

基于受损区域运动特征的时域错误隐藏算法^①

赵连凤^② 郭宝龙 韩合民

(西安电子科技大学机电工程学院 ICIE 研究所 西安 710071)

摘 要 为解决视频压缩数据在网络传输中的误码或丢包问题,提出了一种适用于数字视频编码标准 H.264 的时域错误隐藏算法。该算法结合 H.264 中多尺寸编码块模式的特点,通过对受损块邻域块的模式分析来判定受损块模式,并针对各子块的运动类型分别建立相应的运动矢量候选集,从而选择最优运动矢量重建受损块。实验结果表明,该算法有效增强了对复杂运动区域重建的适应性,在不同的网络丢包率环境下均可实现较好的主观和客观重建图像质量。

关键词 H.264, 错误隐藏, 块模式, 运动矢量候选集

0 引 言

随着视频点播(VOD)和可视电话等多媒体业务的开展,多媒体压缩数据在网络传输中的一个亟待解决的问题就是误码或丢包^[1,2]。由于数字视频编码标准 H.264^[3,4]采用了与以往标准相同的编码框架,因此对网络传输产生的错误同样非常敏感。解决这个问题的方法之一是错误隐藏,其基本思想是解码器端采用后处理手段以尽量减少错误对重建图像质量的影响。错误隐藏可以分为空域错误隐藏和时域错误隐藏。其中时域错误隐藏主要应用于帧间图像发生错误的情况下,利用图像序列的连续性,恢复出受损块的运动矢量并用其运动补偿块来代替受损块进行错误隐藏。

H.264 中采用的时域错误隐藏算法^[1]是基于宏块的,对于图像的静止或运动较小部分,恢复的图像质量较好,但对于复杂运动区域及场景切换的情况下,该算法会导致出现明显的方块效应,进而造成图像的不连续。针对这一问题,本文提出了一种基于受损区域运动特征的时域错误隐藏算法。该算法结合 H.264 多尺寸编码块模式的特点,通过对受损块的邻域块模式进行分析来判定受损块模式;同时充分利用运动矢量(motion vector, MV)的空间与时间相关性,根据帧间运动矢量的变化趋势来更为精确地恢复受损块的运动矢量;最后针对不同的模式类型建立各子块相应的运动矢量候选集并选择最优的运

动矢量重建受损块。通过对受损块内部运动特征的分析 and 仿真实验可以看出,该算法提高了对受损区域内的精细运动的处理能力,增强了对运动细节部分的恢复性能。

1 基于受损区域运动特征的时域错误隐藏算法

基于受损区域运动特征的时域错误隐藏算法是一种利用视频图像时间及空间相关性的后处理错误隐藏算法。其特点是充分考虑了受损区域内部的运动特征,针对受损块的模式类型来建立相应的运动矢量候选集,分别对各子块进行恢复,从而克服基于宏块级的错误隐藏对于运动区域恢复的局限性,提高对运动细节部分的恢复性能。

1.1 受损块模式判定

在 H.264 中采用树状结构运动补偿,即允许当前宏块分割为 16×16 、 16×8 、 8×16 、 8×8 四种模式,而 8×8 模式的每个子宏块还可以进一步进行分割。H.264 运用这种多模式的灵活和细致的运动预测块划分,增强了处理精细运动细节的能力,有效地提高了编码效率。

由于相邻宏块间具有空间相关性,且宏块的树状结构划分进一步显示了各宏块间的关联,所以可通过对受损块邻域块模式的分析来判定受损块模式,从而将宏块中不同运动类型的区域划分开来,分别进行恢复处理。由文献[5]中的分析可知,当存在

^① 863 计划(2006AA01Z127),国家自然科学基金(60572152)和陕西省自然科学基金(2005F26)资助项目。
^② 女,1983 年生,硕士;研究方向:视频压缩编码,多媒体通信;联系人, E-mail: lianfengzhao@163.com
(收稿日期:2007-07-23)

边缘穿越宏块时,该宏块会进行进一步划分。如图1所示,若A块(C块)为 8×16 或 8×8 模式时,受损块内部可能存在垂直方向边缘,相应的模式类型为 8×16 ;同理,若B块(D块)为 16×8 或 8×8 模式时,受损块内部可能存在水平方向边缘,相应的模式类型为 16×8 ;当同时存在水平与垂直方向边缘时,宏块划分为 8×8 模式。

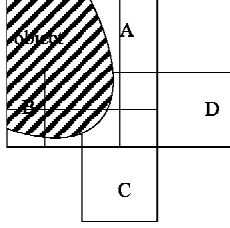


图1 受损块模式划分示例

受损块模式判定具体分析如下:

采用 $flag$ 作为邻域宏块模式类型标识,若A块(C块)为 8×16 或 8×8 模式,设 $flag_A(flag_C)$ 为1,否则设为0;若B块(D块)为 16×8 或 8×8 模式,设 $flag_B(flag_D)$ 为1,否则设为0,受损块模式为 $mode$ 。

则

$mode = 1$ (16×16 模式);

else if ($(flag_A \&\& flag_C == 0) \&\& (flag_B \parallel flag_D == 1)$)

$mode = 2$ (16×8 模式);

else if ($(flag_A \parallel flag_C == 1) \&\& (flag_B \&\& flag_D == 0)$)

$mode = 3$ (8×16 模式);

else

$mode = 4$ (8×8 模式);

通过对受损块模式的分析判定,可以对宏块内部结构进行初步预测,对属于不同对象的部分进行划分,从而有效运用各子块与邻域块间的相关性信息,并以各子块为单位进行错误隐藏来重建受损块。

1.2 帧间预测运动矢量生成

H.264 中的错误隐藏算法利用运动矢量的空域相关性,使用边界匹配准则选择邻域块中最优的运动矢量作为受损块的运动矢量进行错误隐藏。然而在空域低相关区域(如运动对象边缘等),该运动矢量的恢复效果并不理想,这是因为此种情况下邻域运动矢量并不能正确描述当前块的运动。因此,本算法充分考虑了运动矢量从前一帧到当前帧的变化趋势,并基于这种变化趋势插值生成受损块运动矢

量,增强了预测运动矢量的可靠性。

图2示出了帧运动矢量变化的趋势,其中, MV_{CT} 、 MV_{CB} 、 MV_{CL} 、 MV_{CR} 分别为受损块邻域上方、下方、左方、右方的运动矢量, MV_{PO} 、 MV_{PT} 、 MV_{PB} 、 MV_{PL} 、 MV_{PR} 分别为前一帧中相应位置宏块及邻域上方、下方、左方、右方的运动矢量。将上述运动矢量表示为极坐标形式,则当前帧受损块邻域运动矢量表示为 $MV_C^* = (r_C^*, \theta_C^*)$,前一帧中相应位置宏块邻域运动矢量表示为 $MV_P^* = (r_P^*, \theta_P^*)$,前一帧中相应位置宏块运动矢量表示为 $MV_{PO} = (r_{PO}, \theta_{PO})$ 。其中 r 为矢量幅值, θ 为矢量方向。通过对 MV_C^* 、 MV_P^* 的分析,可以分别得到两帧相应位置运动矢量的尺度变化与方向变化信息,则根据变化趋势可插值得到当前受损块预测运动矢量为

$$MV_{CO}^* = \left(\frac{r_C^*}{r_P^*} r_{PO}, \theta_{PO} + \theta_C^* - \theta_P^* \right) \quad (1)$$

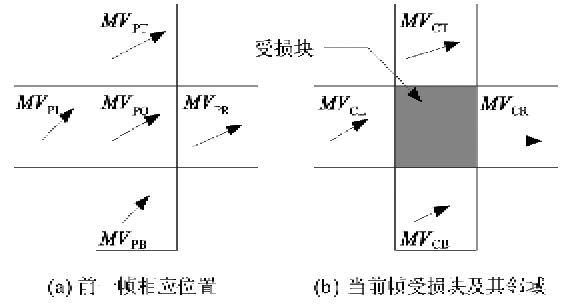


图2 帧间运动矢量变化趋势

在实际应用中,可根据邻域块的可用性(非受损块且符合已划分模式邻域,图2所示为 16×16 模式且邻域块均无损接收)对受损块各子块分别进行的运动矢量预测。引入帧间运动矢量的变化趋势用于预测运动矢量的生成,可进一步利用帧间信息,从而可更为精确地恢复受损块的运动矢量。

1.3 运动矢量候选集的建立及最优运动矢量选择

经过模式判定,受损块可分为 16×16 、 16×8 、 8×16 、 8×8 四种模式类型,相应各子块邻域的可用预测运动矢量如图3所示,同时通过帧间矢量变化分析,可插值生成各子块帧间预测运动矢量,基于以上工作,建立子块运动矢量候选集,其组成包括:

- (1) 零矢量;
- (2) 前一帧相应位置宏块运动矢量 MV_{PO} ;
- (3) 邻域可用预测运动矢量 MV_C^* ;
- (4) 邻域可用预测运动矢量均值 $\overline{MV_C^*}$;
- (5) 帧间预测运动矢量 MV_{CO}^* ;
- (6) 帧间预测运动矢量均值 $\overline{MV_{CO}^*}$ 。

其中(1)和(2)适用于静止及平滑运动区域,其余情况对应复杂运动区域;(3)和(4)主要利用空域相关性,(4)和(5)则考虑了帧间运动矢量的变化特点。针对模式类型建立相应的各子块矢量候选集,可以基于受损区域内部不同对象的运动类型分别进行子块重建,提高对细节运动区域的处理能力。

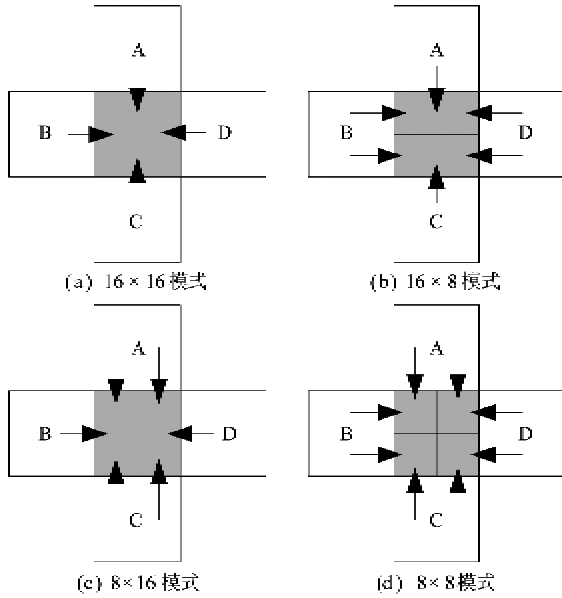


图3 受损块邻域可用预测运动矢量

本算法采用边界匹配算法^[1](border match algorithm, BMA)选择运动矢量候选集中最优的运动矢量用于受损块重建,即首先使用每一个预测矢量进行运动补偿,然后计算补偿块的内外边界像素变化差值,使变化最小的候选运动矢量即为最优的运动矢量。边界匹配函数如下:

$$\begin{aligned} \text{cost}_{\text{BM}} = & \sum_{x=x_0}^{x_0+N-1} (|P_{x,y_0-1}^c - P_{x+MV_x,y_0+MV_y}^p| \\ & + |P_{x,y_0+N}^c - P_{x+MV_x,y_0+N-1+MV_y}^p|) \\ & + \sum_{y=y_0}^{y_0+N-1} (|P_{x_0-1,y}^c - P_{x_0+MV_x,y+MV_y}^p| \\ & + |P_{x_0+N,y}^c - P_{x_0+N-1+MV_x,y+MV_y}^p|) \end{aligned} \quad (2)$$

其中, (x_0, y_0) 为当前受损块左上角像素坐标, P^c 与 P^p 分别为当前帧与前一帧像素。式(2)为 16×16 模式且邻域块均无损接收情况下的边界匹配函数,其他模式子块对应的边界匹配函数应基于空域的平滑性相应进行调整。

1.4 计算复杂度分析

与 H.264 验证模型(JM)采用的时域隐藏方法

相比,本算法计算量主要在以下两方面有所增加。第一个方面是帧间运动矢量的生成。由 1.2 节中可知,将一运动矢量转换为极坐标形式需 3 次乘/除运算及 1 次加/减运算,插值生成 1 帧间运动矢量需 2 次乘/除运算及 2 次加/减运算。因此,要生成 1 组帧间运动矢量(i 个),需要 $(5i + 3)$ 次乘/除运算及 $(3i + 1)$ 次加/减运算。各模式具体计算量见表 1。第二个方面是最优运动矢量的选择。由于边界匹配函数需要计算候选集中各矢量用于补偿后的边界变化差值,所以在矢量的选取过程中需要大量的加、减运算。表 2 所示为各模式获取最优运动矢量所耗损的计算量及与 JM 进行的对比。如表中数据所示,由于本算法对受损块进行模式划分,针对各子块建立运动矢量候选集,并引入了帧间预测矢量来丰富矢量集,使计算量比 JM 有了一定的增加。

表1 生成帧间运动矢量所耗损的计算量

模式类型	帧间运动矢量个数	运算次数	
		乘/除	加/减
16×16	4	23	13
16×8	6	33	19
8×16	6	33	19
8×8	8	43	25

表2 选择最优运动矢量所耗损的计算量

模式类型	运动矢量个数	运算次数		
		加	减	比较
16×16	12	756	768	12
16×8	10×2	620	640	20
8×16	10×2	620	640	20
8×8	8×4	480	512	20
JM	9	567	576	9

根据算法需要,需对前一帧运动矢量进行存储。设宏块尺寸为 $n \times n$, 帧尺寸为 $M \times N$, 每个矢量分量需 d 个字节,则每帧用于运动矢量存储的空间为 $2MNd/n^2$ 字节。对于 QCIF 格式(176×144),宏块大小为 16×16 ,采用 32 位空间存储矢量分量的情况,每帧运动矢量存储需占用约 0.8kB 空间,实际中较容易实现这一要求。

2 仿真实验

本文实验采用 JVT 的 JM90^[6]作为测试平台,分别对 QCIF 格式的 Carphone、Foreman、Salesman 及 Mobile 序列进行测试,编码序列采用 IPPP 的帧结构,测

试序列长度均为 50 帧。为了验证本算法的有效性,选取时域替换方法 (TR)、JM 模型时域隐藏算法 (JM) 与本文算法分别在网络丢包率为 5%、10%、15% 的环境下进行效果比较。

图 4 所示为宏块丢失率为 15% 时采用各方法对 Carphone 第 64 帧的隐藏效果。采用简单的时域的替换会造成边缘明显的不连续,但对于相对静止的部分恢复比较好(图(c))。JM 模型所采用的时域隐藏算法由于对运动矢量进行了有效的预测,图像重建效果较好,但在运动区域边缘处隐藏效果并不理想(图(d))。本文算法由于充分考虑了受损区域内各子块的运动趋势并分别重建,增强了对运动区域恢复的适应性。如图(f)所示的唇部隐藏效果,由于受损区域包括变化较大的唇形及变化较小的面部部分,采用以宏块为单位的隐藏方法忽视了区域内部的运动,导致重建后唇部的变形;而本文算法对受损区域内部不同的运动部分进行了划分,解决了块内的细节运动问题。



(Carphone 第 64 帧,丢包率为 15%): (a) 原始图像; (b) 受损图像; (c) TR 隐藏效果; (d) JM 隐藏效果; (e) 本文算法隐藏效果; (f) 局部隐藏效果比较

图 4 错误隐藏效果比较

表 3 给出了测试序列在不同丢包率情况下的峰值信噪比 (PSNR) 值比较,其中 $PSNR_{TR}$ 为采用时域替换方法的 PSNR 值, $PSNR_{JM}$ 为采用 JM 模型进行错误隐藏所得的 PSNR 值, $PSNR_{PRO}$ 为采用本文算法所得的 PSNR 值。如表中数据所示,对于 Carphone 与 Foreman 这类局部运动较多的序列,本文算法有针对性地解决了细节运动区域的重建问题,重建后 PSNR 值比 TR 方法约提高了 1.5dB,比 JM 方法约提高了 1dB;对于 Salesman 与 Mobile 这类背景复杂且存在大幅度运动(尤其在 Mobile 中存在多个独立运动的物体)的序列,其背景的复杂性会给 JM 与本文算法在最优运动矢量的选择上造成一定的影响,但由于本

文算法对运动区域恢复的适应性,能够有效区分各个部分的运动来分别加以重建,仍可以获得较好的恢复效果。表 3 给出了不同丢包率情况下的峰值信噪比值的比较,其中数据显示,在测试中对于不同运动类型的序列和不同的网络丢包率环境,本文提出的算法性能均优于其他两种算法,并且在网络环境较恶劣的情况下恢复效果仍较理想,是一种鲁棒性很强的算法。

表 3 不同丢包率情况下的峰值信噪比值比较

测试序列	QP	原始 PSNR(dB)	PSNR (dB)	丢包率		
				5%	10%	15%
Carphone	20	43.06	$PSNR_{TR}$	39.43	37.83	36.71
			$PSNR_{JM}$	40.45	38.98	37.92
			$PSNR_{PRO}$	41.14	39.75	39.39
	25	39.54	$PSNR_{TR}$	37.11	36.00	34.89
			$PSNR_{JM}$	38.04	37.40	36.13
			$PSNR_{PRO}$	38.44	38.21	37.16
	30	35.90	$PSNR_{TR}$	34.73	33.75	32.67
			$PSNR_{JM}$	35.10	34.32	33.59
			$PSNR_{PRO}$	35.41	34.91	34.40
Foreman	20	41.93	$PSNR_{TR}$	38.06	36.51	34.96
			$PSNR_{JM}$	38.81	37.40	36.13
			$PSNR_{PRO}$	39.59	38.54	37.35
	25	37.92	$PSNR_{TR}$	34.68	33.28	32.61
			$PSNR_{JM}$	35.20	34.22	33.57
			$PSNR_{PRO}$	35.91	35.37	34.73
	30	34.45	$PSNR_{TR}$	32.22	30.91	30.07
			$PSNR_{JM}$	32.63	31.60	30.78
			$PSNR_{PRO}$	33.35	32.14	31.59
Salesman	20	41.93	$PSNR_{TR}$	38.69	37.26	36.02
			$PSNR_{JM}$	39.41	38.15	37.24
			$PSNR_{PRO}$	40.26	39.21	38.39
	25	38.07	$PSNR_{TR}$	35.05	34.41	33.95
			$PSNR_{JM}$	35.70	35.13	34.76
			$PSNR_{PRO}$	36.42	35.99	35.68
	30	34.07	$PSNR_{TR}$	31.92	31.37	30.85
			$PSNR_{JM}$	32.56	32.09	31.57
			$PSNR_{PRO}$	32.97	32.80	32.24
Mobile	20	40.61	$PSNR_{TR}$	38.71	37.46	36.57
			$PSNR_{JM}$	39.22	38.03	37.26
			$PSNR_{PRO}$	39.95	38.87	38.29
	25	36.29	$PSNR_{TR}$	33.48	32.25	31.18
			$PSNR_{JM}$	33.62	32.64	31.70
			$PSNR_{PRO}$	34.14	33.36	32.63
	30	31.80	$PSNR_{TR}$	29.35	28.52	27.48
			$PSNR_{JM}$	29.64	28.93	28.00
			$PSNR_{PRO}$	30.11	29.49	28.74

3 结 论

本文针对时域错误隐藏算法的应用中经常出现的块效应问题,提出了一种基于受损区域运动特征的错误隐藏算法。该算法利用 H.264 中多尺寸编码块模式的特点,根据受损宏块内部运动类型对宏块进行划分,并充分利用运动矢量的时空相关性,引入帧间预测矢量来丰富运动矢量候选集,从候选集中选择最优矢量对各子块分别重建。通过这种针对各子块运动特征恢复受损区域的方法,大大增强了算法对复杂运动区域重建的适应性。实验结果表明本文算法有效解决了受损区域内部细节运动的恢复问题,对不同类型的视频序列及在不同网络丢包率环境下均有较好的主观和客观重建图像质量。

参考文献

[1] Wang Y, Wenger S. Error resilient video coding techniques.

IEEE Signal Processing Magazine, 2000,17(4): 61-82

- [2] Wenger S. H.264/AVC over IP. *IEEE Transactions on Circuits and Systems for Video Technology*, 2003,13(7): 645-656
- [3] Wilegand T, Sullivan G J. Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14 496210 AVC-2003). Thailand: Joint Video Team of ISO/IEC JTC1/SC29/WG11 and ITU-T SG16/Q. 6 Doc, 2003
- [4] Wilegand T, Sullivan G J, Bjontegaard G, et al. Overview of the H.264 video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 2003,13(7): 560-576
- [5] Huang Y W, Hsieh B Y, Wang T C, et al. Analysis and reduce of reference frames for motion estimation in MPEG-4 AVC/JVT/H.264. In: Proceeding of the 2003 IEEE International Conference on Multimedia and Expo (ICME'2003), Baltimore, USA, 2003. 809-812
- [6] JVT. JM90. http://iphome.hhi.de/suehring/tml/download/old_jm/jm90.zip:JVT,2005

A temporal error concealment algorithm based on motion characteristics of damaged areas

Zhao Lianfeng, Guo Baolong, Han Hemin

(Institute of ICIE, School of Electromechanical Engineering, Xidian University, Xi'an 710071)

Abstract

The paper presents a temporal error concealment algorithm suitable for the H.264 video coding standard for error code and packet loss issues of video compression data in network transmission. The algorithm determines the correct mode for the lost block based on the modes of neighboring macroblocks, and according to the motion of the block, selects the most appropriate motion vector to recover the damaged area among a set of candidate motion vectors. The experimental results show that the complicated motion region can be well recovered with the proposed algorithm and good results can be achieved for video under various conditions of packet loss rate.

Key words: H.264, error concealment, macroblock mode, a set of candidate motion vectors