

MPEG-4 编码中运动估计和补偿算法研究

宁矿凤^{1,2}, 王小玲¹

(1. 中南大学 信息科学与工程学院, 湖南 长沙 410205;

2. 湖南涉外经济学院 计算科学与技术学部, 湖南 长沙 410205)

摘要:对 MPEG-4 视频编码的关键技术——运动估计算法进行了研究和改进。在传统的运动估计算法 MVFAST 的基础上,提出了一种快速运动估计算法,能够在没有明显降低图像质量的条件下,获得更快的编码速率。

关键词:MPEG-4 编码; 运动估计和补偿; MVFAST 算法

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2007)06-0101-03

Research on Motion Estimate and Compensation Algorithm of MPEG-4 CODEC

NING Kuang-feng^{1,2}, WANG Xiao-ling¹

(1. School of Info. Sci. & Eng., Central South University, Changsha 410205, China;

2. Dept. of Computer Sci. & Tech., Hunan College of International Economics, Changsha 410205, China)

Abstract: Focuses on the key technique of MPEG-4 CODEC. Studies and improves motion estimate and compensation algorithm. Based on of conventional motion estimate algorithm (MVFAST), put forwards a motion estimate algorithm and has been able to obtain a faster coding velocity under no decline in quality of the picture.

Key words: MPEG-4 CODEC; motion estimate and compensation; MVFAST algorithm

0 引言

随着网络和数字化时代的飞速发展,使得对视频图像多媒体的处理要求也越来越高。追求高压缩比、高传输效率、高清晰图像质量是视频技术发展的方向。对视频图像的压缩,这几年来已经取得了突飞猛进的发展:小波变换方式的使用,新的运动估计算法的使用,以及新的编码扫描方式的使用等等,都大大提高了视频图像的压缩率、传输效率、图像质量。

在视频压缩系统中使用运动估计是消除图像帧在时间上冗余度的最基本、最有效的一种方法,现有的视频编码国际标准技术框架都将其纳入为重要的组成部分,无论是 H. 263 还是 MPEG1, MPEG2, MPEG4^[1],甚至最新的视频压缩标准 H. 264^[2],无一例外都采用了帧间运动补偿技术。

视频编码压缩系统的效率主要是通过运动估计算法的效率来体现的,运动估计算法速度越快、运动矢量

越准确,那么视频压缩的速率越快、效果越好,而运动估计所用的时间在整个压缩时间中占了很大比例,严重影响着视频编码器是否能实现实时编码,且运动估计的精度也影响着编码的视频质量。因此,有必要对视频压缩中最影响压缩效率的运动估计算法做深入研究,在不影响视频质量的情况下,在传统运动估计算法的基础上寻求改进的方法,寻找到运动估计速度与精度的平衡点,以提高编码效率。

1 常用算法及分析

1.1 全搜索算法

全搜索算法对搜索区域 S 内的所有位置都计算匹配标准函数,搜索过程可以是左到右,从上到下,按扫描顺序搜索,也可以采用以位移(0,0)点为中心的螺旋型搜索。尽管这种算法能够保证找到最佳匹配块,取得令人满意的结果,但由于其运算量十分巨大,内存访问过多,所需运算时间过长,对处理器的运算能力要求非常高。因此为了进一步减少匹配算法的搜索次数,降低运算量,人们研究出一些快速搜索算法,如三步搜索算法、二维对数搜索法、共轭方向搜索法等。

收稿日期:2006-09-08

作者简介:宁矿凤(1980-),女,湖南长沙人,硕士研究生,助教,研究方向为多媒体技术及应用;王小玲,教授,研究方向为数据库技术、数据挖掘技术及网络安全技术。

1.2 MVFAST 算法

MVFAST 算法是在钻石搜索法的基础上进行改进的,它利用了运动矢量的空间相关性^[3],初始化了一系列成为此块运动矢量可能性较大的预运动矢量:(0,0)和左边块、上边块、右上边块的三个运动矢量,并设置了一些阈值作为提前终止搜索的条件,最佳匹配函数采用了绝对差之和 SAD,这些都大大加快了搜索的速度,同时也获得了比较理想的图像质量,已被人们广泛采纳。它的搜索过程如下:

(1) 进行静止块检测,即运动矢量(0,0)的搜索。计算 $SAD(0,0)$,如果 $SAD(0,0)$ 小于阈值 T_1 ,那么就认为运动矢量为(0,0),停止搜索;否则进入(2)。 T_1 的典型值为 512。

(2) 决定当前宏块的运动活动性。设其左边宏块运动矢量为 $V_1(x_1, y_1)$,上边宏块运动矢量为 $V_2(x_2, y_2)$,右上边宏块运动矢量为 $V_3(x_3, y_3)$, $L = \text{MAX}(|x_1| + |y_1|, |x_2| + |y_2|, |x_3| + |y_3|)$,根据 L 的值决定当前宏块的运动活动性。如果 $L < L_1$,则认为当前宏块的运动活动性很低;如果 $L_1 < L < L_2$,则认为当前宏块的运动活动性一般;如果 $L_2 < L$,则认为当前宏块的运动活动性很高。

(3) 决定搜索中心及搜索策略。如果宏块活动性很低,那么搜索中心为(0,0)点,且采用小钻石继续搜索;如果宏块活动性一般,那么搜索中心为(0,0)点,且采用大钻石继续搜索,最后再采用小钻石搜索;如果宏块活动性很高,那么计算 V_1, V_2, V_3 指向的位置的 SAD 值,和 $SAD(0,0)$ 比较,取其使 SAD 最小的位置为搜索中心,采用小钻石继续搜索。

(4) 搜索最佳匹配块。如图 1 所示,在大钻石搜索中,搜索中心点为 (i_0, j_0) ,搜索点 (i_1, j_1) 除中心点外其它必须满足以下条件: $|i_0 - i_1| + |j_0 - j_1| = 2$ 。如果其中使 SAD 最小的位置为钻石中心点,那么切换到小钻石继续搜索;否则以此位置点为中心继续大钻石搜索。在小钻石搜索中,计算中心点以及其上、下、左、右五个点的 SAD,如果使 SAD 最小的点为中心点,那

么搜索结束,运动矢量指向中心点,否则以五点中 SAD 最小的点为中心继续搜索。

2 运动估计和补偿算法的优化

2.1 改进的搜索算法

MVFAST 算法虽然比全搜索快得多,但它并不是最优的算法,笔者在它基础上进行了改进,将当前块的预运动矢量由四个改为了六个,它们分别是:(0,0)、左边块运动矢量 $MV_1(x_1, y_1)$ 、上边块运动矢量 $MV_2(x_2, y_2)$ 、右边块运动矢量 $MV_3(x_3, y_3)$ 和 MV_1, MV_2, MV_3 的中值运动矢量 $MV_4(x_4, y_4)$ 、前一帧对应块的运动矢量 $MV_5(x_5, y_5)$ 。

$$x_4 = \text{median}(x_1, x_2, x_3);$$

$$y_4 = \text{median}(y_1, y_2, y_3);$$

其中,

$$\text{median}(x, y, z) = \begin{cases} x, y \leq x \leq z \text{ 或 } z \leq x \leq y \\ y, x \leq y \leq z \text{ 或 } z \leq y \leq x \\ z, x \leq z \leq y \text{ 或 } y \leq z \leq x \end{cases}$$

在这五个矢量中, MV_4 成为当前块运动矢量可能性最大,其次是(0,0)矢量,最后才是 MV_1, MV_2, MV_3, MV_5 ^[4]。因此,按成为当前块运动矢量可能性由大到小检测这五个运动矢量。另外,在 MVFAST 算法中,钻石搜索的终止条件是最佳匹配位置在钻石的中心,为了更快结束搜索,另加了一个可变阈值 T_1 和一个固定阈值 T_0 ,使其更早结束搜索。如果搜索点的 SAD 值小于 T_0 ,将结束整个搜索过程;如果搜索点的 SAD 值小于 T_1 ,则结束本钻石的搜索,以它为中心进行下个钻石搜索, T_1 为这个钻石的中心点的 SAD 值。考虑到最佳匹配位置在小钻石四个点的可能性很高^[5],直接采用了小钻石,没有使用大钻石。为了防止搜索点过多,在小钻石搜索中设置了搜索点的最多个数阈值 T_3 。这个算法是以牺牲图像质量代价来加快搜索速度的。

具体搜索过程如下:

(1) 检测 MV_4 。计算 MV_4 指向的位置的 SAD 值(记为 SAD_4),如果 $SAD_4 < T_0$,则结束整个搜索过程, MV_4 为当前宏块的运动矢量;否则,跳到(2)。

(2) 检测(0,0)。计算它的 SAD 值(记为 SAD_0),如果 $SAD_0 < T_0$,则结束整个搜索过程,(0,0)为当前宏块的运动矢量;否则,跳到(3)。

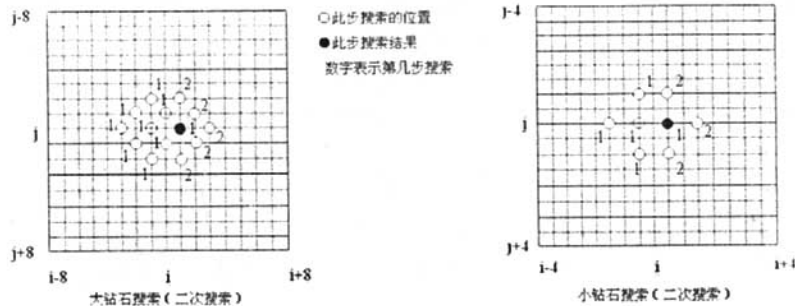


图 1 MVFAST 算法中的大/小钻石搜索

(3) 检测 MV_1, MV_2, MV_3, MV_5 。计算它们指向的位置的 SAD 值 (分别记为 $SAD_1, SAD_2, SAD_3, SAD_5$) , 如果 $SAD_1, SAD_2, SAD_3, SAD_5$ 中仅有一个小于 T_0 , 则结束整个搜索过程, SAD 值小于 T_0 的运动矢量为当前宏块的运动矢量; 如果 $SAD_1, SAD_2, SAD_3, SAD_5$ 中小于 T_0 的不止一个, 则结束整个搜索过程, 取 SAD 值最小的运动矢量为当前宏块的运动矢量; 否则, 跳到 (4)。

(4) 决定搜索中心及搜索阈值, 设置搜索起始点。比较 $SAD_0, SAD_1, SAD_2, SAD_3, SAD_4$, 取 SAD 最小的运动矢量所指的位置为小钻石的中心, T_2 为五个 SAD 值中最小值。搜索起始点的初始值为中心右边的点。

(5) 搜索最佳匹配块。如图 2 所示, 按顺时针顺序计算其右、下、左、上四个点的 SAD, 如果搜索点的 SAD 小于 T_0 , 结束整个搜索过程, 指向这一点的矢量为当前块的运动矢量; 如果搜索点的 SAD 小于 T_2 , 以此点为下一次的搜索中心, T_2 更新为此点的 SAD 值。如果搜索点为上边点, 则下一次搜索的起始点为上边点; 如果搜索点为左边点, 则下一次搜索的起始点为左边点; 以此类推设定下次搜索的起始点继续下一次搜索, 重复 (5); 如果在小钻石搜索过程中, 搜索点的个数大于 T_3 , 结束整个搜索过程。指向 SAD 值最小的点运动矢量为当前块的运动矢量。

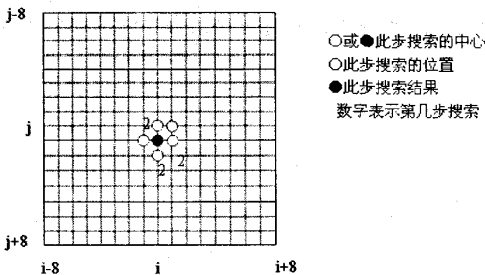


图 2 改进算法中的小钻石搜索 (二次搜索, $T_3 = 5$)

2.2 完整的运动估计算法

根据 MPEG4 的 Simple Profile 可知, 在 P-VOP 中有三种宏块模式: Intra 模式、Inter 模式和 Inter4v 模式, 且支持半像素搜索。Inter 模式宏块需要一个运动矢量, Inter4v 模式含有四个运动矢量, 而 Intra 类型无运动矢量。在 P-VOP 中, 每个宏块都要进行帧间运动估计运算。

为了简化运算、提高精度, 整个搜索过程都是在半像素基础之上的。具体算法如下:

(1) 进行静止块检测, 即运动矢量 (0,0) 的搜索。计算 $SAD(0,0)$, 如果 $SAD(0,0)$ 小于阈值 T , 那么就认为运动矢量为 (0,0), 停止搜索, 此宏块为 Inter 块; 否则进入 (2)。 T 的典型值为 512。

(2) Intra 模式与 Inter 模式的选择。计算此 16×16 宏块自身的复杂度 DEV ; 并按改进的搜索算法以一个像素距离为步长进行 16×16 宏块的搜索, 所得运动矢量为 (x_{16}, y_{16}) , 其所指位置的 SAD 为 SAD_{16} 。如果 $DEV < SAD_{16} - 128$, 此宏块为 Intra 模式, 结束搜索; 否则进入 (3)。

(3) Inter 模式的选择。如果 $SAD_{16} < T$ 并且 $SAD_{16} < 25 \times Qp$ (Qp 为这一帧的量化因子), 选择 Inter 模式, 运动矢量为 (x_{16}, y_{16}) ; 否则进入 (4)。

(4) Inter 模式和 Inter4v 模式的选择。对于这一宏块中的四个 8×8 宏块分别按改进的搜索算法以半像素距离为步长进行搜索, 其中预运动矢量改为了五个, 它们分别是: (x, y) 、左边块运动矢量 $MV_1(x_1, y_1)$ 、上边块运动矢量 $MV_2(x_2, y_2)$ 、右上边块的运动矢量 $MV_3(x_3, y_3)$ 和 MV_1, MV_2, MV_3 的中值运动矢量 $MV_4(x_4, y_4)$ 。其中, (x, y) 为当前块运动矢量可能性最大, 其次是 MV_4 , 最后是 MV_1, MV_2, MV_3 。这四个块的搜索结果分别是运动矢量为 (x_{81}, y_{81}) 和所指位置的 SAD 为 SAD_{81} 、运动矢量为 (x_{82}, y_{82}) 和所指位置的 SAD 为 SAD_{82} 、运动矢量为 (x_{83}, y_{83}) 和其所指位置的 SAD 为 SAD_{83} 、运动矢量为 (x_{84}, y_{84}) 和其所指位置的 SAD 为 SAD_{84} 。如果 $SAD_{16} < SAD_{81} + SAD_{82} + SAD_{83} + SAD_{84} + 16$, 此宏块为 Inter 模式, 运动矢量为 (x_{16}, y_{16}) ; 否则为 Inter4v 模式, 四个块的运动矢量分别为 (x_{81}, y_{81}) 、 (x_{82}, y_{82}) 、 (x_{83}, y_{83}) 、 (x_{84}, y_{84}) 。

2.3 实验结果及分析

我们在 P4 1.7GHz 处理器、256M 内存、Windows2000 操作系统的机器上进行了测试。在改进方法中, $T = 512$, 16×16 宏块搜索时 $T_0 = 524$, 8×8 宏块搜索时 $T_0 = 128$, 测试结果如表 1 所示。

表 1 MPEG-4 编码器中模块测试结果

序列	格式	目标码率 (kps)		全搜索	MVFAST	改进方法
News	QCIF	64	平均 PSNR (dB)	31.191	31.141	31.036
			速度倍数	1	76	101
		96	平均 PSNR (dB)	33.343	33.238	33.058
			速度倍数	1	66	94
foreman	QCIF	64	平均 PSNR (dB)	27.573	27.452	27.218
			速度倍数	1	67	91
		96	平均 PSNR (dB)	28.962	28.854	28.638
			速度倍数	1	60	87
breann	QCIF	64	平均 PSNR (dB)	28.316	28.247	27.984
			速度倍数	1	68	93
		96	平均 PSNR (dB)	29.635	29.627	29.392
			速度倍数	1	58	88
News	CIF	256	平均 PSNR (dB)	37.539	37.484	37.287
			速度倍数	1	60	89
		384	平均 PSNR (dB)	39.001	38.959	38.790
			速度倍数	1	57	82

2 实验结果

本实验采用数字内窥镜摄取的胃镜病灶图像(如图 2a 所示),图 2b 是加入了运动模糊($r = 10$)和 40dB 的高斯白噪声的退化图像,通过信噪比的改善来判断图像复原的优劣:

$$\Delta \text{SNR} = \frac{\|y - x\|^2}{\|\hat{x} - x\|^2} = \frac{\sum (y - x)^2}{\sum (\hat{x} - x)^2}$$

其中, x, y, \hat{x} 分别表示原始图像、退化图像和复原图像。由图 2c, 2d 可以看出,将 NAS-RIF 算法和 Hopfield 神经网络相结合,可以得到较为满意的图像复原效果。

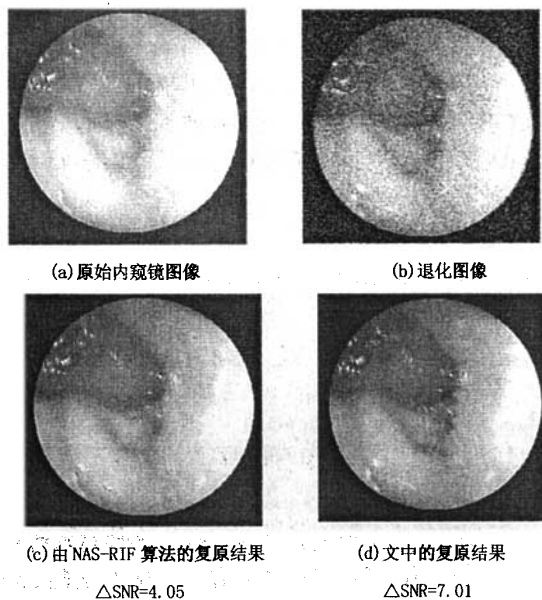


图 2 实验结果

3 结论

综上所述, NAS-RIF 算法在进行图像盲复原的

过程中,若存在观测噪声,复原效果不理想。Hopfield 神经网络算法进行图像复原可以较好地消除复原过程中的震铃效应,但是只能进行已知 PSF 的图像复原,也就是说无法进行图像盲复原。所以将 NAS-RIF 算法和 Hopfield 神经网络算法相结合,先由 NAS-RIF 算法估计出点扩展函数,再由全并行自反馈 Hopfield 神经网络算法进行复原,最后可以得到较理想的复原效果。由于使用了全并行自反馈 Hopfield 神经网络算法,网络自组织和图像同构是同时进行的,网络能量逐步达到精确最小的同时,图像也恢复出来了,从图 2d 即可以看出,图像较清晰,信噪比提高显著。

今后的工作主要是在神经元的取值上做进一步的研究,以减少迭代过程中引入的误差,提高收敛速度。并且将研究的成果进一步运用到其它内窥镜图像的复原上。

参考文献:

- [1] KUNDUR D, HATZINAKOS D. Blind image deconvolution [J]. IEEE Signal Processing Magazine, 1996, 13(3): 43-64.
- [2] 张航, 罗大庸. 图像盲复原算法研究现状及其展望[J]. 中国图象图形学报, 2004(10): 1145-1152.
- [3] KUNDUR D, HATZINAKOS D. A novel blind deconvolution scheme for image restoration using recursive filtering [J]. IEEE Trans on Signal Processing, 1998, 46(2): 375-389.
- [4] ZHOU Y T, CHELLAPPA R, VAID A, et al. Image restoration using a neural network[J]. IEEE Transactions on ASSP, 1988, 36(7): 1141-1151.
- [5] PAIK J K, KATSAGGELOS A K. Image restoration using a modified Hopfield network[J]. IEEE Transactions on Image Processing, 1992, 1(1): 49-63.
- [6] 王磊, 戚飞虎, 莫玉龙. 精确复原退化图像的连续 Hopfield 网络研究[J]. 上海交通大学学报, 1997, 31(12): 43-46.

(上接第 103 页)

3 结论

改进的 MVFAST 算法整个过程都是在半像素基础上进行的,且着重考虑压缩编码的速度率、视频图像的质量、码率控制等要求,经过测试得到:此算法在整个编码速度上提高了 30% 左右, PSNR 值降低了 0.2dB 左右,为广大的实际应用和开发奠定了基础。

参考文献:

- [1] 李斌, 李论. MPEG 1-MPEG7 的分析与评价[J]. 电视技术, 2001(1): 45-48.

- [2] 朱秀昌. 视频编码新标准——H. 264[J]. 电信科学, 2002(12): 28-30.
- [3] 哈力旦. 一种改进的运动矢量编码方法[J]. 西安电子科技大学学报, 2005(4): 60-64.
- [4] Tourapis A M, Au O C, Liou Ming L. Predictive Motion Vector Field Adaptive Search Technique(PMVFAST) Enhancing Block Based Motion Estimation[J]. IEEE Trans. On Circuits & Systems for Video Technology, 2001, 8: 369-377.
- [5] ZHANG Z Y. Camera Calibration with One 2 Dimensional Objects[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 7(26): 892-899.