

# 基于可变尺寸块运动矢量恢复的 H. 264 时域差错掩盖算法

陈 振<sup>1)</sup> 贺建军<sup>2)</sup>

<sup>1)</sup>(湖南涉外经济学院计算机系,长沙 410205) <sup>2)</sup>(中南大学信息学院,长沙 410083)

**摘 要** 针对 H. 264 帧间预测编码的新特点,提出了一种基于可变尺寸块运动矢量恢复的时域差错掩盖算法。该算法首先利用相邻宏块编码模式的相关性,根据周围宏块的编码模式判断受损宏块的编码模式及运动矢量恢复的宏块划分方式,分别对各个划分的子块进行运动矢量的恢复;然后利用相邻块运动矢量参考帧的相关性,根据相邻块运动矢量的参考帧确定匹配使用的参考帧;最后采用边缘失真匹配方法恢复运动矢量。实验结果表明,该算法同传统的差错掩盖算法相比,由于支持不同尺寸块运动矢量的恢复,因此,算法对差错信号能够获得更好的恢复效果。

**关键词** 视频编码 差错掩盖 边缘失真匹配 H. 264

中图法分类号:TN918.81 文献标识码:A 文章编号:1006-8961(2008)07-1244-06

## Temporal Error Concealment Algorithm for H. 264 Based on Variable Size Block Motion Vector Recovery

CHEN Zhen<sup>1)</sup>, HE Jian-jun<sup>2)</sup>

<sup>1)</sup>(Hunan International Economics College, Changsha 410205)

<sup>2)</sup>(Central South University Information Institute, Changsha 410083)

**Abstract** The latest video coding standard H. 264 supports variable size block motion prediction and compensation. Adapt to this new feature, a new temporal error concealment algorithm based on variable size block motion vector resilience, is proposed in this paper. Exploiting the correlation of adjacent macroblocks, the encoding mode of lost macroblock, partition pattern of lost macroblock, and the reference frame used in matching can be decided according to neighboring macroblocks' encoding mode and used reference frame information, and then motion vector is recovered for each partitioned subblocks using side distortion matching method. Finally, simulation results show the proposed algorithm is superior to conventional temporal concealment methods for H. 264 standard.

**Keywords** video coding, error concealment, side distortion matching, H. 264

## 1 引言

伴随计算机和通信技术的快速发展和不断进步,互联网或无线网络上的视频传输日益深入和广泛。为满足网络带宽的约束,必须对原始视频数据进行压缩。由于压缩视频流在网络信道中传输的时

候,会受到各种信道噪声的影响,从而会发生误码或丢包的现象。许多国际编码标准,如 MPEG-1/2/4、H. 261、H. 263 及新一代标准 H. 264<sup>[1]</sup>,都采用基于运动补偿预测编码以消除帧间相关性和通过变换编码以消除帧内相关性的混合编码框架。由于这样不仅导致了压缩视频流信号之间高度的依赖性,而且使得错误的接收信息会在帧内或帧间传播,从而引

起更大的视觉失真。

人们已提出了许多技术,用来提高视频码流的抗干扰性。其中最一种方法是在编码端使用错误恢复机制来提高视频码流的抗干扰性,如使用同步码、重要信息冗余、信道编码等;另一种方法则是在解码端采用差错掩盖技术,使用帧内或帧间的差错掩盖算法将受损的宏块恢复<sup>[2]</sup>。根据受损宏块的编码方式,差错掩盖技术一般分为时域差错掩盖技术和空域差错掩盖技术。其中时域差错掩盖技术是先利用视频序列的时域相关性,通过估计算法来恢复丢失的运动矢量,然后用参考帧中相应的宏块代替当前丢失的宏块;而空域差错掩盖技术则是利用视频序列的空域相关性,通过有效邻域的空间插值算法来估计出当前丢失的宏块。由于差错掩盖技术工作在解码端,并不需要增加额外的码流开销,所以在视频数据的差错恢复中,差错掩盖技术得到了广泛的应用。

当帧间编码宏块丢失时,一般使用时域差错掩盖方法来恢复丢失的宏块。其中最简单直接的时域差错掩盖方法是用与前一帧空间位置对应的宏块来替代受损的宏块,尽管这种方法实现简单,但是当出现运动的时候会导致较大的视觉失真,而此时用运动补偿宏块替代受损的宏块则会显著提高视频质量,其关键是运动矢量的恢复。众所周知,基本的运动矢量恢复方法包括利用相邻宏块运动矢量的均值进行恢复以及利用相邻宏块运动矢量的中值进行恢复等。著名的差错掩盖方法——边界匹配算法(boundary matching algorithm, BMA)<sup>[3]</sup>是基于空间平滑度约束在一组候选的运动矢量(MV)中选择损失的MV。然而,当边界是对角线或反对角线形状的时候,恢复效果会变得比较差。针对最新的视频压缩标准 H. 264,人们提出了新的时域差错掩盖算法,其中包括 H. 264 参考软件采用的算法<sup>[4]</sup>及其改进的算法<sup>[5]</sup>,以及利用小区块特性的 Lagrange 插值算法<sup>[6]</sup>等。这些方法虽然适应 H. 264 标准压缩视频码流的结构,但没有充分利用 H. 264 帧间预测的新特性,也没有考虑帧间预测图像中帧内编码宏块的影响。

为了提高编码效率,H. 264 引入了更加先进和复杂的帧间预测方法,其中包括 7 种不同尺寸和形状块的树状结构运动预测补偿、多参考帧运动预测等。针对这些新特点,本文提出了一种基于可变尺寸块运动矢量恢复的时域差错掩盖算法。该算法首

先利用相邻宏块编码模式的相关性来确定受损宏块的划分方式及使用的参考帧,然后采用边缘失真匹配算法实现各个划分子块的运动矢量恢复。由于运动矢量的恢复使用了更精确的参考帧,并且在判断宏块模式的时候考虑了帧内模式,因此,该算法能够很好地恢复受损的宏块。实验结果验证了算法的优越性。

## 2 H. 264 预测方法概述

新一代视频压缩编码标准 H. 264 使用了许多先进的视频编码技术来提高视频压缩效率,尤其在帧间预测技术上较以往标准有了明显的改进,其主要表现在:(1)支持可变尺寸块的运动预测补偿,并将宏块按照树状结构进行划分(如图 1 所示),由于其在宏块级支持  $16 \times 16$ 、 $16 \times 8$ 、 $8 \times 16$  和  $8 \times 8$  等 4 种划分模式,其中  $8 \times 8$  划分,也称为 P8  $\times$  8 模式,并将其进一步划分为  $8 \times 8$ 、 $8 \times 4$ 、 $4 \times 8$  和  $4 \times 4$  等 4 种模式,因此,一共支持 7 种尺寸块的运动预测补偿;(2)支持多参考帧运动预测(如图 2 所示, $\Delta$  是参考帧与当前帧的相对距离量, $\Delta = 1$  时,表示当前帧的前一帧,…… $\Delta = 4$  时,表示当前帧前的第 4 帧。),且运动矢量搜索不仅可以在前一个重构帧中进行,也可以参考时间上距离更远的重构帧,但需要指出的是,在宏块级上划分的子块,可以分别使用不同的参考帧,而对于  $8 \times 8$  子块进一步划分的子块则需使用相同的参考帧;(3)对于帧间预测帧,不仅考虑了不同尺寸块的帧间预测模式,还考虑了帧内预测模式,然后使用率失真最优化方法在所有的预测模式中选择率失真代价最小的模式。多种模式的使用以及多参考帧的运动预测,由于能够实现对编码宏块进行精确预测,最大程度的发掘帧间和帧内视频数据的相关性,从而可提高视频编码效率。

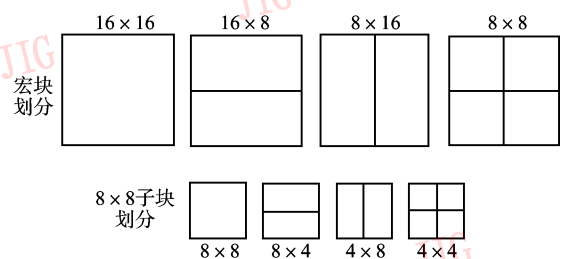


图 1 H. 264 的帧间预测模式

Fig. 1 The different block division modes in H. 264

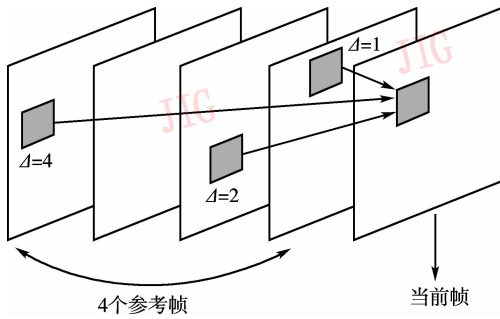


图 2 多参考帧运动预测

Fig. 2 Motion prediction of Multi-reference frame

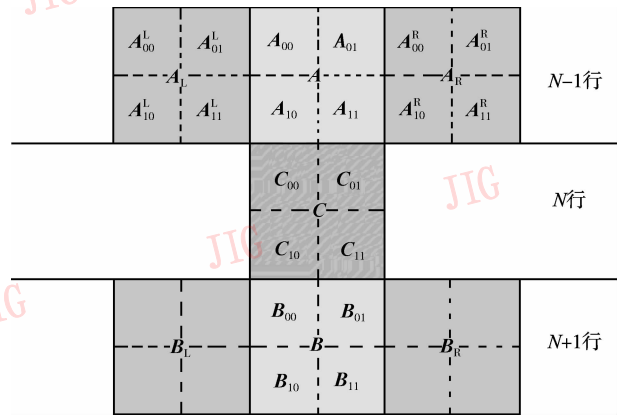


图 3 宏块划分模式及参考帧确定

Fig. 3 The macroblocks division modes and the reference frame determination

### 3 时域差错掩盖算法

本文提出的算法是一种基于受损宏块恢复的时域差错掩盖算法。该算法当解码完一帧之后才启动差错掩盖机制,且差错掩盖的方向不是顺序的方向,而是类似于 H. 264 采纳的差错掩盖算法,使用上下左右向中间夹击的方式,从 4 边向中心做差错掩盖,以避免因从左到右、从上到下的方向进行差错掩盖而带来的错误蔓延。该算法首先根据帧内相邻块之间的编码模式的相关性,判断受损宏块的编码模式是帧内模式还是帧间模式,如果是帧间模式,则进一步确定丢失运动矢量的预测宏块的划分模式;然后根据由受损宏块划分的子块相邻块运动矢量所使用的参考帧信息来确定每个子块运动矢量预测所使用的参考帧;最后,对每一个受损宏块的子块,使用边缘失真匹配方法,在选择参考帧中,对所有候选的运动矢量按照确定的块尺寸进行边缘失真匹配,进而选择具有最小匹配代价的候选运动矢量作为子块的恢复运动矢量。当受损宏块所有子块的丢失运动矢量恢复后,就可以完成整个宏块的差错掩盖。

#### 3.1 受损宏块预测模式及宏块划分模式的确定

与 H. 264 支持的编码模式及宏块划分模式相对应,本文的时域差错掩盖算法在对受损宏块进行差错掩盖之前,需先判断其编码模式,并确定宏块的划分模式,以便实现更精确的丢失运动矢量预测和差错掩盖。这里受损宏块预测模式的判断及宏块划分模式的确定主要是利用相邻宏块间编码模式相关性及运动的相似性。这里只考虑上下相邻的宏块,因为在大部分情况下,左右相邻的宏块都是错误的宏块(如图 3 所示,下标 R, L 分别表示 right 与 left),当前受损宏块为 C,上面相邻和下面相邻的宏块分别为 A 和 B。宏块预测模式及划分模式的判断

方法如下:

- (1) 如果宏块 A 和宏块 B 的编码模式均为帧内 (Intra) 预测编码模式,则宏块 C 的编码模式也为 Intra 预测编码模式,此时使用 H. 264 推荐的算法<sup>[4]</sup>中的空域差错掩盖算法对宏块 C 进行错误恢复;
- (2) 如果宏块 A 和 B 中有一个为 Intra 编码模式,则再考察宏块 A 和 B 左右相邻宏块  $A_L$ 、 $A_R$ 、 $B_L$ 、 $B_R$  4 个宏块的编码模式,如果  $A_L$ 、 $A_R$ 、 $B_L$ 、 $B_R$  4 个宏块中有 2 个以上宏块为 Intra 模式,则宏块 C 为 Intra 编码模式,此时可使用 H. 264 推荐的算法<sup>[4]</sup>中的空域差错掩盖算法对宏块 C 进行错误恢复;否则,宏块 C 编码模式和另一个非 Intra 模式宏块的编码模式相同;
- (3) 如果宏块 A、B 的模式均为  $16 \times 16$  的帧间预测编码模式 (Inter $16 \times 16$ ) 或 SKIP 模式,则宏块 C 为 Inter $16 \times 16$  模式;
- (4) 如果宏块 A、B 中有一个宏块为 Inter $16 \times 16$  模式,则宏块 C 的模式可根据另一个宏块模式进行判断,判断方法是:假设宏块 B 为非 Inter $16 \times 16$  模式的宏块,如果宏块 B 为 Inter $16 \times 8$  或 Inter $8 \times 16$  的帧间预测模式,则宏块 C 的宏块划分模式和宏块 B 一致,如果宏块 B 为  $P8 \times 8$  (包含 4 子块划分模式) 的帧间预测模式,则宏块 C 的划分模式为 4 个  $8 \times 8$  子块划分模式,即采用 Inter $8 \times 8$  的帧间预测编码模式;
- (5) 如果宏块 A 和 B 划分模式均为 Inter $16 \times 8$ 、Inter $8 \times 16$  或 4 种  $P8 \times 8$  模式,且子块划分模式相同,则宏块 C 的划分模式和它们相同;
- (6) 如果宏块 A 和 B 的划分模式中,一个为 Inter $16 \times 8$ ,另一个为 Inter $8 \times 16$  模式,则宏块 C 的划分模式为 4 个  $8 \times 8$  子块划分模式;

(7) 如果宏块  $A$  和  $B$  的划分模式为  $P8 \times 8$  模式,但子块的划分模式不同,则宏块  $C$  为  $P8 \times 8$  模式,其 4 个子块的划分模式和其上(下)相邻的子块划分模式相同;

(8) 如果  $C$  为边界宏块,即宏块  $A$  或宏块  $B$  不存在,则宏块  $C$  的划分模式与其相邻的一个宏块相同。

当判断出宏块  $C$  的预测模式为帧间模式的时候,则可采用本文提出的时域差错掩盖方法对受损的宏块进行差错掩盖。

### 3.2 运动矢量预测的参考帧选择

丢失运动矢量的恢复可根据相邻宏块运动矢量的信息进行预测,考虑到 H. 264 视频编码中多参考帧运动估计的特点,不仅要确定预测的候选运动矢量,还要确定预测的最佳参考帧,才能实现更高质量的差错掩盖。而运动矢量预测参考帧选择的基本思想是利用相邻宏块运动矢量的参考帧之间的相似性来确定丢失运动矢量的参考帧(如图 3 所示)。宏块  $A$  的下面两个子宏块  $A_{10}$  和  $A_{11}$  的运动矢量的参考帧为  $R_{A_{10}}$  和  $R_{A_{11}}$ ,宏块  $B$  的上面两个子宏块  $B_{00}$  和  $B_{01}$  的运动矢量的参考帧为  $R_{B_{00}}$  和  $R_{B_{01}}$ ,如果  $A$  和  $B$  的划分模式不是  $P8 \times 8$ ,则它们分别代表大尺寸划分子块的运动矢量参考帧,例如:如果宏块  $A_{10}$  的预测模式为  $Inter16 \times 16$  模式,且其运动矢量参考帧为  $R_A$ ,则  $R_{A_{10}} = R_{A_{11}} = R_A$ 。此时可在宏块  $A$  上边相邻的所有候选运动矢量中,选取参考帧为  $R_{A_{10}}$  或  $R_{A_{11}}$  的运动矢量作为最终的后续运动矢量,并根据它们本身使用的参考帧来选择  $R_{A_{10}}$  或  $R_{A_{11}}$  运动矢量预测的参考帧;在宏块  $A$  下边相邻的所有候选运动矢量中,选取参考帧为  $R_{A_{10}}$  或  $R_{A_{11}}$  的运动矢量作为最终的后续运动矢量,并根据它们本身使用的参考帧来选择  $R_{A_{10}}$  或  $R_{A_{11}}$  运动矢量预测的参考帧。

### 3.3 候选运动矢量的选择

由于当受损宏块的预测模式确定为帧间预测模式之后,同时也确定了宏块的划分模式,因此需要根据不同的宏块划分模式,为各个子块选择合适的候选运动矢量。候选运动矢量的选择可利用相邻块的空间相关性,选择相邻块的运动矢量作为运动矢量预测的候选。由于不同的宏块划分模式,其子块具有不同的相邻块,且相邻宏块的划分模式也是变化的,因此不能像传统的方法那样,选择宏块  $A$ 、 $B$ 、 $A_L$ 、 $A_R$ 、 $B_L$ 、 $B_R$  的运动矢量作为候选运动矢量,而应对各个子块区别考虑,选择其各自相邻块的运动矢

量作为候选运动矢量。例如在图 3 中,如果宏块  $C$  的预测模式为  $P8 \times 8$ ,则对于它的第 1 个子块  $C_{00}$ ,可选择其相邻的 3 个子块  $A_{11}$ 、 $A_{10}$ 、 $A_{11}$  的运动矢量作为候选运动矢量,这里假设对相邻的 3 个子块没有做进一步的划分。对于宏块  $C$  的更小或更大尺寸的划分,以及相邻块的不同划分模式,其候选运动矢量的选择则使用类似的最近相邻的原则。

### 3.4 边缘失真匹配算法

当候选预测运动矢量确定之后,即可使用边缘失真匹配算法(side distortion matching,SDM)在这些候选运动矢量中选择失真最小的一个作为最终受损块的恢复运动矢量。如图 4 所示, $E$  为受损块, $C$  为与候选预测运动矢量对应的恢复块,块尺寸为  $M \times N$ ,其左上角坐标为  $(x_0, y_0)$ ,候选运动矢量  $v$  的模在  $x$  方向的坐标为  $x_v$ ,在  $y$  方向的坐标为  $y_v$  表示,参考帧为  $F_{ref}$ , $P(x, y)$  代表当前帧的像素点灰度值, $P_{F_{ref}}(x, y)$  代表参考帧的像素点灰度值,则 4 个边缘的失真计算公式分别为

$$D_A = \sum_{i=0}^{M-1} |P(x_0 + i, y_0 - 1) - P_{F_{ref}}(x_0 + x_v + i, y_0 + y_v - 1)| \quad (1)$$

$$D_B = \sum_{i=0}^{M-1} |P(x_0 + i, y_0 + N + 1) - P_{F_{ref}}(x_0 + x_v + i, y_0 + y_v + N + 1)| \quad (2)$$

$$D_L = \sum_{j=0}^{N-1} |P(x_0 - 1, y_0 + j) - P_{F_{ref}}(x_0 + x_v - 1, y_0 + y_v + j)| \quad (3)$$

$$D_R = \sum_{j=0}^{N-1} |P(x_0 + M + 1, y_0 + j) - P_{F_{ref}}(x_0 + x_v + M + 1, y_0 + y_v + j)| \quad (4)$$

总的边缘失真的计算公式为

$$D_{total} = D_A + D_B + D_L + D_R \quad (5)$$

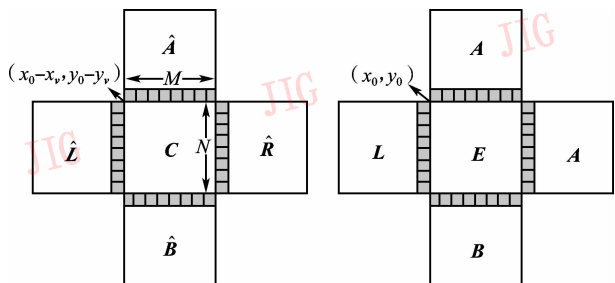


图 4 边缘失真匹配

Fig. 4 Side distortion matching

注意到,如果边缘不存在,则相应的边缘失真为

0。最终,一个  $M \times N$  大小的块的恢复运动矢量为

$$\hat{v} = \arg \min_{v \in V_c} D_{\text{total}}(v) \quad (6)$$

其中,  $V_c$  为候选运动矢量集合。

## 4 实验结果

实验中,测试平台采用 H. 264 参考软件 JM98,利用 JVT 提供的错误样本<sup>[7]</sup>,其测试序列包括 4 个 QCIF 格式的视频序列:Foreman、Football、Garden 和 Tennis,编码结构为 IPPPP,第 1 帧为 I 帧,其余为 P 帧,共编码 150 帧。为验证本文算法的效果,将本文的算法与零矢量方法(ZMV,该算法将零运动矢量作为恢复的运动矢量)、H. 264 中使用的方法<sup>[4]</sup>、H. 264 方法的改进算法<sup>[5]</sup>、Lagrange 插值算法<sup>[6]</sup>在不同的包丢失率环境下进行了比较。比较的指标为亮度分量的峰值信噪比(PSNR),PSNR 取量化参数为 28 时各个帧的 PSNR 平均值。实验结果如表 1 所示。

表 1 各种算法的仿真结果

Tab.1 The simulation results of algorithms

序列	算法	包丢失率(%)				
		0	1	5	10	20
Foreman	ZMV	35.14	28.64	25.80	24.76	23.48
	H. 264	35.14	31.53	29.54	27.54	25.60
	H. 264 改进	35.14	31.76	29.55	27.62	25.82
	Lagrange 插值	35.14	31.03	28.44	27.12	24.86
	本文方法	35.14	31.98	29.56	27.67	26.28
Football	ZMV	27.96	25.72	23.36	23.00	20.53
	H. 264	27.96	26.18	23.58	22.66	21.38
	H. 264 改进	27.96	26.22	23.61	22.78	21.35
	Lagrange 插值	27.96	26.20	23.61	22.68	21.37
	本文方法	27.96	26.28	23.76	23.10	21.53
Garden	ZMV	27.14	23.75	22.92	20.00	19.28
	H. 264	27.14	24.69	27.76	22.63	21.47
	H. 264 改进	27.14	24.83	24.78	22.66	21.52
	Lagrange 插值	27.14	24.88	24.74	22.62	21.50
	本文方法	27.14	25.63	25.07	22.66	21.80
Tennis	ZMV	31.47	29.82	25.77	23.88	22.59
	H. 264	31.47	30.72	28.52	26.66	24.72
	H. 264 改进	31.47	30.92	29.14	26.76	24.98
	Lagrange 插值	31.47	30.73	28.55	26.70	24.68
	本文方法	31.47	31.16	29.42	26.79	25.41

从实验结果可以看出,在各种包丢失率的错误环境下,本文算法所恢复的图像质量都优于其余几种算法,特别是对 Foreman 这种运动较复杂、在编码端小尺寸运动预测模式使用率很高的序列,差错掩盖效果更加明显,其原因是由于本文算法支持多种尺寸运动预测模式的矢量恢复,因此能够更好地适应这种具有复杂运动的视频序列。图 5 以 Tennis 序列的第 3 帧为例,给出了包丢失率为 10% 的条件

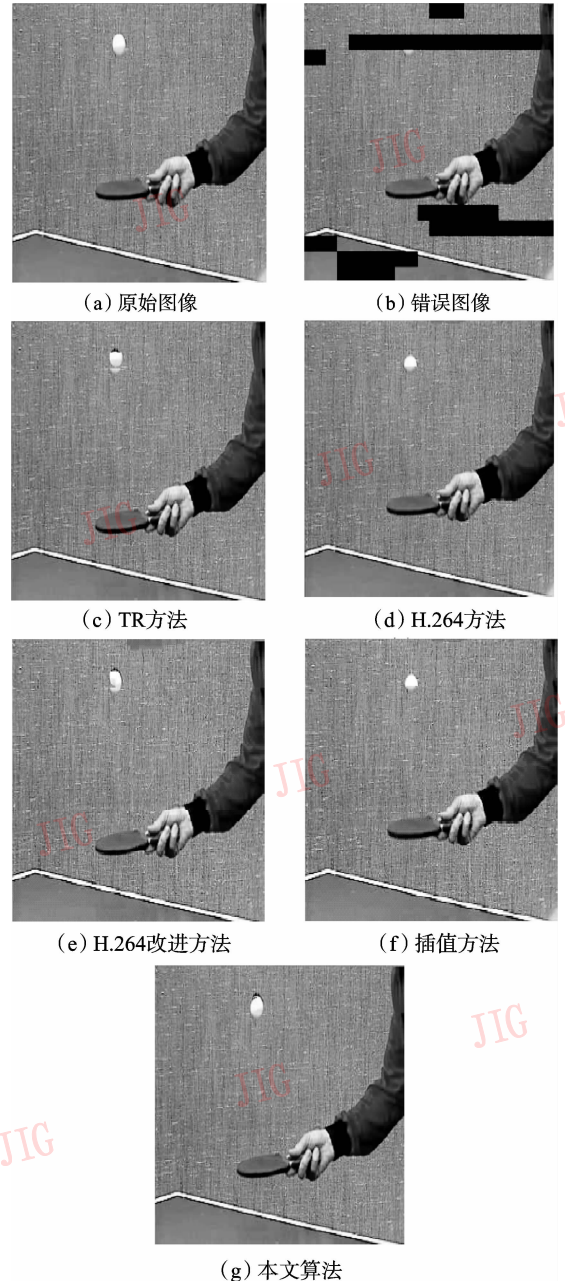


图 5 不同算法主观视频质量的比较 (Tennis 序列,第 3 帧,PLR = 10%)

Fig.5 Comparison of the subjective video quality (Tennis sequence, 3rd, PLR = 10%)

下,各个算法主观视频质量的比较。由图5可以看出,对于背景区域的差错部分,各种算法获得的视频质量差别不大,这是因为这些背景区域运动很少或没有运动,即使是使用直接替换的方法,也不会带来明显的质量损失。但是对于运动比较明显的区域,如乒乓球的运动区域,当出现传输差错时,本文的算法能够获得最好的差错掩盖效果,在主观视频质量上优于另外几种算法。

## 5 结 论

本文提出了一种面向 H.264 视频编码标准的基于可变尺寸块运动矢量恢复的时域差错掩盖算法。该算法先充分利用相邻宏块编码模式及运动矢量参考帧之间的相关性,并根据受损宏块周围宏块的编码模式和参考帧信息来判断受损宏块的编码模式、运动矢量恢复的宏块划分方式以及匹配使用的参考帧,然后分别对各个划分的子块采用边缘失真匹配方法进行运动矢量的恢复。实验结果表明,本文提出算法的性能优于其他的 H.264 时域差错掩盖算法。

## 参考文献 (References)

- 1 Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification [S], IEC 14496-10 AVC. In: 14th Meeting of JVT of ISO/IEC MPEG and ITU-T VCEG [C], Santa Barbara, CA, USA.
- 2 Wang Y, Zhu Q F. Error control and concealment for video communication: a review[J]. Proceedings of IEEE, 1998, **86**(5): 974 ~ 997.
- 3 Lam W M, Reibman A R, Liu B. Recovery of lost or erroneously received motion vectors [A]. In: Proceedings of International Conference on Acoustics Speech and signal process [C], Glasgow, UK, 1993, **5**: 417 ~ 420.
- 4 Varsa V, Hannuksela M M. Non-normative Error Concealment Algorithm [S]. ITU-T SG16, VCEG-N62.
- 5 Xu Y L, Zhou Y H. H.264 video communication based refined error concealment schemes [J]. IEEE Transactions on Consumer Electronics, 2004, **50**(6): 1135 ~ 1141.
- 6 Zheng J, Chau Lap-pui. A Temporal error concealment algorithm for H.264 using lagrange interpolation [A]. In: Proceedings of IEEE International Symposium on Circuits and Systems [C], Hiroshima, Japan, 2004, **2**: 133 ~ 136.
- 7 Roth G, Sjoberg R, Liebl G, *et al.* Common Test Conditions for RTP/IP over 3GPP/3GPP2[S], ITU-T SG16, VCEG-M77.