

# H. 264/AVC 中基于搜索区域划分及评估的运动估计

申 舟 李正明 潘天红

(江苏大学电气信息工程学院, 镇江 212013)

**摘 要** 目前视频编码器中的快速运动估计搜索算法(如三步法、钻石搜索法)搜索点数少、速度快,但易陷入局部最优。另外一些混合模板搜索算法(如 UMHexagons 算法)较为有效地克服了这一缺点,然而这种算法需要处理较多的搜索点数导致搜索速度较慢。为了达到在减少搜索陷入局部最优可能的同时尽量降低搜索点数,提高搜索速度的目的,提出了 PEOSA 算法。首先利用预测运动矢量来区分出需要重点搜索的区域和较为次要的区域,其次依据重要性的不同对区域采用不同的搜索方案。对比实验的结果表明,本文算法在基本不影响图像质量的情况下,使运动估计时间减少了 43.84%,提高了搜索速度。在视频实时编解码方面具有较高的实际应用价值。

**关键词** H. 264 UMHexagons 算法 运动估计 运动矢量

中图分类号: TP37 文献标志码: A 文章编号: 1006-8961(2010)02-0242-05

## Motion Estimation Based on the Partition and Evaluation of the Search Area in H. 264/AVC

SHEN Zhou, LI Zhengming, PAN Tianhong

(College of Electronic and Information Engineering Jiangsu University, Zhenjiang 212013)

**Abstract** Although the search speed of fast motion estimation algorithms such as Three Step Search and Diamond Search is fast, it is easy to trap into local minimum which caused the false motion vector predictor. UMHexagons solved the problem well, however, this algorithm is still needed to search too many points and therefore slowed down the search speed. In order to obtain a good balance between search speed and quality performance, this paper proposed PEOSA algorithm. It divided the search area into four parts firstly, then evaluated the probability of the best motion vector in every part and finally searched every part through different search scheme. The experiment results showed that the PEOSA algorithm save more than 43.84% search time compared with UMHexagons, while the averaging PSNR loss was less than 0.017dB for all tested sequences. The new search algorithm holds high value in real time coding and decoding.

**Keywords** H. 264 UMHexagons motion estimation motion vector

## 0 引言

H. 264 是国际标准化组织 (ISO) 与国际电信联盟 (ITU) 共同成立的联合视频组 (JVT) 于 2005 年公布的新一代视频编码标准<sup>[1]</sup>。这种新的视频编码标准较以往的标准极大地提高了编码效率。在相同

的图像质量下, H. 264 所需码率约为 MPEG-2 的 30%、H. 263 的 51% 以及 MPEG-4 part2 的 61%。

H. 264 的高效是以高度复杂的运算为代价的。运动估计是视频编码器结构中最重要的一部分之一<sup>[2]</sup>, 在 H. 264 编码器的运算总量中所占的比例达到了 60% ~ 80%。运动估计结果的准确程度直接影响码流的大小, 它的运算速度也直接影响帧率的快

基金项目: 国家高技术研究发展计划 (863) 项目 (2008AA10Z208); 江苏大学高级专业人才培养启动基金项目 (08JG046)

收稿日期: 2008-10-13 改回日期: 2008-12-05

第一作者简介: 申舟 (1983—), 男, 江苏大学电气学院研究生。主要研究方向为 H. 264 的研究及其在 DSP 上的实现。E-mail:

shenzhou\_1983@yahoo.com.cn

慢,这两项都是实时视频编码的重要指标。优秀的运动估计算法是提高 H. 264 编码器效率的关键所在。非对称十字形多层次六边形格点搜索算法 (UMH exagons 算法) 是一种优秀的算法,已经正式被 JVT 采纳成为官方算法<sup>[3]</sup>。拥有较高精度的初始点预测,而且其搜索时间比全搜索算法节省了 90%。尽管该算法在一般情况下的运动估计结果已经比较准确了,然而在运动向量较大的块中依然还存在陷入局部最优问题,另外相对于三步法,四步法等算法,它的搜索点数较多,直接减缓了搜索速度,降低了搜索效率。

针对上述问题,提出了 PEOSA (partition and evaluation of search area) 搜索算法。重点搜索最优运动矢量最可能存在的区域是这一算法的核心。首先,以预测运动矢量存在的位置来推断最优运动矢量可能存在的区域,并把这些区域按这种可能性的大小进行划分。接着对这些区域用不同的方案进行处理,在最优运动矢量存在可能性越高的区域,采用较为精细的搜索策略,可能性较低的区域,采用较为粗略的搜索策略,最终找到最优运动矢量。经实验验证,本文算法相对于 UMH exagons 算法,在对 PSNR 影响很小的情况下大幅减少搜索点数,提高了搜索速度,在编码速度与编码质量上取得了良好的平衡。

## 1 UMH exagons 算法

### 1.1 UMH exagons 算法的运动估计准则

H. 264 编码手册<sup>[4]</sup>推荐使用拉格朗日代价函数  $J$  (如式 (1) 所示) 作为衡量宏块之间匹配程度的准则。

$$J(\mathbf{m}, \lambda_{\text{motion}}) = \text{SAD}(s_c(\mathbf{m})) + \lambda_{\text{motion}} \cdot R(\mathbf{m} - \mathbf{p}) \quad (1)$$

式中,  $\lambda_{\text{motion}}$  是拉格朗日常数,  $R(\mathbf{m} - \mathbf{p})$  是采用当前运动矢量时所需的比特数。SAD (sum of absolute difference) 是两个宏块对应像素值的绝对差之和。

$$\text{SAD}(s_c(\mathbf{m})) = \sum_{x=1}^M \sum_{y=1}^N |s(x, y) - c(x - m_x, y - m_y)| \quad (2)$$

式中,  $s(x, y)$  是当前宏块中像素的像素值,  $c(x, y)$  是带搜索位置宏块的像素值,  $m_x, m_y$  是待搜索块相

对于当前块的运动矢量。拉格朗日准则最优的含义是,如果采用当前运动矢量时,源图像与重构图像的差别最小且码率最低,则此时的运动矢量便是最优运动矢量,该准则是对误差和码率综合评价。

### 1.2 UMH exagons 的主要步骤

1) 初始点预测。如图 1(a) 所示,主要利用中值预测和上层预测得到运动矢量点,并从得到这些矢量点得到搜索的起始点  $O$ 。

2) 非对称十字形搜索。以预测得到的起始点  $O$  为中心沿横向搜索  $W/2$  个,纵向  $W/4$  个搜索点 (其中  $W$  代表搜索范围)。找到这些点中代价函数最小的点,作为本步骤的搜索结果  $A$  (如图 1(b))。

3) 多层次六边形搜索。首先以  $A$  为中心,以搜索半径为 2 进行全搜索共计 25 个点,接着以  $A$  为中心进行多层次六边形搜索。搜索层数为  $W/4$ 。搜索点数为  $16 \times W/4 = 4W$ ,本步骤共搜索  $25 + 4W$  个点,从中找出代价函数  $J$  最小的点  $B$  (如图 1(c))。

4) 以  $B$  为中心进行扩展六边形搜索,若代价最小点在模板中央,则进行下一步,否则继续进行扩展六边形搜索。直到代价函数最小点在模板中央为止,最后得到代价函数最小点  $C$ 。以  $C$  为中心进行小菱形搜索,若代价最小点在模板中央,则搜索结束。否则继续进行小菱形搜索,直到代价函数最小点在模板中央为止,最终得到的点  $D$  就是要得到的最优点 (如图 1(d))。

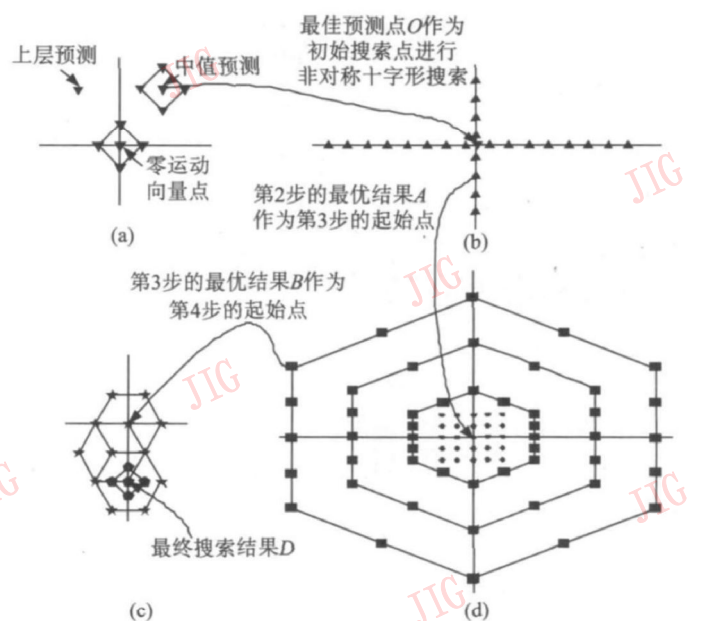


图 1 UMH exagons 算法示意图

Fig 1 Sketch map of the UMH exagons algorithms

### 1 3 UMHexagons算法的性能

高精度的起始点预测、大幅降低搜索陷入局部最优以及相对于 FS 算法大幅提高搜索效率是该算法的三大优点。然而该算法需要较多的搜索点数来保证减少陷入局部最优,以搜索范围  $W = 16$  为例,仅在第 2)步和第 3)步中就共需要搜索  $25 + 25 + 4 \times 16 = 114$  个固定搜索点。再加上初始点预测及第 4)步中不固定点数的搜索,每对一个宏块进行运动估计要搜索的点数在 140 个左右。相对于三步搜法、四步搜索法、对数搜索法等快速算法,该搜索算法搜索点数过多,减缓了搜索速度。

## 2 PEOSA 算法

针对 UMHexagons 算法搜索点数较多的缺点,提出了 PEOSA 算法,在继承 UMHexagons 算法高品质画质的优点的同时,减少了搜索点数,从而直接提高了搜索速度。

### 2 1 PEOSA 算法的相关定义

1) 搜索区域划分。如图 2 所示,将整个搜索区域划分为 4 个象限,并定义象限集合  $X = \{x_1, x_2, x_3, x_4\}$ 。其中  $x_j$  代表 4 个象限。

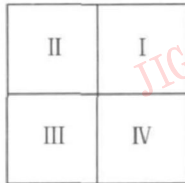


图 2 搜索区域划分

Fig 2 Partition of the search area

2) 预测运动矢量与其所属象限的关系矩阵

$R_{4 \times 3}$ 。

$$R_{4 \times 3} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \\ r_{41} & r_{42} & r_{43} \end{bmatrix} \quad (3)$$

$$r_{ij} = \begin{cases} 1 & r_i \in x_j \\ 0 & r_i \notin x_j \end{cases} \quad (4)$$

式中,  $j$  代表  $mvUpLX, mvMpLX, mvNrpLX$  这 3 个预测运动矢量<sup>[5]</sup>,  $i$  代表象限。

3) 权重向量  $Q = \{1.5, 1, 0.8\}$ 。它的 3 个元素分别代表  $mvUpLX, mvMpLX$  及  $mvNrpLX$  的权重。

4) 代表象限重要性的向量  $M \{m_1, m_2, m_3, m_4\}$ , 如式 (3) 所示, 向量  $M$  中的每一个元素代表该象限存在最优运动矢量的可能性, 数字越高代表这种可能性就越大。

$$M = R_{4 \times 3} \times Q^T \quad (5)$$

5) 方案确定规则。根据向量  $M$  中元素  $m_i$  的取值, 按照式 (4) 规则来确定各个象限的搜索方案。

$$\begin{cases} m_i \leq 1.5 & \text{等级 A 方案 1} \\ 1 \leq m_i < 1.5 & \text{等级 B 方案 2} \\ 0.8 \leq m_i < 1 & \text{等级 C 方案 3} \\ m_i < 0.8 & \text{等级 D 方案 4} \end{cases} \quad (6)$$

### 2 2 PEOSA 算法的搜索方案

方案 1 首先在本象限内以  $W/4$  为步长进行全搜索, 找出结果最优的点, 其次以该最优点为中心用小菱形模板进行搜索。直到拉格朗日代价函数最小点在模板中央为止。

方案 2 在该象限以该象限中心为起点利用三步法进行搜索。

方案 3 在本象限内以预测运动矢量端点为中心用小菱形模板进行搜索, 直到拉格朗日代价函数最小点在模板中央为止。

方案 4 放弃对该区域的搜索。

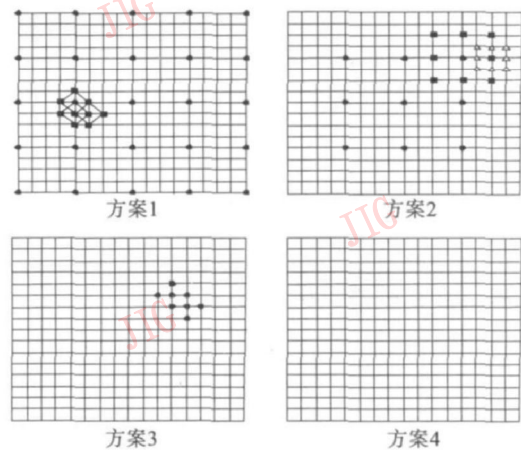


图 3 搜索方案

Fig 3 Search algorithms

### 2 3 PEOSA 算法的步骤

1) 搜索区域划分并评估。根据预测运动矢量

所在位置用式 (5) 计算出向量  $M$ , 分别对这 4 个象限按照式 (6) 给定的规则进行评估, 确定个象限的搜索方案。

2) 对 4 个象限分别根据其所属等级使用对应的搜索方案。每个象限对应找到该象限内的最优结果, 然后在这 4 个象限的最优点中依据拉格朗日代价函数最小的准则找出最优点, 作为整个搜索区域内的最优点即最终结果。

### 2.4 PEOSA 算法性能的理论分析

将搜索区域进行划分并分别用不同的方案进行搜索覆盖了整个搜索范围, 针对搜索范围内最优运动矢量存在的可能性采用复杂度不同的搜索方案, 既减少了搜索点数, 又避免了陷入局部最优, 最大程度上实现编码品质和编码速度的最优, 在搜索范围  $W = 16$  时, 本算法在对一个宏块进行运动估计所需要的搜索点数平均为 67 个左右, 相对于 UMHexagons 算法的 140 个减少了一半, 从而可以大幅减少运动估计时间。

## 3 实验结果及分析

为了测试新算法的性能, 以 JM14.0 位参考软件<sup>[6]</sup>在不同的量化参数下选取了 coastguard highway foreman mobile 4 个标准序列进行测试, 这些序列能够体现画面中大多数的运动情况。同时用全搜索算法 (FS) 作为参考并与 UMHexagons 算法和表 1 列出的测试环境下进行了对比测试, 表 2 和表 3 分别是新旧算法在对不同测试序列不同量化参数情况下<sup>[7]</sup>的运动估计时间以及亮度分量的平均 PSNR, 并着重将新算法与 UMHexagons 算法的实验结果加以对比。

表 1 测试环境

Tab 1 Test environment

环境类别	内容
参考软件	JM14.0
搜索范围	16
编码帧数	50
量化参数	16 22 28 34 40
熵编码	CABAC
帧率	30 fps

表 2 不同搜索算法下的运动估计时间

Tab 2 Contrast of motion estimation time on different search algorithms

测试序列	不同算法下的 MET/s				MET Saved/s
	QP	FS	UMHexagons	PEOSA	
coastguard	16	2 218 423	322 690	183 283	- 139 407
	22	2 083 925	329 686	184 992	- 144 694
	28	1 810 906	326 752	183 203	- 143 549
	34	1 415 872	308 231	174 650	- 133 581
	40	1 036 162	276 688	160 817	- 115 871
highway	16	1 894 477	269 561	150 784	- 118 777
	22	1 397 531	244 388	137 519	- 106 869
	28	957 461	236 560	136 082	- 100 478
	34	666 420	222 711	126 232	- 96 479
	40	472 348	214 000	124 295	- 89 705
foreman	16	1 871 781	283 827	159 873	- 123 954
	22	1 612 577	276 551	160 749	- 115 802
	28	1 293 505	271 593	152 907	- 118 686
	34	999 496	268 729	151 316	- 117 413
	40	767 229	256 665	147 478	- 109 187
mobile	16	1 980 696	299 588	162 826	- 136 762
	22	1 858 348	309 379	163 312	- 146 067
	28	1 665 163	308 505	168 860	- 139 645
	34	1 409 279	310 984	170 568	- 140 416
	40	1 186 983	313 246	173 265	- 139 981
较 UMH 时间节省率				43.84%	

表 3 不同算法下的信噪比对比

Tab 3 Contrast of PSNR on different algorithms

测试序列	亮度平均峰值信噪比 PSNR-Y/dB				PSNR 变化 /dB
	QP	FS	UMHexagons	PEOSA	
coastguard	16	46.24	47.61	46.37	- 0.24
	22	40.47	40.46	40.42	- 0.04
	28	35.14	35.09	35.13	+ 0.04
	34	30.93	30.89	30.89	0
	40	27.28	27.79	27.88	+ 0.09
highway	16	45.93	45.93	45.91	- 0.02
	22	41.51	41.51	41.52	+ 0.01
	28	38.89	38.88	38.88	0
	34	36.18	36.12	36.07	- 0.05
	40	33.43	33.29	33.28	- 0.01
foreman	16	46.04	46.03	45.99	- 0.04
	22	41.20	41.19	41.17	- 0.02
	28	37.27	37.23	37.24	+ 0.01
	34	33.74	33.66	33.68	+ 0.02
	40	30.31	30.25	30.23	- 0.02
mobile	16	46.62	46.62	46.61	- 0.01
	22	40.87	40.85	40.85	0
	28	34.99	34.99	34.96	- 0.03
	34	29.96	29.91	29.93	+ 0.02
	40	25.65	25.63	25.62	- 0.01
PSNR 平均变化				- 0.017	

表中 MET Saved和 PSNR变化分别代表 PEOSA算法和 UMHexagons算法的运动估计时间和 PSNR的变化。由实验的数据结论可以看出在新算法的搜索时间比 UMHexagons算法平均减少 43.32%。而 PSNR 仅仅平均降低了 0.017 dB, 实验结果验证了 PEOSA 算法的合理性和有效性。

## 4 结 论

提出的 PEOSA 搜索算法利用运动矢量存在的可能性, 对整个搜索区域进行划分, 对最优运动矢量存在的可能性较大的区域采用较为细致的搜索方案, 对存在可能性较小的区域采用较为粗略的搜索方案, 这样不仅避免了陷入局部最优的可能, 而且大大降低了搜索点数, 降低了运动估计时间。从实验结果可以看出提出的新算法, 相对于 UMHexagons算法在图像质量仅损失很小的情况下, 较大幅度减少了搜索点数, 提高了搜索速度, 提高了算法的实用性。

## 参考文献 (References)

- [1] ITU-T Rec. H. 264 | ISO/IEC 14496-10 AVC, Draft Text of Final Draft International Standard for Advanced Video Coding[S].
- [2] Wiegand T, Sullivan G J, Bjøntegaard 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.
- [3] JVT-F017, Fast Integer Pel and Fractional Pel Motion Estimation for JVT[S].
- [4] JVT-X101, Text Description of Joint Model Reference Encoding Methods and Decoding Concealment Methods JVT[S].
- [5] Chen Zhibo, Xu Jian-feng, He Yim. Fast integer pel and fractional pel motion estimation for H. 264/AVC[J]. Journal of Visual Communication and Image Representation, 2006, 17(2): 264-290.
- [6] Sühring K. Joint Video Team Software M14.0[CP/OL]. [2008-10-13]. <http://iphome.hhi.de/suehring/tm1/download/>.
- [7] VCEG-N81G, Draft 1, Recommended Simulation Common Conditions for H. 26L Coding Efficiency experiments on Low Resolution Progressive Scan Source Material[S].