

一种基于 H.264/AVC 的视频可逆脆弱水印算法

张维纬* 张茹 刘建毅 伍淳华 钮心忻 杨义先

(北京邮电大学信息安全中心 北京 100876)

(北京邮电大学灾备技术国家工程实验室 北京 100876)

摘要: 结合 H.264 压缩编码标准的特性, 该文提出一种新的基于 H.264/AVC 的视频可逆脆弱水印算法。算法先计算当前宏块预测残差块量化的离散余弦变换(DCT)系数的哈希值生成认证码, 再把认证码作为水印信息嵌入到下一个相邻宏块活性最大的 4×4 块的最后一个非零量化 DCT 系数中。在解码端, 通过比较认证码和提取的水印信息进行视频数据完整性的认证。实验结果表明, 该算法对视频质量和码率的影响较小, 并且能对认证通过的视频数据进行还原和对认证失败的视频 I 帧篡改区域进行有效的定位。

关键词: 视频水印; 脆弱水印; 可逆水印; H.264; 视频认证

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2013)01-0106-07

DOI: 10.3724/SP.J.1146.2012.00726

A Reversible Fragile Video Watermarking for H.264/AVC

Zhang Wei-wei Zhang Ru Liu Jian-yi Wu Chun-hua Niu Xin-xin Yang Yi-xian

(Information Security Center, Beijing University of Posts and Telecommunications, Beijing 100876, China)

(National Engineering Laboratory for Disaster Backup and Recovery,

Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: Combining with the new features of the H.264/AVC video coding standard, a new reversible fragile video watermarking algorithm is proposed. The hash value of quantized Discrete Cosine Transform (DCT) coefficients of predictive residue blocks is performed with the proposed algorithm in current macroblock, and the authentication code is created. Then, the authentication code, as watermark is embedded into the last nonzero quantized DCT coefficient in the most active 4×4 block of the next adjacent macroblock. The integrality authentication is carried out in the decoding end by the comparison of the authentication code with the extracted watermark information. The experimental results show that the proposed algorithm has little impact on the video quality and bit-rate, moreover, the original video can be recovered if the authentication is successful and the tampered area of I frame can effectively be located if the authentication fail.

Key words: Video watermarking; Fragile watermarking; Reversible watermarking; H.264; Video authentication

1 引言

脆弱水印技术的目的是为数字多媒体作品提供完整性和真实性认证并能定位篡改区域和还原原始的内容^[1]。在现有的大多数脆弱水印算法中, 嵌入脆弱水印对视频质量的影响极小。但是对于某些特殊的场合, 比如医疗、军事和司法等领域, 即使由于脆弱水印的嵌入导致视觉上微小的感知也是无法接受的。因此, 一种称为可逆脆弱水印的技术引起了

越来越多的研究与关注。目前作为主流的视频压缩编码标准, H.264/AVC 采用了许多新的技术, 编码效率比以往的标准有了很大的提高^[2], 其优异的压缩性能和良好的网络适应性在视频产品领域扮演着重要的角色。因此, 研究基于 H.264 为压缩编码标准的视频水印技术显得十分迫切^[3]。

近年来, 已有学者对基于 H.264/AVC 压缩编码标准的视频脆弱水印技术进行了研究。Zhang 等人^[4]根据 H.264/AVC 编码时 P 帧和 B 帧编码最佳分块模式选择的脆弱性, 提出了一种水印嵌入在模式 8×8 的运动补偿的非零离散余弦变换(Discrete Cosine Transform, DCT)交流系数上的认证方案。Qiu 等人^[5]提出把脆弱水印嵌在 $1/4$ 像素运动估计搜索过程中运动矢量预测残差的水平和垂直分量中。

2012-06-11 收到, 2012-09-21 改回

国家自然科学基金(61003284), 北京市自然科学基金(4122053), 中央高校基本科研业务费专项资金(BUPT2011RC0210)和新闻出版重大科技工程 (GXTC-CZ-1015004/09, GXTC-CZ-1015004/15-1)资助课题

*通信作者: 张维纬 weiweizh@hqu.edu.cn

Su 等人^[6]提出通过计算每帧图像的奇异值分解(Singular Value Decomposition, SVD)值获得特征码, 并采用最小可视失真(Just Noticeable Distortion, JND)模型把水印嵌入到 DCT 域中的内容认证方案。该方案只能检测视频的帧数是否被增删, 无法检测帧内的内容是否被篡改, 并且该方案计算量大, 无法适用于实时性要求较高的场合。Wang 等人^[7]提出在 H.264/AVC 编码过程中在每个 4×4 的 DCT 块的最后一个量化后的非零系数中嵌入脆弱水印。该算法水印的嵌入容量较大, 但对视频的质量影响也较大。Xu 等人^[8]提出在嵌入水印过程中首先找出合适的参考帧的指数哥伦布码, 然后建立这些码字和水印比特的映射规则。水印的嵌入是通过调制相应的指数哥伦布码字。提取水印时, 不必解码视频, 而只要通过从比特流中解析指数哥伦布码字。

以上所提出的基于 H.264/AVC 压缩编码标准的视频脆弱水印都是不可逆的, 并且这些算法无法实现对篡改区域的定位。其原因是 H.264/AVC 视频编码标准采用的新技术——帧内预测。帧内预测的主要目的是减少帧内图像的空间冗余, 提高编码效率。在帧内预测模式下, 依据先前编码和重建后的块形成一个预测块 P , 当前块减去预测块 P 得到一个差值, 然后对这个差值进行块变换、量化和熵编码等操作。每个 4×4 亮度块有 9 种预测模式, 每个 16×16 亮度块有 4 种预测模式。在对当前块进行编码时, 选择最优的编码模式。

文献[4-8]提出的基于 H.264/AVC 压缩编码标准的视频脆弱水印算法都是采用在 H.264/AVC 编码过程中进行水印的嵌入, 这类算法在编码过程中会造成由于嵌入水印后的误差漂移, 而且由于嵌入水印造成的误差漂移是不可逆的。再者, 这类算法的另一个缺点是嵌入水印需要先完全解码 H.264 码流, 然后再在编码的过程中嵌入水印, 增加了计算量, 这不适用于实时性要求较高的场合。

近年来, 脆弱水印方案与可逆信息隐藏技术的结合在原始图像^[9]、JPEG 图像^[10]、MPEG-4 视频^[11]、2 维矢量地图^[12]、3 维模型^[13]等载体进行应用。在这些应用中, 可逆脆弱水印与压缩编码标准的结合首先假设压缩前后的差别视觉上是无法感知的。目前针对 H.264/AVC 标准的可逆信息隐藏算法研究并不多。文献[14-16]将可逆信息隐藏技术运用于 H.264 标准视频的帧内预测和帧间预测技术领域, 该类算法在一个宏块中嵌入多比特的其他宏块信息, 以在解码时对受损的数据进行恢复或估计。这类算法虽然嵌入信息的容量大, 但在没有还原模块

的解码器中, 解码后对视频质量的影响很大。而即使水印是可逆的, 在认证领域也要使由于嵌入水印引起的失真尽可能小^[17]。因此该类算法不适用于可逆的脆弱水印认证领域。Huo 等人^[18]提出了一种用于版权保护的基于 H.264/AVC 的可逆水印算法。该算法根据 4×4 块的预测模式分布, 选择不会因水印嵌入产生误差传播或具有单一误差传播方向的 4×4 块, 然后根据其传播方向选择能够消除误差传播的 DCT 系数对, 采用“和不变”方法嵌入可逆水印。该算法能消除嵌在 4×4 块水印引起的误差漂移, 提高水印的视频质量, 但算法首先需要进行完全解码, 并且只能选择相邻块满足一定编码模式的 4×4 块嵌入水印, 限制了水印的嵌入容量, 算法只考虑 4×4 编码模式, 这在一定程度上影响了编码效率。Lin 等人^[19]提出一种基于 H.264/AVC 的可逆视频水印算法, 该算法首先利用相邻宏块的残差值量化后的 DCT 值预测当前块的值, 并利用预测差值的直方图平移嵌入水印。算法嵌入的水印容量较大, 但没有考虑人眼的视觉特性, 对水印视频质量的影响也较大。Liu 等人^[20]对水印嵌入导致的帧内误差漂移进行分析, 得到耦合系数对, 即在一个量化后的 DCT 系数上嵌入水印, 需在其耦合的系数上进行补偿, 从而达到消除误差漂移的目的。算法对水印视频的影响较小, 且嵌入的容量较大。但算法同样需要完全解码获得每个块的编码模式, 其次嵌入的条件是选择幅值满足一定条件的系数嵌入水印, 这意味着如果当前宏块的系数都为零, 则无法进行水印的嵌入, 也就无法实现对该宏块的篡改定位。因此不能直接应用于篡改定位的脆弱水印。

针对基于 H.264/AVC 视频脆弱水印算法的不足, 本文提出的方案首先对 H.264 码流进行熵解码, 得到 I 帧亮度分量残差块量化后的 DCT 系数, 先计算当前宏块 DCT 系数的哈希值, 再把该哈希值嵌入到下一个相邻宏块中。水印的嵌入考虑人眼的视觉特性, 将可逆水印嵌在人眼不敏感的 4×4 亮度块的残差量化 DCT 系数中, 水印嵌入后再进行熵编码得到嵌入水印后的码流。在水印嵌入过程中由于没有进行运动补偿, 所以由于水印的嵌入造成当前宏块值的改变不会在帧内预测中造成误差漂移。在解码端, 先对 H.264 码流进行熵解码得到 DCT 系数, 再计算当前宏块 DCT 系数的哈希值及提取下一个相邻宏块中的水印值, 通过比较哈希值和提取的水印值是否相同进行视频数据的完整性认证并对遭篡改的区域进行定位。水印提取的同时对当前块进行数据还原, 这样在解码端进行运动补偿时就不会造成水印的误差漂移。

2 本文提出的水印方案

2.1 水印的生成

在 H.264/AVC 标准中, 编码器是以宏块 (16×16) 为单位进行编码的, 所得的残差系数再进行 4×4 整数 DCT 变换, 变换后每个宏块有 16×16 个 DCT 系数。本文提出的算法把当前预测残差宏块量化后的 256 个 DCT 系数的值作为 HASH 函数 (本文实验选用 MD5 作为哈希函数) 的输入, 输出 128 bit 的 HASH 值, 然后把这 128 bit 的 HASH 值进行 M 等分, 对每一份的 HASH 值分别进行异或操作得到 M bit 的认证码, 再把这个认证码作为水印信息嵌入到相邻的下一个编码宏块, 水印信息的产生过程如图 1 所示。当视频的像素值或量化后的 DCT 系数遭到篡改时, HASH 函数的输出值会产生相应的改变, 就可以在解码端定位篡改区域。

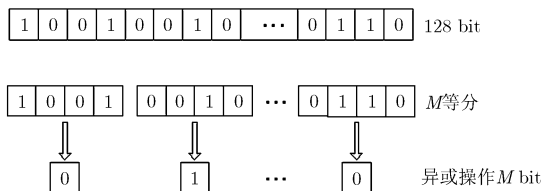


图 1 水印信息的生成

2.2 水印的嵌入

2.2.1 水印嵌入位置的选择 H.264 帧间预测是利用已编码视频帧/场和基于块的运动补偿的预测模式。与以往标准帧间预测的区别在于块尺寸范围更广 (从 16×16 到 4×4)、亚像素运动矢量的使用 (亮度采用 $1/4$ 像素精度) 及多参考帧的运用等等。其压缩效率更高, P, B 帧不适合于水印的嵌入, 因此采取的算法在 I 帧的宏块中嵌入水印。

视频序列与静止图像的不同在于它包含有运动的部分, 具有变化的特性。而人眼在物体发生运动时的空间敏感度有所下降, 可以在运动区域嵌入水印信息^[21]。在 H.264 编码中, 每个宏块中 4×4 子块的非零系数个数代表当前子块的运动活性^[22]。本文算法先统计每个宏块中小块的非零系数个数, 在个数最多的子块中嵌入水印。Wang 等人^[7]提出把水印嵌入到一个宏块中固定的一个 4×4 子块上, 而通过实验发现, 对非零系数个数最多的 4×4 子块中嵌入水印对视频质量和码率的影响更小。

2.2.2 水印的嵌入算法 H.264 宏块的预测残差经过整数 DCT 变换和量化后, 预测残差的能量主要集中在中低频的 DCT 系数中, 大部分的高频系数都是零。因此, 宏块中的 4×4 子块经过 zigzag 扫描后,

1 维分量的后几个系数都为零。图 2 是 Foreman 序列在编码过程中某个宏块的 4×4 子块量化后的 DCT 系数及其 zigzag 扫描图, 其扫描后产生的 1 维分量系数为 $[-3, 2, 4, -1, -2, 0, 3, 0, -1, 1, 0, 0, 0, 0, 0]$ 。

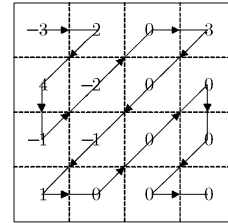


图 2 预测残差 4×4 子块 DCT 系数及 zigzag 扫描顺序

记 zigzag 扫描后的最后一个非零 DCT 系数在 1 维分量中的顺序下标为 LNZ。例如图 2 的 $LNZ = 10$, 当子块所有的系数都为零时, $LNZ = 0$ 。记最后一个非零系数为 C_{LNZ-1} 。本文提出的可逆水印的嵌入算法为首先把最后一个非零系数 C_{LNZ-1} 的 LSB 信息嵌入在其后的一个零系数上, 再把水印信息 w 嵌在 C_{LNZ-1} 的 LSB 上。

具体的嵌入算法可分为以下 3 种情况:

情况 1 $LNZ = 0$

当宏块中 4×4 子块非零系数最多的子块系数都为零时, 直接把水印信息嵌在 zigzag 扫描后的最后一个系数上, 这是因为在 H.264 编码过程中, 不会出现只有 zigzag 扫描后最后一个系数为非零的情况, 这样情况在解码端也易于提取。而 zigzag 扫描后最后一个系数对应于高频区域, 人眼比较不敏感。具体算法为

$$C'_{15} = \begin{cases} 0, & w = 0 \\ 1, & w = 1 \end{cases} \quad (1)$$

情况 2 $1 \leq LNZ \leq 15$

首先, 把最后一个非零系数的 LSB 嵌在其后的零系数上。

$$C'_{LNZ} = \begin{cases} 1, & \text{mod}(C_{LNZ-1}, 2) = 1 \\ -1, & \text{mod}(C_{LNZ-1}, 2) = 0 \end{cases} \quad (2)$$

再把水印信息嵌在最后一个非零系数的 LSB 上。

$$C'_{LNZ-1} = C_{LNZ-1} - \text{mod}(C_{LNZ-1}, 2) + w \quad (3)$$

以图 2 预测残差 4×4 子块 DCT 系数嵌入水印为例, 其嵌入水印后的 DCT 系数如图 3 所示。

情况 3 $LNZ = 16$

由情况 2 可知, 嵌入水印使嵌入的 4×4 子块 DCT 系数最后一个非零系数由嵌入前的 $LNZ = 15$ 变成 $LNZ = 16$ 。在提取水印时, 为区别是由情况 2

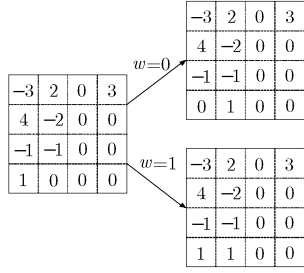


图3 情况2中4×4子块DCT系数水印嵌入图

还是情况3嵌入的水印,嵌入水印前LNZ=16的水印嵌入算法如下:

首先,最后一个非零系数 C_{LNZ-1} 嵌入其前一个系数 C_{LNZ-2} 的LSB信息,并保证嵌入水印后的最后一个非零系数绝对值大于1:

$$C'_{LNZ-1} = 2C_{LNZ-1} + \text{mod}(C_{LNZ-2}, 2) \quad (4)$$

再把水印信息嵌在 C_{LNZ-2} 系数上:

$$C'_{LNZ-2} = C_{LNZ-2} - \text{mod}(C_{LNZ-2}, 2) + w \quad (5)$$

以H.264编码中某个预测残差4×4子块量化DCT系数嵌入水印为例,其嵌入水印后的DCT系数如图4所示。

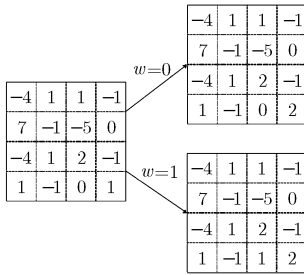


图4 情况3中4×4子块DCT系数嵌入水印

2.3 水印的提取和视频的还原

根据可逆水印的嵌入算法,水印的提取算法如下:首先对嵌入水印的H.264码流进行熵解码,得到每个宏块预测残差的量化DCT系数,再统计每个宏块中4×4子块DCT系数的非零个数,找出非零个数最多的4×4子块,并对其进行水印的提取和视频的还原。由于水印的嵌入会使嵌入水印的4×4子块DCT系数的非零个数增加或者保持不变,因此,只要视频数据没有遭到篡改,水印的嵌入位置和提取位置就会保持同步。记NZ为4×4子块DCT系数的非零个数。水印的提取和视频的还原过程分如下4种情况:

(1) LNZ = 0

$$w = 0 \quad (6)$$

(2) LNZ = 16且NZ = 1

提取水印:

$$w = 1 \quad (7)$$

还原视频:

$$\hat{C}_{15} = 0 \quad (8)$$

(3) $2 \leq \text{LNZ} \leq 15$

提取水印:

$$w = \begin{cases} 0, & \text{mod}(C'_{LNZ-2}, 2) = 0 \\ 1, & \text{mod}(C'_{LNZ-2}, 2) = 1 \end{cases} \quad (9)$$

还原视频:由嵌入水印后的最后一个非零系数 C'_{LNZ-1} 判断嵌入水印之前的最后一个非零系数 C_{LNZ-1} 的LSB值。

由嵌入算法中的式(2)可知,嵌入水印之前最后一个非零系数 C_{LNZ-1} 的LSB值为

$$\text{mod}(C_{LNZ-1}, 2) = \begin{cases} 1, & C'_{LNZ-1} = 1 \\ 0, & C'_{LNZ-1} = -1 \end{cases} \quad (10)$$

还原最后一个非零系数的前一个系数 C'_{LNZ-2} :

$$\hat{C}_{LNZ-2} = C'_{LNZ-2} + \text{mod}(C_{LNZ-1}, 2) - w \quad (11)$$

还原最后一个非零系数 C'_{LNZ-1} :

$$\hat{C}_{LNZ-1} = 0 \quad (12)$$

(4) LNZ = 16且NZ > 1 在这种情况下,要首先判断嵌入水印之前LNZ = 16还是LNZ = 15,提取算法如下:

(a)如果 $|C'_{LNZ-1}| = 1$ 这种情况是嵌入水印之前LNZ = 15。提取水印和还原视频的算法与提取水印的(3)一样,用式(9)提取水印,用式(10)、式(11)、式(12)还原视频数据;

(b)如果 $|C'_{LNZ-1}| > 1$ 这种情况是嵌入水印之前LNZ = 16。提取水印的算法与式(9)相同。

还原视频:由嵌入算法的式(4)可知:

$$\text{mod}(C_{LNZ-2}, 2) = \text{mod}(C'_{LNZ-1}, 2) \quad (13)$$

还原最后一个非零系数的前一个系数 C'_{LNZ-2} :

$$\hat{C}_{LNZ-2} = C'_{LNZ-2} + \text{mod}(C_{LNZ-2}, 2) - w \quad (14)$$

还原最后一个非零系数 C'_{LNZ-1} :

$$\hat{C}_{LNZ-1} = (C'_{LNZ-1} - \text{mod}(C_{LNZ-2}, 2)) / 2 \quad (15)$$

2.4 认证和定位

认证和定位的流程框图如图5所示。

首先对H.264码流进行部分解码获得预测残差的量化DCT系数,对当前宏块相邻的前一个宏块的DCT系数进行HASH变换,获取宏块的HASH值H,再提取当前宏块的水印值w,比较HASH值和提取的水印值是否一致,如果一致,则认证通过,如果不一致,则进行篡改定位。

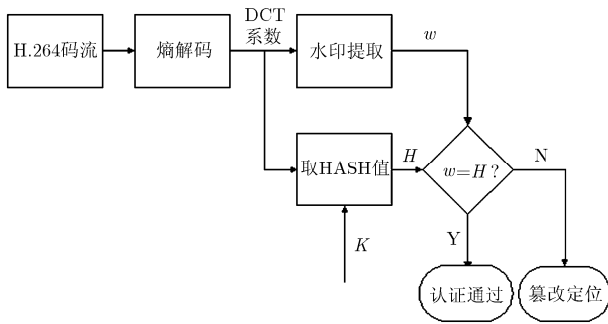


图5 认证和定位的流程框图

3 实验结果及分析

本实验基于 H.264 参考软件模型 JM18.0, 实验所用的序列为 QCIF (176×144) 格式的 Foreman, Carphone, Coastguard, Akiyo, Container, 测试视频既包括了动态性较强的视频序列 Carphone, 也包括了相对动态性较弱的视频序列 Container。每种序列都是在帧率为 30 帧/s, 帧数为 100 帧, 按 IPPPPP... 的顺序进行编码, 每隔 10 帧编码一个 I 帧, 在每个 I 帧中嵌入水印, QP 值设为 28。在一个 I 帧中嵌入的水印比特数为 99 bit。

图 6 和图 7 分别为 Foreman 和 Container 序列在嵌入水印前后第 30 帧视频质量的对比。从图中可以看出, 无论是 Foreman 序列还是 Container 序列, 从主观上判断, 嵌入水印之后图像质量都几乎没有下降。虽然水印的嵌入会使 PSNR 值略有降低, 但在解码端, 可逆水印提取的同时也能获得原始的视频数据。



图6 Foreman 测试序列嵌入水印前后的比较

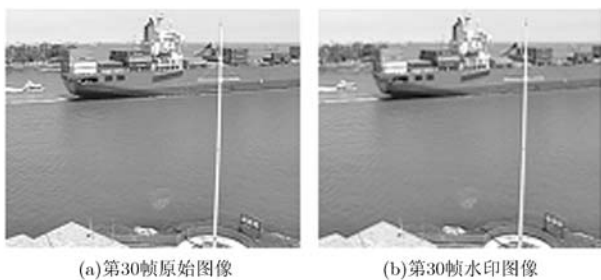


图7 Container 测试序列嵌入水印前后的比较

为了全面衡量嵌入视频的视觉效果, 本文给出 3 种峰值信噪比实验结果。峰值信噪比值 PSNR 指编码后(未嵌入水印)视频与未编码的原始 YUV 视频文件比较计算而得。峰值信噪比值 PSNR1 指编码后嵌入水印视频与未编码的原始 YUV 视频文件比较计算而得。峰值信噪比值 PSNR2 是指嵌入水印视频与嵌入水印的视频经 H.264 解码的视频比较计算而得, 该解码器是普通的解码器, 即没有加入提取水印和视频还原模块的解码器。这两种类型的峰值信噪比值的实验结果均为测试视频序列 10 个 I 帧的亮度峰值信噪比的平均值(由于所有操作均在亮度块中进行, 对色度块没有任何影响)。表 1 为视频测试序列水印嵌入前后对视频序列质量的影响情况。

表 1 测试序列水印嵌入前后 PSNR 值比较

测试序列	PSNR(dB)	PSNR1(dB)	PSNR2(dB)
Foreman	38.989	38.972	43.940
Carphone	41.388	41.360	43.480
Coastguard	38.763	38.753	51.140
Akiyo	46.877	46.840	45.260
Container	41.236	41.192	52.990

从表 1 可以看出, PSNR 和 PSNR1 相差极小, 这是由于本文算法的水印是在熵解码后嵌入水印再进行熵编码, 嵌入水印过程没有进行运动补偿, 所以不会对视频质量造成影响。在一般的应用场合, 即如果解码器中没有视频数据还原模块, 嵌入的水印会在解码运动补偿时造成嵌入水印一定的误差漂移, 但表 1 中 PSNR2 都在 43 dB 以上, 因此水印算法可以保证解码端视频的质量。在一些特殊的应用场合, 要求提取水印后能还原视频, 如果解码器中含有本文算法中的水印提取和数据还原模块, 则解码端的 PSNR2 值为无穷大, 即通过可逆水印会得到原始的视频。

表 2 为嵌入水印前后不同图像码率的变化情况。可以看到水印前后码率的变化并不大, 满足码率稳定的要求。

表 2 测试序列水印嵌入前后码率的变化

测试序列	水印前码率 (kbit/s)	水印后码率 (kbit/s)	码率增量 (kbit/s)
Foreman	270.94	272.98	2.04
Carphone	258.77	258.88	0.11
Coastguard	356.45	356.90	0.45
Akiyo	128.19	128.23	0.04
Container	257.50	257.73	0.23

当篡改者解码已嵌入水印的 H.264 码流，通过改变敏感数据后再进行编码后，通过本文的算法，可以定位到篡改的区域。图 8 为篡改区域的定位结果。从图中可以看出，本文算法能有效定位遭篡改的区域。

表 3 为本文算法与文献[7,19] 在同一个实验平台上的比较结果。所选的测试序列是 foreman 序列。

实验中，3 种算法嵌入的水印比特数是相同的，即 1 个宏块中嵌入 1 bit 的水印信息。从表 3 可以看出，本文算法水印嵌入前后对视频质量的影响与文献[7]相比，PSNR1 相差不大，PSNR2 值及码率的影响比参考文献[7]大。与文献[19]相比，本文算法的 PSNR1 值和 PSNR2 值较大，即对视频质量的影响较小，但码率略大。这是由于文献[7]中的水印是不可逆的，采用的算法是在一个宏块的固定位置嵌入 1bit 的水印信息，因此其对视频质量和码率影响较小。文献[19]是可逆水印，其水印的嵌入方法是利用预测差值的直方图平移嵌入水印，由水印的嵌入算法可知，无论嵌入的水印信息是“0”还是“1”， 4×4 子块的最后一个非零系数都要增“1”或者减“1”，这导致对视频质量的影响较大，特别是当前块的最后一个非零系数是低频系数时，嵌入水印对当前块的影响更明显。本文提出的算法在宏块中嵌入 1 bit 的水印信息和 1 bit 的被嵌入 DCT 系数的 LSB 值，每个宏块比文献[7,19]多嵌入 1 bit 信息，但本文算法在宏块中活性最大的子块中嵌入水印，在视觉上

可以减少水印的可见性。虽然码率有所增加，但本文算法是可逆水印，即解码端可以得到原始视频数据，并可以对遭到篡改的区域进行定位。

4 结束语

针对已有算法的不足，本文根据人眼的视觉特性，结合 H.264 编码标准的特性，通过在一个宏块中选取活性最大的子块，自适应修改 I 帧每个宏块亮度分量量化后的最后一个非零系数，并把最后一个非零系数的 LSB 系数嵌在其后的一个系数，达到嵌入可逆脆弱水印的目的。算法嵌入水印不必进行完全解码，从而减少计算复杂度；考虑人眼视觉特性，增加水印的不可见性。提取端能实现水印的盲提取，而且不需要再输入水印的位置信息，提取水印的同时能实现视频数据的还原。实验结果表明，嵌入的可逆脆弱水印对视频质量和码率的影响较小，而且能有效定位遭篡改的区域。

存在的不足及下一步的工作：(1)进一步降低对视频质量的影响，主要是减少或者去除嵌入的水印在解码端造成误差的漂移；(2)对视频残差系数 DCT 改变从而嵌入可逆的数据，对视频的码率也造成一定的影响；(3)定位的精度不够高。接下来的工作主要是考虑如何控制误差漂移，把水印嵌入造成的视频质量的影响降到最低。另外，本文算法篡改区域的定位是宏块级别的，如何提高定位精度也是今后的工作之一。



图 8 Stefan 序列篡改区域定位

表 3 本文算法与文献[7,19]的比较

算法	PSNR1(dB)	PSNR2(dB)	码率增量(kbit/s)	是否可逆	是否定位
本文	38.972	43.94	2.04	是	是
文献[7]	38.981	44.75	1.46	否	否
文献[19]	38.543	43.01	1.97	是	否

参考文献

[1] Vleeschouwer C D, Delaigle J F, and Macq B. Invisibility and

application functionalities in perceptual watermarking — an overview[J]. *Proceedings of IEEE*, 2002, 90(1): 64-77.

- [2] Kwon S K, Tamhankar A, and Rao K R. An overview of H.264/MPEG-4 part 10[J]. *Journal of Visual Communication and Image Representation*, 2003, 17(2): 186-216.
- [3] Kim D W, Choi Y G, Kim H S, *et al.* The problems in digital watermarking into intra-frames of H.264/AVC[J]. *Image and Vision Computing*, 2010, 28(8): 1220-1228.
- [4] Zhang J and Ho A T S. An efficient authentication method for H.264/AVC[C]. 2nd European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology, London, 2005: 157-164.
- [5] Qiu G, Marziliano, Ho A T S, *et al.* A hybrid watermarking scheme for H.264/AVC video[C]. Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, 2004: 865-869.
- [6] Su P C, Wu C S, Chen I F, *et al.* A practical design of digital video watermarking in H.264/AVC for content authentication[J]. *Signal Processing: Image Communication*, 2011, 26(8-9): 413-426.
- [7] Wang C C and Hsu Y C. Fragile watermarking scheme for H.264 video authentication[J]. *Optical Engineering*, 2010, 49(2): 027003.
- [8] Xu D W and Wang R D. Watermarking in H.264/AVC compressed domain using Exp-Golomb code words mapping[J]. *Optical Engineering*, 2011, 50(9): 097402.
- [9] Qin C, Chang C C, and Chen, P Y. Self-embedding fragile watermarking with restoration capability based on adaptive bit allocation mechanism[J]. *Signal Processing*, 2012, 92(4): 1137-1150.
- [10] Zhang X P, Wang S Z, Qian Z X, *et al.* Reversible fragile watermarking for locating tampered blocks in JPEG images[J]. *Signal Processing*, 2010, 90(12): 3026-3036.
- [11] Chen H, Chen Z Y, Zeng X, *et al.* A novel reversible semi-fragile watermarking algorithm of MPEG-4 video for content authentication[C]. Proceedings of the Second International Symposium on Intelligent Information Technology Application, Shanghai, 2008, 3: 37-41.
- [12] Wang N N and Men C G. Reversible fragile watermarking for 2-D vector map authentication with localization[J]. *Computer-Aided Design*, 2012, 44(4): 320-330.
- [13] Wang J T, Yang Y W, Chang Y T, *et al.* A high verification capacity reversible fragile watermarking scheme for 3D models[J]. *International Journal of Innovative Computing, Information and Control*, 2011, 7(1): 365-378.
- [14] Chung K L, Huang Y H, Chang P C, *et al.* Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC[J]. *IEEE Transactions on Circuits and System for Video Technology*, 2010, 20(11): 1643-1647.
- [15] Lie W N, Lin T C, Tsai D C, *et al.* Error resilient coding based on reversible data embedding technique for H.264/AVC video[C]. Proceedings of IEEE International Conference on Multimedia & Expo, ICME 2005, Amsterdam, 2005: 1174-1177.
- [16] Lin S D, Meng H C, and Su Y L. A novel error resilience using reversible data embedding in H.264/AVC[C]. International Conference on Information, Communications and Signal Processing, Singapore, 2007: 1-5.
- [17] Fridrich J, Goljan M, and Du R. Invertible authentication watermark for JPEG images[C]. IEEE International Conference on Information Technology: Coding and Computing (ITCC), Las Vegas, 2001: 223-227.
- [18] Huo W J and Zhu Y S. A reversible watermarking algorithm with error-drift elimination in H.264/AVC stream[C]. Proceedings of International Conference on Consumer Electronics, Communications and Networks, Xianning, 2011, 4: 2893-2896.
- [19] Lin Y C and Li J H. Reversible watermarking for H.264/AVC videos[C]. International Scientific Research and Experimental Development. VII. International University Conference 2011, Paris, 2011, 78: 828-831.
- [20] Liu Y X, Li Z T, and Ma X J. Reversible data hiding scheme based on H.264/AVC without distortion drift[J]. *Journal of Software*, 2012, 7(5): 1059-1065.
- [21] 刘红梅, 黄继武, 肖自美. 一种小波变换域的自适应视频水印算法[J]. *电子学报*, 2001, 29(12): 1656-1660.
- Liu H M, Huang J W, and Xiao Z M. An adaptive video watermarking algorithm in wavelet domain[J]. *Acta Electronica Sinica*, 2001, 29(12): 1656-1660.
- [22] Mansouria A, Znaveh A M, Torkamani-azar F, *et al.* A low complexity video watermarking in H.264 compressed domain[J]. *IEEE Transactions on Information Forensics and Security*, 2010, 5(4): 649-657.
- 张维纬: 男, 1982年生, 博士生, 研究方向为数字版权管理与数字水印等。
- 张茹: 女, 1976年生, 副教授, 研究方向为数字水印、数字图像取证与密码学等。
- 刘建毅: 男, 1980年生, 副教授, 研究方向为信息内容安全、智能信息处理等。