

针对 H. 264 的基于平坦区域预测的分像素运动估计

方 健 郑 伟 李炳博 王 匡

(浙江大学 信息与电子工程学系, 杭州 310027)

摘 要 运动估计消耗视频编码系统绝大部分的计算量。H. 264 由于采用多模式运动估计和四分之一精度搜索, 不仅增加了整像素运动估计的计算量, 而且也大大增加了分像素搜索运算量。随着快速整像素搜索算法的发展, 整像素计算量不断降低。因此对于 H. 264 的实时应用, 快速分像素运动估计算法变得十分重要。提出了一种全新的基于图像平坦区域预测的分像素运动搜索算法。该算法通过预测图像平坦均匀区域, 有效地减少了分像素搜索点数。实验结果表明, 与全搜索分像素搜索相比, 该算法可以减少 22% ~ 63% 左右的分像素搜索点数, 同时基本不改变图像质量和编码压缩率。

关键词 H. 264 分像素搜索 多模式运动估计 平坦区域

中图分类号: TN919.81 文献标识码: A 文章编号: 1006-8961(2008)12-2286-05

Fractional Pixel Motion Estimation based on Prediction of Flat Regions for H. 264

FANG Jian, ZHENG Wei, LI Bing-bo, WANG Kuang

(Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027)

Abstract Motion estimation (ME) consumes most of the computational resource in a video encoder system. H. 264 adopts multi-mode ME and quarter-pel accuracy fractional pixel search, which brings about large scale of computational cost not only to integer pixel ME but also to fractional pixel ME. However, the computational cost of integer pixel ME has been greatly reduced with the development of fast integer search algorithms. Therefore, fast fractional pixel ME algorithms become significantly important for real-time applications. A novel fractional pixel motion estimation based on prediction of flat regions of an image is first proposed in this paper. It reduces fractional pel search points efficiently by predicting flat homogeneous regions of the image. Experimental results show that compared to the full fractional pixel search, the proposed method can reduce 22% to 63% of fractional pixel search points while keeping the image quality and compression rate almost unchanged.

Keywords H. 264, fractional pixel search, multi-mode motion estimation, flat region

1 引言

与以前的视频标准相比, 新一代视频压缩编码标准 H. 264/AVC^[1], 在相同图像质量条件下可以提高 50% 左右的压缩率^[2]。H. 264 采用 7 种不同大小块的多模式运动估计, 搜索精度细化到 1/4 像素, 这有效增强了匹配块的准确性, 降低了残差数据的

能量, 提高了编码压缩率。但另一方面, 这也增加了整像素运动估计和分像素运动估计的复杂度。因此, 为了实现 H. 264 标准的实时应用, 快速运动估计算法成为目前视频压缩的研究热点之一。

典型的运动估计分两部分: 整像素搜索和分像素搜索。通常, 整像素搜索占运动搜索的绝大部分计算量, 是优化的关键。目前采用的整像素搜索算法主要有: 菱形搜索 (DS)^[3]、六边形搜索 (HBS)^[4]

基金项目: 国家自然科学基金项目 (90307002)

收稿日期: 2006-12-20; 改回日期: 2007-05-31

第一作者简介: 方 健 (1980 ~), 讲师。男。博士。从事视频压缩方面的研究。E-mail: fangjian1980@163.com

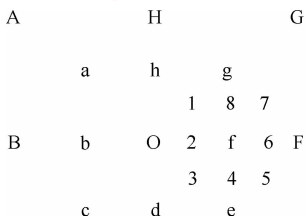
和针对 H. 264 的混合非对称多六边形搜索 (UMHexagonS)^[5]等。这些优化算法有效地降低了整像素搜索的计算量,从而也使得分像素运动估计的复杂度和整像素运动估计复杂度相当,因此有必要对分像素运动估计进行优化。目前分像素运动搜索算法主要有两类:一类是采用优化的搜索模型,通过尽可能少的路径找到匹配点,如基于中心的分像素搜索(CBFPS)^[5],基于抛物面预测的分像素搜索(PPFPS)^[6]等;另一类是提前终止搜索算法,如文献[7]提出的基于方向提取的分像素预测算法(PDR),通过阈值判定,提前结束分像素搜索。文献[8]同时运用了这两种策略,以尽可能地减少没有意义的搜索。

本文结合 H. 264 运动估计的特点,提出了一种基于图像平坦区域预测的分像素搜索算法,简称 FRPS。该算法的基本思想是,根据相邻宏块运动估计的结果和当前宏块第一种模式运动估计的结果,预测图像平坦区域。对于平坦区域的宏块,后续模式的运动估计只进行整像素搜索,不进行分像素搜索。实验结果表明,该算法有效地减少了分像素运动估计搜索点数,同时基本不影响编码图像质量和压缩率。

2 H. 264 中使用的分像素搜索算法

目前,H. 264 采纳了两种分像素搜索算法,分像素全搜索 (FFPS) 和基于中心的快速搜索 (CBFPS)。

分像素全搜索算法如图 1 所示。以最佳整像素位置 O 为中心进行细化的全搜索。首先计算最佳整像素位置周围的 8 个 1/2 像素位置(a,b,⋯,h),



O, A, ⋯, H: 整像素点
a, b, ⋯, h: 半像素点
1, 2, ⋯, 8: 四分之一像素点

图 1 分像素全搜索

Fig. 1 Full fractional pixel search

找到最佳的 1/2 像素匹配点 f;然后计算最佳 1/2 像素位置周围的 8 个 1/4 像素位置(1,2,⋯,8),找到最佳的 1/4 像素匹配点,作为分像素搜索的最佳运动矢量。分像素搜索在计算匹配误差值时增加了 Hadamard 变换,以进行更准确的匹配失真比较。因此最佳整像素点的误差值在分像素运动估计时需要重新计算,加上 8 个 1/2 像素及 8 个 1/4 像素点,全搜索分像素运动估计需要计算 17 个像素位置。

基于中心的快速分像素搜索算法根据块大小分两种情况,对大块(16×16,16×8,8×16)仍然采用全搜索,对于小块(8×8,8×4,4×8,4×4)采用如图 2 所示的搜索策略。首先通过相邻块的运动矢量的中值获得分像素预测运动矢量(Pred_x, Pred_y)。比较它和(0,0)的匹配误差值,其中小的作为分像素搜索的起始点。然后运用菱形搜索获得最终的分像素匹配点(Match_x, Match_y)。和全搜索相比,这种方法可以减少 33% 的计算量^[4]。

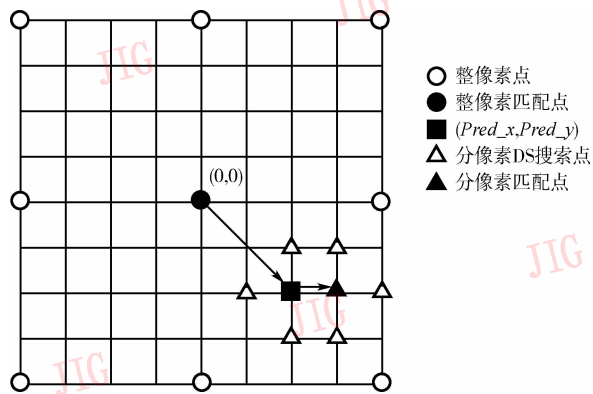


图 2 基于中心快速分像素搜索

Fig. 2 Center based fractional pixel search algorithm

3 FRPS 搜索算法

H. 264 采用 7 种模式的运动估计。如果采用全搜索分像素运动估计,每个模式搜索 17 个像素点,一个宏块分像素搜索共需要计算 119 个点,计算量十分可观。而实际上对于画面平坦区域,分像素搜索对性能的改进并不明显。如果提前预测这些平坦区域块,就可以减少不必要的分像素搜索,这就是 FRPS 算法的切入点。表 1 显示了典型测试码流中,最佳运动矢量落在整像素位置的统计情况。可以发

现,19.83%以上宏块的最佳运动矢量落在整像素位置,特别是对于运动不剧烈的序列,这个比例超过了80%。显而易见,这部分宏块的分像素搜索是没有意义的。进一步研究发现,利用已经进行的运动搜索结果,可以预测这样的整像素运动矢量宏块。

表 1 整像素运动矢量比例
Tab.1 Rate of integer-pel MV

序列	整像素运动矢量比例(%)
claire	85.30
news	80.25
grandma	83.60
silent	71.86
carphone	33.44
foreman	19.83

首先,相邻宏块的搜索结果可以预测区域的平坦程度。通过相邻块的运动矢量的中值获得当前宏块的预测运动矢量,如果预测运动矢量落在整像素位置上,则当前宏块判定为整像素匹配块,简称IMMB(intpel_match_mb)。其次,7种运动估计模式之间也存在相关性。通过模式1(16×16)运动搜索的结果也可以预测当前宏块的平坦程度。如果模式1运动矢量落在整像素位置上,则当前宏块判定为整像素匹配块。表2和表3给出了这两种整像素匹配宏块判定方法的准确率和命中率。其中预测准确率指在判定为整像素匹配的宏块中,实际运动矢量落在整像素上的宏块比率;命中率指在实际运动矢量落在整像素上的宏块中,提前通过预测判定为整像素匹配宏块的宏块比率。PreMV代表相邻块运动矢量中值预测判决,MV_model1代表模式1运动矢量预测判决,Pred & model1表示采用两种预测判决的结果。

表 2 整像素匹配宏块预测方法的准确率

Tab.2 Accuracy-rate of predict method for IMMB
单位:%

序列	PredMV	MV_model1	Pred&model1
claire	91.27	95.96	97.39
news	87.24	94.43	95.77
grandma	90.44	94.78	95.73
silent	83.47	90.44	92.86
carphone	59.77	79.67	87.99
foreman	47.86	68.34	81.63

表 3 整像素匹配宏块预测方法的命中率

Tab.3 Hit-rate of predict method for IMMB
单位:%

序列	PredMV	MV_model1	Pred&model1
claire	98.54	94.77	93.35
news	96.94	99.30	96.40
grandma	96.73	98.88	95.73
silent	96.61	98.11	94.87
carphone	82.05	89.85	74.79
foreman	71.90	91.09	67.10

准确率越高,匹配误差越小,因此编码质量下降的越小,码流变化也越小。命中率越高,跳过分像素搜索的宏块越多,因此分像素运动估计计算量越小。从表中可以看出,单独采用PredMV预测和MV_model1预测准确率都偏低,对于运动比较剧烈的序列更加不理想。把两种预测方法结合使用可以明显提高预测准确率,即使对于剧烈运动序列,准确率也超过80%。因此采用相邻宏块运动矢量中值和当前宏块模式1运动矢量,共同预测整像素匹配块。分像素搜索在计算匹配误差值时增加了Hadamard变换。因此对于判定为整像素匹配的宏块,虽然后续模式不进行分像素搜索,但最佳整像素点的匹配误差值需要重新计算。这里提出的基于平坦区域预测的分像素搜索算法,执行流程如图3所示。

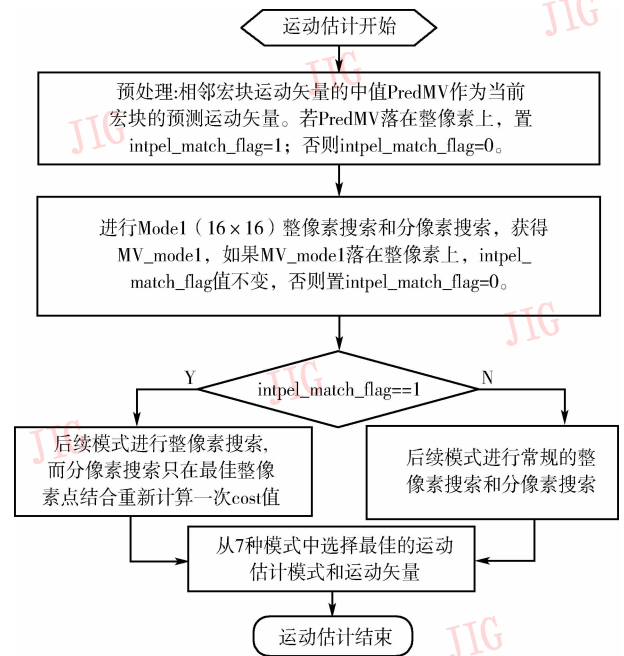


图 3 算法流程

Fig.3 Algorithm procedure

FRPS 快速算法通过检测预测运动矢量和模式 1 运动矢量是否落在整像素点,判定当前宏块是否是整像素匹配块。对于整像素匹配宏块,后续模式进行常规整像素搜索,而分像素搜索只在最佳整像素点重新计算一次误差值。对于普通宏块,进行常规的整像素和分像素搜索。

4 仿真结果和分析

测试序列为标准 QCIF 序列 (claire、news、grandma、silent)、carphone 和 foreman, 帧率均为 30fps。主要编码参数如下:采用 Baseline 框架,无 B 帧,熵编码采用 CAVLC。整像素搜索采用 JM8.4 提供的快速整像素搜索算法 UMHexagonS,半像素搜索使用分像素全搜索 FFPS 和基于中心的快速分像素搜索 CBFPS。FRPS 算法分别和 FFPS、CBFPS 结合,构成 FFPS + FRPS 和 CBFPS + FRPS,分别以 F&F 和 C&F 表示。下面将它们的性能与单独使用 FFPS 的性能进行比较。

首先固定量化步长测试 FRPS 算法的性能。量化步长序号 QP 从 24 到 40,相应量化步长 Qstep 从

10 增加到 64。分别测试编码的峰值信噪比 PSNR, P 帧平均比特数 BPF (Bits Per P-Frame) 和每个宏块平均分像素搜索点数 SPPMB (Search Points Per MB)。计算 FRPS 算法相对于 FFPS 的信噪比改变 Δ_{PSNR} 、P 帧平均比特数改变率 Δ_{BPF} 和平均搜索点数节省率 Δ_{SPPMB} 。表 4 给出了各项指标在 17 组量化步长上的平均值,反映了主要量化区间内的平均性能。图 4 ~ 图 7 显示了 Δ_{PSNR} 和 Δ_{BPF} 在量化区间上的波动情况。

表 4 平均性能

Tab.4 Performance on average

序列	Δ_{PSNR}/dB		$\Delta_{BPF}(\%)$		$\Delta_{SPPMB}(\%)$	
	F&F	C&F	F&F	C&F	F&F	C&F
claire	0.018	0.039	-0.02	-0.21	59.67	63.89
news	0.049	0.058	0.28	0.25	56.18	61.04
grandma	0.024	0.023	0.14	-0.11	54.43	58.72
silent	0.062	0.072	0.02	-0.33	48.79	55.84
carphone	0.015	0.029	-0.19	-0.48	31.66	49.65
foreman	0.008	0.025	-0.07	-0.37	22.73	45.96

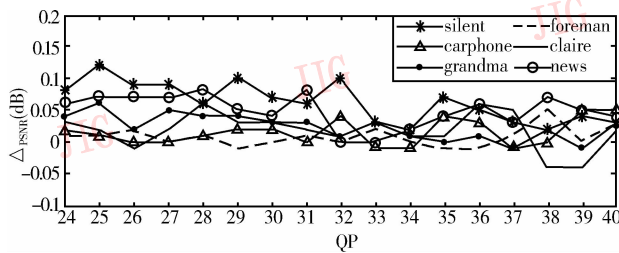


图 4 F&F 算法对 PSNR 的影响

Fig.4 Influence on PSNR by F&F algorithm

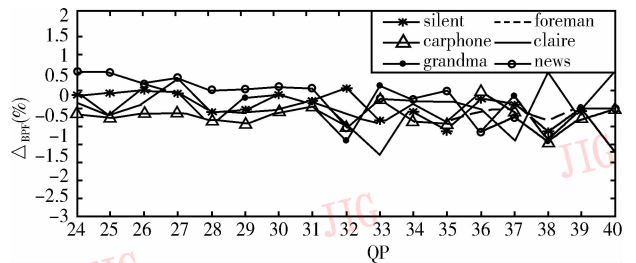


图 5 F&F 算法对 BPF 的影响

Fig.5 Influence on BPF by F&F algorithm

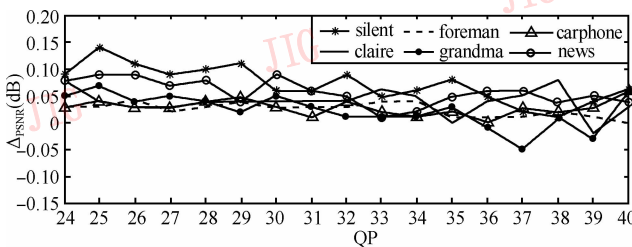


图 6 C&F 算法对 PSNR 的影响

Fig.6 Influence on PSNR by C&F algorithm

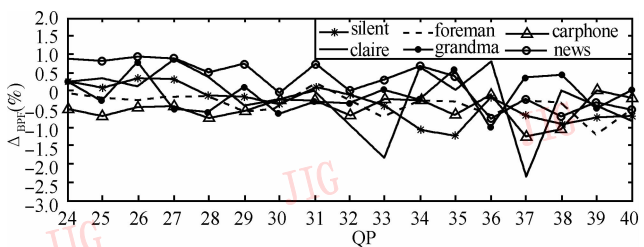


图 7 C&F 算法对 BPF 的影响

Fig.7 Influence on BPF by C&F algorithm

可以发现,FRPS 算法和 FFPS 结合使用,可以降低 22.73% ~ 59.67% 的分像素计算量;平均信噪比下降不超过 0.062dB,最大不超过 0.12dB;平均比特率上升不超过 0.28%,最大不超过 1.06%。如果 FRPS 和 CBFPS 结合使用,可以降低

45.96% ~ 63.89% 的分像素计算量;平均信噪比下降不超过 0.072dB,最大不超过 0.14dB;平均比特率上升不超过 0.25%,最大不超过 0.93%。对于大背景,小运动序列,整像素匹配宏块比例高,分像素搜索点数可以节省 50% 以上。对于运动剧烈的序

列,整像素匹配宏块比例低,但仍可以节省 30% 左右的分像素搜索点数。

其次固定编码码率测试 FRPS 算法的性能。分别测试序列 *claire* 和 *foreman* 在码率为 50kbps、100kbps、150kbps 和 200kbps 时的性能,结果如表 5 和表 6 所示。可以看出,采用 FRPS 算法后,PSNR 下降不超过 0.13dB,每个宏块分像素搜索点数节省 25~70 点。这表明在固定码流应用场合,FRPS 仍然具有稳定的性能,保持良好的编码图像质量的同时,能够有效降低分像素的计算量。

表 5 *Claire* 序列在固定码率下的性能

Tab. 5 Performance of sequence *claire* on fixed bit-rate

比特率 (kbps)	PSNR (dB)			SPPMB		
	FFPS	F&F	C&F	FFPS	F&F	C&F
50	40.92	40.87	40.79	119	50.2	43.2
100	44.04	43.96	43.96	119	52.1	44.1
150	45.83	45.74	45.70	119	53.6	44.3
200	47.06	46.96	46.97	119	54.2	45.0

表 6 *Foreman* 序列在固定码率下的性能

Tab. 6 Performance of sequence *foreman* on fixed bit-rate

比特率 (kbps)	PSNR (dB)			SPPMB		
	FFPS	F&F	C&F	FFPS	F&F	C&F
50	30.07	30.01	30.12	119	87	62.3
100	33.23	33.21	33.21	119	93.3	64.7
150	34.96	34.93	34.96	119	94.4	64.9
200	36.11	36.12	36.13	119	94.9	64.8

5 结 论

针对 H. 264 的基于图像平坦区域预测的分像素搜索算法,充分利用了相邻宏块间相关性和宏块内不同模式间的相关性。根据中值预测运动矢量和模式 1 的运动矢量,预测整像素匹配宏块。对于整

像素匹配宏块,其余 6 种匹配模式只进行整像素搜索,跳过分像素搜索。实验结果表明,该算法在保证图像质量和编码压缩率的前提下,可以进一步减少 22%~63% 左右的分像素搜索点数,特别适用于视频会议和视频监控等应用场合。而且该算法本身具有独立性,可以和其他的快速分整像素搜索算法结合使用,以获得更好的性能。

参考文献 (References)

- 1 Thomas Wiegand. ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H. 264 | ISO/IEC 14496-10 AVC) [A]. In: 7th JVT Meeting and 19th VCEG Meeting [C]. Pattaya, Thailand; 2003.
- 2 Thomas Wiegand, Gary J Sullivan, Overview of the H. 264/AVC video coding stand [J]. IEEE Trans. Circuit Syst. Video Technol, 2003, 13(7): 560~576.
- 3 Zhu Shan, Ma Kai-kuang. A new diamond search algorithm for fast block-matching motion estimation [J]. IEEE Transactions on Image Processing, 2000, 9(2): 287~290.
- 4 Zhu Ce, Lin Xiao, Chau Lap-pui, Hexagon-Based Search Patten for Fast Block Motion Estimation [J]. IEEE Transactions on CSVT, 2002, 12(5): 349~355.
- 5 Chen Z B, Zhou P, He Y. Fast integer pel and fractional pel motion estimation for JVT (JVT-F017) [A]. In: Proceedings of 6th Joint Video Team meeting [C], Awaji, Japan, 2002: 5~13.
- 6 Chen Z B, Du C, Wang J H, et al. PPFPS-A paraboloid prediction based fractional pixel search strategy for H. 26L [A]. In: Proceedings of International Symposium on Circuits and Systems 2002 [C], Scottsdale, Arizona, 2002: 9~12.
- 7 Chen Z B He Y. Prediction based Directional Refinement (PDR) algorithm for fractional pixel motion search strategy (JVT-D069) [A]. In: Proceedings of 4th Joint Video Team meeting [C], Klagenfurt, Austria, 2002: 22~26.
- 8 Yin P, Tourapis Hyc, Tourapis A M, et al. Fast mode decision and motion estimation for JVT/H. 264 [A]. In: Proceedings of International Conference on Image Processing 2003 [C], Barcelona, Spain, 2003: 853~856.
- 9 JVT reference software version 8. 4. [EB/OL]. http://bs.hhi.de/~suehring/tml/download/old_jm/2006-6-3.