

基于自适应块尺寸的 H.264 时域错误隐藏算法

张明, 毕笃彦, 李成, 吕丹

(空军工程大学 工程学院信号实验室, 陕西 西安 710038)

摘要: 为提高在视频通信中重建图像质量, 充分利用运动矢量的空间相关性、相邻宏块的编码模式信息, 提出了一种新的 H.264 时域错误隐藏算法。该算法根据误码块周围的邻块模式, 自适应决定隐藏块的尺寸; 对于每个尺寸块, 都从多个候选运动矢量集合中选取使外边界对应像素差值最小的运动矢量; 并利用运动估计技术更精确地恢复丢失宏块的运动矢量, 从而提高所恢复视频信号的质量。实验结果表明: 对于不同运动类型序列和不同宏块丢失率, 该算法不仅计算量较小, 而且恢复后的图像质量均优于传统的时域错误隐藏算法。

关键词: 时域错误隐藏; H.264; 外边界匹配; 宏块模式; 运动估计

中图分类号: TP391

文献标识码: A

文章编号: 1001-2400(2007)S1-0095-05

Robust temporal error concealment for H.264 with adaptive block sizes

ZHANG Ming, BI Du-yan, LI Cheng, Lü Dan

(Sig. Lab of Engineering School, AFEU, Xi'an 710038, China)

Abstract: To improve the decoded quality of videos when bit stream data incur transmission errors, an effective temporal error concealment algorithm is proposed for H.264 coded video. It uses not only motion vectors and reference frames but the modes of macroblocks adjacent to the lost macroblock as well. Each lost macroblock is concealed on the basis of different block sizes depending on the modes of neighboring macroblocks. Then the motion vector with the minimum value of external boundary match distortion is selected for each block from a list of motion vector candidates. Finally, motion estimation is employed to more accurately recover the lost motion vectors and improve the video quality. For video sequences with different motions and various lost rates of inter-coded macroblocks, experimental results show the proposed technique can obtain a better video quality than conventional temporal concealment techniques and the computational complexity is very low.

Key Words: temporal error concealment; H.264; external boundary matching; macroblock mode; motion estimation

随着移动视频通信的广泛应用, 实现高质量和稳定的视频传输成为研究人员追求的目标。视频码流在有损信道中传输将不可避免会出现误码或丢包现象, 解码错误会在时间和空间上迅速扩散, 严重降低主观视觉效果。错误隐藏技术针对视频应用的特点, 利用视频信号在空间域和时间域存在的相关性, 以及人为添加的冗余信息, 恢复出人眼可以接受的视频图像, 成为近年来移动视频传输研究中的热点。

当帧间编码宏块丢失时, 一般使用时域误码掩盖方法, 其关键是运动矢量的估计。估计运动矢量主要利用运动矢量的空间和时间相关性, 常见方法是令其为相邻宏块运动矢量、它们的均值或中值、零矢量或前帧中对应位置的运动矢量等^[1], 这些方法仅在运动矢量场平滑的情形下隐藏效果较好; 文[2]提出更为精确的基于边界匹配的算法(boundary matching algorithm, BMA), 从多个候选运动矢量中选择最佳运动矢量; 文[3]中进一步将BMA准则替换为外边界匹配准则(external boundary matching algorithm, EBMA), 进一步提高了运动矢量的准确度。在最新的H.264^[4]视频编码标准中, 帧间预测引入了多参考帧、1/4 像素运动估计精度、灵活块分割等新特性, 极大地提高了压缩效率。参考模型JM10.2 采用的错误隐藏算法只是前述方法的简单延伸, 没有充分考虑这些新特性, 图像质量有待进一步改善。

针对 H.264 视频编码标准的特点, 笔者提出的新算法充分利用了块编码模式、运动矢量的空间相关性、1/4 像素运动矢量的精度以及边界像素的信息, 更为精确地恢复丢失宏块的运动矢量, 从而提高所恢复视频信号的质量。

1 JM 算法

在 H.264 参考模型 10.2 中, 时域错误隐藏算法首先要统计当前帧中正常解码宏块的运动强度。如果平均运动

收稿日期: 2007-06-03

基金项目: 国家高技术研究发展计划(863)资助项目(2006AA701121)

作者简介: 张明(1981-), 男, 空军工程大学博士研究生。

矢量小于预定门限, 则所有误码块都为零运动矢量, 即错误数据直接用参考帧中对应位置的图像数据替代; 否则, 使用有运动补偿的时域隐藏方式, 需参考周围邻块的运动矢量及参考帧信息。

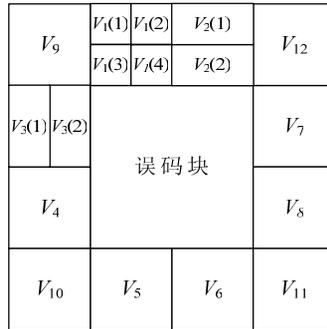


图 1 空域候选运动矢量

如图 1 所示, 与误码块相邻的 8×8 块共有 12 个, 其运动矢量分别用 $V_1 \sim V_{12}$ 表示。由于 H.264 中支持比 8×8 更小尺寸的编码块模式, 因而存在某些 8×8 块中存在指向同一参考帧的多个运动矢量, 如图 1 中 1~3 块所示。此时, V_i 应取其内部子块运动矢量的均值。即

$$V_i = (d_x, d_y, \text{ref}) = (\text{Mean}(V_i(1), V_i(2), \dots)) \quad (1)$$

其中 d_x 、 d_y 分别为运动矢量的水平、垂直分量。ref 为参考帧标号。JM 就从下面集合候选运动矢量中, 采用 BMA 准则选取使边界像素和误码块的相邻像素差值最小的运动矢量。

$$\{ZM, V_1, V_2, \dots, V_8\} \quad (2)$$

其中 ZM 代表零运动矢量 $(0, 0, 0)$ 。匹配误差计算如下

$$C_{\text{BMA}} = \sum_{x=x_0}^{x_0+M-1} \left[\left| f_{x, y_0-1} - f_{x+d_x, y_0+d_y}^{\text{ref}} \right| + \left| f_{x, y_0+N} - f_{x+d_x, y_0+N-1+d_y}^{\text{ref}} \right| \right] + \sum_{y=y_0}^{y_0+N-1} \left[\left| f_{x_0-1, y} - f_{x_0+d_x, y+d_y}^{\text{ref}} \right| + \left| f_{x_0+M, y} - f_{x_0+M-1+d_x, y+d_y}^{\text{ref}} \right| \right] \quad (3)$$

其中 (x_0, y_0) 是误码块左上角像素的坐标。f, f^{ref} 分别为当前帧和参考帧。M, N 分别为的宽度和高度。JM 中时域错误隐藏都是宏块大小, 即 $M=N=16$ 。

2 新算法

文中方法是对 JM 算法的改进和拓展: 根据邻块编码模式决定误码块的错误隐藏模式, 突破了 JM 中只能以宏块为单元的限制; 充分利用了周围 8 个相邻宏块的编码信息, 增加了候选运动矢量数量; 将 BMA 准则替换为 EBMA, 减少了误判的概率; 将运动估计过程引入错误隐藏过程中, 进一步提高了估计精度。

2.1 选择隐藏模式

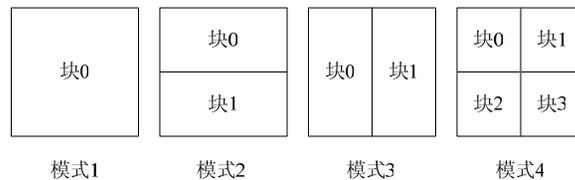


图 2 4 种时域隐藏模式

JM 中总是按照 16×16 大小的块进行错误隐藏, 即每个宏块只估计一个运动矢量。对于既包含运动物体和静止背景的宏块, 恢复效果不佳。这时就可以将宏块分为多个小块(如 4 个 8×8 块), 对每个小块分别预测一个运动矢量, 将会明显提升图像质量。然而, 如果每个误码块都分成多个子块, 对于静止块和准静止宏块, 各子块的运动矢量可能互不相同, 反而会造成一定的块效应, 影响主观图像质量。

H.264 支持 7 种不同尺寸的编码方式，编码宏块时都要选择最好的预测模式作为最终的编码模式。由于知道周围邻块的编码模式，则可以利用这些模式判断误码块的隐藏模式。如图 2 所示，使用 16×16 ， 16×8 ， 8×16 ， 8×8 4 种块尺寸的隐藏模式，分别记为模式 1~4。设上边、左边、下边、右边的相邻宏块分别为 A ， B ， C ， D 。判别过程如下：

- ① 如果 A ， C 都为 16×16 或 16×8 编码模式，且 B ， D 都为 16×16 或 8×16 编码模式（ 16×16 模式包括 skip 和 direct 模式），则使用隐藏模式 1。
- ② 如果 B 和 D 中存在 16×8 编码模式，且其余相邻宏块都为 16×16 模式，则使用隐藏模式 2。
- ③ 如果 A 和 C 中存在 8×16 编码模式，且其余相邻宏块都为 16×16 模式，则使用隐藏模式 3。
- ④ 其余情形下，使用隐藏模式 4。

2.2 候选运动矢量

JM 中仅使用上下左右 4 个邻块的运动信息，即 $V_1 \sim V_8$ 。为了提高估计精度，必须参考更多的运动矢量信息。候选运动矢量可以参考当前帧中邻块的运动信息（空间相关性），也可以参考前一种相同位置块的运动信息（时间相关性）。由于后者需要保存前帧中所有块的运动矢量，内存需求较大，而且运动矢量的时间相关性通常小于空间相关性^[5]，所以本算法只考虑了周围邻块的运动矢量及其加权均值。

算法参考了误码块周围 8 个邻块的运动矢量 $V_1 \sim V_{12}$ ，并针对各种情形对候选运动矢量进行了筛选，每个块只选择 8 邻域相连的运动矢量，在保证图像质量的基础上降低了计算复杂度。即

$$\text{模式 1: 块 } 0, \quad \{ZM, V_1, V_2, \dots, V_{12}\}, \quad (4)$$

$$\text{模式 2: 块 } 0, \quad \{ZM, V_1, V_2, V_3, V_4, V_7, V_8, V_9, V_{12}\}, \quad (5)$$

$$\text{模式 3: 块 } 0, \quad \{ZM, V_1, V_2, V_3, V_4, V_5, V_6, V_9, V_{10}\}, \quad (6)$$

$$\text{模式 4: 块 } 0, \quad \{ZM, V_1, V_2, V_3, V_4, V_9\}, \quad (7)$$

其余块情形可以类推。

在以往的算法中，大都是取 $V_1 \sim V_8$ 或 $V_1 \sim V_{12}$ 的均值和中值作为候选运动矢量。由于参与运算的矢量较多，预测效果不佳。这里则计算水平、垂直方向上的线性插值运动矢量 V_h 、 V_v ，计算如下：

$$\text{模式 1: 块 } 0, \quad V_h = (V_3 + V_4 + V_7 + V_8)/4, \quad V_v = (V_1 + V_2 + V_5 + V_6)/4, \quad (8)$$

$$\text{模式 2: 块 } 0, \quad V_h = (V_3 + V_7)/2, \quad V_v = (2(V_1 + V_2) + V_5 + V_6)/6, \quad (9)$$

$$\text{模式 3: 块 } 0, \quad V_h = (2(V_3 + V_4) + V_7 + V_8)/6, \quad V_v = (V_1 + V_5)/2, \quad (10)$$

$$\text{模式 4: 块 } 0, \quad V_h = (2V_3 + V_7)/3, \quad V_v = (2V_1 + V_5)/3, \quad (11)$$

其余块情形可以类推。

2.3 匹配准则

如前所述，在对每个运动矢量进行评估时，JM 算法中使用了 BMA 准则，就是计算误码块边界像素的平滑度。BMA 的缺点是参与计算的像素个数较少，对 16×16 宏块，只有 64 个，对 16×8 ， 8×16 ， 8×8 块而言，个数更少。因此计算时容易受到噪声的干扰，不能估计出最佳的运动矢量。另外，如果宏块边界处本身就存在物体边缘，BMA 的匹配效果将更差。

文中采用外边界匹配准则（EBMA），计算参考块的相邻像素和误码块相邻像素的绝对差和。即

$$C_{\text{EBMA}} = \sum_{x=x_0-l}^{x_0+M+l-1} \sum_{y=y_0-l}^{y_0+N+l-1} |f_{x,y} - f_{x+d_x, y+d_y}^{\text{ref}}| p(x, y), \quad (12)$$

$$p(x, y) = \begin{cases} 0, & x_0 \leq x < x_0 + M, y_0 \leq y < y_0 + N, \\ 1, & \text{其他,} \end{cases}$$

其中 l 是边界像素的宽度，一般在 1~8 之间变化，通过调整 l 的大小可以灵活控制运算像素的数目。文[6]研究指

出,对于宏块模式而言,宽度 2~8 之间变化时,对估计精度的影响不大。为了便于计算及适应各种隐藏模式,对所有块都使用宽度为 4 的 EBMA 准则。

2.4 运动估计

序列图像的运动矢量在空间和时间上具有很强的相关性,特别是属于同一对象的块运动保持一致的可能性更大。经过前面步骤得到的运动矢量就非常接近于真正的编码运动矢量,以该矢量为中心进行运动估计可以进一步提高图像质量。

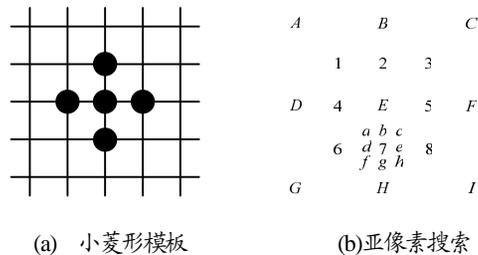


图 3 运动估计示意图

运动估计算法以 EBMA 为匹配准则,分为整像素和亚像素运动估计两个过程。整像素阶段以图 3(a)所示的小菱形模板反复搜索,直到最小误差点位于模板中心时结束。亚像素阶段以得到的最佳整像素精度运动矢量为中心,先进行 1/2 亚像素搜索,再 1/4 亚像素搜索。如图 3(b)所示,大写字母代表整像素位置,阿拉伯数字表示 1/2 像素位置,小写字母表示 1/4 像素位置。假设 E 处为最佳的整像素位置,则在 1/2 像素搜索时将会搜索位置 1, 2, 3, 4, 5, 6, 7, 8; 假设 7 处为 1/2 像素最佳匹配位置,则在 1/4 像素搜索时将会检测 a, b, c, d, e, f, g, h, 所以共有 16 个亚像素位置被搜索。

表 1 4 种算法在不同宏块丢失率下的峰值信噪比

序列	PSNR /dB	宏块丢失率					
		0%	2%	5%	10%	15%	20%
foreman	PSNR _{ZMV}	38.01	35.07	32.83	30.43	29.02	27.99
	PSNR _{JM}		36.49	35.20	33.52	32.23	31.28
	PSNR _{EECMS}		36.89	35.83	34.58	33.21	32.43
	PSNR _{PRO}		36.99	36.02	34.81	33.60	32.74
mobile	PSNR _{ZMV}	35.74	30.71	27.72	25.05	23.29	22.21
	PSNR _{JM}		32.27	29.92	27.54	26.08	24.88
	PSNR _{EECMS}		33.40	31.70	29.77	28.59	27.53
	PSNR _{PRO}		33.69	31.98	29.84	28.53	27.44
stefan	PSNR _{ZMV}	37.04	29.60	26.69	24.29	22.98	22.00
	PSNR _{JM}		33.58	31.19	29.00	27.83	26.71
	PSNR _{EECMS}		34.22	32.38	30.24	28.90	27.77
	PSNR _{PRO}		34.57	32.56	30.40	29.10	27.98
flower	PSNR _{ZMV}	36.74	29.12	25.86	23.47	22.02	21.09
	PSNR _{JM}		33.26	31.03	28.85	27.30	25.94
	PSNR _{EECMS}		34.42	32.86	30.90	29.38	28.37
	PSNR _{PRO}		34.77	33.20	31.39	29.89	28.93

3 实验

采用 H.264 的参考软件 JM10.2 对 CIF 格式序列 foreman, mobile, stefan, flower 进行仿真实验。每个序列编码前 100 帧, IPPP 结构, I 帧周期为 12。QP 为 26, 一个参考帧。I 帧没有包丢失, P/B 帧的宏块丢失率分别有 2%, 5%, 10%, 15%, 20% 5 种情况。采用 4 种算法进行比较: 算法 1 将丢失宏块的运动矢量均恢复为 0, 简称为

ZMV (zero motion vector)。算法 2, 3 分别为 JM 和文献[6]的 EECMS 算法, 算法 4 则是笔者提出的错误隐藏算法。所比较的均为 Y 分量的峰值信噪比 PSNR, 其结果如表 1 所示。

从表 1 可以看出, 在测试中对于不同运动类型的序列和不同的宏块丢失率, 新算法都明显优于其余 3 种算法, 是一种鲁棒性很强的算法。对于运动较为复杂 (如 flower) 和较多细节 (mobile) 的序列, 经 EBMA 准则和运动估计后的运动矢量更为精确, 因此新算法的相对效果更好。flower、mobile 序列在宏块丢失率大于 5% 的时候, 相对 JM 算法亮度信号的 PSNR 能够提高 2dB 以上。图 4 给出了 flower 序列在 10% 丢失率时, 各种算法逐帧的性能比较。I 帧没有宏块丢失, 所以 4 种算法的 PSNR 值都相同。而随着预测帧数目的增加, 文中算法的性能上也越来越优于 ZMV, JM 以及 EECMS 算法。

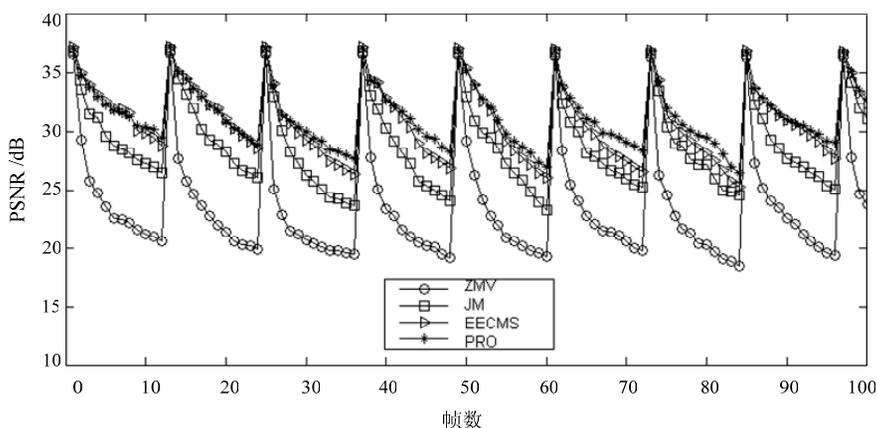


图 4 flower 序列在 10% 宏块丢失率时 4 种算法的逐帧性能比较

4 结束语

提出了一种 H.264 中基于边界匹配准则及运动估计的时域错误隐藏算法, 并充分利用了运动矢量的空间相关性、H.264 中宏块编码模式及 1/4 像素精度等多种信息, 能够获得比传统算法更好的视频质量, 在运动复杂的视频序列及宏块丢失率较高的情形下, 改进效果尤其明显。H.264 码流在高误码信道中传输通信时, 该算法对于提升图像质量具有重要实用价值。

参考文献

- [1] Wang Y, Zhu Q F. Error Control and Concealment for Video Communication: a Review[J]. IEEE Proc, 1998,86(5): 974-997.
- [2] Lam W M, Reibman A R. Recovery of Lost or Erroneously Received Motion Vectors[C]//Proc ICASSP'93. Minneapolis: IEEE, 1993: 417-420.
- [3] Zhang J, Arnold J F, Frater M R. A Cell-loss Concealment Technique for MPEG-2 Coded Video[J]. IEEE Trans on Circuits and Systems for Video Technology, 2000, 10(4): 659-665.
- [4] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 \ ISO/IEC 14 496-10AVC) Joint Video Team (JVT)[S]. 2003, Doc. JVT-G050.
- [5] Antonio C, Clsudia F, Pau D. A Complexity-bounded Motion Estimation Algorithm[J]. IEEE Trans on Image Processing, 2002,11(4): 387-392.
- [6] Agrafiotis D, Bull D R, Canagarajah N. Enhanced Error Concealment with Mode Selection[J]. IEEE Trans on Circuits and Systems for Video Technology, 2006,16(8): 960-973.