基于运动矢量差(MVD)的 CABAC 新算法*

李康顺[1,2], 金 晶[1], 王 峰[3]
(1. 江西理工大学 信息工程学院, 江西 赣州 341000; 2. 华南农业大学 信息学院, 广州 510642; 3. 武汉大学 软件工程国家重点实验室, 武汉 430072)

摘 要: 在 H. 264/AVC 视频编码标准中,基于上下文的自适应二进制算术编码(CABAC)主要应用于主要档次中,并且具有较高的压缩效率。首先分析了 CABAC 编码原理和运动矢量差(MVD)各分量的上下文模型选择原理,而后提出了在帧间编码分割块尺寸下,充分地利用当前块 MVD 与当前块 MV 的相关性、当前块 MVD 与已编码相邻块 MVD 的相关性以及当前块 MVD 中各分量之间相关性的 CABAC 优化算法(CABAC_1 算法)。通过实验表明: 较基准 CABAC 算法,CABAC_1 算法一方面能有效地降低 2% 左右的编码时间及确保了编码序列的视觉质量;另一方面能够有效地节约在编码中帧间编码帧的码流(比特流),其中 P 帧平均节约了 10% 左右的比特流,B 帧节约了 5% 左右的比特流。因此,CABAC_1 算法是一种有效的优化算法。

关键词: 运动矢量差; H. 264; CABAC 编码; 码率; 块分割
中图分类号: TN919. 8 **文献标志码**: A **文章编号**: 1001-3695(2012)02-0744-05
doi:10. 3969/j. issn. 1001-3695. 2012. 02. 092

Novel algorithm for CABAC based on motion vector difference

LI Kang-shun[1,2], JIN Jing[1], WANG Feng[3]
(1. *School of Information Engineering*, *Jiangxi University of Science & Technology*, *Ganzhou Jiangxi* 341000, *China*; 2. *School of Information*, *South China Agricultural University*, *Guangzhou* 510642, *China*; 3. *State Key Laboratory of Software Engineering*, *Wuhan University*, *Wuhan* 430072, *China*)

Abstract: In H. 264/AVC video compression standard, context-based adaptive binary arithmetic coding (CABAC) is mainly used in main profile and has high compression efficiency. At the beginning, this paper introduced and analyzed the theory of CABAC algorithm, the context model selection theory of each component which belonged to MVD, then it proposed an optimization CABAC algorithm (CABAC_1 algorithm) based on motion vector difference (MVD) which were full use of the relevance between the current block MV and the current block MVD, the relevance between the current block MVD and the adjacent encoded block MVD and the relevance between the vertical and horizontal component of MVD. Compared with the standard CABAC algorithm, the CABAC_1 algorithm not only can effectively decrease the encoding time and rate and ensure the visual quality of coding sequence, but also can effectively save the code stream (bit streams) of the inter coding frames. The experiments show that it can save about an average of 10% bit streams of P frames and about an average of 5% bit streams of B frames by using CABAC_1. Therefore, CABAC_1 algorithm is an effective optimization algorithm.

Key words: motion vector difference; H. 264; CABAC encoding; code rate; block partition

0 引言

H. 264/AVC 是由 ITU-T 和 ISO/IEC 联合制定的新一代视频编码标准,其采用两种熵编码方案:基于上下文的自适应二进制算术编码(CABAC)和基于上下文的可变长编码(CAVLC)[1,2]。CABAC 算法在运用高效的算术编码思想的同时,充分地利用视频流上下文信息的相关统计特性对不同的比特流信息自适应地进行上下文调整,并实时更新相应的概率模型,进而提高编码效率[2]。在相同的编码条件下,CABAC 编码与 CAVLC 编码相比,降低 10% ~20% 的平均比特率,并且在低码率情况下的编码效率优势更为明显[3~5]。

作为一种性能优异的熵编码方法,CABAC 受到了学术界普遍的重视,其优化思路主要体现在两方面:a)进一步提高编码效率,以降低码率;b)降低编码的计算复杂度,以提高编码速率。文献[6]在分析 CABAC 中二进制比特位概率估计更新的计算复杂度后,提出了一种基于概率估计更新的 CABAC 加速算法。该算法主要是将比特流以 8 位数据包形式进行概率估计的更新,在一定程度上减少了概率更新的计算复杂度,但却舍弃了比特位概率估计的精确度,对编码后视频序列的主观视觉质量产生较大影响。文献[7]算法主要运用了当前块 MVD 的垂直分量与其对应水平分量的相关性以及与相邻块 MVD 各分量之间的相关性,从而提高对字符的概率估计的准确性,进而在对 MVD 编码中节省了码字,并降低了 5% 左右 P 帧比特流和 3% 左右的 B 帧比特流,但是该算法却忽略与当前块 MV 各分量的相关性。文献[8]提出了一种基于运动矢量差的二进制优化算法,该算法主要是结合不同的量化步长 *QP*

收稿日期: 2011-06-19; **修回日期**: 2011-08-15 **基金项目**: 国家自然科学基金资助项目(70971043);江西省自然科学基金资助项目(2008GZS0028)

作者简介:李康顺(1962-),男, 江西兴国人,教授,博导,博士,主要研究方向为图像视觉、演化计算、演化硬件(likangshun@ sina. com);金晶(1986-),男,硕士研究生,主要研究方向为嵌入式视频编码算法、演化计算;王峰(1981-),女,湖北咸宁人,博士,主要研究方向为演化硬件.

对 MVD 的二进制方式 UEGK(S,K)进行合理的参数选择,并降低了 1.5% 左右的比特流。

1　CABAC 编码算法原理

在 H.264/AVC 视频编码中,CABAC 编码首先对需要编码的非二进制语法元素符号进行二进制化,将其映射为一个由字符"0"和"1"构成的二进制序列。二进制化后的序列分成两部分[6~10]:

a)统计概率近似相等的字符串通过旁路编码器进行等概率($P=1/2$)的算术编码。

b)统计概率不等的字符串,需结合该语法元素的上下文内容进行上下文模型的选择,并结合概率模型码表对输入编码字符进行有条件概率估计,最后在常规编码器中进行二进制算术编码。

1.1　CABAC 编码的概率模型的优化

CABAC 在计算复杂度和编码效率之间作了折中,在 0.01875 ~0.5 范围内建立了一个查表的概率模型,概率量化为 64 个值,这些概率对应于 LPS(least probability symbol)字符,MPS(most probability symbol)字符的概率为 $1-P_{lps}$,则概率的刷新是按照已定义的法则在概率模型表中查找。

如图 1 所示,依据式(1)(2)列出 LPS 被量化后的概率值,并以 σ 编号一一对应于状态 state,实线和虚线指示了概率的刷新值。概率的刷新有两个方向:a)当前字符是 LPS 时,在图上顺着虚线往左寻找;b)当前字符是 MPS 时,在图上顺着实线往右寻找。

$$P_\sigma=\partial\times P_{\sigma-1}\text{ 且 }P_0=0.5\quad \sigma=0,1,2,3,\cdots,63 \tag{1}$$

$$\partial=\left(\frac{0.01875}{0.5}\right)^{\frac{1}{63}} \tag{2}$$

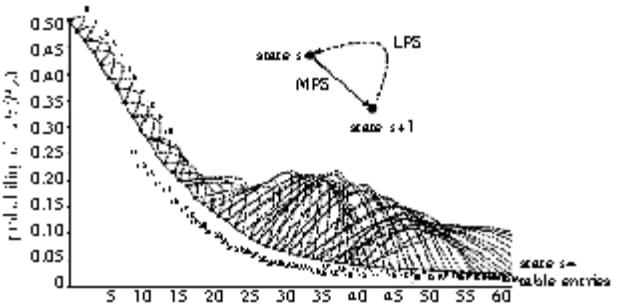


图 1　CABAC 概率估计和刷新模型

为进一步减小运算的复杂度,将区间长度 $R(2^8\leqslant R\leqslant 2^9)$ 按统一的量化步长 $Q(R)$ 分成四个等长的区间 $Q=\{Q_0,Q_1,Q_2,Q_3\}$,进而得到一个 4×64 的二位概率表格。

1.2　CABAC 的初始化及上下文模型

在 H.264/AVC 视频编码中,slice 是自适应编码的基本单位。因此,新 slice 数据到来时,CABAC 要依据各 slice 的量化步长 QP 参数对各概率状态进行初始化。具体的初始化过程可参考文献[6,7]。

H.264/AVC 将一个片内出现的数据划分为 399 个上下文模型,每个模型用索引 CtxIdx 标志,并在每个模型的内部进行概率的查找和更新。同时,每个上下文模型都单独使用所对应的码表进行概率状态维护,并且这些模型的划分精确到了比特,大多数的比特和其邻近的比特位都处于不同的上下文模型之中。在解码器中,对于输入的比特位首先查找它属于哪个上下文模型,然后再查找该上下文模型对应的概率表以及算术编码的递归区间。

2　运动矢量差(MVD)概述

2.1　运动矢量差(MVD)的计算

在 H.264/AVC 标准中,帧间编码帧可分割成 16×16,16×8,8×16,8×8,8×4,4×8 及 4×4 的块[11],每个块都有独立的运动矢量(motion vector, MV),对每个块的运动矢量 MV 编码时需要巨大的比特量,特别是当选择较小的块分割(8×8,8×4,4×8 及 4×4)时,所需的比特量更大[12]。然而,相邻块的 MV 通常有着很大的相关性,如图 2 所示,当前 C 块的 MV 预测需利用相邻已编码块(左边的 A 块,上边的 B 块,右上方的 D 块)的 MV,得到运动矢量的预测值(predicted motion vector, MVP);然后以 MVP 为搜索起始点,在一定的范围内搜索当前编码块的最佳匹配块。当前编码块与最佳匹配块之间的 MV 为最优化的运动矢量 MV,记做 $MV_{optimal}$。运动矢量差 MVD 可由式(3)得到。

$$MVD=MV_{optimal}-MVP \tag{3}$$

从式(3)MVD 的计算式可得出,MVD 不仅与已编码的块(左边的 A 块,上边的 B 块,右上方的 D 块)具有很高的相关性,还跟当前编码 C 块具有很高的相关性。

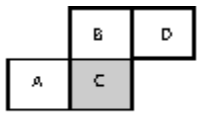


图 2　运动矢量预测模板

2.2　MV 与 MVD 的相互关系

当前编码块的运动矢量差 MVD 可分解成水平方向分量 mvd-x 和垂直方向分量 mvd-y。通过对帧间编码帧中运动矢量 MV 的绝对值和运动矢量差 MVD 的绝对值分布分析(图 3),可以得出|MV|和|MVD|具有近似的概率密度分布,并且其概率值主要集中在小值上,这也从统计角度体现了 MVD 和 MV 的相关性[13]。由于 MVD 和 MV 的相关性,再者,由于当前编码块 MVD 的熵值小于运动矢量 MV 的熵值[14],因此应用 MVD 进行熵编码要比应用 MV 进行熵编码更能节省码流。

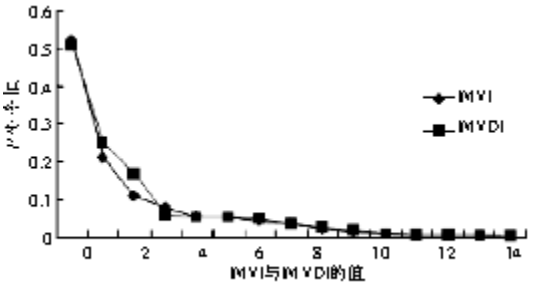


图 3　|MV|与|MVD|的概率分布

综合上述分析,CABAC 编码算法选择 MVD 而非 MV 作为主要的编码语法元素之一,主要体现在三个方面:

a)当前块的 MVD 的取值取决于当前块的 MV 和通过相邻块得到的 MVP,这就从线性的计算角度说明了当前块中 MVD 和 MV 的相关性。

b)在帧间编码帧中,运动矢量 MV 和运动矢量差 MVD 的

绝对值分布极为近似,进一步从统计的角度说明了 MVD 和 MV 的相关性。

c)当前块运动矢量差 MVD 的熵值小于运动矢量 MV 的熵值,编码时可节约码流。

2.3　基准 CABAC 算法中 MVD 各分量的上下文选择

作为 H.264/AVC 视频编码标准中 CABAC 编码的主要语法元素之一,运动矢量差 MVD 中水平分量 mvd-x 和垂直分量 mvd-y 的上下文选择需要精心设计,且各分量的上下文模型选择是由该分量相邻的左边和上边的已编码块(图 2 中的 A 块和 B 块)来决定的。首先,需对 MVD 的水平分量 mvd-x 和垂直分量 mvd-y 各自进行 UEGK(9,3)二进制化;其次,为 MVD 的各分量来的比特流在 7 个上下文模型中进行合适的模型选择。其中 mvd-x 的上下文模型是 CtxIdx = 40 ~ 46,mvd-y 的上下文模型是 CtxIdx = 47 ~ 53。对于 MVD 水平和垂直分量第一个比特位进行上下文选择时需利用以下方法:

$$\mathrm{mvd_inc}(C,\mathrm{cmp}) = \begin{cases} 0 & \mathrm{mv_err}(C,\mathrm{cmp}) < 3 \\ 1 & 3 \leqslant \mathrm{mv_err}(C,\mathrm{cmp}) \leqslant 32 \\ 2 & \mathrm{mv_err}(C,\mathrm{cmp}) \geqslant 32 \end{cases} \tag{4}$$

$$\mathrm{mv_err}(C,\mathrm{cmp}) = |\mathrm{mvd}(A,\mathrm{cmp})| + |\mathrm{mvd}(B,\mathrm{cmp})| \tag{5}$$

在此设置 mvd(K,cmp)代表 K 块中 MVD,其中 cmp ∈ {x, y},且 x、y 分别代表水平和垂直方向。因此,如图 2 中的当前块 C 的上下文索引增量 mvd_inc(C,cmp)可由式(4)(5)计算得出。式中,mv_err(C,cmp)代表依据当前块 C 的相邻块得到的上下文索引增量参考值。

综上分析可知,在基准 CABAC 算法中,在对运动矢量差 MVD 的各分量进行上下文模型选择时,仅仅结合了相邻已编码块的 MVD 相关性,忽略了当前块自身 MV 各分量的相关性,这必定给 MVD 各分量比特流准确的上下文模型选择带来不确定性,进而影响比特流的概率更新和二进制算术编码速率。因此,对于帧间预测编码块的 CABAC 编码,需改进更准确的算法对 MVD 各分量进行上下文模型选择,以达到可进一步节约编码码流的目的。

3　基于 MVD 的 CABAC 优化算法

3.1　块分割类型与运动矢量差 MVD 的关系

从理论上讲,在帧间编码中,对于 8 × 8 块,或者更小的8 × 4、4 × 8 及 4 × 4 块(下文称做小分割块),由于块与邻近的块通常属于相同的运动物体,通过相邻块来预测当前块的运动矢量,可以得到与当前块的近似运动矢量[14],根据式(3)得到的运动矢量差 MVD 较小;但对于 16 × 8、8 × 16 及 16 × 16 的块(下文称做大分割块)来说,由于宏块内不同的块之间可能分属于不同的物体,那么就需要考虑用不同方法处理块之间运动矢量的水平分量和垂直分量,根据式(3)得到的运动矢量差 MVD 较大。

因此,如果当前编码块是小分割块,由于 MVD 各分量与已编码相邻块所对应的分量相关性高,所以其各分量比特流的上下文模型选择可忽略当前块的运动矢量 MV;同理,如果当前编码块是大分割块,其运动矢量差 MVD 各分量与已编码相邻块所对应的分量相关性低,其上下文模型的选择不可忽略当前块运动矢量 MV。

3.2　块分割尺寸与编码比特的关系

通过文献[15]分析,并结合 H.264Visa 分析器对同一视频序列的帧间编码帧(P 帧和 B 帧)统计可知[15]:a)对于 P 帧,小分割块占据着整个编码 P 帧块分割总数的 29.49% ~ 33.67%,运动矢量差 MVD 的比特数占据着整个编码帧比特总数的 26.79% ~ 36.58%,整数 DCT 变换系数的比特数占据着整个编码帧比特总数的 52.44% ~ 61.53%;b)对于 B 帧,其小分割块占据着整个编码 B 帧块分割总数的 1.21% ~ 2.67%,运动矢量差 MVD 的比特数占据着整个编码帧比特总数的 64.44% ~ 72.58%,整数 DCT 变换系数的比特数占据着整个编码帧比特总数的 10.32% ~ 13.73%;c)对于无运动矢量差 MVD 编码的帧内预测编码帧(Ⅰ帧),其整数 DCT 变换系数的比特数占据着整个编码帧比特总数的 90% 以上。

从理论上分析可知:a)对于相同的分割块尺寸,单向预测的 P 帧较双向预测的 B 帧更不准确,所以需要更多的比特量对残差系数进行编码;b)P 帧的小分割块占据了分割总数的 30% 左右,而 B 帧中小分割块占据的比例仅在 1.5% 左右,所以 P 帧中小分割块的编码比特数远远多于 B 帧中小分割块的编码比特量。通过 H.264Visa 分析器对同一视频序列相邻的 B 帧和 P 帧中各分割块的编码比特量统计(图 4)可知:P 帧内各分割块的编码比特量远比 B 帧多,且主要集中在小块分割块中;而 B 帧中小分割块的编码比特量较少。

综合上述分析可知:除去少量的标志位外,整数 DCT 变换系数和运动矢量差 MVD 是 CABAC 编码的主要部分:I 帧中进行整数 DCT 变换系数的编码达 90% 以上;P 帧作为单向的帧间预测编码帧主要是进行整数 DCT 变换系数和运动矢量差 MVD 的编码;B 帧作为双向的帧间预测编码帧,较 P 帧有更高的运动估计精度,所以其仅有极小部分的残差系数,进而主要进行运动矢量差 MVD 的编码,但 P 帧中 MVD 的编码比特量要远多于 B 帧中 MVD 的编码比特量。

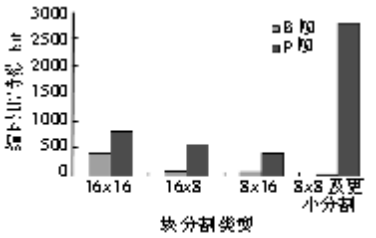


图 4　P 帧与 B 帧中各块分割的编码比特数对比

3.3　块分割类型下基于 MVD 的 CABAC 优化算法

由上述对 2.1 和 2.2 节的分析可知:当前块 MVD 与 MV 的极高相关性分别体现在当前块 MVD 的计算式以及当前块的|MVD|和 |MV| 近似的概率分布的统计。本文 2.3 节说明了在基准 CABAC 算法中,当前块 MVD 各分量的上下文模型选择仅考虑了当前块 MVD 与已编码相邻块 MVD 的相关性,而忽略了与当前块 MV 的相关性。本文 3.1 节从理论角度分析了在分割块尺寸大小中,小分割块下可忽略当前块的运动矢量 MV,而大分割块是不可忽略当前块的运动矢量 MV。本文 3.2 节分析得出了:帧间编码帧中,P 帧中对 MVD 编码的比特数占据着整个编码帧比特总量的 30% 左右,并且其编码的比特总量远远比 B 帧多;B 帧中对 MVD 编码的比特数占据着整个编码帧比特总量的 70% 左右。因此,可以在分割块尺寸大

小模式下,充分地利用当前块 MVD 与当前块 MV 的相关性、当前块 MVD 与已编码相邻块 MVD 的相关性,然后对当前块 MVD 各分量进行准确的上下文模型选择,进而提高对比特流进行概率更新的准确性,以有效地节约编码码流和提高二进制算术编码速率。

在文献[16]中,通过实验证明了当前块 MVD 的垂直分量的绝对值|mvd-y|与两倍的水平分量绝对值 2×|mvd-x|具有相同的概率分布,这就说明当前块 MVD 中各分量之间具有一定的相关性。文献[7]利用了文献[16]中的这个结论,并结合基准 CABAC 算法中 MVD 各分量的上下文模型选择算法,提出了利用式(6)来替代式(5)中当前块 MVD 的垂直分量的上下文索引增量参考值的新算法,并通过实验证明了其有效性(其中式(6)充分利用式(5)中各分量系数关系、|mvd-y|与2×|mvd-x|的系数关系,并选择对权值系数对半来均衡各分量之间的相关性)。

$$\mathrm{mv_err}(C,y)=\frac{|\mathrm{mvd}(A,y)|}{2}+\frac{|\mathrm{mvd}(B,y)|}{2}+|\mathrm{mvd}(C,x)| \quad (6)$$

通过上述分析可知,文献[16]算法只考虑了当前宏块 MVD 中垂直分量 mvd-y 与水平分量 mvd-x 的相关性,可以作进一步改进。因此,本文提出一种在分割块尺寸大小模式下,充分地利用当前块 MVD 与当前块 MV 的相关性、当前块 MVD 与已编码相邻块 MVD 的相关性及当前块 MVD 中各分量之间的相关性的新算法(命名为 CABAC_1 算法),以进一步提高 MVD 各分量上下文模型选择的准确性,达到节约编码码流和提高编码效率的目的。在 CABAC_1 算法中,各式子权值系数同式(6)一样,选取对权值系数对半来均衡各分量之间的相关性。其具体的算法步骤如下:

a)首先判断当前块 C 是否为小分割块(8×8,8×4,4×8 及 4×4 的块)。

b)若是小分割块,则无须考虑当前块 C 的 MVD 与当前块 C 的 MV 的相关性,并用式(6)(7)分别替代式(5)中当前块 MVD 的垂直和水平分量的上下文索引增量参考值。

$$\mathrm{mv_err}(C,x)=|\mathrm{mvd}(A,x)|+|\mathrm{mvd}(B,x)| \quad (7)$$

c)若不是小分割块,则需考虑当前块 C 的运动矢量差 MVD 与当前块 C 的 MV 的相关性,那么用式(8)(9)分别替代式(5)中当前块 MVD 的垂直和水平分量的上下文索引增量参考值。

$$\mathrm{mv_err}(C,y)=\frac{|\mathrm{mvd}(A,y)|}{4}+\frac{|\mathrm{mvd}(B,y)|}{4}+\frac{|\mathrm{mvd}(C,x)|}{2}+\frac{|\mathrm{mv}(C,y)|}{2} \quad (8)$$

$$\mathrm{mv_err}(C,x)=\frac{|\mathrm{mvd}(A,x)|}{2}+\frac{|\mathrm{mvd}(B,x)|}{2}+\frac{|\mathrm{mv}(C,x)|}{2} \quad (9)$$

d)通过步骤 b)和 c)得出的各分量的上下文索引增量参考值在代入式(4)中,求得上下文索引增量 mvd_inc(C,cmp)(cmp ∈{x,y},且 x、y 分别代表水平和垂直方向),然后选择合适的上下文模型。

e)选择合适的上下文模型后,对比特流进行概率的估计更新,最后进行二进制的算术编码。

4　实验仿真与分析

本实验以联合视频编码组 JVT 提供的 H.264/AVC 视频压缩标准的参考测试代码 JM9.5 作为测试工具,并选取不同的测试序列对本文提出的基于运动矢量差的 CABAC 新算法进行验证。表 1 列出了实验测试的环境。

表 1　实验环境

系统平台	奔腾 4 3.20 GHz CPU;1 GB 内存;Windows XP(SP2)
仿真平台	VS2008
标准参考代码	JM9.5(C)编码器
encoder_main.cfg 文件配置	量化参数 QP:I 帧 28,P 帧 28,B 帧 30 搜索范围:全搜索 编码方法:CABAC 序列类型:IBPBP… 其他默认
测试序列	slient:qcif(100 帧),176×144 akiyo:qcif(100 帧),176×144 foreman:qcif(100 帧),176×144 container:qcif(100 帧),176×144 carphone: qcif(100 帧),176×144

本文测试中,将 I 帧、P 帧和 B 帧的量化步长 QP 分别设置为 28、28 和 30。通过实验测试的基准 CABAC 编码算法和本文提出的 CABAC_1 编码算法数据如表 2 和 3 所示,表 4 是由表 2 和 3 中的实验数据得出的实验分析数据。综合各数据及对图 5 的解码帧图像分析可知:

a)在图 5(以 carphone 序列为例)中,(a)与(b)具有相同的主观视觉效果;并且对于相同的视频测试序列,基准 CABAC 算法测试得到的视频各分量(Y,U,V)的信噪比与通过 CABAC_1算法测试得到的基本一致,这就说明 CABAC_1 算法没有增加编码过程中的噪声,能够有效地确保编码过程中各个图像帧的视觉质量。

表 2　基准 CABAC 算法的实验测试结果

序列名称	帧类型比特数/Byte	比特率/kbps	总编码时间/s	各分量(Y,U,V)信噪比/dB
foreman	I 帧:21016 P 帧 11832 B 帧:1816	208.99	10.751	Y:36.73 U:40.96 V:42.83
silent	I 帧:26248 P 帧:984 B 帧:197	165.55	13.308	Y:36.38 U:39.27 V:40.28
akiyo	I 帧:18544 P 帧:664 B 帧:208	117.50	9.141	Y:38.98 U:41.09 V:42.28
container	I 帧:24176 P 帧:1224 B 帧:192	154.56	9.345	Y:37.17 U:41.47 V:40.85
carphone	I 帧:21928 P 帧:9520 B 帧:2672	205.73	9.468	Y:37.20 U:40.88 V:41.88

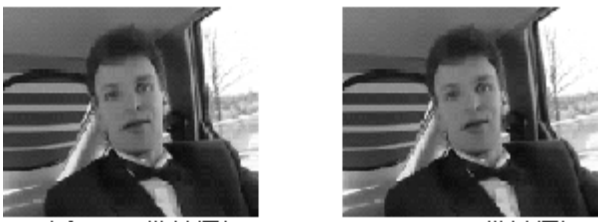
表 3　CABAC_1 算法的实验测试结果

序列名称	帧类型比特数/Byte	比特率/kbps	总编码时间/s	各分量(Y,U,V)信噪比/dB
foreman	I 帧:20996 P 帧 10840 B 帧:1753	204.32	10.541	Y:36.73 U:40.98 V:42.80
silent	I 帧:26209 P 帧:840 B 帧:182	158.16	12.969	Y:36.47 U:39.19 V:40.31
akiyo	I 帧:18490 P 帧:616 B 帧:198	113.59	8.965	Y:39.02 U:40.99 V:42.32
container	I 帧:24172 P 帧:1106 B 帧:182	148.27	9.138	Y:37.12 U:41.30 V:40.80
carphone	I 帧:21921 P 帧:8512 B 帧:2520	196.97	9.264	Y:37.19 U:40.72 V:41.67

表 4　基准 CABAC 与 CABAC_1 算法的实验测试结果对比

%

序列名称	帧类型比特数节省率	比特率节省率	总编码时间节省率
foreman	I 帧:0.092 P 帧:8.393 B 帧:3.475	2.234	1.953
silent	I 帧:0.144 P 帧:14.634 B 帧:7.614	4.464	2.546
akiyo	I 帧:0.291 P 帧:7.229 B 帧:4.818	3.328	2.009
container	I 帧:0.017 P 帧:9.641 B 帧:5.208	4.070	2.211
carphone	I 帧:0.031 P 帧:10.588 B 帧:5.689	4.257	2.149



(a) 基准 CABAC 算法解码帧　　(b) CABAC_1 算法解码帧

图 5　解码帧图像

b)对于作为帧内预测编码的 I 帧,表 4 的统计数据表明其编码比特数的节省量几乎为 0,主要是因为 CABAC_1 算法改进帧间编码帧(P 帧和 B 帧)中 MVD 各分量上下文模型选择的准确性,所以对 I 帧编码无影响。

c)表 4 的统计数据表明,CABAC_1 算法可使 P 帧的比特数节约量达到 10% 左右,B 帧的比特数节约量为 5% 左右。显然,CABAC_1 算法能有效地降低帧间编码帧的比特流。这主要体现在 CABAC_1 算法不仅考虑了当前块中 MVD 与相邻已编码块 MVD 之间的相关性和当前块 MVD 各分量之间的相关性,还充分地考虑了当前块 MVD 与当前块 MV 之间的相关性,使其有效地提高了对 MVD 中各比特流上下文模型选择的准确性,进而提高了帧间编码帧的编码效率。但是在同一序列中,较 P 帧而言,由于双向预测的 B 帧的 MVD 的编码比特数更少,其每个分割块的 MVD 预测更为准确且运动估计的时间较长,所以运用 CABAC_1 算法中各个相关性对 B 帧的 MVD 的比特位进行上下文模型选择的作用不如 P 帧,因而其提高的编码效率较 P 帧小。

d)表 4 的统计数据还表明,CABAC_1 算法对比特率和编码时间有了一定的提高:比特率提高了 4% 左右,编码时间提高了 2% 左右。这主要是由于 CABAC_1 算法比基准 CABAC 算法能更准确地对 MVD 各分量进行上下文模型选择,进而准确地对比特流进行概率更新,以有效地节约编码码流和提高二进制算术编码速率。

总之,CABAC_1 算法既确保了编码图像的视觉质量、编码比特率和时间,又节约了帧间编码帧的比特流。

5　结束语

本文在对 H.264/AVC 中 CABAC 算法原理及其运动矢量差 MVD 中各分量的相关性分析的基础上,在帧间编码分割块尺寸下,充分利用当前块 MVD 与当前块 MV 的相关性、当前块 MVD 与已编码相邻块 MVD 的相关性及当前块 MVD 中各分量之间的相关性,提出了一种新的基于运动矢量差的 CABAC 优化算法(CABAC_1 算法)。通过实验结果表明,CABAC_1 算法不影响已有的码率、编码时间和图像质量,成功实现了节约帧间编码帧的编码比特数。因此,CABAC_1 算法是一种有效的改进算法,并为节约视频序列的编码码流提供了一种新思路。

参考文献:

[1] RICHARDSON I E G. H.264 and MPEG-4 video compression video coding for next-generation multimedia [M]. England: Wiley,2003.

[2] 毕厚杰. 新一代视频压缩编码标准——H.264/AVC[M]. 2 版. 北京:人民邮电出版社,2009.

[3] MARPE D,SCHWARZ H,WIEGAND T. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard [J]. IEEE Trans on Circuits and Systems for Video Technology,2003,13(7):620-636.

[4] MARPE D,WIEGAND T,SULLIVAN G J. The H.264/MPEG4 advanced video coding standard and its applications [J]. IEEE Communications Magazine,2006,44(8):134-144.

[5] SCHWARZ H,MARPE D,WIEGAND T. Overview of the scalable video coding extension of the H. 264/AVC standard [J]. IEEE Trans on Circuits and Systems for Video Technology,2007,17(9):1103 -1120.

[6] 郭斌,王维东,叶青青,等. 基于概率估计更新的 CABAC 加速算法[J]. 中国图象图形学报,2009,14(2):281-285.

[7] 孙畅. H.264/AVC 标准中 CABAC 熵编码和率失真优化技术的研究与优化[D]. 济南:山东大学, 2008.

[8] 郭文豪. 运动矢量差的 CABAC 二进制化方法研究[D]. 西安:西安电子科技大学,2010.

[9] 孙书为, 陈书明. 高效的 H.264 并行编码算法[J]. 电子学报, 2009,37(2):357-361.

[10] HAHM J,KYUNG C M. Efficient CABAC rate estimation for H.264/AVC mode decision[CJ]. IEEE Trans on Circuits and Systems for Video Technology,2010,20(2):310-316.

[11] LEE J,KIM S,LEE S. Fast inter mode decision using motion vector-based moving window [J]. IEEE Trans on Consumer Electronics, 2010,56(3):1942-1949.

[12] KIM T J,HONG J E,SUH J W. A fast intra mode skip decision algorithm based on adaptive motion vector map [J]. IEEE Trans on Consumer Electronics,2009,55(1):179-184.

[13] CHIU M Y,SIU W C. Computationally-scalable motion estimation algorithm for H.264/AVC video coding[J]. IEEE Trans on Consumer Electronics,2010,56(2):895-903.

[14] LAROCHE G,JUNG J,PESQUET-POPESCU B. RD optimized coding for motion vector predictor selection [J]. IEEE Trans on Circuits and Systems for Video Technology,2008,18(9):1247-1257.

[15] MARPE D M,SCHWARZ H,BOSSE S. Video compression using nested quad-tree structures, leaf merging, and improved techniques for motion representation and entropy coding [J]. IEEE Trans on Circuits and Systems for Video Technology,2010,20(12):1676-1687.

[16] GHANDI M,GHANDI M M,SHAMSOLLAHI M B. A novel context modeling scheme for motion vectors context-based arithmetic coding [C]//Proc of the Canadian Conference on Electrical and Computer Engineering. 2004:2021-2024.