

# H. 264 中带有中途停止的多层逐次 消元运动估计算法

薛全 刘济林 陈建乐

(浙江大学信息与通信工程研究所, 杭州 310027)

**摘要** H. 264 视频压缩标准采用多模式运动估计, 可以有效减少块匹配预测误差, 但随着模式选择的增多, 算法计算量成倍增加。为此, 提出一种带有中途停止的多层逐次消元运动估计算法(MSEHS)。该算法根据模式分布规律, 使用多层逐次消元法加速大块模式的搜索过程, 并且提出中途停止准则, 判断是否继续进行小块模式的搜索。这样既保证了多模式运动估计的优点, 又减少了冗余计算。实验结果表明, 该算法比全搜索算法整体速度提高了近4倍, 同时可以保持与全搜索算法非常接近的图像质量和比特率。

**关键词** 视频压缩 H. 264 运动估计 块匹配 中途停止

中图分类号: TP391.41 文献标识码: A 文章编号: 1006-8961(2004)06-0712-05

## Multilevel Successive Elimination with Half-Stop Motion Estimation Algorithm in H. 264

XUE Quan, LIU Ji-lin, CHEN Jian-le

(Institute of Information and Communication Engineering, Zhejiang University, Hangzhou 310027)

**Abstract** H. 264 video coding standard uses motion estimation with multiple block sizes to reduce block-matching errors for higher rate-distortion performance. But the more modes are used, the more computation is needed linearly with the number of allowed block sizes. So a fast block motion estimation algorithm named multilevel successive elimination with half-stopping (MSEHS) is provided in this paper. According to the analysis of statistical characteristics about the best modes, multilevel successive elimination algorithm is described to speed up the searching process for the larger block modes firstly, then using the middle results SAD (sum of absolute difference) and motion vector, a half-stopping principle is defined as a threshold to judge whether it is necessary to continue the motion estimation for the smaller block modes. MSEHS algorithm not only reserves the advantage of multiple block modes in motion estimation but also deletes the redundant computation in the searching process. Experimental results show that the proposed technique provides competitive performance compared with full search algorithm, that is, highly reduced computational complexity, very similar image quality and bitrate.

**Keywords** video compression, H. 264, motion estimation, block matching, half-stopping

## 1 引言

基于块匹配的运动估计是视频压缩的关键技术之一。利用已编码的参考帧预测当前帧, 使当前帧内每个块都可用一个残差块和运动矢量表示, 可以有效消除视频序列中的时间相关性。显然, 运动估计越

精确, 残差块系数越小, 码率越低, 因此 ITU-T/ISO/IEC 联合提出的新一代视频压缩标准 H. 264<sup>[1]</sup>引入了多模式运动估计, 解决了以往单一模式无法准确描述同一块内多物体、多速度、多方向运动的困难, 并且还可以精确表示具有不同像素精度的纹理。

目前有很多基于单模式块匹配的运动估计算法, 全搜索算法<sup>[2]</sup>可以得到最精确的运动矢量, 但是

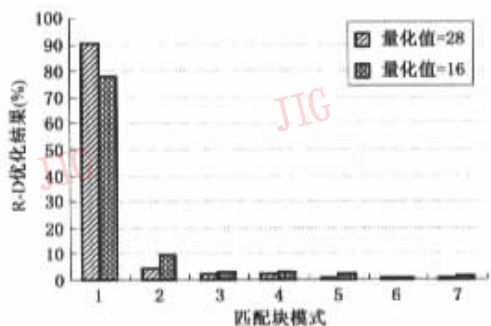
随着模式选择的增多,算法复杂度成倍增加,采用全部模式的运动估计占整个 H.264 编码器计算量的 80% 以上<sup>[3]</sup>。典型的快速算法主要有新三步搜索法<sup>[4]</sup>、钻石搜索法<sup>[5]</sup>等,一般是在搜索范围内沿着梯度下降的方向寻找最优匹配块,算法速度很快,但搜索误差较大,容易陷入局部极值,在提高速度的同时,导致了图像质量降低。因此快速高质量的多模式运动估计算法成为 H.264 实时编码器实现的关键。

## 2 多模式运动估计

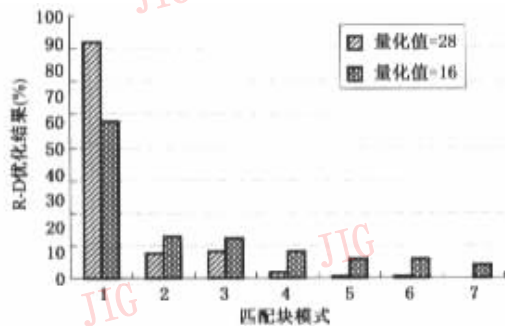
MPEG-4 和 H.263 等视频压缩标准在运动估计中使用  $16 \times 16$  或  $8 \times 8$  2 种块模式。为了提高运

动估计的精度,H.264 运动估计中定义了 7 种块模式( $16 \times 16$ 、 $16 \times 8$ 、 $8 \times 16$ 、 $8 \times 8$ 、 $8 \times 4$ 、 $4 \times 8$ 、 $4 \times 4$ ),其中称  $16 \times 16$ 、 $16 \times 8$ 、 $8 \times 16$ 、 $8 \times 8$  为大块模式, $8 \times 4$ 、 $4 \times 8$ 、 $4 \times 4$  为小块模式。对于一个  $16 \times 16$  的块,可以由 7 种不同大小和形状的块模式组成,分别有 1、2、4、8、16 个不同运动矢量。H.264 可以从模式 1 到模式 7 中分别进行运动估计,得到每个模式对应的运动矢量,然后从中选择最佳的块模式和相应的运动矢量。

针对不同运动特征的序列,在不同量化参数下,对运动估计最优块模式选择结果进行了统计。图 1 给出了 akiyo 和 stefan 两种序列的模式选择结果,其中标号 1 至 7 分别代表  $16 \times 16 \sim 4 \times 4$  7 种块模式。



(a) akiyo



(b) stefan

图 1 模式选择统计结果

可以看出块模式的最终选择和序列的运动特征有关:背景简单、低速运动的序列多采用较大的块模式;具有高速运动、复杂纹理的序列采用更多的是小块模式,且细节越丰富,小块模式越多;同时,随着量化值的减少,为得到更精确的运动矢量,提高图像质量,需要使用更多的小块模式。

## 3 带有中途停止的多层逐次消元算法

如果只选择对减少预测误差有益的块模式,跳过无用的块模式搜索过程,就可以既提高图像质量,保证多模式运动估计的特点,又降低算法复杂度。

### 3.1 逐次消元法<sup>[6]</sup>

利用当前帧和参考帧中像素的三角不等式,可以得到搜索过程中最优匹配块具备的必要条件,推导结论为

$$\|X\|_1 - \|Y(x, y)\|_1 \leq$$

$$\sum_{i=1}^N \sum_{j=1}^N |f(i, j, t) - f(i - x, j - y, t - 1)| \quad (1)$$

其中,

$$\|X\|_1 = \sum_{i=1}^N \sum_{j=1}^N f(i, j, t)$$

$$\|Y(x, y)\|_1 = \sum_{i=1}^N \sum_{j=1}^N f(i - x, j - y, t - 1)$$

式(1)右侧是当前块与参考块像素的绝对误差和(SAD),左侧为当前块像素之和与参考块像素之和差的绝对值,记 SAD\_SUM。如果一个块是最优匹配块,那么它需要满足上述条件,否则可以判断出该点不是最优点,直接淘汰。算法的基本步骤是:

(1)选择初始搜索点对应的 SAD 作为当前最小 SAD(curr\_min\_SAD)的初始值;

(2)遍历搜索窗口内的点,计算其 SAD\_SUM,与 curr\_min\_SAD 比较;

(3)如果 SAD\_SUM 比 curr\_min\_SAD 大,可以推断出该参考块与当前块的 SAD 一定大于 curr\_min\_SAD,因此这个点一定不是最优点,直接淘汰掉否则继续下一步;

(4)计算当前块和参考块的 SAD,如果比 curr\_

min\_SAD 小, 则计算该块的运动矢量, 并更新 curr\_min\_SAD。

逐次消元算法本质上是一种快速全搜索算法, 通过遍历整个搜索窗口, 运用最优匹配块具备的必要条件淘汰冗余点, 减少搜索过程中计算量最大的 SAD 计算, 有效地降低算法复杂性, 同时可以提供与全搜索算法相同的结果。

### 3.2 多层逐次消元算法

如图 2 所示, H. 264 多模式运动估计的结构为金字塔形状。因此将逐次消元法扩展到大块模式运动估计搜索过程中, 即多层逐次消元算法。以模式 1 (16×16) 为例进行推导, 结果同样适用于其他大块模式。

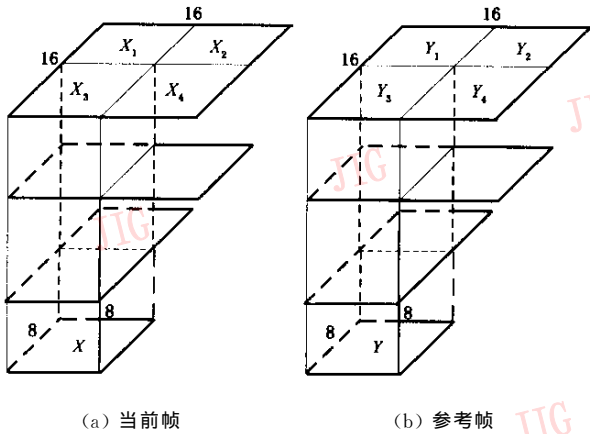


图 2 大块模式金字塔结构

一个 16×16 块由 4 个 8×8 块组成, 当前帧内块分别表示为  $X_1, X_2, X_3, X_4$ , 参考帧内块表示为  $Y_1, Y_2, Y_3, Y_4$ 。

利用参考块和当前块内像素之间的三角不等式可以得到如下结果:

$$\begin{aligned} \left| \|X_1\|_1 - \|Y_1\|_1 \right| &\leq \|X_1 - Y_1\|_1 \\ \left| \|X_2\|_1 - \|Y_2\|_1 \right| &\leq \|X_2 - Y_2\|_1 \\ \left| \|X_3\|_1 - \|Y_3\|_1 \right| &\leq \|X_3 - Y_3\|_1 \\ \left| \|X_4\|_1 - \|Y_4\|_1 \right| &\leq \|X_4 - Y_4\|_1 \end{aligned} \quad (2)$$

合并上式可以得到

$$\left| \sum_{k=1}^4 \|X_k\|_1 - \sum_{k=1}^4 \|Y_k\|_1 \right| \leq \sum_{k=1}^4 \|X_k - Y_k\|_1 \quad (3)$$

左侧为当前帧和参考帧内对应的各自 16×16 块的像素之和的差的绝对值 SAD\_SUM, 右侧为当前块和参考块内 4 个 8×8 块组成的像素之和的差的绝对值, 记为 SAD\_N (N=4)。可以得到最优匹配块满足的必要条件:

$$\text{SAD\_SUM} \leq \text{SAD\_N} \quad (4)$$

同时由式(1)又可以得到

$$\text{SAD\_N} \leq \text{SAD} \quad (5)$$

这说明通过将大块分割成若干小块的和, 得到的当前块与参考块像素和的 SAD\_N, 比直接计算 SAD\_SUM 更精细, 以 SAD\_N 作为阈值可以将搜索范围进一步缩小, 与直接在各个大块模式上利用逐次消元算法搜索运动矢量相比, 匹配的计算量更小。

使用多层逐次消元算法的另一个好处是, 在实际计算 16×16, 16×8 和 8×16 块模式的搜索匹配过程中, 可以利用 8×8 块中的 SAD\_SUM 直接求和, 得到其他大块模式的 SAD\_SUM。这样充分利用搜索过程的中间结果, 使计算当前块模式像素和与参考块模式像素和的计算量很少。

### 3.3 中途停止准则

模式分布统计结果表明: 16×16 块模式应用最多, 越小的块模式选择的比例越小, 因此进行大块模式的运动搜索后, 可以判断是否有向小块模式发展的趋势。若没有的话, 停止其他模式搜索, 直接跳到下一个块; 如果发现图像有运动细节和复杂纹理, 则需要继续进行其他模式的搜索。

中途停止判别函数定义如下:

$$\text{Cost}(i) = \text{SAD}(i) + (\omega \times MV_{\text{bits}}(i)) \gg \lambda \quad (6)$$

其中,  $\text{SAD}(i)$  为模式  $i$  ( $i=1, 2, 3, 4$ ) 得到的最小 SAD,  $MV_{\text{bits}}(i)$  为对应的最优运动矢量比特数,  $\omega$  为自适应权重因子,  $\lambda$  为常数 256。

判别准则: 如果判别函数具有单调递减性, 则继续搜索其余模式, 否则停止搜索。

$$\text{Cost}(4) \leq \min\{\text{Cost}(2), \text{Cost}(3)\} \leq \text{Cost}(1) \quad (7)$$

利用大块模式运动估计得到的中间结果, 应用中途停止准则进行小块模式选择, 既可以准确地描述运动和纹理, 又可以减少冗余的搜索计算。

### 3.4 算法描述

带有中途停止的多层逐次消元运动估计算法的主要步骤是:

(1) 初始搜索起点预测。

(2) 应用逐次消元算法, 计算初始搜索起点对应 8×8 块模式的最优运动矢量, 存储该块的 SAD\_SUM。

(3) 对于 16×16, 16×8, 8×16 3 个模式, 利用 8×8 模式计算中得到的结果, 通过直接求和得到其余大块模式的 SAD\_SUM, 应用多层逐次消元算法搜索相应模式内的最优运动矢量, 同时保存各自的 SAD 和运动矢量。

(4)应用中途停止准则判断是否继续进行小块模式搜索。如果满足条件,执行第 5 步,否则执行第 6 步。

(5)使用全搜索算法搜索  $8 \times 4, 4 \times 8, 4 \times 4$  3 个小块模式的最优运动矢量。

(6)判断  $16 \times 16$  块的最优组合形式,得到整体最优运动矢量。

### 4 实验结果与分析

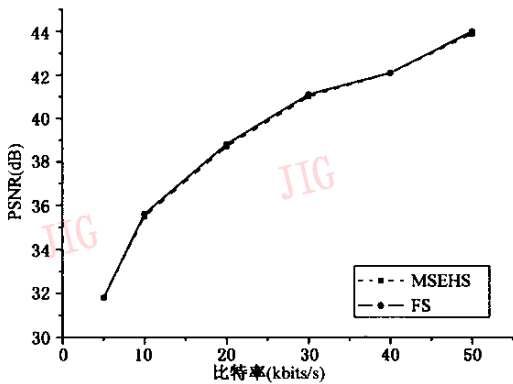
在 PC 机(Pentium IV 1.8G 主频,256M 内存)上,将本文算法实现于 H. 264 标准模型 JM61e<sup>[7]</sup>

中,仿真实验选择了具有不同空间细节和运动剧烈程度的 3 个 CIF 格式序列 akiyo, foreman, stefan ( $352 \times 288, 4:2:0, 300$  帧)。运动估计搜索范围为  $-15/+15$ , 编码帧率为 30Hz, 使用单参考帧, 每 15 帧做一次帧内编码, 编码帧格式为 IPPP。

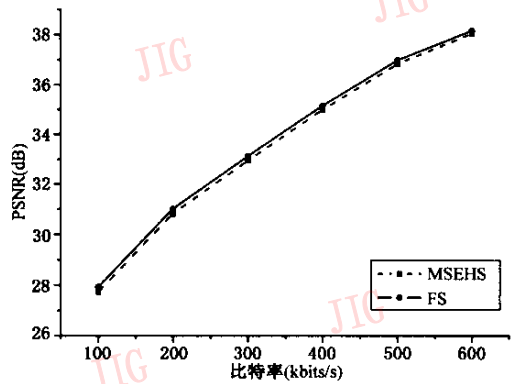
表 1 给出了 2 种算法的质量、比特率、平均搜索耗时 3 个方面的整体性能比较结果。图 3 给出了本文算法与全搜索算法在比特率失真上的比较, 图 4 以 foreman 序列为例, 进一步给出了原图像和解码后图像的主观质量对比结果。

表 1 图像质量、比特率、平均搜索时间对比

序列	算法	PSNR			比特率 (kbits/s)	平均耗时 (s/f)
		Y(dB)	U(dB)	V(dB)		
akiyo	FS	40.06	42.52	43.98	30.51	3.03
	MSEHS	39.99	42.13	43.97	31.02	0.85
foreman	FS	37.21	40.69	42.49	116.18	3.96
	MSEHS	37.11	40.65	42.43	119.89	1.12
stefan	FS	35.62	37.85	37.96	435.3	4.75
	MSEHS	35.44	37.76	37.87	439.7	1.32



(a) akiyo



(b) stefan

图 3 比特率失真比较



(a)原图像



(b)解码后图像

图 4 图像主观质量比较(foreman 第 17 帧)

计算机仿真结果表明,本文提出的算法有效地减少了块匹配的计算量,而且可以获得与全搜索算法相当的图像质量、信噪比和比特率。进一步的分析发现,对于运动较少的图像,本文算法得到的结果几乎和全搜索算法完全相同;对于运动较剧烈的图像,图像的编码质量会有所下降,PSNR 最多下降了 0.18dB,但原图像与解码图像的主观质量几乎相同。这是因为运动剧烈的图像细节较多,在中途停止判决的过程中有些小块模式没有被有效地检测到。当量化值减小,码率升高时,本文算法得到的重建图像质量和码率与全搜索算法越来越接近。同时,在相同图像质量下,使用本文算法得到的码率比全搜索算法略高,这是由于某些应该使用小块模式的运动估计使用了大块模式,残差块系数偏大。

## 5 结 论

针对 H.264 编码中多模式运动估计计算量过大的问题,提出了一种新的带有中途停止的多层逐次消元运动估计算法,综合运用了多层逐次消元法和中途停止准则等策略。实验结果表明,该方法可以有效地减少运动估计块匹配的计算量,而且可以获得与全搜索算法非常近似的图像质量和码率。由于多层逐次消元算法主要使用了加法和比较运算,减少了绝对值等耗时运算,并且结构简洁,适合 DSP 或 ASIC 等专用处理器并行计算。因此本文算法经过适当的修改后,可以在多种专用平台下应用,实现 H.264 编码器对 CIF 格式的实时编码。

## 参 考 文 献

- 1 ITU-T Rec. H. 264/ISO/IEC 11496-10. Advanced Video Coding-Final Committee Draft [S]. Document JVT-G050, March 2003.
- 2 Tekalp A M. Digital Video Processing[M]. Englewood Cliffs: Prentice Hall, New Jersey, USA, 1995:101~108.

- 3 Lappalainen, V, Hailapuro, A, Hamalainen T D. Optimization of emerging H. 26L video encoder [A]. In: Proceedings of IEEE Workshop on Signal Processing Systems Design and Implementation, SiPS'2001 [C], Antwerp, Belgium, 2001: 406~415.
- 4 Li R, Zeng B, Liou M L. A new three-step search algorithm for block motion estimation[J]. IEEE Transactions on Circuits and System for Video Technology, 1994, 4(4):438~442.
- 5 Zhu S, Ma K K. A new diamond search algorithm for fast block matching motion estimation [A]. In: Proceedings of 1997 International Conference Information, Communications and Signal Processing, ICICS'1997 [C], Beijing, China, 1997, 1: 292~296.
- 6 Li W, Salari E. Successive elimination algorithm for motion estimation [J]. IEEE Transactions on Image Processing, 1995, 4(1):105~107.
- 7 JVT Reference Software Official version JM61e [CP/OL]. <http://bs.hhi.de>



薛全 1975年生,浙江大学通信与信息系统专业博士研究生。主要研究方向为视频压缩、网络流媒体传输和模式识别。



刘济林 1947年生,教授,博士生导师。主要研究领域为计算机视觉、多媒体处理、图像通信和智能交通等。



陈建乐 1979年生,浙江大学通信与信息系统专业博士研究生。主要从事视频压缩和图像处理的研究。