

一种利用预测运动矢量改进的 H. 264 快速运动估计算法

郭继昌 邱琳耀 张 雪

(天津大学电子信息工程学院, 天津, 300072)

摘 要: 针对 H. 264 标准推荐使用的快速运动估计算法——非对称十字型多层次六边形格点搜索 (Unsymmetrical cross multi-hexagon grid search, UMHexagonS) 算法搜索速度慢的问题, 提出了一种改进算法。在起始搜索点的预测环节, 建立预测矢量集, 并根据预测矢量集的长度信息决定后续的搜索策略; 在全局搜索环节, 利用预测运动矢量之间的相关性适当跳过某些搜索步骤, 并更换一些搜索模板; 并且, 根据整数变换和量化的特性检测全零系数块, 提前终止运动估计过程。实验结果表明, 在量化步长为 28 时, 本文算法比 UMHexagonS 算法平均减少了 34.80% 的运动估计时间, 同时编码性能基本不变。该算法在不同量化步长的条件下能够适应不同运动强度的视频序列, 是一种适合 H. 264 的速度快且性能好的快速运动估计算法。

关键词: 视频编码; 运动估计; 预测运动矢量; H. 264; 非对称十字型多层次六边形格点搜索

中图分类号: TN919.81; TP301.6 **文献标志码:** A

Improved Method of H. 264 Fast Motion Estimation Using Predictive Motion Vector

Guo Jichang, Qiu Linyao, Zhang Xue

(School of Electronic Information Engineering, Tianjin University, Tianjin, 300072, China)

Abstract: An improved algorithm is proposed based on unsymmetrical cross multi-hexagon grid search (UMHexagonS), which is a fast motion estimation algorithm recommended by H. 264. In the prediction link for initial search points, predictive vector sets are established, and the follow-up search strategy is determined according to the length of prediction vector sets. In the global search link, the correlation between prediction motion vectors is used to skip some search steps, and some templates are replaced. In addition, the block of zero coefficients is detected to terminate the motion estimation process in advance, according to characteristics of integer transform and quantization. Experimental results show that, when the quantization step size is 28, the proposed algorithm reduces the motion estimation time by 34.80% compared with the UMHexagonS algorithm, while maintaining the performance. Finally, the algorithm can adapt to video sequences with different motion intensity under different quantization steps, which is a fast motion estimation algorithm with fast speed and good performance suiting for H. 264.

Key words: video coding; motion estimation; predicted motion vector (PMV); H. 264; unsymmetrical cross multi-hexagon grid search

引言

运动估计是 H.264 编码器中最耗时的环节之一。块匹配法由于算法简单、易于硬件实现,成为应用最广泛的运动估计算法。在块匹配运动估计算法中,传统的全搜索法通过检查搜索范围内的每一个点,可以得到最佳运动矢量。全搜索法能够获得图像质量上的高性能,但是计算复杂度过高。采用全搜索法的运动估计过程占据了 H.264 编码器超过 50% 的编码时间^[1],为了解决编码器对计算能力和内存容量要求非常高的问题^[2],许多快速搜索算法被提出。这些快速搜索算法大体可以分为几类,第 1 类采用特定的搜索模板,其中比较有代表性的算法是菱形搜索算法、六边形搜索算法等。由于搜索模板较为简单,这类算法对于运动缓慢的视频序列能够取得比较好的效果,但其没有从全局考虑运动方向,容易陷入局部最优化。第 2 类快速搜索算法采用动态的搜索范围,其主要思想是根据当前块的运动状态自适应地选择搜索窗口的尺寸^[3]。第 3 类快速搜索算法采用提前终止策略,提前终止策略有很多,包括零运动矢量检测、最佳运动矢量检测以及全零系数块的检测等^[4-5]。近年来,基于宏块时域和空域上的特性,一些放宽匹配条件的算法涌现出来,比如:逐次消元算法^[6]和局部失真搜索算法^[7]等。这些算法能够在保持图像质量的同时降低运动估计的运算量,但是硬件实现较为困难。

目前 H.264 标准推荐采用的快速运动估计算法是非对称十字型多层次六边形格点搜索(Unsymmetrical cross multi hexagon grid search, UMHexagonS)算法^[8]。该算法是一种混合编码技术,相对全搜索法可节约 90% 的运算量,大大降低了计算复杂度,同时能够保持较好的编码效率和图像质量,比较好地兼顾了编码性能和编码速度的统一。但是 UMHexagonS 算法在全局搜索、提前终止策略等方面仍然存在一定的问题,因此针对 UMHexagonS 算法的改进成为近年来的研究热点^[9-10]。本文以 UMHexagonS 算法为基础,利用预测运动矢量与最佳运动矢量的高度相关性,在起始搜索点的预测、全局搜索以及全零系数块的检测 3 个方面进行改进。

1 UMHexagonS 算法

UMHexagonS 算法是联合视频专家组(Joint video team, JVT)提供的 JM 模型中推荐使用的运动估计算法,其采用多种预测方式进行起始搜索点的预测,并基于运动对象的特点,混合采用多种模板进行运动矢量的搜索。

UMHexagonS 算法的第 1 步为起始搜索点的预测,采用 5 种方式预测起始的搜索中心。在这 5 种预测方式中,既有利用运动矢量时间相关性的预测方式(前帧对应块预测、相邻参考帧预测),也有利用运动矢量空间相关性的预测方式(中值预测、上层块预测),还有利用运动矢量中心偏移特性的预测方式((0,0)预测)。但是该算法并没有将这 3 种特性有效地结合起来,把这个标记为问题 1。

在精确地找到起始搜索点之后,UMHexagonS 算法进行全局搜索。全局搜索依次采用:(1)非对称十字型模板搜索;(2)5×5 螺旋方形模板搜索;(3)多层次大六边形模板搜索。在全局搜索完成之后,UMHexagonS 算法以得到的全局最优点为中心,进行局部搜索。局部搜索依次采用:(1)小六边形模板搜索;(2)小菱形模板搜索。全局搜索和局部搜索的模板如图 1 所示。UMHexagonS 算法的全局搜索过程虽然很好地避免了局部最优化的问题,但是仍然会进行大量不必要的搜索,把这个标记为问题 2。

另外,UMHexagonS 算法还在多处进行了提前终止检测,具体来说就是设定阈值,将当前搜索点的拉格朗日(Lagrangian)编码代价与阈值相比较,如果小于阈值就跳过某些搜索步骤。具体的提前终止策略如下:

如果 $J < Th1$,则进入小菱形搜索;如果 $Th1 \leq J < Th2$,则进入小六边形模板搜索。其中 J 为前搜索点所在块的 Lagrangian 编码代价, $Th1$, $Th2$ 设定的阈值,阈值的设定与当前块大小和量化步长的大小有关。

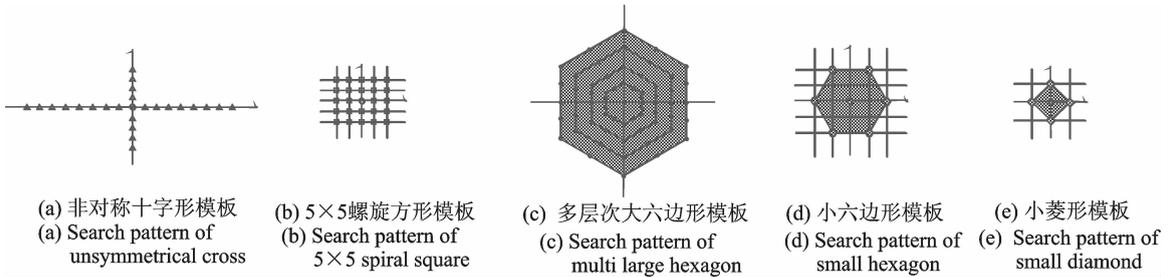


图1 UMHexagonS算法的搜索模板

Fig. 1 Search patterns of UMHexagonS algorithm

UMHexagonS算法虽然采用了提前终止策略,但是没有充分利用整数变换和量化的特点、预测 MV 处绝对之和(Sum of absolute difference, SAD)的分布特性以及率失真判决的条件,把这个标记为问题3。

2 UMHexagonS 算法改进

2.1 预测矢量集

针对前文分析中提出的问题1,本文通过建立预测矢量集的方法将预测运动矢量的时间相关性、空间相关性以及中心偏移特性有效地结合起来。从空间、时间等维度选取预测运动矢量,建立预测矢量集的思想可以取得较好的预测效果^[1]。

具体思路为:在起始搜索点预测之后,为当前块建立一个支持区域上的预测矢量集 $\{MV_1, MV_2, MV_3\}$,根据预测矢量集的长度信息决定后续搜索策略, MV_1 从空间维度选取(中值 MV), MV_2 从时间维度选取(前帧对应块 MV), MV_3 为5个预测 MV 中最优的预测 MV (其包含 $(0,0)MV$),这3个预测 MV 构成的集合充分利用了运动矢量的时空特性和偏移特性,是预测最佳运动矢量比较有效的的基本预测矢量组合。计算预测矢量集中 MV 的最大长度 L ,长度信息可由式(1~3)得到

$$MV_i = (x_i, y_i) \quad i=1,2,3 \quad (1)$$

$$l_i = |x_i| + |y_i| \quad i=1,2,3 \quad (2)$$

$$L = \max(l_1, l_2, l_3) \quad (3)$$

条件1 如果 $L \leq 1$,表明当前编码块的运动状态缓慢,则跳过全局搜索,直接进入小菱形搜索;

条件2 如果 $1 < L \leq 2$,表明当前编码块的运动状态适中,则依次进行大菱形搜索和小菱形搜索;

条件3 如果 $L > 2$,表明当前编码块处于快速运动状态,则进入全局搜索。小菱形模板和大菱形模板如图2所示。

2.2 自适应的全局搜索

针对前文分析中提出的问题2,本文采用自适应的全局搜索方法进行改进。一般来说,相邻帧之间的相关性很大,这意味着精确的起始搜索点很有可能就是全局最优点,即使不是全局最优点,那么也有很大可能落入全局最优点的附近。因此在大部分情况下,全局搜索环节只需要进行 5×5 螺旋方形模板搜索。记准确度最高的两个预测 MV 的水平分量差和垂直分量差中的最大值为 ΔMV ,当 $\Delta MV \leq 3$ 时,可以认为这两个预测 MV 的差值非常小、起始搜索点的预测非常精确^[12]。在这种条件下,

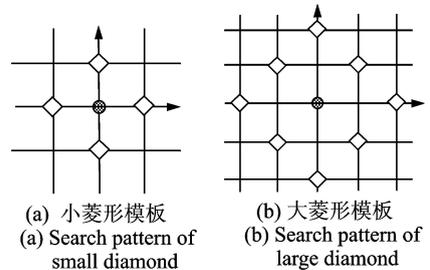


图2 小菱形模板和大菱形模板

Fig. 2 Search pattern of small diamond and large diamond

可跳过非对称十字型模板搜索和多层次大六边形模板搜索,只进行 5×5 螺旋方形搜索。 $\Delta \mathbf{MV}$ 由式(4)计算

$$\Delta \mathbf{MV} = \max(\text{abs}(\mathbf{MV}_{1x} - \mathbf{MV}_{2x}), \text{abs}(\mathbf{MV}_{1y} - \mathbf{MV}_{2y})), \quad (4)$$

式中: \mathbf{MV}_1 和 \mathbf{MV}_2 分别为准确度最高的两个预测 \mathbf{MV} , x 为 \mathbf{MV} 的水平方向, y 为 \mathbf{MV} 的竖直方向。

在 5×5 螺旋方形搜索中,模板中每个点成为最佳点的概率并不相同^[13],因此本文采用改进的螺旋方形模板搜索:对 16×16 块取消搜索,对 4×4 块采用 5×5 螺旋方形模板搜索,其余尺寸的块均采用 3×3 螺旋方形模板搜索。

另外,本文根据当前编码块的运动状态为全局搜索过程自适应地选择搜索窗尺寸。搜索窗尺寸(Search range, SR)由式(5)计算^[14]

$$\text{SR} = \left[\frac{\text{SAD}_{\text{initial}}}{n_1 \times n_2} \left(\frac{a}{P_{\text{max}}} + b \right) \right] + c \quad (5)$$

式中: $\text{SAD}_{\text{initial}}$ 是起始搜索点的 SAD 值; n_1, n_2 分别为当前块的宽度和高度; P_{max} 表示配置文件中的搜索窗尺寸; a, b, c 均为常量,分别设为 1, 0.75 和 0; $[\cdot]$ 为取整操作,同时 SR 不能大于 P_{max} 。

2.3 全零系数块检测

针对在前文分析中所提出的问题 3,本文通过提前判定全零系数块来终止没有意义的运动搜索过程。全零系数块是指经过变换和量化之后系数全部变为零的块,对于全零系数块做运动估计不会提高编码效率。因此,如果在运动估计时检测到全零系数块,那么就没有必要搜索更精确的匹配块。

H.264 标准采用基于拉格朗日优化算法的率失真优化技术实现对视频编码的控制。率失真优化的关键就是不断计算每个 \mathbf{MV} 对应的 Lagrangian 编码代价,然后选出代价最小的 \mathbf{MV} 。Lagrangian 编码代价定义为

$$J(M, \lambda) = \text{SAD} + \lambda \times R(\mathbf{MV} - \mathbf{PMV}) \quad (6)$$

式中: SAD 为当前块和候选块对应像素差的绝对值之和; λ 为 Lagrangian 常数; \mathbf{MV} 为当前块的运动矢量; \mathbf{PMV} 为预测运动矢量。

H.264 标准采用 4×4 整数变换对帧内和帧间预测的差值块进行编码^[15],经过整数变换和量化后 4×4 块内所有系数全部变为零的充分条件为

$$\sum_{m=0}^3 \sum_{n=0}^3 |x_{mn}| < \frac{25QP}{12} \quad (7)$$

式中: x_{mn} 为差值块的系数, QP 为量化步长。式(7)左边是运动估计时 4×4 块的 SAD 值,右边是全零块判决的阈值,如果当前块内所有的 4×4 块都满足式(7),说明当前块经过变换和量化后系数全为零,对当前块进行运动估计没有意义。由式(6)可知,如果在预测 \mathbf{MV} 处进行全零系数块的判决,这也是 $J(M, \lambda)$ 上的最优判决。

因此,本文在中值预测结束、得到中值 \mathbf{PMV} 之后,计算当前编码的差值块包含的所有 4×4 块的 SAD,并与阈值比较,如果 SAD 全部小于阈值,那么将 \mathbf{MV} 设成中值 \mathbf{PMV} ,结束运动估计;否则进行正常的运动搜索过程。式(7)中的阈值过小,为了加速运动搜索,经过大量的实验,在综合考虑编码性能和计算复杂度之后,在一定误判率的基础上,将全零块判决的阈值设定为 $3QP + 90$ ^[16]。

根据 2.1~2.3 节的描述,本文算法的流程如图 3 所示。

3 实验结果与性能分析

3.1 测试平台及其配置

实验计算机的配置如下:处理器为 Intel Core 2,主频为 2.93 GHz,内存为 2 GB,操作系统为 Windows

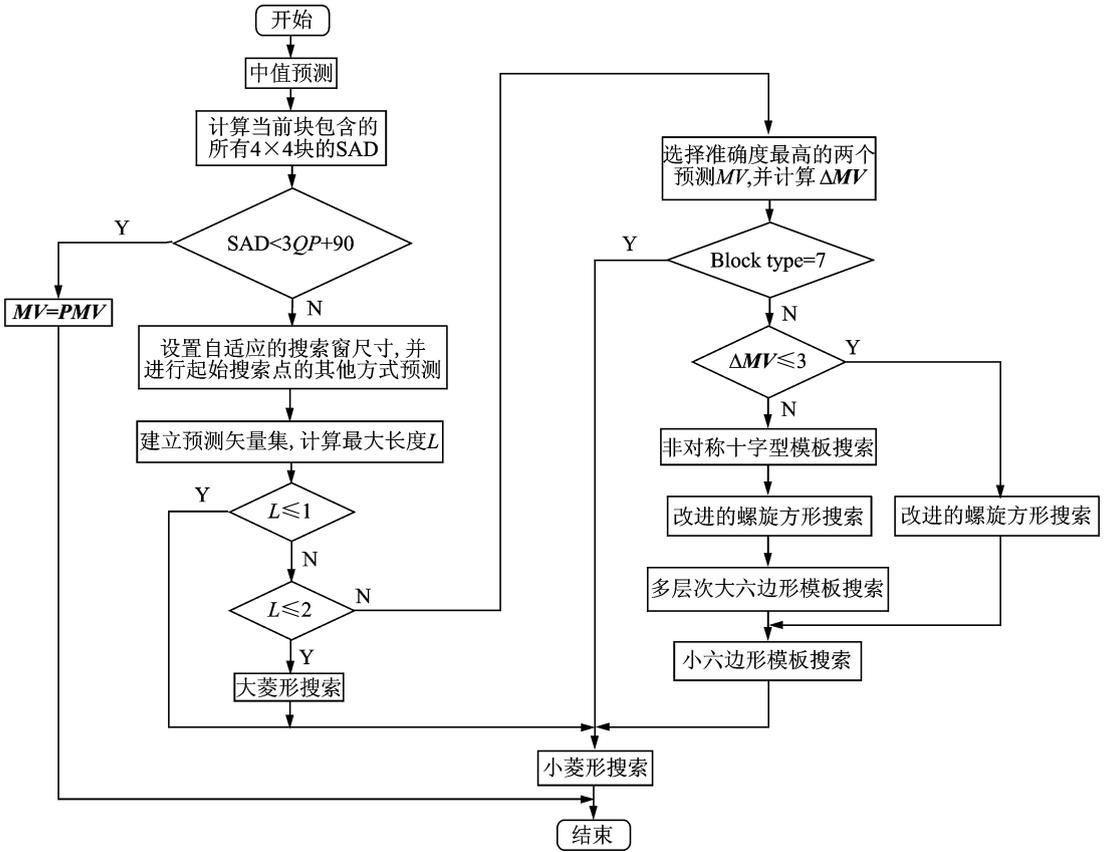


图3 改进算法的流程图

Fig. 3 Flow chart of improved algorithm

XP。实验软件平台为 Visual Studio 2010, 测试软件为 H. 264 标准的参考校验模型 JM10.2, 改进算法用 C 语言实现。编码器配置采用 JM10.2 的主类别, 主要编码参数设置如下: 编码 80 帧, 帧率为 30 fps, 使能哈达玛变换, 运动估计搜索半径为 16, 参考帧为 5 帧, 熵编码类型为 CABAC, 帧类型为 IBPBP, 其他参数为缺省设置。实验中选取了 News(QCIF), Foreman(QCIF), Coastguard(QCIF) 和 Waterfall(CIF) 4 个视频序列作为编码器的输入。其中 News 代表慢速运动序列, Foreman 代表中速运动序列, Coastguard 代表快速运动序列, Waterfall 代表垂直方向运动远大于水平方向运动的非常规运动序列。

3.2 测试结果与分析

全搜索法(Full search)简记为 FS, UMHexagonS 算法简记为 UMHS, 简化的 UMHexagonS(Simplified-UMHexagonS)算法简记为 S-UMHS, 本文改进算法命名为 IMP。在同一测试环境下, 对这 4 种算法进行测试, 结果如表 1 所示。表 1 中, QP(Quantization step)代表量化步长; PSNR(Power signal-to-noise ratio)代表峰值信噪比, BR(Bit rate)代表比特率, ENT(Encoding time)代表编码时间, MET(Motion estimation time)代表运动估计的时间。从表 1 可以看出, 无论对于慢速、中速和快速运动序列, 还是对于非常规运动序列, 本文算法相比 UMHS 算法均能够大幅度地减少编码时间和运动估计时间。

表 2 是本文算法相对于 FS 算法以及 S-UMHS 算法的变化情况。由表 2 可知, 相比 FS 算法, 本文算法在 PSNR 平均减少 0.022 dB 的情况下, 平均节省了 79.62% 的运动估计时间。相比 S-UMHS 算法, 本文算法除了在慢速运动序列上运动估计的时间有所增加外, 在其他序列上运动估计的时间都能取

得不同程度的降低。表 3 是本文算法相对 UMHS 算法的变化情况以及文献[9]算法相对 UMHS 算法的变化情况。由表 3 可知,相比 UMHS 算法,文献[9]算法平均减少了 23.16%的运动估计时间,本文算法平均节省了 34.80%的运动估计时间,优于文献[9]算法。另外,文献[9]算法相比 UMHS 算法在 PSNR 上的波动性较大(最大 0.05 dB),本文算法在这方面保持了较好的性能。

表 1 实验结果记录(QP=28)

Tab. 1 Experimental result records (QP=28)

测试序列	测试项	FS	UMHS	S-UMHS	IMP
News	PSNR/dB	37.89	37.89	37.88	37.88
	BR/(kbit/s)	179.58	179.46	179.47	179.45
	ENT/s	123.918	58.468	47.689	49.411
	MET/s	89.683	21.577	12.779	15.189
Fore man	PSNR/dB	37.05	37.04	37.04	37.03
	BR/(kbit/s)	190.18	189.76	189.90	189.31
	ENT/s	128.626	68.064	57.782	53.296
	MET/s	95.257	33.169	23.834	20.530
Coast guard	PSNR/dB	34.75	34.74	34.73	34.71
	BR/(kbit/s)	277.91	276.91	276.21	275.85
	ENT/s	152.999	73.862	63.062	59.390
	MET/s	110.956	38.013	25.916	23.274
Water fall	PSNR/dB	36.00	35.99	35.98	35.98
	BR/(kbit/s)	807.29	805.87	806.38	806.62
	ENT/s	498.987	260.640	227.920	218.467
	MET/s	357.310	117.018	85.389	78.753

表 2 本文算法与 FS,S-UMHS 算法的编码性能比较(QP=28)

Tab. 2 Comparison of coding performance between the proposed algorithm and FS,S-UMHS algorithms

测试序列	本文算法与 FS 算法比较结果				本文算法与 S-UMHS 算法比较结果			
	差值		变化率/%		差值		变化率/%	
	PSNR/dB	BR/(kbit · s ⁻¹)	BR	MET	PSNR/dB	BR/(kbit · s ⁻¹)	BR	MET
News	-0.01	-0.13	-0.072	-83.06	0	-0.02	-0.011	+18.86
Foreman	-0.02	-0.87	-0.457	-78.45	-0.01	-0.59	-0.311	-13.86
Coastguard	-0.04	-2.06	-0.741	-79.02	-0.02	-0.36	-0.130	-10.19
Waterfall	-0.02	-0.67	-0.083	-77.96	0	+0.24	+0.030	-7.77
Average	-0.022	-0.933	-0.338	-79.62	-0.008	-0.183	-0.106	-3.24

表 3 本文算法与文献[9]算法的编码性能比较(QP=28)

Tab. 3 Comparison of coding performance between the proposed algorithm and the algorithm in Ref. [9]

测试序列	本文算法与 UMHS 算法比较结果				文献[9]与 UMHS 算法比较结果			
	差值		变化率/%		差值		变化率/%	
	PSNR/dB	BR/(kbit · s ⁻¹)	BR	MET	PSNR/dB	BR/(kbit · s ⁻¹)	BR	MET
News	-0.01	-0.01	-0.006	-29.61	-0.04	-0.06	-0.226	-13.66
Foreman	-0.01	-0.45	-0.237	-38.10	+0.01	+0.77	+0.882	-20.07
Coastguard	-0.03	-1.06	-0.383	-38.77	+0.05	-0.13	-0.070	-27.72
Waterfall	-0.01	+0.75	+0.093	-32.70	+0.03	-8.35	-1.421	-31.20
Average	-0.015	-0.193	-0.133	-34.80	+0.013	-1.942	-0.209	-23.16

为了分析本文算法在不同序列上的效果,对 News, Foreman, Coastguard 三个序列在不同量化步长 (QP=24, 28, 32, 36 和 40) 下各编码 80 帧。在不同序列、不同量化步长的条件下,图 4 为相比 UMHS 算法,本文算法 IMP 在运动估计时间上的下降率。由图中可以看出,相比 UMHS 算法,本文算法针对运动强度不同的序列,能够减少不同比率的运动估计时间。具体来说,针对运动强度较大的视频序列(如 Coastguard),运动估计时间的下降率在量化步长从 24 至 36 的条件下均超过了 30%;而对于运动强度较小的视频序列(如 News),运动估计时间的下降率大致分布在 20%~30%之间。这是因为,针对运动强度较小的序列,各个预测 MV 的值以及各预测 MV 之间的差值均较小,而改进 1 和改进 2 主要是根据预测 MV 及其差值的大小做改进的;因此在运动强度较低的情况下,改进 1 和改进 2 所起的作用相对大一些。随着运动强度的增大,预测 MV 及其差值逐渐增大,满足阈值条件的概率不断下降,改进 1 和改进 2 的作用逐渐减小;相应地,改进 3 所起的作用也就越来越大。改进 3 中全零系数块的检测建立在一定误判率的基础之上。随着运动强度的增大,改进 3 所起的作用逐渐增大,在加速运动搜索的同时,全零系数块的误判率也随之增加,这是造成峰值信噪比下降的主要原因。同时,从表 1~3 等实验数据知,运动强度越大的序列,本文算法与 UMHS 算法的 PSNR 之差也越大,这也进一步验证了改进 3 设定的阈值建立在一定误判率的基础上,在加速运动搜索过程的同时造成了 PSNR 一定程度的损失。为了进一步比较 IMP 和 UMHS 算法的性能,选取运动强度较小的 News 序列和运动强度较大的 Coastguard 序列,对每个序列在不同量化步长 (QP=24, 28, 32, 36 和 40) 下各编码 80 帧,分别绘制这两个序列下两种算法的率失真(Rate-distortion, R-D)性能曲线图,如图 5, 6 所示。从图中可以看出,无论对于 News 序列,还是对于 Coastguard 序列,这两种算法的率失真性能曲线均相差无几,这表明改进算法在编码质量和编码效率方面损失很小。

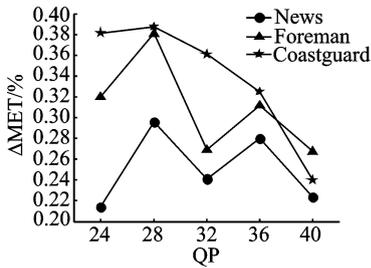


图 4 相比 UMHS, IMP 的运动估计时间下降率

Fig. 4 Comparison of UMHS, IMP decline rate of motion estimation time

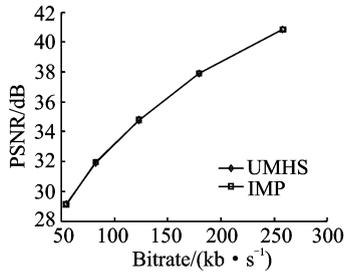


图 5 News 序列的率失真性能曲线

Fig. 5 Rate-distortion curve of News

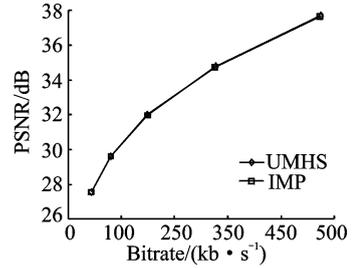


图 6 Coastguard 序列的率失真性能曲线

Fig. 6 Rate-distortion curve of Coastguard

4 结束语

在 H. 264 编码器中,运动估计是最耗时的环节之一,其计算复杂度随着编码效率的提高成倍增加,这使得编码器的实时性很难实现。针对这个问题,本文利用预测运动矢量与最佳运动矢量的高度相关性,在起始搜索点的预测、全局搜索和局部搜索等环节适当跳过一些搜索步骤,并更换一些搜索模板,以达到简化搜索的目的;另外,本文还根据整数变换和量化的特性检测全零系数块,提前终止没有意义的运动搜索过程。实验结果表明,在重建图像质量和码率接近的情况下,相比 UMHexagonS 算法,本文算法平均减少了 34.80% 的运动估计时间(量化步长为 28)。另外,在不同量化步长的条件下,本文算法针对不同运动强度的视频序列均能减少一定比率的运动估计时间,同时率失真性能与 UMHexagonS 算法相差无几,因此本文算法是一种适合 H. 264 的搜索效率高且编码损失小的快速算法。最后要说明的

是,本文改进 3——全零系数块的检测环节建立在一定误判率的基础上,在加速运动搜索的同时造成了 PSNR 一定程度的损失,值得进一步的研究。

参考文献:

- [1] Li L, Liu S, Chen Y, et al. Motion estimation without integer-pel search [J]. Image Processing, IEEE Transactions on, 2013, 22(4): 1340-1353.
- [2] Wiegand T, Sullivan G J, Bjøntegaard G, et al. Overview of the H.264/AVC video coding standard[J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2003, 13(7): 560-576.
- [3] Ko Y H, Kang H S, Lee S W. Adaptive search range motion estimation using neighboring motion vector differences [J]. Consumer Electronics, IEEE Transactions on, 2011, 57(2): 726-730.
- [4] Yang L B, Yu K M, Li J, et al. An effective variable block-size early termination algorithm for H.264 video coding [J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2005, 15(6):784-788.
- [5] Sarwer M G, Wu Q M J. Adaptive variable block-size early motion estimation termination algorithm for H.264/AVC coding standard [J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2009, 19(8): 1196-1201.
- [6] Choi C, Jeong J. Extended successive elimination algorithm for fast optimal block matching motion estimation [C] // The Sixth International Conferences on Advances in Multimedia. Nice, France: MMEDIA, 2014: 33-36.
- [7] Chen H M, Chen P H, Lin C T, et al. An adaptive macroblock-mean difference based sorting scheme for fast normalized partial distortion search motion estimation [J]. Computers & Electrical Engineering, 2013, 39(5): 1409-1421.
- [8] Chen Z, Xu J, He Y, et al. 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.
- [9] 李子印,杨齐. 基于运动信息自适应的快速运动估计算法 [J]. 中国图象图形学报,2012,17(9):1069-1074.
Li Ziyin, Yang Qi. Fast motion estimation algorithm based on motion information adaptation [J]. Journal of Image and Graphics, 2012, 17(9): 1069-1074.
- [10] 刘英哲,王进祥. H.264 中一种基于搜索范围自适应调整的运动估计算法 [J]. 电子与信息学报,2013,35(6):1382-1387.
Liu Yingzhe, Wang Jinxiang. Motion estimation algorithm based on adaptive search range adjustment for H.264 [J]. Journal of Electronics & Information Technology, 2013, 35(6): 1382-1387.
- [11] Zeng H, Cai C, Ma K K. Fast mode decision for H.264/AVC based on macroblock motion activity [J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2009, 19(4): 491-499.
- [12] 吴银花,金龙旭,张宁,等. 针对 H.264 改进的快速整像素运动估计算法 [J]. 光学精密工程,2013,21(4):1017-1025.
Wu Yinhu, Jin Longxu, Zhang Ning, et al. Improvement of fast integer pixel motion estimation algorithm for H.264 [J]. Optics and Precision Engineering, 2013, 21(4): 1017-1025.
- [13] 丁燕,宋雪桦,闫述,等. 基于快速运动估计 UMHexagonS 算法的改进 [J]. 数据采集与处理,2009,24(5):660-663.
Ding Yan, Song Xuehua, Yan Shu, et al. Improvements of UMHexagonS algorithm based on fast motion estimation [J]. Journal of Data Acquisition and Processing, 2009, 24(5): 660-663.
- [14] Ismail Y, McNeely J B, Shaaban M, et al. Fast motion estimation system using dynamic models for H.264/AVC video coding [J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2012, 22(1): 28-42.
- [15] 周欣,段哲民,周巍. 一种适用于 H.264/AVC 的新型整数变换与量化算法 [J]. 数据采集与处理,2011,26(6):619-625.
Zhou Xin, Duan Zhemin, Zhou Wei. A new integer transform and quantization algorithm for H.264/AVC[J]. Journal of Data Acquisition and Processing, 2011, 26(6): 619-625.
- [16] 方健,郑伟,王匡. 针对 H.264 的自适应提前终止搜索算法 [J]. 浙江大学学报(工学版),2007,41(4):411-415.
Fang Jian, Zheng Wei, Wang Kuang. ATETS: Adaptive early termination search algorithm for H.264 [J]. Journal of Zhejiang University(Engineering Science), 2007, 41(4): 411-415.

作者简介:



郭继昌(1966-),男,教授、博士生导师,研究方向:数字图像处理、滤波器理论及设计,E-mail:jcguo@tju.edu.cn。



邱琳耀(1992-),男,硕士研究生,研究方向:视频编解码,E-mail:qiulinyao88@163.com。



张雪(1990-),女,硕士研究生,研究方向:音频处理,E-mail:snow_zhang201209@163.com。

