安全的 MPEG 压缩视频数字隐写算法

徐长勇 平西建 刘翠卿
(解放军信息工程大学信息工程学院, 郑州 450002)

摘 要 视频数字隐写具有隐藏容量大的优点,但是通常在获得隐藏容量的同时,却忽视了安全性。为了在二者之间取得平衡,利用纠错码信息隐藏的原理,提出了一种以压缩视频为载体的数字隐写算法。该算法在进行秘密信息嵌入时,采用了二次嵌入的策略,即首先将秘密信息嵌入到纠错码码字,然后将得到的载密纠错码码字嵌入到与不采用差分编码的系数相对应的行程幅度对的幅度值,同时采用比特率控制策略来减小信息嵌入前后的视频流长度变化。实验及分析表明,该算法的隐写结果不仅具有较好的视觉和统计上的不可感知性,而且满足密码学中的"Kerchhoff 准则"。此外,该算法在确保一定的嵌入容量的情况下,能够使视频流长度在信息嵌入前后保持近似不变。

关键词 信息隐藏 数字隐写 压缩视频 纠错码 安全性
中图法分类号: TP309, TN918.74 **文献标识码**: A **文章编号**: 1006-8961(2009)11-2237-10

Secure Steganographic Algorithm for MPEG Compressed Video

XU Chang-yong, PING Xi-jian, LIU Cui-qing
(*Institute of Information Engineering, PLA Information Engineering University, Zhengzhou* 450002)

Abstract Video steganography has the advantage of large embedding capacity, but the security is usually neglected when obtain the capacity. In order to balance the two aspects, a secure steganographic algorithm for embedding secret information in compressed video is proposed utilizing the principle of steganography based on error-correcting code. The strategy of two-time embedding is designed to embed secret bits. In other words, the secret bits are firstly embedded into the error-correcting codeword. Then by modifying the level value of run-level pair corresponding to the DCT coefficient which is not differentially encoded, the stego error-correcting codeword obtained by first-time embedding is secondly embedded. At the same time, measures are taken for the purpose of preserving video stream size. Experimental results show that the algorithm has a good visual and statistical imperceptibility, and satisfies the Kerchhoff rule in cryptography. Furthermore, the algorithm has a considerable embedding capacity, and can keep the nearly invariant size of video stream.

Keywords information hiding, steganography, compressed video, error-correcting code, security

1 引 言

数字隐写(steganoraphy)是信息隐藏技术的重要分支之一,它是将秘密信息嵌入到公开的数字媒体中(如图像、音频、视频等),以不引起第 3 方注意的方式来进行隐密传输的一种通信手段。一般来说,数字隐写应具有安全性和嵌入容量两个方面的重要特性。前者确保了隐密通信过程无法被发现,或者更进一步来说,即使发现了隐密通信,也无法准确提取出秘密信息;而嵌入容量则可保证通信时能够传输足够多的数据。与信息隐藏技术的另一重要

基金项目:国家自然科学基金项目(60473022)
收稿日期:2008-07-08;**改回日期**:2008-10-07
第一作者简介:徐长勇(1980~),男。解放军信息工程大学信号与信息处理专业博士研究生。主要研究方向为信息隐藏、图像处理、网络信息安全等。E-mail:chyong80@163.com

分支——数字水印不同,后者强调鲁棒性,要求嵌入的水印能够抵抗各种攻击,而数字隐写则要求安全性是第 1 位的。它首先要求隐写不对载体造成人为的修改痕迹;其次要求隐写后,载体的统计特性保持不变;最后,它是在秘密信息嵌入与提取过程中使用密钥,以确保即使隐蔽通信被发现,也无法准确提取信息。近年来,图像数字隐写得到了快速发展,已有相当多的算法被提出。尽管受计算复杂度等因素的制约,视频数字隐写的发展相对落后,但是由于视频具有更大的可用载体空间,能够嵌入更多的秘密信息,因此无疑使得视频数字隐写也具有很高的研究价值。

根据存在形式的不同,视频数字隐写可以分为针对未压缩视频的和压缩视频的两类。文献[1]~[3]讨论了未压缩视频的隐写方法,然而为了便于存储和传输,通常需要对未压缩视频进行编码,因此该类算法需要能抵抗视频压缩。而压缩视频中的隐写算法则不存在这一问题,但这类算法除了安全性及嵌入容量等基本要求外,还需使得秘密信息嵌入前后的视频流长度尽可能保持不变。现有的压缩视频中的隐写和水印算法,通常都是根据视频编码标准的特点来嵌入信息,例如,它们都是修改 I 帧编码时产生的离散余弦变换(discrete cosine transform, DCT)系数[4-5]或运动预测时产生的运动向量[6-7],但是这类方法在嵌入信息后还需进行差值补偿,这不仅增加了计算复杂度,而且有可能对视频质量造成较大影响。而在变长编码(variable length coding, VLC)时产生的可变长码中嵌入信息,则不需要这一过程。Langelaar 等人提出的 lc-VLC 算法就是在修改后的码字长度不变的可变长码中嵌入信息[8],这些码被称为 lc-VLC,其隐藏容量取决于视频流中 lc-VLC 的个数。Mobasseri 等人提出的 VLC map 算法是通过构造变长编码对(VLC pair)来嵌入信息[9],其隐藏容量是每块 1 bit。Liu 等人提出的 A/S 算法[10]是在块的最后一个 AC 系数上基于 A/S 树来嵌入信息,其隐藏容量是每两块 1 bit。HCIS 算法[11]同样是通过修改变长码来嵌入信息,其隐藏容量较大,但是对视频质量的影响也较大。纵观这些通过修改变长码实现的隐藏算法,普遍存在着安全性问题。例如对于 lc-VLC 算法,只需找到视频流中的各个 lc-VLC,即可提取出信息。而对于 HCIS 算法,其 0 bit 填充策略可能会留下固有的标识特征。

针对现有算法存在的安全性问题,本文利用纠错码信息隐藏的原理,提出了一种安全的以压缩视频为载体的数字隐写算法。该算法首先将秘密信息嵌入到纠错码码字,然后将载密纠错码码字嵌入到与不采用差分编码的系数相对应的行程幅度对(run-level pair)的幅度值。同时在信息嵌入过程中还采用比特率控制策略来减少隐写造成的视频流长度变化。

2 MPEG 压缩视频数据分析

在 MPEG 压缩编码标准中,宏块是基本的编码单元,其编码方式有帧内编码和帧间编码两类。其中帧内编码宏块主要存在于 I 帧,同时也少量存在于 P 帧和 B 帧。对于帧内编码宏块,可先将其分为 8×8 的块进行 DCT,并对所得 DCT 系数进行量化,然后对量化后的 DC 系数进行差分编码;而对于 AC 系数,则是以 zigzag 扫描的顺序进行行程编码,最后通过对差分编码后的 DC 系数和行程编码后的 AC 系数进行变长编码来得到编码比特流。帧间编码宏块存在于 P 帧和 B 帧中,编码时,先对其进行运动补偿预测;然后对得到的运动矢量进行差分编码;同时对预测误差按 8×8 块进行 DCT,并对 DCT 系数进行量化;接着对量化后的 DC 和 AC 系数均进行行程编码;最后通过变长编码得到比特流。

差分编码是指对当前宏块与参考宏块之间相应数据的差值进行编码,以达到数据压缩的目的。因此对于数字隐写来说,如果修改采用了差分编码的系数,将会造成误差累积。为了消除误差的累积效应,需要在嵌入信息后进行差值补偿。而如果只修改帧内编码宏块中的 AC 系数以及帧间编码宏块中的 DC 和 AC 系数等没有采用差分编码的系数,则不存在前述问题。同时由于视频中存在着大量不采用差分编码的系数,因此如果在其中嵌入信息,则不仅能够保证嵌入信息后的视频质量,而且使嵌入容量也得到了保障。

对于不采用差分编码的 DCT 系数,都是先由行程编码得到行程幅度对(*run*, *level*),然后再通过进行变长编码来得到编码比特流。在行程幅度对中,幅度值 *level* 是量化后非零的 DCT 系数,行程值 *run* 表示按照 zigzag 扫描的顺序在该非零 DCT 系数之前存在的值为 0 的 DCT 系数的个数。因此,如果修改行程值,也就是说,如果该非零 DCT 系数之前的值为 0 的个数发生变化,则将会使得后续所有的非

零 DCT 系数的位置发生变化,从而导致解码视频严重失真。而如果修改幅度值,则只改变当前被修改的 DCT 系数,不会影响其他的系数。因此,本文通过修改行程幅度对中的幅度值来嵌入信息。表 1 是 MPEG-2 标准中的部分变长编码码表,该码表给出了行程幅度对与变长编码码字之间的编码关系。

表 1　部分变长编码码表(s 是符号位:"0"正"1"负)

Tab. 1　Part VLC table(s denotes sign bit)

(*run*, *level*)	变长码	码长
(0, 1)	11 s	3
(0, 2)	0100 s	5
(0, 3)	00101 s	6
(0, 4)	0000110 s	8
(0, 5)	00100110 s	9
(0, 6)	00100001 s	9
(0, 7)	0000001010 s	11
⋮		
(1, 1)	011 s	4
(1, 2)	000110 s	7
(1, 3)	00100101 s	9
(1, 4)	0000001100 s	11
⋮		

通过观察变长编码码表可见,通常来说,在行程相等的情况下,幅度值越小,对其进行改变所引起的变长码码长的变化越大。但由于随着幅度值的逐渐增大,其相应的码长变化逐渐减小,因此如果通过对较大的幅度值进行修改来嵌入信息,则可以减小信息嵌入对视频流长度的影响。同时,由于只需对视频进行部分解码即可得到行程幅度对,而不需要繁琐的 DCT 和逆 DCT,从而可以提高运算速度,而且由于行程幅度对所对应的码字长度可以通过变长编码码表得到,因此可以较好地对比特率进行控制。

3　纠错码数字隐写原理

现有的利用纠错码的信息隐藏方法主要有以下几种:(1)为了提高鲁棒性而对秘密信息先进行纠错编码再嵌入载体数据的方法[12],其主要用于数字水印;(2)在有噪信道中利用纠错能力强的纠错码携带秘密信息的方法[13],但是该方法嵌入容量较小;(3)利用纠错码的某些思想,以提高嵌入效率的研究,如矩阵编码等[14]。虽然这些方法从不同侧面应用了纠错码的检纠错性能,但并没有将其编码机制应用于信息隐藏。由于信息隐藏技术是利用数字媒体中存在的冗余来隐藏信息的,而纠错码是为了提高信息传输的可靠性而人为地加入冗余的具有严密数学结构的编码数据,因此纠错码在可靠传输环境下的编码冗余便成为一种可以利用的资源。文献[15]提出了利用纠错码的编码冗余来对纠错码码字的错误图样进行映射编码的信息隐藏方法。常见的纠错码包括汉明码、格雷码、RS 码,等等。

定义 1:设(n,k)纠错码的校验矩阵是$\boldsymbol{H}$,$\boldsymbol{c}$是该纠错码中的任一个码字,若用$\boldsymbol{c}$作为载体码字,载密码字用$\boldsymbol{r}$表示,错误图样为$\boldsymbol{e}=\boldsymbol{r}-\boldsymbol{c}$,则$\boldsymbol{S}^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{r}^{\mathrm{T}}=\boldsymbol{H}(\boldsymbol{c}+\boldsymbol{e})^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{c}^{\mathrm{T}}+\boldsymbol{H}\boldsymbol{e}^{\mathrm{T}}=\boldsymbol{0}+\boldsymbol{H}\boldsymbol{e}^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{e}^{\mathrm{T}}$就称$\boldsymbol{S}$为$\boldsymbol{r}$或者$\boldsymbol{e}$的伴随式。

由此可见:$\boldsymbol{S}^{\mathrm{T}}$仅与错误图样$\boldsymbol{e}$有关,而与发送的码字无关,即$\boldsymbol{S}$完全由$\boldsymbol{e}$确定。也就是说,如果在码字$\boldsymbol{c}$中嵌入信息,且其引入的错误在纠错范围之内,那么是可以通过纠错译码算法来进行识别,并获得错误图样的。由此就得到了如下利用纠错码的数字隐写原理。

设$\boldsymbol{m}$是要嵌入纠错码各个码字的秘密信息,如果在$\boldsymbol{m}$与错误图样$\boldsymbol{e}$之间建立一一对应的映射编码关系,并用$\boldsymbol{c}$作为载体码字,载密码字用$\boldsymbol{r}$表示,则可由$\boldsymbol{r}=\boldsymbol{c}+\boldsymbol{e}$完成秘密信息的嵌入(对于二元域上的纠错码来说,这里的"+"指的是模 2 加,下同)。由于纠错码编码体制为收发双方共知,因此当接收到载密码字$\boldsymbol{r}$时,即可通过计算$\boldsymbol{S}^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{r}^{\mathrm{T}}$来得到伴随式$\boldsymbol{S}$。由于$\boldsymbol{S}^{\mathrm{T}}$仅与$\boldsymbol{e}$有关,因此可以根据$\boldsymbol{S}^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{e}^{\mathrm{T}}$利用各种已有的译码方法求解出$\boldsymbol{e}$,再由其与秘密信息间的映射关系得到秘密信息。

大家知道,安全的数字隐写系统应该满足密码学中的"Kerchhoff 准则",即除了密钥,数字隐写设计与实现的全部细节公开外,没有密钥就不能提取秘密信息。由于通信双方是知道信息隐藏的编码机制的,如果监视通信的第 3 方也知道该编码机制的话,则上述数字隐写的安全性仅仅依赖映射编码的复杂性,显然是没有安全保障的,因此,根据"Kerchhoff 准则"的要求,应使用共享密钥来增加安全性。

设(n,k)纠错码可以纠正t个错误,且收发双方每码字共享密钥的长度满足$v\geqslant t+1$,则用于隐藏信息的载体码字$\boldsymbol{c}_{\mathrm{u}}$的长度为$n-v$,共享密钥$\boldsymbol{k}_{\mathrm{u}}$与$\boldsymbol{c}_{\mathrm{u}}$

共同组成纠错码码字 $\boldsymbol{r}$。$\boldsymbol{e}_u$ 是与载体码字相应的错误图样，$\boldsymbol{m}_u$ 是要嵌入载体码字的秘密信息，如果在 $\boldsymbol{m}_u$ 与 $\boldsymbol{e}_u$ 之间建立一一对应的映射关系，把 $\boldsymbol{c}_u$ 作为载体码字，载密码字用 $\boldsymbol{r}_u$ 表示，则 $\boldsymbol{r}_u = \boldsymbol{c}_u + \boldsymbol{e}_u$。当接收到载密码字 $\boldsymbol{r}_u$ 时，若将其与共享密钥 $\boldsymbol{k}_u$ 组合得到完整的纠错码码字 $\boldsymbol{r}$，则可计算伴随式 $\boldsymbol{S}$。同样可先根据 $\boldsymbol{S}^T = \boldsymbol{H}\boldsymbol{e}^T$ 求解 $\boldsymbol{e}$，再由映射关系得到秘密信息。

如果纠错码的信息位全部采用共享密钥的话，即共享密钥的长度 $v = k$，则共享密钥共有 2^k 种组合，对其进行 (n,k) 纠错码编码，即可得到 n 位的纠错码码字。此时，由于 v bits 的信息位全部取自共享密钥，因此只能把秘密信息嵌入到纠错码的校验数据位，其对应的载体纠错码码字长度为 $n-k$，其错误图样的个数为

$$N = C_{n-k}^0 + C_{n-k}^1 + \cdots + C_{n-k}^t \tag{1}$$

与其相对应的最大编码子集的比特长为

$$h = \lfloor \mathrm{lb}\ N \rfloor = \lfloor \mathrm{lb}(C_{n-k}^0 + C_{n-k}^1 + \cdots + C_{n-k}^t) \rfloor \tag{2}$$

其中，$\lfloor \cdot \rfloor$ 表示向下取整。

4 安全的压缩视频数字隐写算法

4.1 隐写算法框架

图 1 给出了本文提出的压缩视频数字隐写算法框架。视频流解析过程是通过对压缩视频进行变长解码来得到与不采用差分编码的 DCT 系数相对应的行程幅度对，视频流合成是通过对嵌入信息后的行程幅度对进行变长编码来得到载密视频流的过程。秘密信息嵌入时，首先对其进行映射编码得到错误图样，同时对共享密钥进行纠错编码得到载体码字，并将错误图样和载体码字通过第 1 次嵌入来得到载密码字；然后由第 2 次嵌入将载密码字嵌入到行程幅度对的幅度值中；最后由视频流合成过程来得到载密视频流。

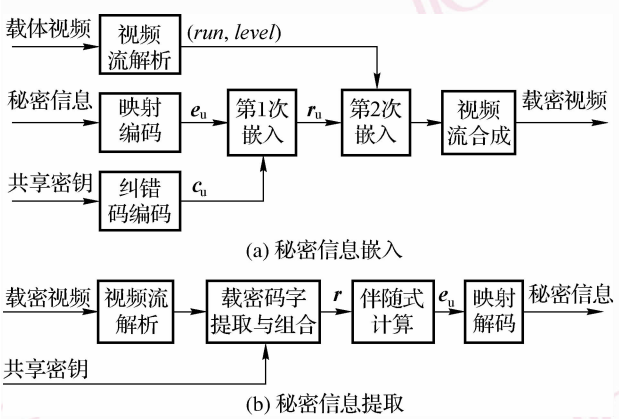

图 1 压缩视频数字隐写算法框架

Fig. 1 Sketch map of the proposed algorithm

秘密信息提取时，同样先由视频流解析得到行程幅度对，同时从中提取载密码字，并和共享密钥组合得到纠错码码字；然后计算伴随式，并求解错误图样；最后由其与秘密信息间的映射关系来得到秘密信息。

本文中的载体视频是指未嵌入信息的压缩视频流，载密视频是指嵌入信息后的视频流。为了减小秘密信息嵌入对视频质量的影响，本文仅在亮度分量(Y 分量)中嵌入信息，对色度分量不进行修改。

4.2 秘密信息嵌入

秘密信息嵌入之前，应先确定共享密钥的长度。本文中纠错码的信息位全部取自共享密钥，即共享密钥长度为 $v = k$。以下是秘密信息嵌入的详细过程：

(1)在秘密信息和错误图样之间建立映射编码关系。对于秘密信息序列 $\boldsymbol{M}$，将其按照 h 个一组进行分组，先得到秘密信息组合 $\boldsymbol{m} = (m_1 m_2 \cdots m_h)$，设码字校验位错误图样为 $\boldsymbol{e}_u = (e_1 e_2 \cdots e_{n-v})$；然后根据秘密信息与错误图样之间一一对应的映射编码表，将秘密信息序列映射编码成错误图样序列。

(2)先利用密钥生成长度足够长的伪随机 0,1 序列，并以其作为共享密钥序列；然后从该共享密钥序列中依次取 v bits，并对其进行纠错编码，即得到长度为 n bits 的纠错码码字 $\boldsymbol{c} = (c_1 c_2 \cdots c_n)$。

(3)用 $\boldsymbol{c}$ 的校验位 $\boldsymbol{c}_u = (c_{v+1} c_{v+2} \cdots c_n)$ 作为载体码字，载密码字用 $\boldsymbol{r}_u$ 表示，由 $\boldsymbol{r}_u = \boldsymbol{c}_u + \boldsymbol{e}_u$ 嵌入秘密信息。至此，第 1 次嵌入完成，即已把秘密信息嵌入到载体纠错码码字，但是还没有嵌入到载体视频中。

(4)对载体视频流进行解析，以寻找可嵌入信息的块。解析时，首先根据校验位的长度来选择合适的嵌入单元，即确定是以块，还是以宏块为单元来嵌入 $n-v$ bits 的载密码字。当校验位较短时，可以以块为单元；而当校验位较长时，如果仍以块为单元，则会由于在很多块中不存在如此多的满足条件的系数值，从而影响嵌入容量。所以，此时应以宏块为单元，将一个宏块中的 4 个块作为整体来进行考虑。

确定嵌入单元后，再以序列开始码为起点，对视频流进行解析，即得到片(slice)起始码。对于片中的一个嵌入单元，先对其进行变长解码得到行程幅

度对,然后从中提取与不采用差分编码的系数相对应的行程幅度对的幅度值,并通过取绝对值来得到幅度值序列 $\boldsymbol{L}=(level_1,level_2,\cdots,level_K)$,记为 $\boldsymbol{L}=(l_1,l_2,\cdots,l_K)$,$K$ 为该嵌入单元中不采用差分编码的系数个数。同时记录下 $\boldsymbol{L}$ 中各个值的符号(正或者负)。

设定阈值 T,统计 $\boldsymbol{L}$ 中所有不小于阈值 T 的幅度值,记为 $\boldsymbol{L}_s=(l_1^s,l_2^s,\cdots,l_D^s)$。显然,$\boldsymbol{L}_s\subseteq\boldsymbol{L}$,$D$ 表示 $\boldsymbol{L}$ 中不小于 T 的幅度值的个数。

①如果 $D\geqslant n-v$,则从 $\boldsymbol{L}_s$ 中选取幅度值最大的 $n-v$ 个值,并按其在 $\boldsymbol{L}$ 中的顺序排列得 $\boldsymbol{L}_u=(l_1^u,l_2^u,\cdots,l_{u_{n-v}})$。显然,$\boldsymbol{L}_u\subseteq\boldsymbol{L}_s$。如果 $\boldsymbol{L}_s$ 中存在多个与 $\boldsymbol{L}_u$ 中的幅度值相等的值,则取在 $\boldsymbol{L}$ 中排列靠前的那些幅度值。

②如果 $D<n-v$,则不在该单元中嵌入信息,继续对下一个单元进行处理。

(5)通过改变幅度值的最不重要比特(least significant bit, LSB)来将载密码字 $\boldsymbol{r}_u$ 嵌入到 $\boldsymbol{L}_u$ 中。

设 $l_i(l_i\in\boldsymbol{L}_u)$ 是当前单元中用于嵌入信息的幅度值,$\mathrm{LSB}(l_i)$ 是其 LSB 值,$r_j(r_j\in\boldsymbol{r}_u)$ 是要嵌入 l_i 的载密码字,$\tilde{l}_i$ 为修改后的幅度值,$\tilde{\boldsymbol{L}}_u$ 是对 $\boldsymbol{L}_u$ 中各幅度值进行修改后所得到的序列。同时,用 $\hat{\boldsymbol{L}}_u$ 表示 $\boldsymbol{L}$ 中除 $\boldsymbol{L}_u$ 以外的所有幅度值组成的集合,$\hat{\boldsymbol{L}}_u$ 中各幅度值的最大值是 $l_{\max}$,显然 $l_i\geqslant l_{\max}$。设嵌入信息前的行程幅度对所对应的变长码码长为 d_i,嵌入信息后,其码长变为 $\tilde{d}_i$,则信息嵌入引起的码长改变量 $\delta=\tilde{d}_i-d_i$(均以 bit 为单位)。分以下几种情况对幅度值 l_i 进行修改,并在每次修改时对 δ 值进行累加。此外,每当开始在一个新的单元嵌入信息时,令 $\delta=0$。

①当 $l_i=T$ 时,如果 $r_j=\mathrm{LSB}(l_i)$,则不需修改幅度值,即

$$\tilde{l}_i=l_i \tag{3}$$

如果 $r_j\neq\mathrm{LSB}(l_i)$,则为了实现盲提取,必须使修改后的幅度值 l_i 不小于阈值 T,只能将其加 1,即

$$\tilde{l}_i=l_i+1 \tag{4}$$

②当 $l_i>T$ 时,如果 $r_j=\mathrm{LSB}(l_i)$,则

$$\tilde{l}_i=l_i \tag{5}$$

如果 $r_j\neq\mathrm{LSB}(l_i)$,则根据 δ 值的大小以及 $l_{\max}$ 和 l_i 值的大小关系来决定如何对该幅度值进行修改:

1)当 $l_i-l_{\max}>1$ 时,可根据 δ 值的大小修改 l_i:

如果 $\delta\geqslant0$,则应通过将 l_i 减 1 来减少总码长,即

$$\tilde{l}_i=l_i-1 \tag{6}$$

如果 $\delta<0$,则应通过将 l_i 加 1 来增加总码长,即

$$\tilde{l}_i=l_i+1 \tag{7}$$

2)当 $l_i-l_{\max}=1$ 时,可根据等于 $l_{\max}$ 的幅度值在 $\boldsymbol{L}$ 中所处位置来决定如何修改 l_i:

如果 $\boldsymbol{L}$ 中,在 l_i 之前存在等于 $l_{\max}$ 的幅度值,则只能将 l_i 加 1,即

$$\tilde{l}_i=l_i+1 \tag{8}$$

如果 $\boldsymbol{L}$ 中等于 $l_{\max}$ 的幅度值都在 l_i 之后,则嵌入策略与 $l_i-l_{\max}>1$ 时相同,即

若 $\delta\geqslant0$,则通过将 l_i 减 1 来减少总码长,即

$$\tilde{l}_i=l_i-1 \tag{9}$$

若 $\delta<0$,则通过将 l_i 加 1 来增加总码长,即

$$\tilde{l}_i=l_i+1 \tag{10}$$

3)当 $l_i-l_{\max}<1$,由于 $l_i\geqslant l_{\max}$,则 $l_i=l_{\max}$。此时,为了使修改后仍能将该幅度值识别为最大的 $n-v$ 个幅度值之一,无论 δ 取值如何,只能将其加 1,即

$$\tilde{l}_i=l_i+1 \tag{11}$$

对于 $\boldsymbol{L}_u$ 中的各个幅度值,可根据上述各种不同情况进行修改来得到 $\tilde{\boldsymbol{L}}_u=(\tilde{l}_1^u,\tilde{l}_2^u,\cdots,\tilde{l}_{n-v}^u)$,并通过将修改后的结果放回 $\boldsymbol{L}$ 中来得到 $\tilde{\boldsymbol{L}}=(\tilde{l}_1,\tilde{l}_2,\cdots,\tilde{l}_K)$。至此,就完成了第 2 次嵌入,即把载密纠错码码字嵌入到行程幅度对的幅度值中。

(6)对嵌入信息后的幅度值序列 $\tilde{\boldsymbol{L}}$ 进行优化,以减小信息嵌入前后的视频流长度变化。

若定义单元长为该单元的第 1 个块的第 1 个系数到该单元的最后一个块的块结束符前的变长码码字长度,则由信息嵌入引起的单元长变化即为在该单元内嵌入信息后累加得到的码长改变量 δ,此时 $\delta=\sum_{i=1}^{n-v}(\tilde{d}_i-d_i)$。嵌入信息后,$\delta$ 不一定为 0。此时,可根据 δ 的大小来对 $\tilde{\boldsymbol{L}}$ 进行优化。

① 当 $\delta=0$ 时,不需再进行优化。

② 当 $\delta<0$ 时,说明信息嵌入使得单元长度减小。由信息嵌入过程可见,$\tilde{\boldsymbol{L}}_u$ 仍是由 $\tilde{\boldsymbol{L}}$ 中除 $\tilde{\boldsymbol{L}}_u$ 以

外的其他幅度值组成的集合。为了使得 $\delta=0$,需要对 $\bar{\boldsymbol{L}}_{\mathrm{u}}$ 中的某个幅度值进行调整。设 $l_j\in\hat{\boldsymbol{L}}_{\mathrm{u}}$ 是要调整的幅度值,$\tilde{l}_j$ 是调整后的幅度值,同时设 $\tilde{\boldsymbol{L}}_{\mathrm{u}}$ 中的最小幅度值是 $l_{\min}$,且 l_j 和 $\tilde{l}_j$ 所属的行程幅度对所对应的码长分别为 d_j 和 $\tilde{d}_j$,则调整原则是 $\tilde{l}_j=l_j+q$,且 $\tilde{l}_j<l_{\min}$(q 是正整数,一般取 1),同时应使得 $\tilde{d}_j-d_j\rightarrow|\delta|$。也就是说,$\tilde{l}_j$ 应小于 $\tilde{\boldsymbol{L}}_{\mathrm{u}}$ 中的最小幅度值,否则将可能造成信息无法准确提取,同时调整后的码长增加量应尽可能等于 $|\delta|$。如果在 $\bar{\boldsymbol{L}}_{\mathrm{u}}$ 中找不到满足上述条件的幅度值,则采用增加高频系数的策略,即在该嵌入单元的某个块的最后增加一个码长等于或接近于 $|\delta|$ 的行程幅度对,其幅度值一般为较小的正整数(如 1,2 等),而行程值则取较小的整数(如 0,1 等)。

③ 当 $\delta>0$ 时,信息嵌入使得单元长度增加,应该使调整后的单元长度减小。设 $l_j\in\hat{\boldsymbol{L}}_{\mathrm{u}}$ 及 $\tilde{l}_j$ 分别是调整前后的幅度值,其相应的码长分别为 d_j 和 $\tilde{d}_j$,则调整原则是 $\tilde{l}_j=l_j-q$(一般取 $q=1$),且 $d_j-\tilde{d}_j\rightarrow\delta$。也就是说,对 l_j 进行调整后的码长减少量应该尽可能地等于 δ。如果在 $\hat{\boldsymbol{L}}_{\mathrm{u}}$ 中找不到满足上述条件的幅度值,则采用丢弃高频系数的策略:从该嵌入单元中寻找一个块,如果该块中的最后一个幅度值不属于 $\tilde{\boldsymbol{L}}_{\mathrm{u}}$,且其所属的行程幅度对的对应码长等于或接近于 δ,则将该幅度值所属的行程幅度对删除。

如果对单元中的一个幅度值进行修改后仍不能使得 $\delta=0$,则不再对其他幅度值进行修改。这是因为:如果再修改其他系数,则将可能使视频质量进一步下降。而从安全性角度来说,与视频文件大小的轻微变化相比,视频质量的下降更容易被察觉。

将经过上述调整后的幅度值放回 $\bar{\boldsymbol{L}}_{\mathrm{u}}$ 即可得到 $\bar{\tilde{\boldsymbol{L}}}_{\mathrm{u}}$,然后和 $\tilde{\boldsymbol{L}}_{\mathrm{u}}$ 一起放回 $\tilde{\boldsymbol{L}}$,即可得到最终的载密幅度值序列 $\hat{\boldsymbol{L}}$。

(7)将 $\hat{\boldsymbol{L}}$ 中的幅度值乘以其各自的符号,并放回其各自所属的原行程幅度对,然后进行变长编码,并将编码所得码字放回其在原视频流中的相应位置,即可得到该嵌入单元所对应的载密视频流。

最后对视频流中其他所有符合条件的嵌入单元进行上述运算,直至到达序列结束码,就完成了整个压缩视频的秘密信息嵌入。

4.3 秘密信息提取

秘密信息提取的步骤如下:

(1)先根据纠错码的校验位长度,确定单元的大小,即块或宏块;然后由视频流解析过程得到与不采用差分编码的系数相对应的行程幅度对的幅度值,并计算绝对值,以形成幅度值序列 $\boldsymbol{L}=(l_1,l_2,\cdots,l_K)$。

(2)使用同样的阈值 T 计算 $\boldsymbol{L}$ 中所有不小于阈值 T 的幅度值,记为 $\boldsymbol{L}_{\mathrm{s}}=(l_1^{\mathrm{s}},l_2^{\mathrm{s}},\cdots,l_D^{\mathrm{s}})$。若 $D<n-v$,则该单元中不存在秘密信息,继续对下一个单元进行处理。若 $D\geqslant n-v$,则从 $\boldsymbol{L}_{\mathrm{s}}$ 中取幅度值最大的 $n-v$ 个值并按其在 $\boldsymbol{L}$ 中的顺序排列,先得到 $\boldsymbol{L}_{\mathrm{u}}=(l_1^{\mathrm{u}},l_2^{\mathrm{u}},\cdots,l_{n-v}^{\mathrm{u}})$,然后提取各个值的 LSB,即得到载密纠错码码字 $\boldsymbol{r}_{\mathrm{u}}$。

(3)先利用与发送端相同的密钥和相同的伪随机数产生方法来生成同样的共享密钥序列,然后依次取 v bits,并和 $\boldsymbol{r}_{\mathrm{u}}$ 组合在一起,即得到 n bits 的纠错码码字 $\boldsymbol{r}$。

(4)根据纠错码的校验矩阵 $\boldsymbol{H}$,由 $\boldsymbol{S}=\boldsymbol{r}\boldsymbol{H}^{\mathrm{T}}$ 计算伴随式 $\boldsymbol{S}$。由于 $\boldsymbol{S}^{\mathrm{T}}$ 仅与错误图样 $\boldsymbol{e}$ 有关,因此根据 $\boldsymbol{S}^{\mathrm{T}}=\boldsymbol{H}\boldsymbol{e}^{\mathrm{T}}$ 即可求解出 $\boldsymbol{e}$,再由其与秘密信息间的映射关系得到秘密信息,即可完成该单元的秘密信息提取。

最后对视频流中的其他单元进行上述运算,即可完成整个视频流的秘密信息提取。

4.4 秘密信息嵌入与提取示例

下面以一个例子来阐明秘密信息的嵌入和提取过程。若使用(7,4)汉明码,则码长 $n=7$,信息位长 $k=4$,则 $n-k=3$。码字的校验矩阵取如下形式:

$$\boldsymbol{H}=\begin{bmatrix}1&1&0&1&1&0&0\\1&0&1&1&0&1&0\\0&1&1&1&0&0&1\end{bmatrix}$$

相应的生成矩阵为

$$\boldsymbol{G}=\begin{bmatrix}1&0&0&0&1&1&0\\0&1&0&0&1&0&1\\0&0&1&0&0&1&1\\0&0&0&1&1&1&1\end{bmatrix}$$

用 $v=k=4$bits 长的共享密钥作为信息位进行汉明码编码,可得到 7bits 的码字 $\boldsymbol{c}$。码字校验位中可用于嵌入信息的错误图样数为 $N=C_{n-v}^{0}+C_{n-v}^{1}=4=2^2$,分别对应 3 个校验位发生错误以及没有错误的情况。同时,$h=\lfloor\mathrm{lb}N\rfloor=2$,即可以先将秘密信息序列以 2 bits 为 1 组进行分组;然后在秘密信息和错误图样之间建立一一对应的映射编码表。在错误图样和校验矩阵确定后,根据定义 1,伴随式也就确

定了。一种错误图样与秘密信息间的映射编码以及与伴随式之间的对应关系如图 2 所示。

秘密信息		错误图样		伴随式
0 0		0 0 0 0 0 0 0 1		0 0 1
0 1		0 0 0 0 0 1 0 0		0 1 0
1 0	↔	0 0 0 0 1 0 0 0	↔	1 0 0
1 1		0 0 0 0 0 0 0 0		0 0 0

图 2　错误图样与秘密信息间及伴随式之间的关系

Fig. 2　Mapping encoding between error-map and secret bits as well as relationship between error-map and syndrome

需要指出的是:秘密信息与错误图样之间的映射编码关系是可以由通信双方共同商定,并根据需要定时更换的,而错误图样与伴随式之间的对应关系则是由定义 1 确定的。

设共享密钥为[1 100],对其进行编码得到的汉明码码字为 $\boldsymbol{c}=[1\ 100\ 011]$,取其校验位 $\boldsymbol{c}_u=[011]$作为载体码字。设秘密信息 $\boldsymbol{m}=[00]$,其对应的部分错误图样为 $\boldsymbol{e}_u=[001]$,则 $\boldsymbol{r}_u=\boldsymbol{c}_u+\boldsymbol{e}_u=[010]$。

当以块为嵌入单元,设某块中与不采用差分编码的系数相对应的行程幅度对分别为(0,9)、(0,8)、(1,3)、(1,1)、(1,2)、(2,3)、(5,1)、(0,1),则通过提取各个幅度值,并取绝对值,可得 $\boldsymbol{L}=(9,8,3,1,2,3,1,1)$。设阈值 $T=1$,则 $\boldsymbol{L}_s=(9,8,3,1,2,3,1,1)$,$\boldsymbol{L}_u=(l_1^u,l_2^u,l_3^u)=(9,8,3)$,$\overline{\boldsymbol{L}}_u=(1,2,3,1,1)$,$l_{max}=3$。根据 $\boldsymbol{L}_u$ 的选取原则可知,$\boldsymbol{L}_u$ 中的幅度值(3)是行程幅度对(1,3)中的幅度值,$\overline{\boldsymbol{L}}_u$中的幅度值(3)是 $\boldsymbol{L}$ 中的第 2 个幅度值(3)。$\boldsymbol{L}_u$ 中各幅度值的 LSB 分别为 101,而 $\boldsymbol{r}_u=[010]$,所以 $\boldsymbol{L}_u$ 中的 3 个值都需要修改。

首先令 $\delta=0$,对于 $l_1^u=9$,应该根据式(6)进行修改,即 $\tilde{l}_1^u=8$,同时累加 δ 的值,则 $\delta=0$;对于 $l_2^u=8$,同样由式(6)可得 $\tilde{l}_2^u=7$,同时更新 δ,则 $\delta=-2$;对于 $l_3^u=3$,由于 $l_{max}=3$,则由式(11)可得 $\tilde{l}_3^u=4$,此时 $\delta=-2+2=0$。从而 $\tilde{\boldsymbol{L}}_u=(8,7,4)$,$\tilde{\boldsymbol{L}}=(8,7,4,1,2,3,1,1)$。由于 $\delta=0$,因此不需再对 $\tilde{\boldsymbol{L}}$ 进行优化。可先将 $\tilde{\boldsymbol{L}}$ 中各幅度值放回其各自所属的行程幅度对,并进行变长编码,然后将编码码字放回其在原视频流中的相应位置,即完成了该块的秘密信息嵌入。

接收方对与上述块对应的视频流进行部分解码后,若得到的 $\boldsymbol{L}=(8,7,4,1,2,3,1,1)$,则 $\boldsymbol{L}_u=(8,7,4)$,再提取其 LSB 得 $\boldsymbol{r}_u=[010]$。使用与发送端相同的共享密钥并和 $\boldsymbol{r}_u$ 组合得 $\boldsymbol{r}=[1\ 100\ 010]$。由定义 1 通过计算伴随式可得 $\boldsymbol{S}=\boldsymbol{r}\boldsymbol{H}^T=[001]$,由图 2 知,其所对应的错误图样 $\boldsymbol{e}=[0\ 000\ 001]$,再由其与秘密信息间的映射关系得 $\boldsymbol{m}=[00]$,即可准确提取出秘密信息。

5　实验与分析

为了验证本文算法的性能,使用标准的视频测试序列 mobile 和 guard 进行了实验。序列长度分别为 40 帧和 64 帧,帧大小分别为 720×576 和 352×288,色度格式均为 YCbCr4∶2∶0。实验时,首先采用标准 MPEG2 视频编码算法将其压缩成帧率为 25 fps、码率分别为 8 Mbps 和 4 Mbps 的视频文件;然后在其中嵌入信息,嵌入的信息为伪随机的 0,1 序列。此外,纠错码使用(7,4)汉明码和(23,12)格雷码,并分别以块和宏块为单元进行秘密信息的嵌入与提取,取阈值 $T=2$。结果显示,载密视频可以正常解码播放,并且在其未受攻击时,嵌入的信息可以准确提取。以下重点就嵌入信息后的视频质量、隐藏容量以及安全性等方面进行分析。

5.1　嵌入信息后的视频质量

对于两段视频嵌入信息后的质量评价,分别对没有嵌入信息以及本文算法使用(7,4)汉明码和(23,12)格雷码嵌入信息后的视频流进行解码,并对相应的帧进行比较。图 3 显示了视频 mobile 的第 25 帧和视频 guard 的第 41 帧在秘密信息嵌入前后的变化情况。(由于本文算法是在亮度分量中嵌入信息,因此这里只显示亮度分量的变化情况)。由图 3 可以看出,信息嵌入没有对视频质量造成视觉上可见的影响。对于两段视频中的其他帧也进行了比较,并得到了相同的结果。

本文使用峰值信噪比(peak signal to noise ratio, PSNR)来定量的衡量秘密信息嵌入对视频质量的影响,即分别对没有嵌入信息以及嵌入信息后的视频进行解码,并计算同一段视频中具有相同预测类型的帧与相对应的原始未压缩视频帧之间的平均 PSNR,计算得到的平均 PSNR 如表 2 所示。由表 2 可见,与不嵌入信息时相比,嵌入信息后各帧的 PSNR 下降均在 1dB 左右。这说明载密视频与没嵌入信息的视频相比,虽有一定的差别,但是由于这种差别主要存在于运动物体的边缘以及具有较复杂纹

(a) 无信息嵌入 (b) 使用(7，4)汉明码 (c) 使用(23，12)格雷码

(d) 无信息嵌入 (e) 使用(7，4)汉明码 (f) 使用(23，12)格雷码

图 3 秘密信息嵌入前后的视频质量变化情况

Fig. 3 Difference of video quality before and after embedding

表 2 秘密信息嵌入前后的 PSNR

Tab. 2 Average PSNR of different frames

视频	信息嵌入与否	不同类型帧的平均 PSNR(dB)		
		I	P	B
mobile	无信息嵌入	40.808 9	40.778 1	37.555 8
	使用(7,4)汉明码嵌入	39.419 5	39.738 4	36.724 0
	使用(23,12)格雷码嵌入	39.390 4	39.720 0	36.694 3
guard	无信息嵌入	41.069 3	40.946 8	33.836 5
	使用(7,4)汉明码嵌入	39.587 9	39.830 3	32.914 2
	使用(23,12)格雷码嵌入	39.561 7	39.782 6	32.867 7

理的区域,因此在视频播放时不易被人眼察觉,也就是说,隐写算法满足视觉不可感知的需求。

5.2 隐藏容量

对于本文算法来说,在采用(7,4)汉明码和(23,12)格雷码时,在每个可嵌入单元中所嵌入的信息分别为 2 bits 和 7 bits。以每秒内嵌入的比特数作为隐藏容量的度量,本文算法以及其他不同算法的隐藏容量对比见表 3。

从表 3 可以看出,本文算法在使用(7,4)汉明码和(23,12)格雷码时所嵌入的数据量相当,并且均高于 lc-VLC 算法、VLC Map 算法以及 A/S 算法。

表 3 不同算法的隐藏容量

Tab. 3 Embedding capacity of different algorithms

算法	隐藏容量(bps)	
	mobile	guard
lc-VLC 算法[8]	17 545	6 923
VLC Map 算法[9]	84 315	23 737
A/S 算法[10]	42 157	11 868
HCIS 算法[11]	609 231	274 292
本文算法使用(7,4)汉明码	123 066	41 478
本文算法使用(23,12)格雷码	115 333	40 332

由于 HCIS 算法基本上是在各个块的所有系数中都嵌入了信息,因此其嵌入容量高于本文算法。不过较大的嵌入量也必然导致对视频质量产生较大影响,与不嵌入信息时的视频帧相比,使用 HCIS 算法嵌入信息后,各帧的 PSNR 平均值下降了约 4 dB。

5.3　安全性

隐写算法的安全性首先取决于嵌入秘密信息后载密对象的视觉不可感知性,即首先要求从视觉上无法区分载密对象与载体对象之间的区别。由 5.1 节可见,无论是使用(7,4)汉明码,还是使用(23,12)格雷码得到的载密视频帧,与未嵌入信息时得到的视频帧均是无法区分的。由此可见,本文算法能够保证载密对象的视觉不可感知性。

在视觉不可感知性得到保证的前提下,统计不可感知性就成为另一个需要考虑的因素,也就是说,秘密信息嵌入不应改变载体数据的统计特性。考虑到本文算法实质上是通过对量化后的 DCT 系数进行修改来嵌入信息,而现有的针对 DCT 域嵌入的隐写分析方法通常是通过秘密信息嵌入前后 DCT 系数的统计特性的变化来进行的,并且 I 帧采用了与静止图像相类似的编码方式,因此可以用嵌入信息后对 I 帧进行编码的 DCT 系数直方图的变化情况来度量其统计特性的变化。图 4 是对信息嵌入前后视频 mobile 中的某一 I 帧进行编码所得的 DCT 系数的直方图的变化情况。由图 4 可见,直方图的变化非常小。

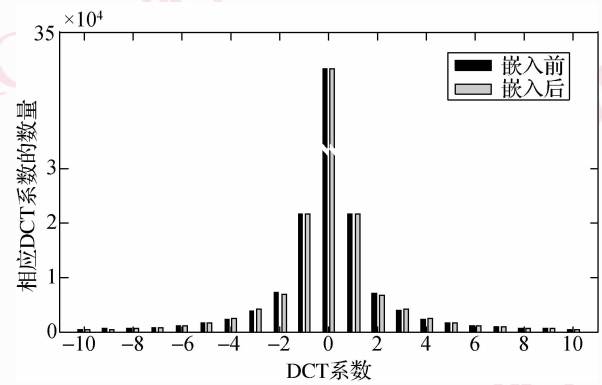

图 4　秘密信息嵌入前后的 DCT 系数直方图

Fig. 4　Histogram of DCT coefficients

密钥的使用对数字隐写的安全性来说,亦是一个非常重要的因素,它能确保即使隐蔽通信被发现,传输的秘密信息也无法被准确提取。现从以下两个方面对密钥的使用进行分析:

首先,在共享密钥错误的情况下,接收方无法准确提取出秘密信息。设(n,k)纠错码可以纠正 t 个错误,本文使用的共享密钥长度为 $v=k$,显然 $v>t$,此时即使攻击者掌握隐写所使用的纠错码编码机制,但由于用于嵌入信息的载体中的纠错码部分码字处于不可纠错范围,因此也无法准确提取出秘密信息。例如,对于 4.4 节中的示例,假设接收端使用了错误的共享密钥,如[1 110],则将其和 $\boldsymbol{r}_u$ 组合可得 $\boldsymbol{r}$=[1 110 010],通过计算伴随式即可得 $\boldsymbol{S}$=[010]。由图 2 知,对应的错误图样为 $\boldsymbol{e}$=[0 000 010],再由秘密信息与错误图样之间的映射关系可知,嵌入的秘密信息为[10],而发送端实际嵌入的是[00],两者并不一致,即秘密信息没有被准确提取。这说明共享密钥的使用增强了算法的安全性,即在共享密钥错误的情况下,接收端也无法准确提取出秘密信息。

其次,秘密信息和错误图样之间的映射编码关系以及各个可嵌入单元的选择顺序都是可以由密钥控制的。当然,也可以通过对秘密信息进行预处理来进一步增强安全性。

综合以上分析可见,本文提出的压缩视频数字隐写算法具有较好的视觉不可感知性和统计不可感知性,并且满足密码学中的“Kerchhoff 准则”。

此外,秘密信息嵌入前后的视频流长度如表 4 所示。由表 4 可见,由于采用了比特率控制策略,视频流长度的变化是非常小的。实际上,从安全性角度来说,视频流长度的轻微变化并不容易被第 3 方发现。这是因为即使是在相同的编码条件下,对同样大小、同样长度的视频序列进行编码,只要视频内容不同,编码得到的视频流长度就不是相同的。

表 4　不同情况下的视频流长度

Tab. 4　Video stream size in different cases(Bytes)

情况	视频流长度(Bytes)	
	mobile	guard
无信息嵌入	1 624 068	1 300 484
使用(7,4)汉明码	1 623 556	1 299 583
使用(23,12)格雷码	1 623 438	1 301 158

6　结　论

本文提出并实现了一种以压缩视频为载体的数字隐写算法。该算法具有如下特点:

(1)通过在秘密信息与错误图样之间建立映射编码关系,同时由于是使用由对共享密钥进行纠错码编码得到的纠错码的校验位码字作为载体码字,从而增强了算法的安全性。

(2)由于是将载密码字嵌入到与不采用差分编码的系数相对应的行程幅度对的幅度值,同时嵌入是通过修改块中较大的幅度值来实现的,因此较好地保证了嵌入信息后的视频质量。

(3)秘密信息嵌入时,由于采取了比特率控制策略,从而可使得秘密信息嵌入前后的视频流长度保持近似不变。

进一步的研究工作是对视频流中编码的 DCT 系数进行更加充分的利用,以提高嵌入容量;同时可考虑使用其他的纠错码,以增加安全性。

参考文献(References)

1 Chen Tian-hang, Liu Shao-hui, Yao Hong-xun, *et al*. Spatial video watermarking based on stability of DC coefficients [A]. In: Yeung D S(eds): Lecture Notes in Artificial Intelligence 3930 [C], Berlin, German: Springer-Verlag, 2006: 1033-1042.

2 Lancini R, Mapelli F, Tubaro S. A robust video watermarking technique in the spatial domain [A]. In: Proceedings of International Symposium on Video/Image Processing and Multimedia Communications [C], Zadar, Croatia, 2002: 251-256.

3 Chae J J, Manjunath B S. Data hiding in video [A]. In: Proceedings of International Conference on Image Processing [C], Kobe, Japan, 1999, **1**: 311-315.

4 Simitopoulos D, Tsaftaris S A, Boulgouris N V, *et al*. Compressed domain video watermarking of MPEG streams [A]. In: Proceedings of International Conference on Multimedia and Expo [C], Lausanne, Switzerland, 2002: 569-572.

5 Liu Hong-mei, Shao Feng-lian, Huang Ji-wu. A MPEG-2 video watermarking algorithm with compensation in bit stream [A]. In: Safavi-Naini R, Yung M(eds): Lecture Notes in Computer Science 3919 [C], Berlin, German: Springer-Verlag, 2006: 123-134.

6 Dai Yuan-jun, Zhang Li-he, Yang Yi-xian. A new method of MPEG video watermarking technology [A]. In: Proceedings of International Conference on Communication Technology [C], Beijing, China, 2003: 1845-1847.

7 Zhang Gui-dong, Mao Yao-bin, Wang Zhi-quan. A video watermarking scheme based on motion vector [J]. ACTA Scientiarum Naturalium Universitatis Sunyatseni, 2004, **43**(Suppl. 2): 117-119. [张桂东, 茅耀斌, 王执铨. 一种基于运动矢量的视频水印方案[J]. 中山大学学报(自然科学版), 2004, **43**(增刊. 2): 117-119.]

8 Langelaar Gerhard C, Setyawan Iwan, Lagendijk Reginald L. Watermarking digital image and video data [J]. IEEE Signal Processing Magazine, 2000, **17**(5): 20-46.

9 Mobasseri Bijan G, Marcinak Michael P. Watermarking of MPEG 2 video in compressed domain using VLC mapping [A]. In: Proceedings of ACM Workshop on Multimedia and Security [C], New York, USA, 2005: 91-94.

10 Liu Bin, Liu Fen-lin, Lu Bin, *et al*. Real-time steganography in compressed video [A]. In: Gunsel B(eds): Lecture Notes in Computer Science 4105 [C], Berlin, German: Springer-Verlag, 2006: 43-48.

11 Sun Yi-feng, Liu Fen-lin, Wang Guo-dong, *et al*. A new MPEG2 video steganography [A]. In: Proceedings of 7th CIHW [C], Nanjing, China, 2007: 84-88. [孙怡峰, 刘粉林, 王国栋等. 一种基于 MPEG2 视频编码的隐写算法[A]. 见: 第七届全国信息隐藏暨多媒体信息安全学术大会论文集[C], 南京, 2007: 84-88.]

12 Albanesi M G, Guerrini F. Authentication and recovery of digital images using error correcting codes [A]. In: Proceedings of ACM Workshop on Multimedia and Security: Authentication, Secrecy, and Steganalysis [C], Juan-les-Pins, France, 2002: 23-26.

13 Wang Wei-xiang, Liu Yu-jun, Li Wen-xiong. Implementation of information hiding technology in channel coding using LDPC code [J]. Journal of Information Engineering, 2006, **7**(1): 49-50,53. [王伟祥, 刘玉君, 李文雄. 利用 LDPC 码实现信道编码信息隐藏技术[J]. 信息工程大学学报, 2006, **7**(1): 49-50,53.]

14 Crandall Ron. Some Notes on Steganography [EB/OL]. http://os. inf. tu-dresden. de/ ~ westfeld/crandall. pdf.

15 Liu Cui-qing, Ping Xi-jian, Wang Yun-he, *et al*. Research on LSB steganography method in images based on parity-check matrix of perfect codes [J]. Journal of Harbin Institute of Technology, 2006, **38**(Sup): 795-799. [刘翠卿, 平西建, 王云鹤等. 基于完备码监督矩阵的图像 LSB 信息隐藏方法[J]. 哈尔滨工业大学学报, 2006, **38**(增刊): 795-799.]