

一种基于概率矩阵的快速运动估计算法

乐培玉 张太镒 王洁

(西安交通大学电子与信息工程学院, 西安 710049)

摘 要 基于块匹配的运动向量估计算法已被多种国际视频编码标准所采用,但其计算复杂度一直是一个研究热点。为了提高运动向量估计算法的速度和精度,提出了一种新的基于概率矩阵的快速块匹配运动估计算法,该算法首先根据之前宏块的运动向量来估计当前宏块各可能的运动向量对应的概率值,以组成和搜索窗口同样大小的概率矩阵,然后依据概率大小限制搜索的次数,以平衡算法的速度和精度。仿真实验结果表明,和标准菱形搜索法相比,该算法在精度略有提高的同时,还有效地提高了搜索效率。

关键词 视频压缩 运动估计 块匹配算法 概率矩阵

中图分类号: TN919.81 **文献标识码:** A **文章编号:** 1006-8961(2007)01-0032-05

A Fast Motion Estimation Algorithm Based on Probability Matrix

YUE Pei-yu, ZHANG Tai-yi, WANG Jie

(School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049)

Abstract The complexity of video encoders in the international standards of H. 263, MPEG-4, H. 264, depends mainly on the block motion estimation algorithm. In order to make motion estimation algorithm faster and more accurate, a fast motion estimation algorithm based on probability matrix is proposed in the paper. In the algorithm, every possible motion vector's probability of current macro block is estimated by the previous macro blocks' motion vectors, and constitutes the probability matrix whose size is as the same as the search window's. The algorithm restricts the number of pixels to search by probability. From the results of experiment, comparing the proposed algorithm and diamond search algorithm, it shows that the veracity is improved a little, and the speed grown faster at the same time.

Keywords video compression, motion estimation, block matching algorithm, probability matrix

1 引 言

由于视频图像序列存在着很大的时间相关性,而运动估计(motion estimation, ME)及运动补偿(motion compensation, MC)技术将有效地减小这些时间冗余,因此被广泛地应用于各种视频编码标准中,例如 MPEG-1、MPEG-2^[1]、MPEG-4^[2]、H. 261、H. 263 等。

运动估计方法有块匹配算法(block matching algorithm, BMA)和像素递归算法(pixel recursive algorithm, PRA)。其中块匹配算法是将图像划分为

许多互不相叠的子块,并认为子块内所有像素的位移都相同,且只做平移运动,其块匹配的准则主要有绝对误差和(sum absolute deviation, SAD)、平均绝对误差(mean absolute deviation, MAD)、互相关系数(cross correlation parameter, CCP)、均方误差(mean square error, MSE)和最小均方误差(minimum mean error, MME)等,其中 SAD 准则计算量最少,目前大部分快速块匹配算法都使用 SAD 准则。全搜索(full search, FS)是块匹配算法中应用最广泛的一种技术,它是对给定的搜索窗口内的所有点进行搜索。它的优点是产生的残差系数最小,但由于其巨大的时间开销和计算量是实时视频编码系统所不能接受的,为此人

收稿日期:2005-07-20; 改回日期:2005-12-20

第一作者简介:乐培玉(1981~),男,2006年于西安交通大学获通信与信息系统专业硕士学位。主要研究方向为数字图像处理、音视频编解码等。E-mail: cnstarlg@163.com

们提出了很多快速搜索算法,如:三步搜索法(three step search, TSS)^[3]、菱形搜索法(diamond search, DS)^[4]、四步搜索法(four step search, 4SS)^[5]、2 维对数搜索法(log search, LOGS)^[6]等等。由于这些算法减少了运动估计的计算量,因此可得到近似于全搜索算法的运动向量。其中 DS 算法由于其综合性能优越而被 MPEG4 国际标准采纳,并收入验证模型(verification model, VM)^[7]。

上述快速搜索算法实施的措施有:①通过限制搜索点的个数来降低 BMA 的复杂度的方法,如三步搜索法、四步搜索法、新三步搜索法等;②运用时空相关性,从得到的运动向量选择块匹配的初始点的方法,如基于起始点预测的快速运动估计算法。这些算法各有各的优势,又有各自的不足。如果将限制搜索点的思想与时空相关性联系起来,则算法将更加优越。为此,本文提出了一种基于概率矩阵的快速运动估计算法。仿真结果表明,与 DS 算法相比,该算法不仅复杂度明显降低,而且精度还略有提高。

2 本文算法

众所周知,视频图像序列中某一需要进行运动估计的宏块或块,其可能出现的运动向量有若干个,每一个可能出现的运动向量对应一个相应的概率。如果知道每个可能出现的运动向量的先验概率,那么就可以取概率较大的若干向量组成一个集合,称为待匹配运动向量集。若对集合中的向量逐一进行匹配,则 SAD 值最小的向量即为运动估计的结果。如果集合中的运动向量越多,则运动估计越准确。当集合中的向量为所有可能的运动向量时,则运动估计的结果与 FS 算法的结果一致。

概率矩阵 P 是指用来表示每个运动向量的概率的矩阵。当搜索窗的半径为 r 时,概率矩阵为 $(2r+1)^2$ 阶。

$$P = [P_{u,v}], \quad u, v = 0, 1, \dots, 2r$$

概率 $P_{i,j}$ 和运动向量 $V^{\text{motion}} = (V_i^{\text{motion}}, V_j^{\text{motion}})$ 的对应关系为

$$u = i + r; \quad v = j + r$$

例如当 $r=15$ 时,则概率矩阵中第 15 行,15 列的元素就表示运动向量为(0,0)的概率。

由于宏块或块的运动向量的先验概率是无法事先知道,因此首先要对运动向量的先验概率进行估

计。宏块或块的运动向量在时间和空间上有很大的相关性。表 1^[7]给出了当前宏块与相邻宏块以及前一帧相邻宏块之间的相关性。用这种相关性,即可以从已得到的运动向量中估计当前的运动向量。

表 1 宏块运动向量之间的时空相关性

Tab. 1 Space-time relativity between motion vectors of two consecutive macro blocks

t 帧与 t-1 帧的时间相关系数			
水平运动向量	0.140	0.161	0.141
垂直运动向量	0.168	0.186	0.168
宏块	M	I	N
水平运动向量	0.151	0.166	0.160
垂直运动向量	0.186	0.218	0.201
宏块	H	G	J
水平运动向量	0.148	0.179	0.163
垂直运动向量	0.146	0.166	0.159
宏块	L	K	O
空间相关系数			
水平运动向量	0.385	0.520	0.426
垂直运动向量	0.395	0.521	0.377
宏块	C	B	D
水平运动向量	0.488		
垂直运动向量	0.487		
宏块	A		

根据表 1 中的时间相关性来统计 $t-1$ 帧的运动向量,就可以估计得到 t 帧的运动向量的概率分布,但这种方式需要对每个宏块单独统计一次,复杂度非常大。当 r 较大的时候,由于它的计算量甚至可以超过运动估计的计算量,非常不实用,因此本文假设帧内宏块的运动向量基本相同,即前景和背景的运动相差不大。在这样的假设下, $t-1$ 帧各宏块的运动向量与 t 帧各宏块运动向量的相关性相同,于是可以通过简单的对 $t-1$ 帧进行统计来得到 t 帧运动向量的概率分布。如果用 $P(t)$ 表示为 t 帧的概率矩阵,用 P_0 表示第 $t-1$ 帧的统计结果,则

$$P(t) = P(t-1) \times 0.3 + P_0 \times 0.7$$

其中,0.7 和 0.3 是由经验获得。

根据表 1 中的时间相关性,就可以通过对 $P(t)$ 进行修正来得到第 t 帧第 n 个宏块的概率分布。本文中是通过 A, B, D 这 3 个宏块的运动向量来修正的。本文中修正的方法是将 $P(t)$ 中 A, B, D 宏块运

动向量的对应概率值加 0.1。如果用 $P(t, n)$ 表示第 t 帧第 n 宏块的概率矩阵, $\hat{P}(n)$ 表示第 n 个宏块的修正概率矩阵, 那么

$$P(t, n) = P(t) + \hat{P}(n)$$

其中, 0.1 为经验值。

有了 $P(t, n)$, 就可以对第 n 个宏块进行搜索了。为了降低运算复杂度, 不可能对每个运动向量进行匹配计算, 本文依据概率大小来限制搜索的次数。搜索时, 首先将 $P(t, n)$ 的各元素按概率大小顺序排列, 并按一定准则取若干点, 例如可取前 10 个元素, 也可取前 m 个元素, 且满足 $\sum P_{u,v} < T_p$ (下角 P 代表 probability), 本文按后者取, 且 $T_p = 0.97$; 然后按顺序对各运动向量进行匹配, 以找到 SAD 最小的运动向量。这样搜索精度固然高, 但每个宏块都需要搜索 m 个点, 特别是已经得到很小的 SAD 时, 再往下搜索就显得很不明智。为了加快搜索速度, 可设定一个阈值 T_{SAD} , 搜索过程中一旦 SAD 小于阈值 T_{SAD} , 即可以跳出搜索。

由于一般序列无法完全满足该假设条件, 故上面得到的 SAD 最小的运动向量可能不是最优的, 甚至也不是局部最优的, 但是由于它与全局最优点非常接近, 因此, 可以先使用 DS 算法中的小菱形模板搜索一次, 再通过对上述运动向量进行修正来得到最优的运动向量。综上所述, 其具体步骤如下:

- (1) 编码开始, 第 1 帧为 I 帧, 不使用运动估计;
- (2) 根据上一帧运动向量的统计特性 P_0 来估计本帧的运动向量概率分布矩阵

$$P(t) = P(t-1) \times 0.3 + P_0 \times 0.7$$

- (3) 根据第 n 个宏块的左边宏块、上面宏块和右上的运动向量估计本宏块运动向量的分布概率矩阵

$$P(t, n) = P(t) + P_k(n)$$

- (4) 在 $P(t, n)$ 中选取概率值较大的 m 个运动向量, 使它们的概率之和小于阈值, 即

$$\sum P_{u,v} < T_p$$

- (5) 计算 m 个运动向量的 SAD 值, 然后取 SAD 最小的运动向量为起始点, 并用小菱形模板搜索一次来得到最终的运动估计结果;

- (6) 重复步骤 3 ~ 步骤 5, 直到本帧所有宏块编码完成, 并统计本帧运动向量的概率分布 P_0 ;

- (7) 重复步骤 2 ~ 步骤 6, 直到所有帧编码完成。基于概率矩阵算法的流程图如图 1 所示。

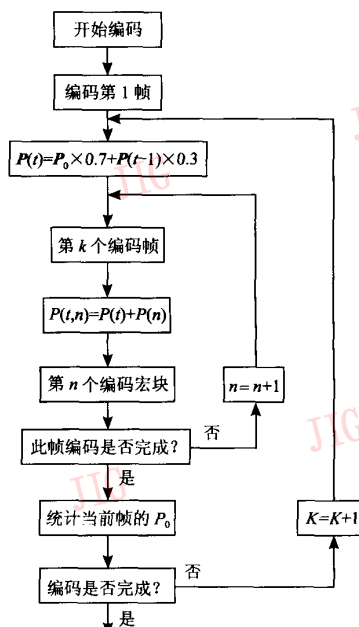


图 1 算法流程示意图

Fig. 1 Flow chart of the algorithm

和标准菱形算法一样, 本文算法也是通过限制搜索点的个数来减少搜索的次数。不同的是, 标准菱形算法是建立在运动估计预测误差单峰假设的基础上。一半情况下, 这个假设是成立的, 然而当假设不成立时, 标准菱形算法很容易陷入局部极小点。由于本文算法是依据序列内容的特点来估计概率矩阵, 并根据概率值大小选定搜索点, 搜索点可以在搜索窗的任何位置, 因此降低了陷入局部极小点的可能。

标准菱形算法, 以及三步搜索算法、四步搜索算法的共同特点是有各自固定的搜索形状, 即搜索模板。如果此类算法中搜索模板过大, 则对于小运动矢量来说, 会造成搜索次数增加; 若搜索模板过小, 则对于大运动矢量来说, 因搜索步长太小, 不能及时跟踪运动矢量, 同样会造成搜索次数增加。由于标准菱形算法同时使用了大菱形模板和小菱形模板, 且对于较大的运动矢量和较小的运动矢量分别对待, 从而减少了搜索次数。然而在极端情况下, 例如当运动矢量为 0 时, 标准菱形算法需要搜索 9 个点; 又如果 SAD 最小点在搜索窗口的边缘, 那么标准菱形算法的大模板对于大的运动矢量的跟踪还是比较迟钝。本文算法是先对概率矩阵进行估计, 并在此基础上选择若干点进行搜索。当运动矢量很小的时候, 概率矩阵分布集中, 例如对于运动矢量为 0 的

点,一般只需要计算 5 个点。当运动矢量很大的时候,概率分布分散,此时选出来的点也比较分散,可使得搜索次数减少。

综上所述,由于本文算法根据编码序列的实际特点,选取运动估计的搜索点,因此可以在运动估计精度略有提高的基础上,改进搜索效率。

3 实验仿真

为了比较算法的优劣,下面对 3 种算法(全搜索算法,标准菱形算法和本文算法)进行了对比实验。实验使用 Visual C++ 6.0 编程实现,在 MPEG2 的 MP@LL 下进行,搜索范围为 15 × 15,采用 SAD 匹配准则,色度格式为 4:2:0,帧频为 25fps,压缩码率为 1.152Mbps, GOP 长度为 12, I 帧和 P 帧距离为 1。

实验从以下 3 个方面考察算法的有效性:

- (1) 复杂度,即平均每个宏块搜索的次数;
- (2) 平均峰值信噪比 (peak signal noise ratio, PSNR);

(3) 准确性,即相对于 FS 算法的平均运动矢量偏移距离。

实验选取 5 个 Qcif 格式的标准测试序列的前 100 帧:News, Trevor, Suzie, Salesman 和 Foreman。其中 News 序列和 Salesman 序列都是背景和前景几乎相对静止的,符合本算法的假设条件;而 Trevor、Suzie 和 Forman 序列的前景则运动较大,背景几乎不动,不符合本算法的假设条件。

3.1 复杂度对比

复杂度可以用平均每个宏块的搜索次数来衡量,即搜索次数越少,计算的复杂度就越小,且运动估计花费的时间也越少。表 2 给出了不同序列不同搜索算法每宏块的平均搜索次数。

表 2 各算法下每个宏块平均搜索次数

Tab.2 Average search time with each algorithm for one block

算法	每个宏块平均搜索次数				
	News	Trevor	Suzie	Salesman	Foreman
FS	782.21	782.21	782.21	782.21	782.21
DS	11.66	12.27	13.69	11.53	14.14
本文算法	4.07	9.37	12.02	4.81	13.04

由表 2 可以看出,FS 算法的搜索次数非常之

大,相比之下,DS 算法大大减少了每个宏块的搜索次数,也降低了运算复杂度,而本文算法的平均搜索次数最少,复杂度最低。其中 News 和 Salesman 序列,因为符合假设条件,所以平均搜索次数非常之低,效果明显;其余序列虽不符合本算法的前提条件,但复杂度也有降低。

3.2 平均峰值信噪比对比

平均峰值信噪比是从客观上评价重建图像准确性的一种方法。每一帧图像的峰值信噪比为

$$PSNR = 10 \times \lg \frac{255 \times 255}{MSE}$$

其中, $MSE = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (f(m,n) - \hat{f}(m,n))^2$, M , N 分别为图像的长宽, $f(m,n)$, $\hat{f}(m,n)$ 分别为原图和重建图像对应的像素灰度值。表 3 给出了各算法下不同序列重建图像的平均峰值信噪比。平均峰值信噪比是每一帧重建图像峰值信噪比的平均。图 2 给出了 News 序列和 Trevor 序列重建图像峰值信噪比随帧数变化的关系。

表 3 各算法下各序列重建图像的平均峰值信噪比

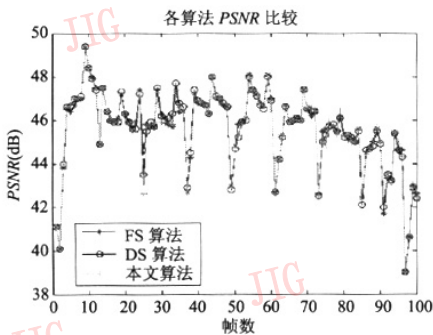
Tab.3 Average PSNR with each algorithm

算法	各序列重建图像的平均峰值信噪比 (dB)				
	News	Trevor	Suzie	Salesman	Foreman
FS	47.058	44.939	46.957	45.654	42.951
DS	47.060	44.921	46.955	45.637	42.945
本文算法	47.062	44.922	46.956	45.645	42.946

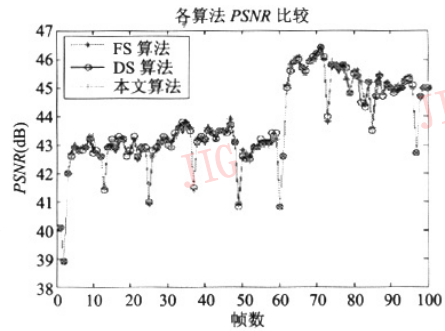
由表 3 和图 2 可以看出,FS 算法重建图像的峰值信噪比值最高,本文算法各帧重建图像的峰值信噪比都超过了 DS 算法,与 FS 接近。可见相对于 DS 算法,本文算法在运动估计的精度上略有改善。其中对 News 序列,DS 算法和本文算法重建图像的峰值信噪比值都比 FS 算法高,这是因为本文采用 SAD 准则,而峰值信噪比值则是根据 MSE 计算的,故不能借此否定 FS 算法的全局最优性。

3.3 准确性对比

准确性比较将采用相对于 FS 算法的平均运动矢量偏移距离来衡量。如果将 FS 算法得到的是最优运动矢量记为 (F_x, F_y) , 而将新算法得到的运动矢量记为 (M_x, M_y) , 则偏移距离 $Dist = \sqrt{(M_x - F_x)^2 + (M_y - F_y)^2}$, Dist 值越小,说明新的运动矢量越接近全局最优点。表 4 给出了各算法下的不同序列平均 Dist 值。



(a) News



(b) Trevor

图 2 News 序列和 Trevor 序列 3 种算法重建图像的峰值信噪比

Fig. 2 PSNR of News and Trevor sequence with each algorithm

表 4 各算法下的运动矢量平均偏移距离

Tab. 4 Average excursion distance of the motion vectors with each algorithm

算法	不同序列运动矢量平均偏移距离				
	News	Trevor	Suzie	Salesman	Foreman
FS	0.000	0.000	0.000	0.000	0.000
DS	0.040 1	0.044 2	0.097 8	0.001 7	0.504 5
本文算法	0.039 2	0.042 5	0.069 8	0.000 8	0.351 1

由表 4 可以看出,与 DS 算法相比,本文算法找到的运动矢量更接近全局最优点,即本文算法的准确性略有提高。

通过以上的分析,说明本文提出的算法与 FS、DS 算法相比,在准确性略有改善的前提下,还有效降低了了运动估计的复杂度。当满足本文假设时,算法复杂度改善明显;即使不满足假设时,算法复杂度也有一定改善。

4 结 论

本文提出了一种基于概率矩阵的快速块匹配运动估计算法。该算法先综合运用时空相关性来对宏块的运动向量的概率分布进行估计,再基于概率矩阵,对概率较大的运动向量进行块匹配运算,找到最

小 MAD 点后,再使用小菱形搜索算法进行修正来得到最终运动估计结果。该算法与标准的菱形算法相比,在精度略有提高的基础上,搜索速度也有了较大改进。

参考文献 (References)

- 1 ITU-T Rec. Generic Coding of Moving Pictures and Associated Audio Information: Video[S]. H. 262, ISO/IEC 1381-2-1995.
- 2 ITU-T. Information Technology—Generic Coding of Audio-visual Objects(final Draft of International Standard)[S]. ISO/IEC JTC1 IS 14496-2 (MPEG-4), Oct,1998.
- 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 Po Lai-man, Ma Wing-chung. A novel four-step search algorithm for fast block motion estimation[J]. IEEE Transactions on Circuit System Video Technology, 1996, 6(3): 313 ~ 317.
- 5 Jain J, Jain A. Displacement measurement and its application in interframe image coding[J]. IEEE Transactions on Communication, 1981, 29(12):1799 ~ 1808.
- 6 Kossentini F, Lee Y W, Smith J T, et al. Predictive RD optimized motion estimation for very low bit-rate video coding[J]. IEEE Journal Selected Areas in Communication, 1997, 15(12):1752 ~ 1763.
- 7 Liu Limg-kuo, Feig E. A block-based gradient descent search algorithm for block motion estimation in video coding[J]. IEEE Transactions on Circuits Systems Video Technology, 1996, 6(4): 419 ~ 422.