

一种 H.264/AVC 中的快速运动估计算法

段青青, 宋学瑞

(中南大学信息科学与工程学院, 长沙 410083)

摘要: 在UMHexagonS算法的基础上, 最新的H.264/AVC视频编码采用一种简单、有效的混合快速运动估计算法。该文介绍一种新的快速搜索算法, 初步研究和分析其综合性能的优越性, 并提出改进策略, 如增加终止搜索的判定、进一步划分运动类型、采用自适应的方向性搜索模板。实验结果表明, 改进的策略能减少“简化UMHexagonS算法”8%~30%的搜索时间, 而码率和率失真性能的变化可以忽略不计, 对一些特殊序列, 率失真性能略优于“简化UMHexagonS算法”。

关键词: H.264 标准; 快速运动估计; 简化UMHexagonS算法; 方向性搜索模板

Fast Motion Estimation Algorithm in H.264/AVC

DUAN Qing-qing, SONG Xue-rui

(College of Information Science and Engineering, Central South University, Changsha 410083)

【Abstract】 The latest video coding standard proposes a simple yet efficient hybrid FME scheme based on the existing UMHexagonS scheme. This paper introduces this new fast search algorithm. Some improved schemes are proposed based on the study and analysis of the advantages of simplified UMHexagonS algorithm, such as introduces more early terminate criterions, further division to the motion type, adopts directional search patterns for different blocktype adaptively, etc. Experimental results show that the improved strategies can reduce about 8%~10% of motion estimation time, while the R-D performance and bit rate are almost the same as simplified UMHexagonS algorithm.

【Key words】 H.264; fast motion estimation; simplified UMHexagonS; directional search pattern

1 概述

运动估计是视频压缩编码的关键部分, 能有效地去除序列图像的帧间冗余。目前各种视频编码标准中广泛应用的运动估计算法是块匹配算法(BMA), 匹配的准则是块的失真度。

近年来, 典型的基于块匹配的快速运动估计算法主要有: 三步搜索法(TSS), 基于块的梯度下降搜索法(BBGDS), 四步搜索法(FSS), 六边形搜索法 HEXBS(Hexagon-Based Search)等。这些算法主要采用具有不同步骤和搜索模板的搜索策略, 以达到较小的计算复杂度, 在相对小的搜索范围和图像尺寸时, 这些算法可以达到比较好的效果, 但在处理某些大尺寸图像和较大的搜索范围时, 很容易落入局部最小点, 影响编码的效果。

H.264/AVC 是 ITU-T 和 MPEG 联合制定的最新的视频编码标准, 它采用 UMHexagonS 算法作为整像素的快速运动估计算法, 该算法较好地解决了局部最小点等问题, 在保持较好率失真性能的前提下, 运算量也十分低。但为了保证较高的率失真性能, UMHexagonS 采用不同搜索模板和一些局部全搜索, 尽管也采用了早期终止机制, 它在速度方面的提高是非常有限的。

为了进一步降低编码时间、提高率失真性能, 目前 JVT 参考软件在 UMHexagonS 算法的基础上提出改进和简化策略。实验结果表明, 改进和简化的 UMHexagonS 算法(以下简称 SUMHexagonS)相对于 UMHexagonS 算法, 可以平均减少 55%左右的运动估计时间, 同时其具有较低的码率和运算复杂度。

2 UMHexagonS 算法

目前H.264/AVC正式采用的UMHexagonS算法的运算量相对于快速搜索算法可减少 90%以上^[1], 而且在这种低运算量下, 率失真性能明显优于其他快速算法, 能较好地满足低码率和实时性要求。

但 UMHexagonS 算法利用了空间域、时间域和上层块的 MV 和 SAD, 这些预测矢量的存储占用了大量内存资源。这对于应用到内存有限的硬件资源上是不理想的。而且也未充分使用这些预测集, 一些高准确率预测矢量可以作为搜索的目标矢量, 而无须进行任何搜索, 但按 UMHexagonS 的“早停止”判断, 其仍需继续搜索, 这不但增加了搜索时间, 有时甚至降低了匹配度, 因此, “早停止”门限值须进一步完善。UMHexagonS 算法使用了小六边形和小菱形模板来确定局部最优点, 所以, 可以减少其中 25 点的局部全搜索, 以提高速度。

3 “简化 UMHexagonS 算法”的介绍

JVT 提出了一种改进和简化的快速运动估计方案。在 SUMHexagonS^[2], 具体操作如下:

(1) 在对预测矢量进行搜索后, 增加了直接停止搜索的判断, 省去了局部的全搜索, 降低了搜索点数。

(2) 利用非常简单的早期终止技术。早期终止门限值的计

作者简介: 段青青(1984 -), 女, 硕士研究生, 主研方向: 视频压缩编码, 大规模集成电路; 宋学瑞, 副教授

收稿日期: 2007-10-28 **E-mail:** qingjin-163397@163.com

算, 通过移位和比较操作来简化和替代原来含有浮点乘法运算的算法。

(3)只使用上层预测矢量和中值预测矢量, 在很大程度上降低了对内存空间的要求。

SUMHexagonS 算法采用的匹配准则仍为 UMHexagonS 算法中率失真最优化准则, 达到在目标数据率的条件下传输的视频信号失真最小。匹配误差函数为

$$J(MV, \lambda_{MOYION}) = SAD(s, c(MV)) + \lambda_{MOYION} \times R(MV - PMV) \quad (1)$$

其中, 绝对差值和 SAD 为

$$SAD(s, c(\overline{MV})) = \sum_{x=1, y=1}^{Bx, By} |s[x, y] - c[x - MVx, y - MVy]| \quad (2)$$

其中, s 是当前进行编码的原始数据; c 为已经编码重建的用于进行运动补偿的参考帧数据; MV 为候选的运动矢量; λ_{MOYION} 为拉格朗日常数; PMV 为预测矢量; $R(MV - PMV)$ 代表了运动矢量差分编码可能消耗的比特数, 可在查表中得到。

搜索步骤如下:

步骤 1 准静止块检测

搜索中值预测矢量和原点, 通过是否满足收敛收缩的条件来判断是否为准静止块, 若满足, 进行收敛搜索后停止搜索, 否则进入步骤 2。

步骤 2 运动类型判定

进行一次小的局部搜索, 获得最佳匹配点后, 对运动类型进行判断: 若满足大运动块的要求进入步骤 3; 否则进入步骤 4。

步骤 3 大运动块搜索

以获得的最佳匹配点为中心, 进行对称的十字形搜索, 获得最佳匹配点后进入一次 6 点的六边形搜索, 获得最佳匹配点后, 采用不断扩大一倍搜索半径的 16 点的大六边形模板进行搜索, 直至搜索超出范围结束步骤 3。

步骤 4 中运动块搜索

(1)检测上层块预测矢量, 并进行一次小的局部搜索, 获得最佳匹配点。再次判断是否满足收敛搜索的条件: 满足则进入步骤 5, 进行收敛搜索后结束搜索; 否则进入(2)。

(2)扩展的六边形搜索 以步骤 4 确定的最佳匹配点为中心进行连续的中六边形搜索, 直至其最佳的运动矢量位于六边形的中心。获得最佳匹配点后进入步骤 5。

步骤 5 小运动块的精细搜索

以当前最佳匹配点为中心进行小菱形模板搜索, 直至最佳匹配点为菱形中心。

收敛搜索即一次小菱形模板的搜索。收敛搜索的判定定义为式(3), 设当前最佳匹配点的 SAD 值为 min_mcost 。

$$min_mcost < (ConvergeThreshold \gg blocktype_shiftfactor[blocktype]) \quad (3)$$

大运动块的判定定义为

$$(blocktype == 1 \& \& min_mcost < (CrosssearchThreshold1 \gg blocktype_shiftfactor[blocktype])) \quad (4)$$

$$(min_mcost < (CrosssearchThreshold2 \gg blocktype_shiftfactor[blocktype])) \quad (5)$$

对于 16×16 (skip), 16×16 , 16×8 , 8×16 , 8×8 , 8×4 , 4×8 , 4×4 块, $blocktype_shiftfactor[8] = \{ 0, 0, 1, 1, 2, 3, 3, 1 \}$, 根据经验, 设 $CrossSearchThreshold1 = 800$, $CrossSearchThreshold2 = 7000$, $ConvergeThreshold = 1000$ 。对于不同类型的块, 判定的阈值可以通过简单的移位得到。

若假定搜索半径为 12, 则 SUMHexagonS 搜索算法的界

面图如图 1 所示。其中, 灰色搜索点为当前搜索步骤的最佳匹配点, 并作为下步的搜索中心。

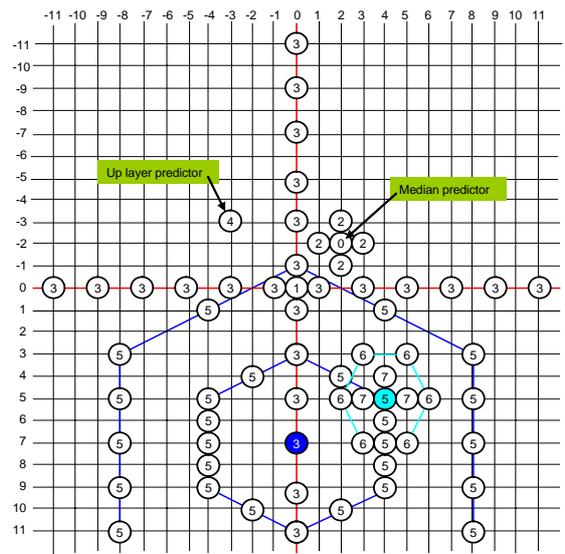


图 1 SUMHexagonS 运动搜索算法界面图

4 基于 SUMHexagonS 算法的分析和改进

实验结果证明, 改进的算法相对于 UMHexagonS, 能有效地降低了运算复杂度, 且在编码时间和运动估计时间上分别减少 46%, 57% 左右, 码率也在一定程度上得到了降低, 同时保持了较高的 R-D 性能。此算法生成的运动矢量更加平滑, 运动矢量残差也更小, 这对在易错环境中的数据分割和非对称的错误检测是非常有利的。

但 SUMHexagonS 算法未采用的两个视频序列的重要特征如下:

(1)一些甚至所有运用的预测运动矢量相互之间都是相同或者相近的, 在某种情况下, 目标运动矢量就在预测矢量的一个很小的临近区域内^[3], 因此, 可以引用了一些额外的可靠性判断标准, 来进一步加速运动估计。

(2)运动矢量场分布具有空间方向性, 尤其是在水平和垂直方向。针对不同类型的宏块, 引入不同类型的搜索模板进行匹配, 能保证搜索精度和搜索速度的一致性。

另外只使用空间预测, 对于高纹理或者锐化的运动序列, 会预测失败, 容易陷入局部最小, 造成很大的匹配误差。高准确率起点预测, 能增大开始就找到最佳匹配点的概率, 及时结束搜索, 搜索时间的减少完全能够弥补用于预测和判断所花费的时间。

针对以上特点, 进行了以下几点改进:

(1)根据相邻块之间的相关性引入了非常可靠的早期停止标准。在检测中值预测矢量后, 引入立即停止搜索标准, 满足标准则无须进行收敛搜索, 而直接终止搜索。

(2)引入了对视频序列小运动类型的判定。判定为小运动类型后, 无须再进行时空域矢量预测, 直接进入步骤 5。

(3)大中运动搜索前的矢量预测, 除上层块预测外增加了加权的时间域预测。

(4)考虑到自然图像序列在水平方向的运动要比垂直方向的运动更普遍, 通过非对称十字形就可比较准确地预测到最佳运动矢量。实验结果表明, 采用非对称的十字形搜索, 搜索点减少, 但精度变化可以忽略不计。对于一些有大垂直运动的特殊序列, 垂直的扩展六边形搜索能使之仍然保持较高

精度。

(5)在SUMHexagonS算法中,对中运动类型的精确定位方法沿用了UMHexagonS 中的方法,搜索中一直采用固定的六边形模板。本文采用不同搜索模式采用^[4]的方向性搜索模板,能自适应地对宏块或亚宏块进行搜索,加快搜索速度。具体的搜索模板分类如下:

1)在 16×16, 8×8 搜索时,采用方向性均匀六边形模板 Hexagon,如图 2(a)所示。

2)在 16×8, 8×4 搜索时,采用方向性水平六边形模板 HHexagon,如图 2(b)所示。

3)在 8×16, 4×8 搜索时,采用方向性垂直六边形模板 VHexagon,如图 2(c)所示。

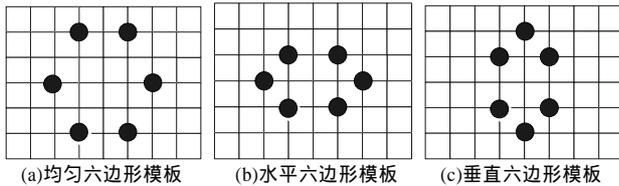


图 2 3 种六边形搜索模板

在 4×4 搜索时,直接进入步骤 5 进行 SDSP 搜索。

HHexagon 对水平运动显著的图像序列有着良好的搜索精度和速度,与之对应,VHexagon 对垂直运动的序列有着最优的性能。

5 实验结果与分析

为了验证算法改进的有效性,采用了 JVT 的参考软件 JM9.6 作为实验平台,并进行了大量的实验。

实验采用不同运动类型的标准视频序列:

- (1)剧烈运动序列: bus_cif, coastguard_qcif;
- (2)中运动序列: Froeman_cif, mobile_cif;
- (3)小运动序列: slient_qcif, mother_cif。

编码控制参数如表 1 所示。

表 1 仿真 JM 的编码控制参数

Name	Value	Name	Value
ProfileIDC	66	LevelIDC	30
UseHadamard	ON	SearchRange	32
NumberReferenceFrames	3	SymbolMode	CAVLC
RDOptimization	1	FramesToBeEncoded	30
FrameRate	30	FrameSkip	0
QP	28	GOP structure	IPPP

笔者分别利用上述两种算法对不同视频序列的前 30 帧进行编码比较。表 2 给出了视频编码耗时和运动估计耗时的比较。不难看出,改进的算法能有效地降低运动估计时间和编码时间,尤其对于剧烈运动的序列效果更为明显,可减少 30% 以上的搜索时间。表 3 给出了两种算法编码的平均比特率比较。可以看出,由于快速算法预测精度有细微的降低,因此视频编码后的比特数一般都会增加,即快速的运动估计算法通常以提高码率为代价来大幅度地提高搜索速度的。但本文的改进算法进行编码时,产生的比特率变化很小,在实际视频编码、传输中,其是基本不会造成影响的。表 4 给出了使用两种算法后的平均峰值信噪比的比较。可以看出,采用改进的算法,PSNR 的变化可以忽略不计,甚至对于一些方向性运动比较强的序列,PSNR 有一定程度的增加。这说明,改进的算法能较好地保持原算法率失真性能,对图像质量影响很小。

表 2 编码 30 帧图像的编码耗时和运动估计耗时比较

测试序列	Improved SUMHS		SUMHS		耗时比例	
	编码时间/s	运动估计时间/s	编码时间/s	运动估计时间/s	编码/(%)	运动估计/(%)
Silent_qcif	70.765	2.329	72.455	2.558	-2.33	-8.95
Mother_cif	203.126	8.563	212.017	9.418	-4.20	-9.07
Foreman_cif	250.111	12.634	256.657	14.133	-2.33	-10.61
Mobile_cif	441.456	11.005	472.724	17.598	-6.15	-37.46
coastguard_qcif	79.136	3.056	82.419	3.759	-3.98	-18.70
Flower	397.551	11.678	415.332	16.905	-4.28	-30.92

表 3 图像帧间编码的平均比特率比较

测试序列	Improved SUMHS/(Kb·s ⁻¹)	SUMHS/(Kb·s ⁻¹)	耗时比例/(%)
Silent_qcif	114.32	113.92	+0.350
Mother_cif	150.54	150.59	-0.027
Foreman_cif	462.10	460.19	-0.020
Mobile_cif	1 925.69	1 920.40	+0.275
coastguard_qcif	232.90	232.37	+0.228
Flower	1 681.12	1 681.11	0.000

表 4 图像亮度分量的平均峰值信噪比比较

测试序列	PSNR/dB	Improved SUMHS/(Kb·s ⁻¹)	SUMHS/(Kb·s ⁻¹)	耗时比例/(%)
Silent_qcif	Y	36.15	36.15	0.000 0
	U	39.05	39.06	-0.025 0
	V	40.04	40.07	-0.074 0
Mother_cif	Y	39.35	39.34	+0.025 0
	U	43.47	43.47	0.000 0
	V	44.48	44.47	+0.022 0
Foreman_cif	Y	36.98	36.98	0.000 0
	U	40.87	40.87	0.000 0
	V	43.78	43.78	0.000 0
Mobile_cif	Y	34.30	34.30	0.000 0
	U	36.51	36.51	0.000 0
	V	36.30	36.29	+0.028 0
coastguard_qcif	Y	34.18	34.18	0.000 0
	U	44.17	44.18	-0.022 6
	V	44.97	44.99	-0.044 0
flower	Y	35.31	35.32	-0.028 0
	U	37.26	37.27	-0.027 0
	V	38.36	38.37	-0.026 0

6 结束语

本文对 H.264 近期所采用的“简化 SUMHexagonS 算法”进行了研究和分析,提出了改进方法。改进算法充分考虑视频序列时域和空域的相关性,利用可靠的预测技术和早期停止判断标准,加快对静止或准静止块的搜索,大大减少了不必要的搜索点数。同时,该算法根据运动矢量分布的方向性,使用具有方向性的六边形搜索模型进行搜索,加快了精细搜索速度。

参考文献

- [1] Chen Zhibo, Zhou Peng, He Yun. Fast Integer and Fractional Pel Motion Estimation for JVT[C]//Proc. of the 6th JVT-F017 Conference. Awaji, Japan: [s. n.], 2002.
- [2] Yi Xiaoquan, Zhang Jun. Improved and Simplified Fast Motion Estimation for JM[C]//Proc. of the 16th Conference on ISO/IEC MPEG & ITU-T VCEG. Poznan, Poland: [s. n.], 2005.
- [3] Tourapis1 A M, Oscar C. Predictive Motion Vector Field Adaptive Search Technique——Enhancing Block Based Motion Estimation Visual Communications and Image Processing[C]//Proc. of 2001 Visual Communications and Image Processing Conference. San Jose, CA: [s. n.], 2001.
- [4] Cheung C H. Novel Cross-diamond-hexagonal Search Algorithms for Fast Block Motion Estimation[J]. IEEE Transaction on Multimedia, 2005, 7(1): 16-22.