

# H.264 中基于分裂与合并的宏块模式快速选择算法

李晓彬<sup>1,2</sup>, 董绍武<sup>1,2</sup>

(1. 中国科学院国家授时中心, 临潼 710600; 2. 中国科学院研究生院, 北京 100049)

**摘要:** H.264 是目前最新的视频编码标准。同以往标准相比, 一个明显的改进就是使用了多种宏块分块模式从而使运动估计更为精确。但是计算量非常惊人。虽然 JVT 采用了一个快速运动估计(FME)的提案能在码率和效果基本无变化的情况下极大地提高搜索速度, 但计算量仍很大。该文提出了一个对基于分裂与合并的宏块模式快速选择算法, 在  $8 \times 8$  块搜索基础上先由运动向量剧烈程度决定合并或分裂, 而对中间的运动向量推导出一个阈值来决定是否合并。实验证明该算法能在码率和效果基本不受影响的情况下极大地提高编码的速度。

**关键词:** H.264; 宏块模式; 分裂; 合并

## Fast Variable-size Block Select Algorithm Based on Merge and Split Procedures for H.264

LI Xiaobin<sup>1,2</sup>, DONG Shaowu<sup>1,2</sup>

(1. National Time Service Center, Chinese Academy of Sciences, Lintong 710600;

2. Graduate School, Chinese Academy of Sciences, Beijing 100049)

**【Abstract】** H.264 is the latest under development video coding standard. It uses variable-size block to improve the motion estimation accuracy of inter-coding blocks. But it is extremely computational intensive. Though JVT adopts a proposal about fast motion estimation(FME) which shows very good capability in keeping the rate distortion performance and a great computation reduction can be achieved, still computes complex. This paper proposes and investigates fast variable block-size motion estimation algorithms based on merge and split procedures for H.264 video encoding. Experiments results show that the proposed method can save about 24.7% to 29.7% computations while keeping similar picture quality.

**【Key words】** H.264; Variable-size block; Merge; Split

### 1 概述

H.264/AVC 是目前最新的视频编码的国际标准。它是由 ISO/IEC 运动图像专家组(MPEG)和 ITU-T 视频编码专家组(VCEG)共同组成的联合视频小组所完成的。同以往的标准相比, 同等的视频质量它只需要 H.263 一半的码流, MPEG-2 的 1/4 的码流。因此, 这个新标准不仅能带来更加清晰的画面质量, 而且为不同带宽的视频应用提供了一个好的解决方案。多参考帧、多种宏块分块模式、率失真优化(RDO)、循环滤波器和基于上下文的自适应算术编码(CABAC)被引入 H.264 用于视频编码是其能有如此优异性能的原因。相比于 H.263 的  $16 \times 16$  和  $8 \times 8$  两种宏块模式, 每个  $16 \times 16$  宏块能被分解成 1 个  $16 \times 16$  块、2 个  $8 \times 16$  块、2 个  $16 \times 8$  块、4 个  $8 \times 8$  块。如果  $8 \times 8$  块被选择了, 还可以进而分解成 4 个  $4 \times 4$ 、2 个  $8 \times 4$  和 2 个  $4 \times 8$  小块。由于宏块模式的增多, 率失真优化(RDO)取代以往的 SAD 值而得到较低的码率和更好的画面质量之间的平衡。在率失真优化过程中, 编码器将遍历多个参考帧的情况下还需依次测试不同宏块模式的组合, 从而找出具有最小 RDO 花费的组合。但此过程中的计算量非常惊人, 据统计如果 7 种宏块模式全开运动估计的时间约占整个编码时间的 60%, 所以加快宏块模式的选择对 H.264 的实用化有非常重要的意义。

虽然 JVT 采用了一个快速运动估计(FME)的提案能在码率和效果基本无变化的情况下极大的提高搜索速度, 但它主要加速了一个参考帧上的搜索, 整体的计算量仍很大。目前

有不少快速方法被提出来加速 H.264 编码过程中的运动搜索。目前主要分为 2 类: 一种思路是向上合并法, 文献[1,2]在把整个帧分成  $4 \times 4$  块, 在此搜索结果基础块被向上合并成大块, 但由于一般情况下帧内只有很少部分运动剧烈而大部分  $4 \times 4$  块需要合并, 每次合并还需重新搜索并计算 RDO 花费, 计算量仍很大; 另外一个思路是向下分裂法, 在文献[3]中它根据块内信息来确定块内运动的均匀性, 如果不均匀的话, 将从  $16 \times 16$  块向下分裂直到块满足均匀的条件。但这种方法会产生“大部分效应”, 即如果一个大块中虽然有小的运动剧烈块但整个块的 RDO 花费在允许范围内的话, 将不会往下分裂。为避免这两种思路的缺陷, 本文提出了一种结合了分裂与合并的宏块模式快速选择方法, 先按  $8 \times 8$  块对整个帧搜索, 在此搜索基础上对块间运动向量 (MV)相似和变化剧烈的分别进行合并或分裂, 而对介于二者之间的情况本文推导出一个阈值来决定是否合并。实验证明, 本算法能在码率和效果基本不受影响的情况下很大的提高运动搜索的速度, 达到提高编码速度的目的。

### 2 H.264 帧间宏块模式选择

在 H.264 帧间编码中有 7 种不同的宏块分块模式。如图 1 所示, 目前在 JM92 中依次遍历每个宏块模式并计算它们 RDO 花费, 最佳分块模式将选择其中 RDO 花费最小的分块

**作者简介:** 李晓彬(1981—), 女, 硕士生, 主研方向: 时间频率测量和控制方法及应用; 董绍武, 研究员、在读博士生

**收稿日期:** 2005-11-17 **E-mail:** lixiaobin@mails.gscas.ac.cn

模式。由图 1 中可见编码器的计算量非常大。

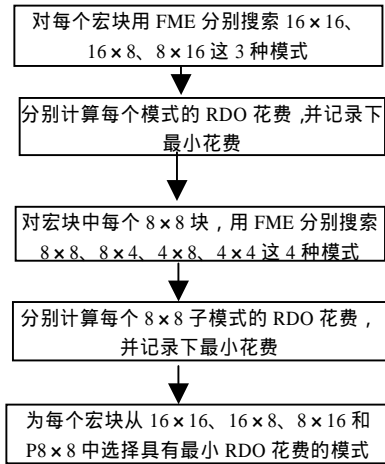


图 1 H.264 最佳宏块模式选择

### 3 快速宏块模式选择

#### 3.1 分析

在概述中介绍了已有的向上合并法和向下分裂法，这两类有个共同毛病就是初始搜索的块的大小不合适，很多块需反复计算才能得到最佳宏块模式。本文提出的快速宏块模式选择算法采用  $8 \times 8$  块作为初始搜索，因此只要根据一定的条件判断很容易能合并到  $16 \times 16$  的大块或分裂到  $4 \times 4$  的小块，无须反复计算 RDO 花费。此外从图 2 中可以看到， $8 \times 8$  块是固定模式中表现最好的分块方式。

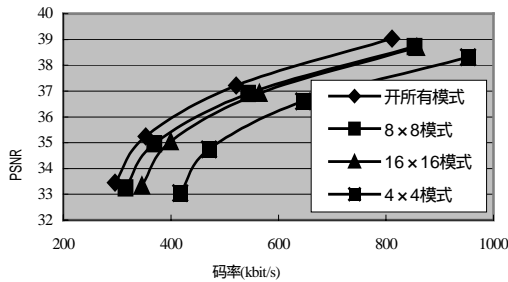


图 2 H.264 表现(Foreman.cif)

首先分析  $8 \times 8$  块的分裂。如图 3 所示，将  $8 \times 8$  分裂为  $8 \times 4$  或  $4 \times 8$ ，进而分裂为  $4 \times 4$  块。在这里使用不同块间的运动向量(MV)差进行判断。公式如下：

$$\begin{cases} |MV_{i,x} - MV_{j,x}| > d_{split} \\ OR \\ |MV_{i,y} - MV_{j,y}| > d_{split} \end{cases} \Rightarrow Split(i, j \in A, B, C, D) \quad (1)$$

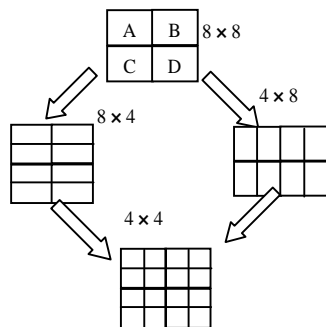


图 3 从  $8 \times 8$  分裂到  $8 \times 4$ 、 $4 \times 8$  和  $4 \times 4$

其中  $MV_{i,x}$  和  $MV_{i,y}$  是第  $i$  个块的水平和垂直方向运动向量， $d_{split}$  是阈值。如果两个块之间的运动向量差大于此阈值说明

块与块之间运动比较剧烈，需要尝试用小的块搜索。在仿真实验中，使用  $1/4$  高精度像素搜索，经实验测定  $d_{split}$  为 6 比较合适。

接着将讨论  $8 \times 8$  的合并。这里定义如下：

$$\begin{cases} |MV_{i,x} - MV_{j,x}| < d_{merge} \\ AND \\ |MV_{i,y} - MV_{j,y}| > d_{merge} \end{cases} \Rightarrow Merge(i, j \in A, B, C, D) \quad (2)$$

在实验中使用  $1/4$  高精度像素搜索，定义  $d_{merge}$  为 2，即两个块之间的运动向量最多只相差半个像素。对合并后的块需要重新进行运动搜索，其初始搜索点是二者和的均值。如果 AB 与 AC 都能合并，由对自然图像序列一个公认的结论，水平方向上的运动要比垂直方向上的运动剧烈的多，则优先合并 AB 两个块。

对于运动向量之差介于  $d_{split}$  和  $d_{merge}$  的块，很难光从运动向量上判断其是否需要合并或者直接使用  $8 \times 8$  块即可。因此，我们提出了一个阈值判断法来决定是否合并。其基本思想如下：在运动估计后 H.264 将对残差使用  $4 \times 4$  的整数变换，对于  $4 \times 4$  的块，量化后的 DC 值为

$$DC = \sum_{i=0}^3 \sum_{j=0}^3 f(x, y) / ((2^{qbits} - f) / QE[q_{rem}][0][0]) \quad (3)$$

可以假设，如果变换系数的 DC 值为零，则其它的 AC 系数也为零，即当前块是零块。因此如果条件(4)成立，量化后的 DC 系数将为零。

$$\sum_{i=0}^3 \sum_{j=0}^3 f(x, y) < ((2^{qbits} - f) / QE[q_{rem}][0][0]) \quad (4)$$

其中， $qbits = 15 + QP/6$ ， $q_{rem} = QP \% 6$ ， $f = (1 \ll qbits) / 6$ ，QE 为量化系数表 QP 为输入的量化参数。而  $\sum_{i=0}^3 \sum_{j=0}^3 |f(x, y)|$  为  $4 \times 4$  块的差值的绝对值之和(SAD)，因此  $4 \times 4$  块的零块的 SAD 阈值可以定义为

$$SAD_{4 \times 4} = (2^{qbits} - f) / QE[q_{rem}][0][0] \quad (5)$$

对于两个  $8 \times 8$  块的合并，其中一共有 8 个  $4 \times 4$ ，在得到  $4 \times 4$  零块的 SAD 阈值后如果两个块的 SAD 之和小于  $8 * SAD_{4 \times 4}$ ，则说明  $8 \times 16$  或  $16 \times 8$  残差非常平滑，可以忽略不计，在实际合并中，还需乘上一个比例因子  $\beta$  (仿真实验中定为 1.05)，从而使虽然不属于零块但仍很平滑的块合并。故 A、B 合并的条件如下：

$$(SAD_A + SAD_B) < 8 * \beta * (2^{qbits} - f) / QE[q_{rem}][0][0] \quad (6)$$

#### 3.2 提出的算法

根据上面的观察和分析，提出了以  $8 \times 8$  块搜索结果为基础的分裂与合并算法，具体如下：

(1) 使用 JM92 提供的快速搜索算法 FME 搜索每个  $8 \times 8$  块，搜索精度为  $1/4$  像素。

1) 如果  $|MVA_x - MVB_x| < 2$  且  $|MVA_y - MVB_y| < 2$  (由于  $1/4$  像素搜索，2 即代表半个像素)，合并两个块返回二者和的均值前往步骤(2)。

2) 如果  $|MVA_x - MVB_x| > 6$  或  $|MVA_y - MVB_y| > 6$  (即代表一个半个像素)，分裂两个块成  $8 \times 4$  或  $4 \times 8$  前往步骤(3)。

3) 如果介于二者之间，使用式(6)，如果小于阈值则前往步骤(2)，否则保持  $8 \times 8$  块。

(2) 对合并后的块进行运动估计，用步骤(1)中 1)，3) 条件判断是否需要合并到  $16 \times 16$  块，3) 使用的式(6)中 8 应为 16 (共 16 个  $4 \times 4$  块)。

(3) 对分裂后的块进行运动估计，用步骤(1)中 2) 条件判断是否需要分裂到  $4 \times 4$  块。

### 4 实验与结果

实验中,采用运动搜索所耗时间、峰值信噪比(PSNR)和码率(bit-rate)这 3 个测度来验证本算法的性能,并用 news.qcif、Mother\_daughter.qcif 和 silent.qcif 这 3 个 QCIF 标准序列(177×144)和 foreman、paris、bus、football、mobile 这 5 个 CIF 标准序列(352×288),采用 16×16、16×8、8×16、8×8、8×4、4×8、4×4 这 7 种宏块分块模式,搜索范围为 33×33,即±16 个像素点,使用 P、B 帧,参考帧数目为 5 帧,搜索方式采用里面的快速搜索(FME)而不是全搜索,其它参数设置使用其 main 档次默认设置。

仿真实验在 P4 2.8G 512MB 内存的 PC 机,用 Visual C++ 6.0 在 JM92 上使用标准 C 编程实现,未使用 MMX 优化。为了测试时间的准确性,分别测试以上 8 个序列的 150 帧。将本算法和 JM 中原有的用 FME 依次遍历宏块模式,表 1、表 2 和表 3 分别给出了这 8 个序列 150 帧的平均 PSNR、码率和总的运动搜索时间,每个表的最后一列是本算法同 FME 算法比较的结果,其中:-表示减少而+表示增加。

从表 1 中可以看到,此算法的 PSNR 却只降低了 0.09 到 0.14,而从表 2 中可以看到码率也只是略上升了 1.2%到 2.1%。由于运动估计占编码的很大比例,从表 3 种可以看到整个编码时间减少了 24.7%到 29.7%,编码效率得到了极大地提高。

表 1 150 帧 PSNR (dB)

	FFS	Proposed	Results
news	37.71	37.62	- 0.09
Mother_daughter	38.25	38.17	- 0.08
silent	36.54	36.47	- 0.07
foreman	37.21	37.08	- 0.13
paris	36.18	36.07	- 0.11
bus	34.76	34.62	- 0.14
football	36.18	36.14	- 0.14
mobile	33.90	33.78	- 0.12

表 2 150 帧码率(kbit/s)

	FFS	Proposed	Results
news	174.28	176.78	+1.4%
Mother_daughter	104.91	106.93	+1.9%
silent	177.17	179.38	+1.2%
foreman	521.32	532.11	+2.1%
paris	970.09	987.34	+1.4%
bus	1 295.22	1 344.40	+1.8%
football	1 222.52	1 244.40	+1.8%
mobile	1 870.01	1 895.17	+1.3%

表 3 150 帧时间(s)

	FFS	Proposed	Results
news	249.541	187.997	- 24.7%
Mother_daughter	243.734	178.958	- 26.6%
silent	257.903	190.706	- 26.1%
foreman	1099.57	800.041	- 27.2%
paris	1092.90	823.073	- 24.7%
bus	1302.37	918.324	- 29.5%
football	1276.99	898.036	- 29.7%
mobile	1312.41	962.158	- 26.7%

以上实验结果证明了本算法能在压缩质量基本无变化的情况下极大的加快运动搜索。

## 5 结论

H.264 能提供比 H.263 更好的视频质量和更低的码率,但它花费了很大的编码时间。本文提出了一种结合了分裂与合并的宏块模式快速选择方法。实验证明了本算法能在压缩质量基本无变化的情况下极大的加快运动搜索。

## 参考文献

- 1 Wu D, Wu S, Lim K P, et al. Block Inter Mode Decision for Fast Encoding of H.264[C]. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004: III181-III184.
- 2 Rhee I, Martin G R, Muthukrishnan S, et al. Quadtree-structured Variable-size Block Matching Motion Estimation with Minimal Error[J]. IEEE Trans. Circuits Syst. Video Technol., 2000, 10(1): 42-50.
- 3 Tu Y K, Yang J F, Shen Y N, et al. Fast Variable-size Block Motion Estimation Using Merging Procedure with an Adaptive Threshold[C]. IEEE International Conference on Multimedia and Expo., 2003: II789-II792.
- 4 Wiegand T, Sullivan G, Bjontegaard G, et al. Overview of the H.264 /AVC Video Coding Standard[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(7): 560-576.
- 5 Joint Video Team Software JM 92[EB/OL]. <http://bs.hhi.de/~suehring/tml/download/>.
- 6 Richardson I. H.264 and MPEG-4 Video Compression[M]. Wiley, 2003.

(上接第 235 页)

## 5 结语

基于 IP 网络的多媒体信息传输系统,用纯软件方法实现了以视频信息为主的多媒体信息在 IP 网络中的实时传输,给出了图像与数据的采集、处理、传输及设备的远程控制的完整解决方案。将视频和各类数据信息混合压缩处理,打包实时传输,协议透明,维护简单,实践证明是行之有效的。

## 参考文献

- 1 Kuo F, Effelsberg W. 龙晓苒译. 多媒体通信协议与实现[M]. 北京: 清华大学出版社, 2000.
- 2 毕厚杰. 多媒体信息的传输与处理[M]. 北京: 人民邮电出版社, 1999.
- 3 王汇源. 数字图像通信原理与技术[M]. 北京: 国防工业出版社,

2000.

- 4 Charles P. Windows95 程序设计[M]. 北京: 清华大学出版社, 1997.
- 5 ITU-T Recommendation H.263. Video Coding for Low Bit Rate Communication[S]. 1996-03.
- 6 ITU-T Recommendation H.263. Video Coding for Low Bit Rate Communication[S]. 1998-01.
- 7 ITU-T Study Group 16. Video Codec Test Model Near Term (Version8) [S]. Q15-B-46, 1997.
- 8 ITU-T Study Group 16. H.26L Test Model Long Term Number 5 Draft0[S]. Q15-K-59, 2000.
- 9 骆立俊, 邹采荣, 高西奇等. 视频编码中的块运动估计算法[J]. 电视技术, 1997, (12): 2-8.