

一种快速自适应全局运动估计算法的研究¹

张 方 肖 嵩 吴成柯

(西安电子科技大学 ISN 国家重点实验室 西安 710071)

摘 要: 该文提出一种快速自适应全局运动估计算法。该方法从分析块匹配运动矢量统计特性出发, 通过引入一个可预设置参数, 计算当前帧与参考帧之间的基本平移矢量, 并自适应地采取混合全局运动、局部运动估计的算法处理不同特征的源数据。仿真结果表明, 与传统方法相比, 该文算法复杂性低、处理延时小、更利于硬件实现, 在 PSNR 变化不大的情况下, 编码效率有较大的改善。

关键词: 全局运动估计, 基本平移矢量, 运动矢量场, H.263+

中图分类号: TN911.22 **文献标识码:** A **文章编号:** 1009-5896(2004)04-0574-06

Study on a Fast Adaptive Global Motion Estimation Algorithm

Zhang Fang Xiao Song Wu Cheng-ke

(ISN National Key Lab., Xidian University, Xi'an 710071, China)

Abstract An adaptive global motion estimation algorithm is proposed in this article. Based on the statistic feature of block matching moving vector, the method calculates the basic translation vector between current frame and referenced frame through a revisable parameter, and adaptively chooses processing algorithm for image sources with different characteristics. Simulation results show that the proposed algorithm not only has the advantages of low complication, short delay and simple implementation, but also can highly improve the coding efficiency when PSNR varies slightly.

Key words Global motion estimation, Basic translation vector, Motion vector field, H.263+

1 引言

随着视频压缩技术的不断发展及其应用领域的拓宽, 全局运动估计 (Global Motion Estimation, GME) 和全局运动补偿 (Global Motion Compensation, GMC) 技术正受到越来越多的关注。传统算法通常通过构造多参数的仿射或双线性图像运动模型^[1,2], 依靠迭代法从连续几帧图像中计算出全局运动参数, 用来分隔全局运动矢量和局部运动矢量并分两步分别加以处理。为了加快全局运动估计速度, 文献 [3] 针对两步法提出一种快速局部运动估计算法, 文献 [4] 则提出采用估计精度分层递增的方法, 文献 [5] 力图通过稠密估计及 Gauss-Newton 和 Levenberg-Marquadet 优化算法来寻找估计精度和处理速度的平衡, 这些尽管都取得了不同程度的效果, 但仍然无法解决迭代特征点的严格要求、运算占用资源大、高频分量丢失、重建噪声较高的问题。文献 [6] 提出一种复合全局运动、局部运动估计的方法, 较大程度地减小了内插值噪声, 特别是提高了低分辨率图像的重建峰值信噪比, 尽管同时提出了快速算法, 但其运算复杂性及处理延时使其仍不适宜用于实时压缩系统。相对于视频实时压缩编码而言, 传统算法更适合用于图像分割、相机运动参数还原及三维重建等领域。

¹ 2002-07-04 收到, 2003-02-25 改回

国家自然科学基金资助课题 (No.60002007)

ITU-T 提出的 H.263+ 的标准中, 可以通过参考图像重采样 (附件 P) 来引入全局运动估计, Kui Zhang^[7] 描述了具体的实现算法, 尽管在编码效率及运动补偿效果方面取得明显提高, 但运算量及复杂性仍然太大。H. Richter 等人^[8] 引入一个运动估计中的低通滤波器, 实现了一种基于 MPEG-4 而优于其标准算法的实时快速全局运动估计方法, 取得了很好的效果, 但仍然是从计算 6 参数图像仿射模型出发, 分割背景运动矢量并用来估计局部图像匹配搜索范围。本文提出一种新的快速自适应混合全局运动、局部运动估计算法, 并将其实现在基于 H.263+ 标准的编码器中。通过软件仿真, 将其与 H.263+ 标准算法、基于 Kui Zhang 和 H. Richter 等人传统算法的 H.263+ 编码器做了比较, 结果表明带有本文算法的 H.263+ 编码器与标准 H.263+ 的编码器相比, 编码效率有约 20% 的提高, 在保证相同压缩比时, 重建图像质量约有 0.5 dB 的改善; 与传统方法相比, 本文算法复杂性低、处理延时小、更利于硬件实现, 不存在选取匹配特征点问题, 在 PSNR 变化不大的情况下, 编码效率有较大的改善。

2 视频序列的基本平移矢量

传统全局运动估计方法首先构造多参数图像运动模型, 然后通过序列图像特征点迭代的方法分隔开全局运动和局部运动, 本文则仅仅从提高相同条件下数据压缩比的角度, 不考虑运动模型参数的计算, 只是利用全局运动造成时间上的冗余性在宏块运动矢量及其差分量的统计特性上的反映, 来达到全局运动估计的目的。为了便于描述, 我们引入如下概念:

设同一帧内第 i 个宏块的运动矢量为

$$\mathbf{MV}_i = [\mathbf{MV}_{(i,x)}, \mathbf{MV}_{(i,y)}], \quad i = 1, 2, \dots, M$$

必然存在:

$$\mathbf{MV}_i = \mathbf{T}_C + \mathbf{M}_i, \quad i = 1, 2, \dots, M \quad (1)$$

当满足 $\sum_{i=1}^M |\mathbf{M}_i| = \text{Min}$ 时, 称式 (1) 中的 $\mathbf{T}_C$ 为基本平移矢量。同一帧内所有运动矢量构成一个运动矢量场 (Motion Vector Field, MVF)^[9], 其矢量顶点构成一个三维曲面 α , 而表示图像全局运动的基本平移矢量 $\mathbf{T}_{mi}$ 的顶点则位于一个与图像平面 β 相互平行的二维平面 γ 上, 并且 α 上所有点到 γ 平面的垂直距离和小于到任意平行于 β 的二维平面的垂直距离和。当式 (1) 中基本平移量 $\mathbf{T}_C$ 占到 $\mathbf{MV}_i$ 一定的比重时, 如果仍对每个宏块都传送这一帧内恒定量, 则无疑降低了数据压缩率, 或者在同样压缩率时降低了重建图像的质量。据此, 本文提出一种低复杂度 $\mathbf{T}_C$ 的计算方法, 对于压缩后的数据码流只在当前帧内第一宏块传输此基本平移矢量 (也就是全局运动矢量), 其余宏块则只传输 $\mathbf{M}_{mi}$, 从而达到提高编码效率和压缩质量的目的。

根据基本平移矢量的定义, 存在如下方程式:

$$\left. \begin{aligned} \mathbf{MV}_i &= [\mathbf{MV}_{(i,x)}, \mathbf{MV}_{(i,y)}], & i &= 1, 2, \dots, M \\ \mathbf{T}_C &= [T_x, T_y] \\ \sum_{i=1}^M [|\mathbf{MV}_{(i,x)} - T_x|^2 + |\mathbf{MV}_{(i,y)} - T_y|^2] &= \text{Min}, & M &\text{为一帧图像总的宏块数} \end{aligned} \right\} \quad (2)$$

假定运动矢量场内任意量其 x 分量与 y 分量之间均相互独立, 则 T_x 和 T_y 要使方程式组 (2) 中第 3 式左边的两项同时取最小值, 根据极值点的微分特性, 有如下式 (3):

$$\left. \begin{aligned} \frac{\partial}{\partial T_x} \sum_{i=1}^M |\mathbf{MV}_{(i,x)} - T_x|^2 &= 0 \\ \frac{\partial}{\partial T_y} \sum_{i=1}^M |\mathbf{MV}_{(i,y)} - T_y|^2 &= 0 \end{aligned} \right\} \quad (3)$$

据此, 本文提出基本平移矢量计算方法如下:

步骤 1 统计同一帧内所有宏块运动矢量 x 和 y 分量的正负值个数 N_{x+} , N_{x-} , N_{y+} 及 N_{y-} ;

步骤 2 参考式 (3), 如果 $N_{x+} \geq N_{x-}$ (条件 (a)), 则取 $T_x = \frac{1}{N_{x+}} \sum_{i=1}^{N_{x+}} MV_{(i,x+)}$; 如果 $N_{x+} < N_{x-}$ (条件 (b)), 则取 $T_x = \frac{1}{N_{x-}} \sum_{i=1}^{N_{x-}} MV_{(i,x-)}$. 其中 $MV_{(i,x+)}$ 及 $MV_{(i,x-)}$ 分别表示 $MV_{(i,x)}$ 中的所有正数值和所有负数值;

步骤 3 参考步骤 2 中所述, 计算 T_y ;

步骤 4 得到最终的基本平移矢量 $T_C = [T_x, T_y]$.

3 自适应混合全局运动、局部运动估计方法

如果不存在全局运动的序列图像采用全局运动估计算法, 则必然会造成算法复杂化、处理延时等方面的损失, 因此算法应该能自动检测基本平移矢量的特性来判断采用那种估计方法, 从而实现自适应混合全局运动、局部运动估计, 这样才会使其具有更为广泛的应用领域. 同时出于算法灵活性的考虑, 引入一个可设置阈值参数 T_{frame} 和压缩数据流 Picture 层标志位 frame_x , frame_y 以及宏块层标志位 mb_x 和 mb_y . 具体描述如下:

算法 1

步骤 1 将第 2 节中步骤 2 的条件 (a) 改为 $N_{x+} \geq T_{\text{frame}} > N_{x-}$ (条件 (a')), 条件 (b) 改为 $N_{x-} \geq T_{\text{frame}} > N_{x+}$ (条件 (b')) 来计算 T_x , 可以取初值 $T_{\text{frame}} = M_{\text{num}}/2$, 其中 M_{num} 表示当前处理帧的总宏块数. 若条件 (a) 和 (b) 都不满足, 取 $T_x = 0$; 同理计算出 y 方向基本平移量 T_y ;

步骤 2 仍沿用运动矢量场内任意量其 x 分量与 y 分量之间均相互独立的假定, 若 $T_x = 0$, 表示当前帧相对于参考帧不存在 x 方向的全局运动, 置 $\text{frame}_x = 0$, 否则 $\text{frame}_x = 1$; 同样根据 T_y 处理 frame_y ;

步骤 3 如果 $\text{frame}_x = 1$ 或者 $\text{frame}_y = 1$, 则在帧编码初传送宏块运动矢量 $M = [M_x, M_y]$ 之前先传送基本平移矢量 $T_C = [T_x, T_y]$;

步骤 4 如果 $\text{frame}_x = 0$, 则当前帧内传送的所有宏块运动矢量 x 分量 $M_x = MV_x$, 且置所有宏块的标志位 $mb_x = 0$; 否则, 如果条件 (a') 满足且当前宏块运动矢量 x 分量 $MV_{(mi,x)} \geq 0$, 或者条件 (b') 满足且 $MV_{(mi,x)} < 0$, 则置当前宏块 $mb_x = 1$, 且 $M_{(mi,x)} = MV_{(mi,x)} - T_x$, 否则 $mb_x = 0$ 且 $M_{(mi,x)} = MV_{(mi,x)}$. y 方向分量的处理方法与此相同. 这是因为考虑到同一帧内 $MV_{(mi,x)}$ 和 $MV_{(mi,y)}$ 可能有正有负, 为了避免用全局运动估计后的宏块局部运动矢量绝对值反而增大甚至超出运动矢量的范围 (比如 $[-16, +15.5]$, 这与算法选用的搜索窗口大小有关), 使得后续处理模块如 VLC 等无法操作, 便将与基本平移矢量对应分量符号相反的运动矢量看作是奇异值, 对其相应宏块不做全局运动估计;

步骤 5 采用步骤 4 处理完当前帧内所有宏块后回到步骤 1.

依据文献 [10], 为了进一步减小数据量, 要将计算出各宏块的运动矢量与空间相邻模块做差分编码后传送, 图 1 是 H.263+ 标准运动矢量差分编码的示意图. 这样通过消除宏块运动矢量之间的空间冗余性进一步提高了整个系统的编码效率, 也在一定程度上减小了全局运动带来编码效率上的损失 [7], 因此单纯分离运动矢量中的基本平移矢量并不能最大程度提高数据压缩比, 基于此, 本文对算法 1 做了修改, 通过计算一帧内运动矢量差分值之间的全局不变量 $MVD_{\text{const}} = [MVD_x, MVD_y]$ 代替算法 1 中的基本平移矢量 T_C 进行编码, 同时引入一个可预设置参数 MVD_{frame} 及宏块层矢量溢出标志 $\text{global}_x, \text{global}_y$. 具体描述如下:

算法 2

步骤 1 对于当前帧内所有确定为帧间编码的宏块, 计算出对应的运动矢量差分值: $D_i =$

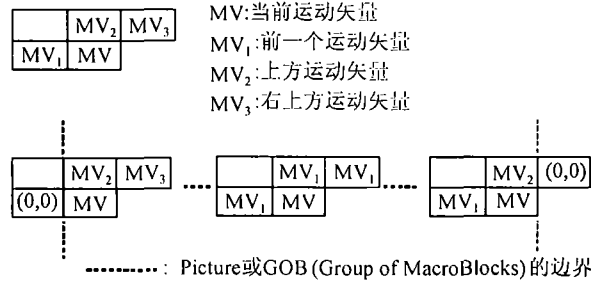


图1 H.263+ 运动矢量差分编码示意图

$[D_{(i,x)}, D_{(i,y)}], i = 1, 2, \dots, M;$

步骤2 分别统计同一帧内 D_i 的 x, y 分量正负值个数 $dN_{x+}, dN_{x-}, dN_{y+}$ 及 dN_{y-} ;

步骤3 若 $dN_{x+} > \text{MVD}_{\text{frame}} \geq dN_{x-}$, 则 $dN_x = dN_{x+}$; 若 $dN_{x-} > \text{MVD}_{\text{frame}} \geq dN_{x+}$, 则 $dN_x = dN_{x-}$; 若这两者都不满足, 则 $dN_x = 0$ 。同理计算 dN_y ;

步骤4 若 $dN_x = 0$, 则 $\text{MVD}_x = 0$; 若 $dN_y = 0$, 则 $\text{MVD}_y = 0$; 若这两者都不满足, 则取 $\text{MVD}_{\text{const}} = [\text{MVD}_x, \text{MVD}_y]$ 如下:

$$\frac{\partial}{\partial \text{MVD}_x} \sum_{i=1}^{dN_x} |D_{(x,i)} - \text{MVD}_x|^2 = 0 \quad (4a)$$

$$\frac{\partial}{\partial \text{MVD}_y} \sum_{i=1}^{dN_y} |D_{(y,i)} - \text{MVD}_y|^2 = 0 \quad (4b)$$

若 $dN_x = dN_{x+}$, 则式 (4a) 中的 $D_{(x,i)}$ 为当前帧内所有为正数的运动矢量 x 分量; 若 $dN_x = dN_{x-}$, 则式 (4a) 中的 $D_{(x,i)}$ 为当前帧内所有为负数的运动矢量 x 分量。同理计算 $D_{(y,i)}$ 。则有

$$\left. \begin{aligned} \text{MVD}_x &= \frac{1}{dN_x} \sum_{i=1}^{dN_x} D_{(x,i)} \\ \text{MVD}_y &= \frac{1}{dN_y} \sum_{i=1}^{dN_y} D_{(y,i)} \end{aligned} \right\} \quad (5)$$

步骤5 参考算法1, 在运动矢量差分值前传送依据式 (5) 得到的帧内全局不变量 $\text{MVD}_{\text{const}} = [\text{MVD}_x, \text{MVD}_y]$;

步骤6 如果 $\text{sgn}(D_{(d,i)}) = \text{sgn}(\text{MVD}_d)$, 则计算 $\text{TD}_{(d,i)} = D_{(d,i)} - \text{MVD}_d$ 且置 global_d 为1; 否则计算 $\text{TD}_{(d,i)} = D_{(d,i)}$ 且置 global_d 为0。这里 d 表示 x, y 两个分量, i 表示当前处理的宏块序号;

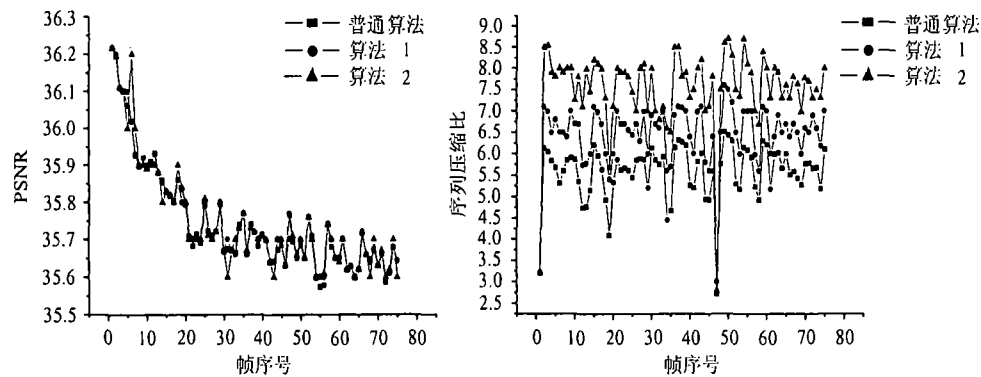
步骤7 传送 $\text{TD}_i = [\text{TD}_{(x,i)}, \text{TD}_{(y,i)}]$; 用步骤6 处理完当前帧所有宏块后回到步骤1。

4 仿真结果

为了验证本文算法的适用性及实际性能, 我们利用 C++ 在 PC 机上实现了分别采用算法1, 算法2 的基于 H.263+ 标准的视频压缩软件平台。试验所用计算机为 PIII 1GHz, 256M 内存, 软件环境为 C++ Builder 6.0, 通过对部分标准图像序列的测试, 当采用变速率输出模式

(VBR)，量化步长取定值 5，帧内 / 帧间编码判断阈值为 20 时，每个序列处理 75 帧所需时间如表 1 示。针对存在全局运动的 Garden 序列，其压缩比及信噪比曲线如图 2 示。

表 1 算法处理时间对比表 (单位: s)				
图像序列	算法 1	算法 2	H.263+ 算法	说明
Claire	278	243	208	CIF 格式、灰度
Garden	272	230	174	SIF 格式、彩色、全局运动
Football	254	221	179	SIF 格式、彩色
Mobile	258	218	162	SIF 格式、彩色、全局运动



(a) Garden 序列 PSNR-帧序号曲线 (b) Garden 序列压缩比-帧序号曲线

图 2

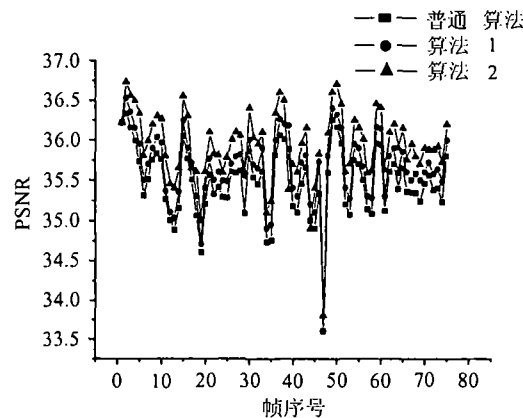


图 3 相同压缩比 Garden 序列 PSNR-帧序号曲线

可以看出，由于是 VBR 模式，量化步长及帧内 / 帧间编码判断阈值保持不变，使得重建图像 PSNR 没有多大变化，但压缩比即编码效率则有较大程度的提高。如果采用恒定码速率输出模式 (CBR) 即通过缩短量化步长、减小帧内 / 帧间编码判断阈值、提高强制帧内编码的频率等方法保证相同数据压缩比时，算法 1 及算法 2 的重建图像 PSNR 值分别有平均 0.2dB 和 0.5dB 的提高，如图 3 示。试验中发现，修改 T_{frame} 及 MVD_{frame} ，会得到不同的算法性能。以

算法 1 为例，即使是对 Garden 及 Mobile 等存在全局运动的序列，仍然会采用 $T_{\text{frame}} = M_{\text{num}}/2$ ，则一些存在全局运动的序列会被误判为普通图像，从而限制了本文算法的应用领域。修正后的算法只判断 N_+ 、 N_- 差的绝对值，只要大于一个定值，则认为该图像序列存在全局运动。同理算法 2 也可以被修改为只判断 dN_+ 、 dN_- 差的绝对值，只要大于阈值 MVD_{frame} ，则采用相应的全局运动估计方法。

将本文算法与 Kui Zhang 及 H. Richter 的方法相比，采用 Garden 标准图像序列，VBR 模式下仿真结果如表 2 示。显然，本文提出的自适应混合全局运动、局部运动估计算法更具有运算复杂性低、处理延时小、编码效率高、利于硬件实现等方面的优势。

表2 不同算法的编码性能比较

算法	第10帧			第20帧		
	处理时间(s)	PSNR(dB)	压缩比	处理时间(s)	PSNR(dB)	压缩比
算法1	3.2	35.89	6.72	3.4	35.79	6.40
算法2	2.8	35.92	7.06	3.0	35.80	6.90
Kui Zhang	15.2	36.02	6.65	15.6	35.82	6.32
H. Richter	4.6	35.96	6.76	4.6	35.83	6.44

5 小结

传统的全局运动估计算法通常是通过构造仿射参数或双线性图像运动模型，依靠连续多帧图像中特征点数据，采用迭代或待定系数方法求解运动参数，然后将运动完整分隔成全局运动和局部运动并分别加以处理，其算法复杂性高、运算量大、处理延时长，精度受迭代特征点选取的影响，且无法消除高频信息丢失的现象，不利于实时视频压缩。基于此，本文摒弃构造及分析图像帧运动模型的方法，最先提出利用全局运动带来的时间冗余性在帧内运动矢量及其差分编码量统计特性上的反映，通过求解基本平移矢量和帧内全局不变性，采用混合全局运动、局部运动估计的方法，自适应地处理各种不同特征的源数据。通过软件仿真，将本文提出的自适应全局运动估计算法实现在基于 H.263+ 的视频编码器中，并与 H.263+ 标准算法、基于 Kui Zhang 和 H. Richter 等人传统算法的 H.263+ 编码器做了比较，结果表明与标准 H.263+ 的编码器相比，本文算法编码效率有约 20% 的提高，在保证相同压缩比时，重建图像质量约有 0.5dB 的改善；与传统方法相比，本文算法复杂性低、处理延时小、更利于硬件实现，不受匹配特征点选取的影响，在 PSNR 变化不大的时，编码效率有较大的改善。

参 考 文 献

- [1] Tse Y T, Baker R L. Global zoom/pan estimation and compensation for video compression. Proc. ICASSP, Toronto, Canada, May 1991: 2725-2728.
- [2] Hoetter M. Differential estimation of the global motion parameters zoom and pan. *IEEE Trans. on Signal Processing*, 1989, 16(2): 249-265.
- [3] Kook-yeol Yoo, Jae-kyoon Kim. A new fast local motion estimation algorithm using global motion. *Signal Processing*, 1998, 68(2): 219-224.
- [4] Dufaux F, Konrad J. Efficient, robust and fast global motion estimation for video coding. *IEEE Trans. on Image Processing*, 2000, IP-9(3): 497-501.
- [5] Yuwen He, Bo Feng, Shiqiang Yang, Yichuo Zhong. Fast global motion estimation for global motion compensation coding. ISCAS 2001, Sydney, Australia, May 2001: 233-236.
- [6] Kook-yeol Yoo, Jae-kyoon Kim. New motion estimation and compensation algorithms for video compression combining global and local motions. *Signal Processing: Image Communication*, 1999, 11(15): 201-216.
- [7] Kui Zhang, Josef Kittler. Global motion estimation and robust regression for video coding. Proc. Int'l Conf. Acoustics, Speech, and Signal Processing, Seattle, WA, May 1998: 241-244.
- [8] Richter H, Smolic A, Stabemack B, Muller E. Real time global motion estimation for MPEG-4 video encoder. Proc. PCS'2001, Picture Coding Symposium, Seoul, Korea, April 2001: 197-200.
- [9] Joerg Heuer, Andre Kaup. Global motion estimation in image sequences using robust motion vector field segmentation. Proceedings ACM Multimedia 99, Orlando, Florida, 30 October-5 November 1999: 261-264.
- [10] ITU-T. Draft ITU-T recommendation H.263: video coding for low bitrate communication. International Telecommunication Union, 26 September 1997.

张 方：男，1977 年生，博士生，主要从事图像压缩编码、网络视频流业务及无限视频等方面的研究工作。
 肖 蕊：女，1977 年生，在职博士生，主要从事图像压缩编码、联合信源信道编码等方面的研究工作。
 吴成柯：男，1938 年生，教授，博士生导师，主要从事图像处理、图像通信、计算机视觉等方面的研究。