧、3 个参考帧的 RDO 算法之间,但在编码总时间加速比例上提高非常明显,平均编码时间减少达到了 71.20%。

具体到不同序列的实验结果来看,文献[12]算法对序列运动剧烈程度和纹理丰富程度依赖性很强,具体表现在对运动平缓和纹理简单的序列加速比例较大,而对运动较剧烈的序列和纹理丰富的序

列加速比例较小。这主要是因为编码模式选择过程中利用的是宏块之间的空间相关性,而纹理简单,运动平缓的序列空间相关性很强。序列 hall_monitor, container 和 akiyo 存在大量静止背景,这些静止背景大部分采用了 skip 模式编码,所以实验结果反映出来的 ΔT 就比较高,分别达到了 42.83%、50.16%和 47.76%。而对纹理丰富的 mobile 序列、运动剧烈的 coastguard 序列, ΔT 仅有 3.54%和 10.47%。本文算法能将比特率增加率、亮度 PSNR 下降量保持在很低的同时,大幅度降低编码时间,这主要是由于本文算法利用的是编码宏块在 $P16 \times 16$ 模式编码过程中得到的参考帧、比特率和 CBP 信息,而这些信息来源于编码宏块本身,并不依赖于宏块之间的时空相关性。所以对运动剧烈的 coastguard 序列和纹理丰富的 mobile 序列也能节省 60%以上的编码时间。最后需要指出的是,文献[12]也采用了与本文算法相同的快速运动搜索算法和优先采用 skip 模式对宏块进行编码的方法,

所以 skip 模式对本文算法在编码时间减少上的贡献是比较小的。总的来说,本文算法对各种序列在保证不明显降低率失真性能的情况下,能起到十分明显的降低计算复杂度的作用。

4 结 论

H. 264/AVC 支持多参考帧和多种编码模式,其 RDO 编码过程需要遍历所有参考帧和模式,每种模式又都需要在 RDO 框架下进行完整的编解码,这大大增加了编码的复杂度。针对这一问题,本文通过从 MB 的 $P16 \times 16$ 模式编码过程中得到的 MB 自身的参考帧、CBP 和比特率等编码特征,提出了一种基于 MB 级编码特征的低复杂度 RDO 算法。算法实现了运动搜索和编码模式选择的低复杂度 RDO,克服了采用时空相关性的低复杂度算法所固有的过于依赖在时空上相邻 MB 信息的缺陷。实验结果表明,该算法有效降低了对序列的运动剧烈程度和纹理丰富程度的依赖性,在不同比特率下,对各种测试序列都极大地提高了编码速度。

参考文献 (References)

- [1] Keng Pang Lim, Gary Sullivan, Thomas Wiegand. Text description of joint model reference encoding methods and decoding concealment methods [EB/OL]. (2005-01-21) [2008-09-21]. http://wftp3.itu.int/av-arch/jvt-site/2005_01_HongKong/JVT-N046r1.doc.
- [2] Chen Zhibo, Zhou Peng, He Yun, et al. Fast integer pel and fractional pel motion estimation for JVT [EB/OL]. (2002-12-05) [2008-10-11]. <http://wftp3.itu.int/av-arch/jvt-site/>.
- [3] Kangjun L, Gwanggil J, Falcon R, et al. An adaptive fast multiple reference frame selection algorithm for H. 264/AVC using reference region data [EB/OL]. (2009-10-07) [2009-11-05]. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5336231.
- [4] Zhang Shufang, Li Hua. A multi-reference frames fast selection algorithm for H. 264 [J]. Acta Electronica Sinica, 2009, 37(1): 62-66. [张淑芳, 李华. 基于 H. 264 的多参考帧快速选择算法 [J]. 电子学报, 2009, 37(1): 62-66.]
- [5] Chang Kan, Men Aidong, Zhang Wenhao. A novel fast multiple reference frames selection method for H. 264/AVC [EB/OL]. (2009-10-11) [2009-11-14]. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5329402.
- [6] Ji Lei, Su Xiuqin, Li Xiang. A low complexity fast intra mode decision algorithm for H. 264/AVC [EB/OL]. (2009-10-17) [2009-11-20]. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5304048.
- [7] Wei Zhenyu, Li Hongliang, King N N. An efficient intra mode selection algorithm for H. 264 based on fast edge classification [EB/OL]. (2007-05-27) [2009-08-28]. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4253467.
- [8] Han Yu, Men Aidong, Quan Ziyi. Combined spatial correlation of coded neighboring blocks for intra-frame coding mode prediction [EB/OL]. (2009-06-04) [2009-11-12]. <http://ieeexplore.ieee.org/iel5/5318882/5318883/5318969.pdf?arnumber=5318969>.
- [9] Ren Fang, Dong Jinming. Fast and efficient intra mode selection for H. 264/AVC [EB/OL]. (2010-01-22) [2010-02-21]. <http://www.computer.org/portal/web/csdl/doi/10.1109/ICCMS.2010.148>.
- [10] Yuan Qingsheng, Chen Yiqiang, Kang Yi. A fast region-based inter mode selection algorithm [EB/OL]. (2008-06-23) [2009-08-02]. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4607563.
- [11] Yu A C, Martin G R. Advanced block size selection algorithm for inter frame coding in H. 264/MPEG-4 AVC [EB/OL]. (2004-04-27) [2009-08-9]. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1418698.
- [12] Inchoon Choi, Woong Choi, Jeyun Lee, et al. The fast mode decision with fast motion estimation [EB/OL]. (2005-01-17) [2008-04-16]. http://wftp3.itu.int/av-arch/jvt-site/2005_01_HongKong/JVT-N013.doc.
- [13] Chen Boyuan, Yang Shih-Hsuan. Using H. 264 coded block patterns for fast inter-mode selection [EB/OL]. (2008-06-23) [2009-06-27]. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4607536.