TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 07/09 pp198–209
Volume 22, Number 2, April 2017

Novel Video Steganography Algorithm Based on Secret Sharing and Error-Correcting Code for H.264/AVC

Yingnan Zhang*, Minqing Zhang, Xiaoyuan Yang, Duntao Guo, and Longfei Liu

Abstract: In this paper, we analyze the video steganography technique, which is used to ensure national security and the confidentiality of the information of governmental agencies and enterprises. Videos may be used to transmit secrets and conduct covert communication. As such, we present an algorithm based on a secret sharing scheme and an Error-Correcting Code (ECC), which combines Grey Relational Analysis (GRA) with a partition mode in video compression standard H.264/AVC. First, we process secret information by secret sharing, and then use an ECC to process the obtained information. Moreover, we choose the Discrete Cosine Transform (DCT) blocks using GRA, and then use rules to hide the pretreated information in DCT coefficients of the video frames. Experimental results indicate that our algorithm has good invisibility, better robustness, good anti-steganalysis ability, and little influence on the bit rate of the video carrier. In addition, the bit error rate is low after attacks such as noise, filtering, or frame loss in the simulation environment.

Key words: steganography; video; secret sharing; error-correcting code; robustness; grey relational analysis; partition mode

1 Introduction

With the rapid development of the Internet, any information can be transferred, including information that should otherwise be secure. To ensure the security of important information, steganography techniques have emerged.

Steganography is a method for hiding secret messages in ordinary carriers without revealing their existence[1]. The carrier (or cover file) may be a digital image, audio file, or video file. If secret information has been hidden in a cover file, it can be transferred across public channels, whether they are secure or not.

In recent years, the development of steganography based on digital images has made strong advances. Researchers have proposed many high-performance algorithms, such as LSB[2], LSB matching[3], BPCS[4], F5[5], nsF5[6], MME[7], and OutGuess[8], as well as a number of recent adaptive algorithms such as HUGO[9], WOW[10], MVGG[11], and UED[12]. However, size limitations currently restrict the volume of information that can be hidden. For this reason, steganography based on digital video has recently been developed[13–16]. Compared with traditional media like digital images, the capacity of video is much greater, which makes video steganography very convenient, as well as offering greater redundancy and high communication quality and robustness.

The video compression standard H.264/AVC is the most mature standard available, and has high compression efficiency and transmission reliability. Furthermore, it is well adapted for network transmission. Since its introduction, H.264/AVC has replaced most other standards, and today many

• Yingnan Zhang, Minqing Zhang, Xiaoyuan Yang, Duntao Guo, and Longfei Liu are with Key Laboratory of Network & Information Security of PAP, Engineering College of PAP, Xi'an 710086, China. E-mail: zyn583@163.com; api_zmq@126.com; xyyangwj@126.com; gdt1979@qq.com; ya_zhou_521@163.com.

∗ To whom correspondence should be addressed.

Manuscript received: 2016-04-20; revised: 2016-05-23; accepted: 2016-06-30

videos are compressed using H.264/AVC. Many video websites also use the H.264/AVC as their main compression standard[17]. As such, research efforts to advance our understanding of H.264/AVC data hiding methods are of vital importance.

Today's highly developed Internet environment is a challenging one. When being transmitted, stego-carriers (carriers embedded with information) are easily influenced by noise, filtering, frame losses, packet losses, and other various attacks that can cause serious damage to the information hidden in the carrier, as illustrated in Fig. 1. To better guarantee success in covert communication, various proposals have been made to make the steganographic algorithm more robust.

Zhang et al.[18] proposed a robust video steganography scheme based on H.264/AVC. Embedded into video data in the compressed domain, this scheme has high robustness and good visual quality without increasing the overall bit rate, but cannot completely recover after frame loss attack.

Singh and Siddiqui[19] also proposed a robust scheme in which a chaotic system is used to generate a random sequence to hide data in the middle frequency coefficients of Discrete-Cosine-Transform (DCT) blocks, and the data are pretreated with an Arnold transform. Experimental results demonstrate that while the proposed algorithm achieves higher security and robustness, it does not have the ability to make error corrections.

Liu et al.[20] proposed a novel robust data-hiding algorithm based on an Error-Correcting Code (ECC). This method uses a Bose-Chaudhuri-Hocquenghem (BCH) ECC code to correct the error bits caused by network transmission. First, the BCH code encodes the data to be embedded, and then hides the data in the paired-coefficients of the block after performing the DCT. This code also induces distortion drift by intra-frame prediction. The experimental results show that this algorithm achieves greater robustness and high visual quality, but the computation complexity is high due to the difficulty in encoding the data with the BCH code.

Zhang et al.[21] proposed a novel video steganography algorithm based on grey relational analysis that combines the partition modes with features of the H.264/AVC. First, the algorithm computes the grey relevancy of the block and determines the presence of any texture features. Then, it embeds information in the DCT coefficients. Experimental results show that the proposed algorithm has little impact on video quality and bit rates, and has the advantages of having anti-noise, anti-filter, and high embedding capacities.

As two important branches of information security, cryptography and steganography have a similar purpose that is realized in different ways, and each method has its own advantages. Today, more and more schemes combine these two techniques[22, 23].

The secret sharing scheme in cryptography is superior in that it requires only some so-called sub-secret pieces to obtain the original secret information. Furthermore, many of the steganography algorithms have features that minimize the loss of the secret information to be transmitted, such as those in

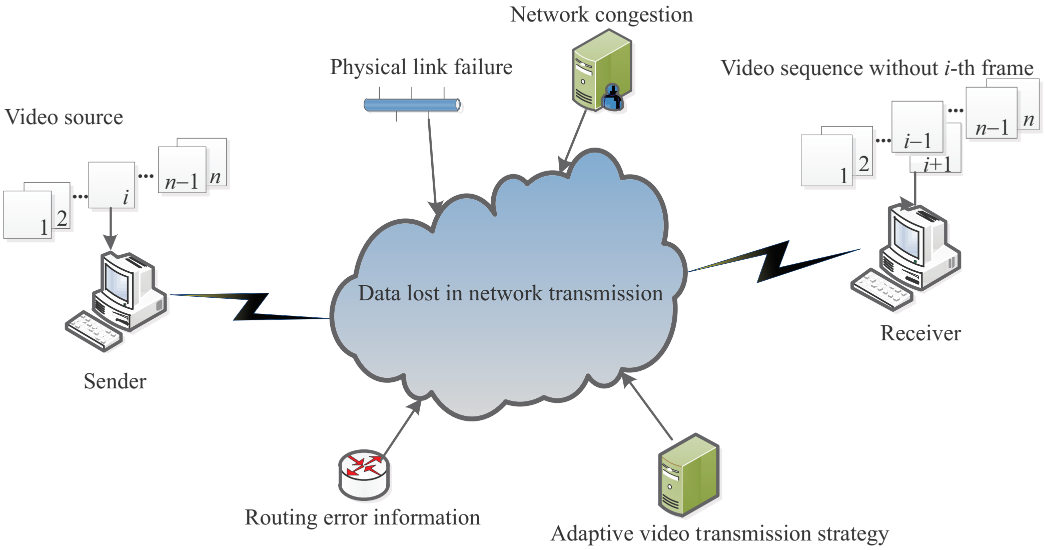

Network congestion
Physical link failure
Video sequence without *i*-th frame
Video source
1
2
...
i
...
n−1
n
1
2
...
i−1
i+1
...
n−1
n
Data lost in network transmission
Sender
Receiver
Routing error information
Adaptive video transmission strategy

Fig. 1 The condition of the network transmission.

Refs. [24–28].

In this paper, we present a video steganography algorithm that is robust, error-concealed, and can be used in covert communication. The main contributions of this work are as follows. First, we use Shamir's secret sharing scheme to reconstruct the original secret information bit after frame loss. Next, we use the Hamming code prior to embedding the data to correct the error bits caused by network transmission, noise, and filtering. Third, we choose the embedding blocks by Grey Relational Analysis (GRA), and use rules to hide the bits to ensure minimal distortion and blind extraction.

The rest of this paper is organized as follows: In Section 2, we describe the preliminaries of our scheme. In Section 3, we describe in detail our embedding and extraction mechanisms. We present our experimental results and analysis in Section 4, and in Section 5 we draw our conclusions and share our plans for future work.

2 Preliminaries

2.1 Partition mode

In the H.264/AVC coding standard, each macro-block (16×16 pixels) can be divided into four modes: 16×16, 16×8, 8×16, and 8×8, as shown in Fig. 2. As shown in Fig. 3, when using the 8×8 mode, the split mode has four sub-macro-block modes: 8×8, 8×4, 4×8, and 4×4. This segmentation improves the correlation between each macro-block.

H.264/AVC uses a model based on the Lagrangian rate-distortion mode[29], which traverses all of the current macro-block modes and calculates its Lagrangian cost to determine which macro-block partition mode is optimal. The Lagrangian formula is as follows:

$$J_{\text{MODE}}(S_i, I_i|Q, \lambda_{\text{MODE}}) = D_{\text{REC}}(S_i, I_i|Q)+ \lambda_{\text{MODE}} \times R_{\text{REC}}(S_i, I_i|Q) \qquad (1)$$

where λ_{MODE} is the Lagrangian parameter, $D_{\text{REC}}(S_i, I_i|Q)$ is the bit rate of the encoded bit stream, $R_{\text{REC}}(S_i, I_i|Q)$ is the distortion after coding, S_i is the i-th macro-block, I_i is the current coding mode, and Q is the quantization step size. The partition mode obtains the minimum cost of the current mode. The H.264/AVC then calculates the current block cost function, and selects the best mode with respect to the cost function, and marks it[17]. We typically take the more symmetrical macro-block 4×4, 8×8, or 16×16 as the modulating block.

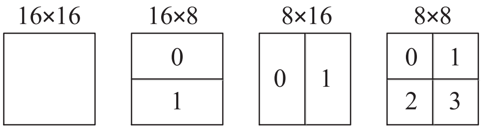


Fig. 2 Macro-block level mode.

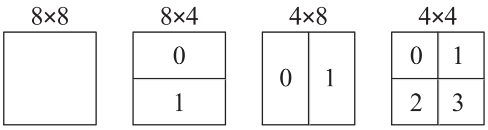


Fig. 3 Sub-macro-block level mode.

2.2 Secret sharing

In 1979, Shamir developed a cryptographic scheme, known as the secret sharing or threshold scheme, that divides data D into n pieces in such a way that D can be easily reconstructed from any t pieces, whereby even a complete knowledge of $t-1$ pieces reveals absolutely no information about D. This technique enables cryptographic systems that are supported by the construction of robust key management schemes to function securely and reliably, even when some of the pieces are destroyed or security breaches expose all but one of the remaining pieces[30]. The details of this scheme are as follows.

Assume that the secret data to be transmitted is $a_0 = D$. The sender builds one polynomial $p(x)$ of degree $t-1$. Assume the data D is a number. To divide D into pieces labelled D_i, we pick a random $t-1$ degree polynomial:

$$p(x) = D + a_1 x + \cdots + a_{t-1}x^{t-1} \bmod p \qquad (2)$$

where, p is a prime number, which is positive, and the coefficients $a_1, \ldots, a_{t-1}$ in $p(x)$ are randomly chosen from a uniform distribution of the integers in $[0, p)$ (which can also be described as $a_i \in Z(p)$, $i = 1, 2, \ldots, t-1$). Assuming that we want to divide the secret D into n pieces, the sender calculates $y_i = p(x_i)$, $i = 1, 2, \ldots, n$, with (x_i, y_i) being the secret data to be transmitted, for which the polynomial can be public or not.

Assuming the receiver has t pieces, denoted as $(x_i, y_i), i = 1, 2, \ldots, t$, then we can calculate the secret D. The details of this calculation are as follows:

We obtain t equations:

$$\begin{cases} a_0 + a_1 x_1 + a_2 x_1^2 + \cdots + a_{t-1} x_1^{t-1} = y_1, \\ a_0 + a_1 x_2 + a_2 x_2^2 + \cdots + a_{t-1} x_2^{t-1} = y_2, \\ a_0 + a_1 x_3 + a_2 x_3^2 + \cdots + a_{t-1} x_3^{t-1} = y_3, \\ \vdots \\ a_0 + a_1 x_t + a_2 x_t^2 + \cdots + a_{t-1} x_t^{t-1} = y_t \end{cases} \quad (3)$$

The linear equation has t unknown numbers in t equations, and x can be any positive integer, where the number is not important. Then, we can prove the secret sharing procedure as follows. We obtain the determinant of the coefficients from the following linear equations:

$$\begin{vmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{t-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{t-1} \\ 1 & x_3 & x_3^2 & \cdots & x_3^{t-1} \\ & & \vdots & & \\ 1 & x_t & x_t^2 & \cdots & x_t^{t-1} \end{vmatrix} \quad (4)$$

This is a Vandermonde determinant[31] whose value is not zero when $x_i \neq x_j,\ i, j = 1, 2, \ldots, t$ and $i \neq j$, and we can know that it has only one answer by the Cramer rule[32]. If we have t or more pieces, we can correctly obtain the secret, but with fewer than t pieces, we cannot.

We can determine the secret using a Lagrange differential equation. If we get t pieces, such as $(x_i, y_i),\ i = 1, 2, \ldots, t$, we can construct the Lagrange differential equation as

$$f(x) = \sum_{i=1}^{t} y_i \prod_{j=1, j \neq i}^{t} \frac{x - x_j}{x_i - x_j} \quad (5)$$

The constant term can be described as

$$D = a_0 = f(0) = \sum_{i=1}^{t} y_i \prod_{j=1, j \neq i}^{t} \frac{-x_j}{x_i - x_j} \quad (6)$$

Then, we can obtain the secret as

$$D = \sum_{i=1}^{t} b_i y_i,\ b_i = \prod_{j=1, j \neq i}^{t} \frac{-x_j}{x_i - x_j} \quad (7)$$

2.3 Error correcting code

The ECC[33] can detect and even correct an error interrupted by the real network or attackers. To do so, we must lengthen the distance between the code elements by adding more elements after the original code.

There are many kinds of ECCs, including the BCH code[34], Hamming code, LDPC[35], LRPC[36], and Goppa code[37].

In video steganography processing, unlike image processing, the scheme must have low complexity to minimize the time consumed.

Considering the correcting ability, complexity, and cover file, the Hamming code has lower complexity than other ECCs[38]. As such, to lower the complexity of the embedding process, we chose the Hamming code as the error-correcting method in our algorithm.

Since we mainly use the Hamming code[39] in our work, we introduce it below.

The Hamming code, proposed by mathematician Richard Wesley Hamming in 1947, can correct one error.

We denote the code as C, which contains n bits and k information bits, all of which belong to the vector space F^n of degree n.

The bit rate is $R = 1 - \dfrac{n-k}{n}$, and, obviously, when $n \to \infty$, $R \to 1$.

Definition 1 A binary system code's length satisfies $n = 2^{n-k} - 1$

Definition 2 A code's check matrix has three conditions:

(1) Every row is independent;

(2) The degree is $(n-k) \times n$;

(3) No row consists of all zeros.

Based on the above, this code can be called an (n, k) Hamming code in the domain GF(2).

For example, a $(7, 4)$ Hamming code's encoding process can be described as follows:

$$\boldsymbol{c} = \boldsymbol{mG} \quad (8)$$

where $\boldsymbol{m}$ is the original information bit of degree k and $\boldsymbol{c}$ is the code obtained. Matrix $\boldsymbol{G}$ is the generator matrix of code C, and can be described as follows:

$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

For code $\boldsymbol{c}$, the check matrix $\boldsymbol{H}$ of the degree $(n-k) \times n$ satisfies

$$\boldsymbol{cH}^{\mathrm{T}} = \boldsymbol{0} \quad (10)$$

where the matrix $\boldsymbol{H}$ is as follows:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \quad (11)$$

Now, we can obtain $\boldsymbol{m}$ and its corresponding $\boldsymbol{c}$, as shown in Table 1.

The difference between the receiver and sender codes is called the error map, E, as shown in Table 2.

2.4 Analysis of feasibility

In this section, we describe Shamir's secret sharing

Table 1 Values of *m* and its corresponding *c*.

No.	*m*	*c*	No.	*m*	*c*
1	0000	0000000	9	1000	1000111
2	0001	0001011	10	1001	1001100
3	0010	0010101	11	1010	1010010
4	0011	0011110	12	1011	1011001
5	0100	0100110	13	1100	1100001
6	0101	0101101	14	1101	1101010
7	0110	0110011	15	1110	1110100
8	0111	0111000	16	1111	1111111

Table 2 Error map.

Syndrome	Error map	Error bit location
001	0000001	B0
010	0000010	B1
011	0001000	B2
100	0000100	B3
101	0010000	B4
110	0100000	B5
111	1000000	B6

method.

First, we assume that each group of the secret s is divided into n pieces, so we build the polynomial as follows:

$$p(x) = a_0 + a_1 x + \cdots + a_{t-1} x^{t-1} \bmod p \tag{12}$$

where $a_0 = s$ and the coefficients $a_1, \cdots, a_{t-1}$ in $p(x)$ are randomly chosen from a uniform distribution of the integers in $Z(p)$. We choose the prime number $p = 17$.

Next, we assume that the secret information is $s = 5$ and $t = 3$, and we can build the polynomial as follows:

$$p(x) = 5 + 4x + 2x^2 \tag{13}$$

We then divide the secret into six pieces, and in order to clearly illustrate this process, we let the x values of these six polynomials be 1, 2, 3, 4, 5, and 6, respectively. Then, we can calculate the six sub-secrets as follows:

$$\begin{cases} p(1) = 11 \bmod 17 = 11, & x = 1; \\ p(2) = 21 \bmod 17 = 4, & x = 2; \\ p(3) = 38 \bmod 17 = 1, & x = 3; \\ p(4) = 53 \bmod 17 = 2, & x = 4; \\ p(5) = 82 \bmod 17 = 7, & x = 5; \\ p(6) = 101 \bmod 17 = 16, & x = 6 \end{cases} \tag{14}$$

Now, we can hide the sub-secrets (11, 4, 1, 2, 7, and 16) into different frames, $f(1) = p(1)$, $f(2) = p(2)$, $f(3) = p(3)$, $f(4) = p(4)$, $f(5) = p(5)$, and $f(6) = p(6)$, and label them as a frame group. In this frame group, we need only three frames to reconstruct the original secret information.

Assuming we obtain six frames in this group, we use three to reconstruct the information, and randomly choose $f(1)$, $f(3)$, and $f(5)$.

According to the Lagrange differential equation, the original secret information can be rewritten as in Formula (15):

$$\begin{aligned} a_0 = s = f(1) \cdot \frac{(0-3)(0-5)}{(1-3)(1-5)} + \\ f(3) \cdot \frac{(0-1)(0-5)}{(3-1)(3-5)} + \\ f(5) \cdot \frac{(0-1)(0-3)}{(5-1)(5-3)} \bmod 17 = \\ 11 \times \frac{(0-3)(0-5)}{(1-3)(1-5)} + 1 \times \frac{(0-1)(0-5)}{(3-1)(3-5)} + \\ 7 \times \frac{(0-1)(0-3)}{(5-1)(5-3)} \bmod 17 = \\ \frac{176}{8} \bmod 17 = 22 \bmod 17 = 5 \end{aligned} \tag{15}$$

So, by the above steps we have obtained the original secret information, based only on three frames from the frame group.

We then construct the video by frames, and the size is typically big. We can use Shamir's secret sharing scheme to pretreat the original secret information to be hidden, and then hide the sub-secrets in different frames, which are then labelled as a frame group. When frame loss occurs, we can obtain the rest of frame group to reconstruct the original secret information.

2.5 Grey relational analysis

Grey relational analysis, a method for analyzing various factors in a system, is an important branch of the grey theory system[40].

The analysis procedure is described below.

2.5.1 Grey correlation coefficient

Set the reference sequence $x_0 = \{x_0(k) | k = 1, 2, \ldots, N\}$ and comparative sequence $x_i = \{x_i(k) | k = 1, 2, \ldots, N\}$, where N represents the number of sequences. Define the correlation formula as follows:

$$\xi_{i,0}(k) = \frac{\min\limits_i \min\limits_k |x_0(k) - x_i(k)| + \rho \max\limits_i \max\limits_k |x_0(k) - x_i(k)|}{|x_0(k) - x_i(k)| + \rho \max\limits_i \max\limits_k |x_0(k) - x_i(k)|} \tag{16}$$

where $\rho \in [0, 1]$ is the distinguishing factor, and $\xi_{i,0}(k)$ is used to reflect the similarity of the comparative sequence x_i and reference sequence x_0 at the same point.

$\xi_{i,0}(k)$ is in the range $\left[\frac{\rho}{1+\rho}, 1\right]$, when $\min_i \min_k |x_0(k) - x_i(k)| = 0$ and $|x_0(k) - x_i(k)| = \max_i \max_k |x_0(k) - x_i(k)|$, $\xi_{i,0}(k)$ obtains the minimum value, and when $|x_0(k) - x_i(k)| = \min_i \min_k |x_0(k) - x_i(k)|$, $\xi_{i,0}(k)$ obtains the maximum value.

2.5.2 Grey correlation degree

The grey correlation degree formula is as follows:

$$r(x_i, x_0) = r_{i,0} = \frac{1}{N} \sum_{i=1}^{N} \xi_{i,0}(k) \tag{17}$$

where $r(x_i, x_0)$ reflects the overall similarity of the comparative sequence x_i and the reference sequence x_0 at the same point.

From Eq. (17), we can also obtain the grey relational degree in the range $\left[\frac{\rho}{1+\rho}, 1\right]$.

A sketch of our proposed scheme is provided in Fig. 4.

3 Video Steganography Algorithm Based on Secret Sharing and ECC

3.1 Selecting hiding regions

Generally, steganography algorithms hide secret information in non-smooth regions. In this paper, we use grey relational analysis to determine whether a region is smooth or not.

When we obtain the partition modes of the current block, we can perform the following steps (using $X \times X$ ($X = \{4, 8, 16\}$) to represent the chosen block): (1) confirm the comparative and reference sequences and (2) select hiding regions.

3.1.1 Confirm the comparative and reference sequences

The ideal smooth region is the region in which there is no difference in pixel values, assuming that the pixel values of ideal smooth region are all identical.

(1) Comparative sequence: Scan each pixel value of the $X \times X$ block by zigzag.

(2) Reference sequence: Use ideal smooth areas as the reference sequence. In our algorithm, we take the average value of all the pixels in the block as the comparison sequence.

3.1.2 Select hiding regions

To select and then label the modulating regions, we use Eqs. (16) and (17) to calculate the grey correlation coefficients of the current block, with a threshold $T, T \in (0, 1)$.

Generally, the threshold T may vary with the number of secret information bits, the number of frames, and the size of the video sequence. In this paper, we define the threshold T as follows:

$$T = \frac{S}{B_4 + 4B_8 + 16B_{16}} \times 1.1 \tag{18}$$

where S is the number of information bits, and B_4, B_8, and B_{16} are the number of 4 × 4, 8 × 8, and 16 × 16 blocks, respectively. Some redundancy is also necessary, so we multiply by 1.1.

3.2 Embedding algorithm

The details of the embedding algorithm are as follows.

Step 1 Use Shamir's secret sharing scheme to pretreat the secret information, by dividing the secret information into n_1 pieces, for which only k_1 pieces are needed for reconstruction. The key is (k_1, n_1), and this step is carried out in the decimal system.

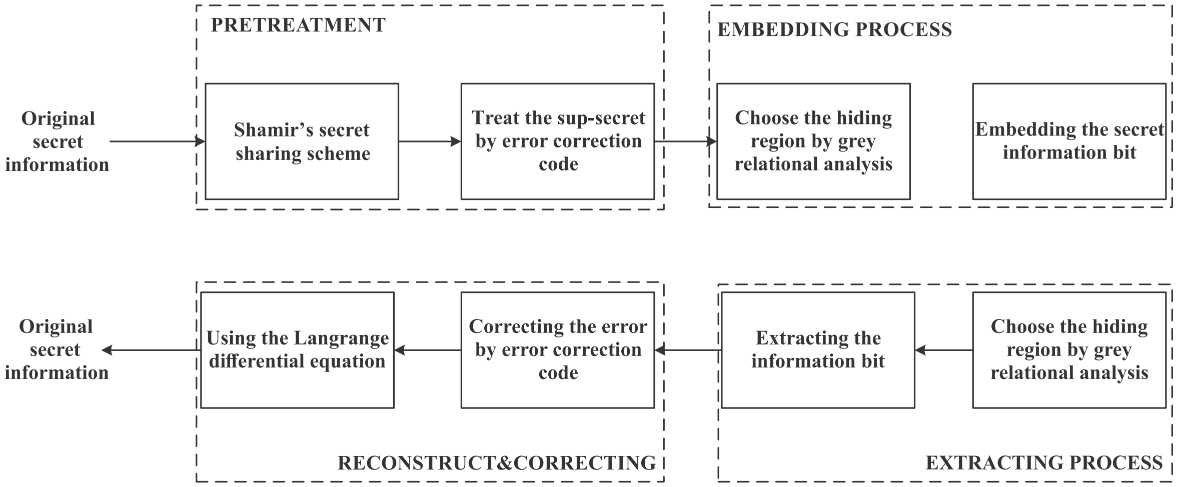

PRETREATMENT
EMBEDDING PROCESS
Original secret information
Shamir's secret sharing scheme
Treat the sup-secret by error correction code
Choose the hiding region by grey relational analysis
Embedding the secret information bit
Original secret information
Using the Langrange differential equation
Correcting the error by error correction code
Extracting the information bit
Choose the hiding region by grey relational analysis
RECONSTRUCT&CORRECTING
EXTRACTING PROCESS

Fig. 4 Sketch of our proposed scheme.

Step 2 Change the number of each frame in Step 1 from decimal to binary, and combine it with (n_2, k_2). Use the Hamming code to treat the sub-secrets, where the code's length is n_2 and the original information bit is k_2. This step and the following steps use the binary system.

Step 3 Perform a DCT on these frames, and from Section 3.1, we can obtain the region in which to hide the information. Then, scan these DCT coefficients by zigzag into a one-dimensional array where:

l, the length of the vector to be embedded;

T, quantization threshold;

round(•), rounding operator, using the computational method in Eq. (19).

$$\text{round}(x) = \begin{cases} 0, & 0 < x < 0.5; \\ 1, & 0.5 \leqslant x < 1 \end{cases} \tag{19}$$

By modifying the first eight coefficients into an 8-dimensional vector, $\boldsymbol{C} = (c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7)$,

(1) $l = |\boldsymbol{C}| = \sqrt{\sum_{j=0}^{7} c_j^2}$;

(2) $l' = \begin{cases} \left(\text{round}\left(\dfrac{l}{T}\right) + \alpha\right) \cdot T, & \text{if } M(i) = 1; \\ \left(\text{round}\left(\dfrac{l}{T}\right) - \alpha\right) \cdot T, & \text{if } M(i) = 0. \end{cases}$

$M(i)$ is the bit obtained in Step 2, l' is the modified vector length, and α ($\alpha \in (0, 0.4)$) is generated by the key k_3 to ensure the randomness of the embedding process;

(3) $\boldsymbol{C}' = \dfrac{l'}{l}\boldsymbol{C}$; return vector $\boldsymbol{C}'$ to the DCT blocks.

Step 4 Determine the embedding number based on the mode flag. The 4 × 4 block embeds one bit, the 8 × 8 block embeds four bits, and the 16 × 16 block embeds 16 bits.

Step 5 Repeat Steps 2–4 until all the secret information bits are embedded.

3.3 Extracting algorithm

Step 1 First, reconstruct the stego-carrier, judge the partition modes, label the 8 × 8, 4 × 4, and 16 × 16 blocks; calculate the grey relational degree again, make a judgment according to the standard of the embedded algorithm, and mark it again.

Step 2 Perform a DCT on the video frame, scan the DCT coefficients marked in Step 1 by zigzag, and enter them into a one-dimensional array.

Step 3 Convert the first eight coefficients into an 8-dimensional vector $\boldsymbol{S}$;

(1) $\boldsymbol{S} = (s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7)$,

(2) $L = \sqrt{\sum_{j=0}^{7} s_j^2}$,

(3) $I = \dfrac{L}{T} - \text{round}\left(\dfrac{L}{T}\right)$,

(4) $M(i) = \begin{cases} 1, & \text{if } I > 0; \\ 0, & \text{if } I < 0. \end{cases}$

where s_i is the DCT coefficients of the stego-video, L is the vector length of vector $\boldsymbol{S}$, and $M(i)$ is the extracting bit of the i-th block. A 4 × 4 block can extract only one bit, a 8 × 8 block can extract four bits, and 16 × 16 block can extract 16 bits.

Step 4 Repeat Step 3 until the secret information extraction is complete.

Step 5 Divide the secret information from Step 4 into groups, in which every group has n_2 bits. Compare them with the information in Table 2, and correct the error generated by the network transmission to obtain k_2 bits. This step and the previous step are performed in the binary system.

Step 6 Add all the bits in the same frame from Step 5, and change these bits from the binary system to the decimal system.

Step 7 Each n_1 frame is a secret frame group, and using Shamir's secret sharing method, we can use only k_1 of them to reconstruct the original secret information by the Lagrange differential equation.

In our algorithm, there are many keys, including (k_1, n_1), (n_2, k_2), k_3, and T. For secret sharing, we divide the information into n_1 pieces, and use k_1 pieces for reconstruction. We then use the ECC and the information bit number k_2 and add the bit number $n_2 - k_2$; in the grey relational analysis, we use the threshold T; and in the embedding process, we generate the pseudo random sequence α by the key k_3, and do not make these keys public.

4 Experimental Results

Our experimental platform is the X.264, and we used Microsoft Visual C++ and MATLAB 2012 software, with a computer configuration as follows: Core i5, 2.40 GHz, 3.0 GB RAM. We downloaded the video sequences from the website “media.xiph.org” , the format of the sequences include the QCIF (176 × 144) sequence Grandma, Carphone, Container, Miss-America, Soccer, and the CIF (352 × 288) sequence include Stefan, Foreman, News, Paris, and Mobile. In every sequence, there were 300 frames, encoding the I frame per 15 frames, the grey distinguishing coefficient ρ is 0.25, and the secret information to be hidden was a pseudo random 01 sequence generated by the key K.

In this study, we used the Shamir secret sharing $(3, 5)$ threshold scheme and the $(7, 4)$ Hamming code to construct an example, which can also be changed to enhance security.

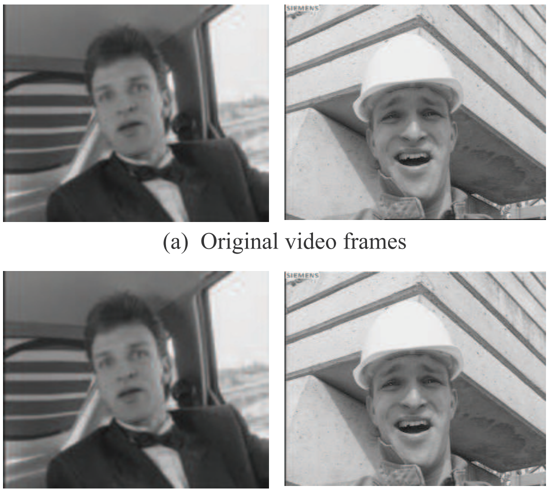
4.1 Subject analysis of invisibility

Figure 5 shows the comparison of the original frame and the frame after embedding. From Fig. 5, we can see that the video using our algorithm to embed has good invisibility. The video sequences after embedding satisfy the Human Visual System (HVS).

4.2 Objective analysis of invisibility

To use the Peak Signal-to-Noise Ratio (PSNR) as the measurement of the invisibility, the PSNR can be described as follows:

$$\text{PSNR} = 10 \times \lg \frac{255^2}{\text{MSE}} \tag{20}$$



(a) Original video frames

(b) Video frames after embedding

Fig. 5 Comparison of original and embedded frames.

where MSE is the mean square error between the original and stego video images, as determined using Eq. (21) as follows:

$$\text{MSE} = \frac{1}{KMN} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{n=1}^{N} [f_1(m, n, k) - f_2(m, n, k)]^2 \tag{21}$$

where f_1 is the original video image, f_2 is the stego-video image, K is the frame number, and $M \times N$ is the size of the video sequence.

In our experience, when the PSNR value is above 36 dB, the sequence is clear and fluent. The PSNR values of the test sequences are listed in Table 3, and the results show that the decrease in the PSNR is small after embedding, with an average decline in the PSNR value of about 0.912 dB. In Ref. [21], the average decline of the PSNR value is about 1.316 dB, thus our algorithm performs better with respect to invisibility.

4.3 Anti-steganalysis ability

Heidari and Gaemmaghami[41] proposed an SVD-based DCT domain steganalysis algorithm, based on the modification of DCT coefficients, which has a high detection rate for steganography. In image detection technology, the video frame itself can be seen as an image. In this study, we extracted each frame, and then tested its error detection rate.

A False Positive (FP) indicates an occurrence of a non-stego video being classified as a stego video. A False Negative (FN) indicates that a stego video has been classified as a non-stego video. In the detection, we mainly relied on FPs and FNs to judge the security of the steganography algorithm. The higher the number of FPs and FNs, the better is the steganographic algorithm.

Table 3 Comparison of original and embedded PSNR values.

Video	Frame	Original	Our algorithm		Ref. [21]	
sequence	size	PSNR (dB)	PSNR after embedding (dB)	Decrement (dB)	PSNR after embedding (dB)	Decrement (dB)
Grandma	176 × 144	36.87	36.10	0.77	35.39	1.48
Carphone	176 × 144	39.31	38.63	0.68	37.11	2.20
Container	176 × 144	39.15	38.42	0.73	38.34	0.81
Miss-America	176 × 144	41.57	40.55	1.02	40.59	0.98
Soccer	176 × 144	39.21	38.45	0.76	38.19	1.02
Stefan	352 × 288	37.86	36.88	0.98	36.47	1.39
Foreman	352 × 288	38.44	37.32	1.12	37.11	1.33
News	352 × 288	40.28	38.89	1.39	39.30	0.98
Paris	352 × 288	39.80	38.92	0.88	37.98	1.82
Mobile	352 × 288	40.69	39.90	0.79	39.54	1.15

The results are shown in Table 4.

From this table, we can see that the number of FPs and FNs after detection is high, and the error detection rate is also high. So, we can conclude that our algorithm has high security.

4.4 Analysis of the bit rate

To detect the Bit Rate Increase (BRI) of our algorithm after embedding, we used the following BRI formula:

$$B_{\mathrm{BRI}} = \frac{B'_{\mathrm{rate}} - B_{\mathrm{rate}}}{B_{\mathrm{rate}}} \times 100\% \qquad (22)$$

where B'_{rate} is the bit rate after embedding and B_{rate} is the bit rate of the original video sequence. B_{BRI} is the increase in the bit rate. The results are shown in Table 5.

From this table, we can see that the BRIs of our algorithm are mostly lower than those of Ref. [21] after embedding the secret information. Therefore, our scheme has a lower bit rate increase after embedding.

4.5 Analysis of robustness

In this section we analyze the robustness of our scheme.

When transmitting a video sequence through a real network, the sequence is inevitably impacted by attacks such as noise and filtering.

Table 4 Error detection rates of steganalysis algorithm.

Video sequence	Frame size	FP (%)	FN (%)	Error detection (%)
Grandma	176 × 144	58.27	54.33	56.30
Carphone	176 × 144	54.63	57.49	56.06
Container	176 × 144	57.11	51.22	54.17
Miss-America	176 × 144	60.53	54.98	57.76
Soccer	176 × 144	47.29	50.61	48.95
Stefan	352 × 288	60.26	48.32	54.29
Foreman	352 × 288	46.52	56.49	51.51
News	352 × 288	50.33	47.67	49.00
Paris	352 × 288	63.14	56.86	60.00
Mobile	352 × 288	58.82	46.48	52.65

Sometimes the video loses a number of frames when transmitting. We used salt-and-pepper noise and Gaussian filtering to simulate real network attacks, and randomly dropped a few frames of the test videos to simulate another attack.

In the experiment, we used salt-and-pepper noise[42] with an intensity of 0.05, and for the 3 × 3 Gaussian filter, we used the Bit Error Rate (BER) as the measurement. We detected the survival rate of the algorithm by simulating the frame loss of a real network.

The results are shown in Tables 6–8.

From these three tables, we can see that the BERs after attack by noise or filtering is very low, and the survival rate of our algorithm is higher than that of Ref. [21]. Reference [21] used only grey relational analysis to choose the blocks to be hidden, and when frame loss occurs in a transmit network, the hiding blocks are lost, and the secret information cannot easily be reconstructed. In our algorithm, based on the secret sharing scheme, not all the frames are needed to extract the secret, so we can tolerate some frame loss during network transmission.

Typically, the network frame loss rate can reach 10%[20]. As shown in Table 8, our algorithm can work within these conditions very well.

Based on the above results, our algorithm demonstrates better robustness.

5 Conclusion and Future Work

In this paper, we proposed a novel video steganography algorithm based on the secret sharing scheme and the ECC. Using secret sharing and the ECC to pretreat

Table 5 BRI values.

Video sequence	Frame size	Before embedding (Kb/s)	Our algorithm		Ref. [21]	
			After embedding (Kb/s)	B_{BRI} (%)	After embedding (Kb/s)	B_{BRI} (%)
Grandma	176 × 144	602.15	608.59	1.07	610.88	1.45
Carphone	176 × 144	674.23	679.69	0.81	682.86	1.28
Container	176 × 144	568.31	576.10	1.37	576.95	1.52
Miss-America	176 × 144	624.17	637.33	2.11	636.72	2.01
Soccer	176 × 144	596.28	601.82	0.93	604.99	1.46
Stefan	352 × 288	1823.89	1836.83	0.71	1842.68	1.03
Foreman	352 × 288	2664.00	2680.25	0.61	2690.37	0.99
News	352 × 288	1768.38	1783.23	0.84	1790.66	1.26
Paris	352 × 288	1695.47	1708.19	0.75	1720.39	1.47
Mobile	352 × 288	1633.84	1652.30	1.13	1660.63	1.64

Table 6 BERs of the secret information after salt-and-pepper attack.

Video sequence	Frame size	BER (%)
Grandma	176 × 144	0.006
Carphone	176 × 144	0.010
Container	176 × 144	0.013
Miss-America	176 × 144	0.017
Soccer	176 × 144	0.012
Stefan	352 × 288	0.019
Foreman	352 × 288	0.021
News	352 × 288	0.016
Paris	352 × 288	0.011
Mobile	352 × 288	0.020

Table 7 BERs of the secret information after Gaussian filter attack.

Video sequence	Frame size	BER (%)
Grandma	176 × 144	0.013
Carphone	176 × 144	0.006
Container	176 × 144	0.012
Miss-America	176 × 144	0.018
Soccer	176 × 144	0.016
Stefan	352 × 288	0.011
Foreman	352 × 288	0.012
News	352 × 288	0.015
Paris	352 × 288	0.013
Mobile	352 × 288	0.012

Table 8 Comparison of survival rates after frame loss.

Video sequence	Frame loss rate (%)	Survival rate of Ref. [21] algorithm (%)	Our algorithm (%)
Grandma	7	73.32	100.00
	15	38.39	89.21
Carphone	7	78.98	100.00
	15	40.21	88.32
Container	7	71.34	100.00
	15	36.53	90.11
Miss-America	7	75.65	100.00
	15	32.11	91.54
Soccer	7	80.33	100.00
	15	30.75	94.67
Stefan	7	69.21	100
	15	42.31	89.21
Foreman	7	72.76	100.00
	15	43.42	85.35
News	7	74.52	100.00
	15	29.81	83.78
Paris	7	78.93	100.00
	15	35.65	87.64
Mobile	7	73.21	100.00
	15	34.77	89.47

the information bits to be hidden, we then used grey relational analysis to choose the block that would hide the secret information, and then embedded the secret information bit by the given rules. Our experimental results indicate that our algorithm achieves high robustness, and has other advantages including good anti-steganalysis, a low bit rate increase, and good invisibility.

In this study, although we solved the problem of attacks when transmitting in the real network, to achieve the goal of covert communication, we have not fully developed the video characteristics. For example, videos have great capacity to hide information, such as with increased block sizes, and we have only used symmetric blocks. In future work, we will divide frames into pieces, and hide information using the secret sharing scheme. Additional work will involve ECC research to identify more effective error-correcting codes.

Acknowledgment

The author would like to thank the reviewers for their insightful comments and helpful suggestions. This research was supported by the National Natural Science Foundation of China (Nos. 61379152 and 61403417).

References

[1] T. Filler, J. Judas, and J. Fridrich, Minimizing additive distortion in steganography using syndrome-trellis codes, *IEEE Trans. on Information Forensics and Security*, vol. 6, no. 3, pp. 169–174, 2011.

[2] Y. Lee and L. Chen, An adaptive image steganographic model based on minimum-error LSB replacement, in *Proc.of the Ninth National Conference on Information Security*, Taichung, China, 1999, pp. 8–15.

[3] J. Mielikainen, LSB matching revisited, *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285–287, 2006.

[4] E. Kawaguchi and R. Eason, Principles and applications of BPCS steganography, in *Proceedings of SPIE—the International Society for Optics and Photonics East*, 1999, pp. 464–473.

[5] A. Westfeld, F5-a steganographic algorithm, *Lecture Notes in Computer Science*, vol. 2137, pp. 289–302, 2001.

[6] J. Fridrich, T. Pevny, and J. Kodovsky, Statistically undetectable jpeg steganography: Dead ends challenges, and opportunities, in *Proc. of the 9th Workshop on Multimedia & Security*, Dallas, TX, USA, 2007, p. 3–14.

[7] Y. Kim, Z. Duric, and D. Richards, Modified matrix encoding technique for minimal distortion steganography, *Lecture Notes in Computer Science*, vol. 4437, pp. 314–327, 2007.

[8] N. Provos, Defending against statistical steganalysis, in *SSYM'01 Proceedings of the 10th Conference on USENIX Security Symposium*, Washington DC, USA, 2001, pp. 323–336.

[9] T. Pevny, T. Filler, and P. Bas, Using high-dimensional image models to perform highly undetectable steganography, *Lecture Notes in Computer Science*, vol. 6387, pp. 161–177, 2010.

[10] V. Holub and J. Fridrich, Designing steganographic distortion using directional filters, in *IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, 2012, pp. 234–239.

[11] J. Fridrich and J. Kodovsky, Multivariate Gaussian model for designing additive distortion for steganography, in *2013 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, 2013, pp. 2949–2953.

[12] L. Guo, J. Ni, and Y. Shi, Uniform embedding for efficient JPEG steganography, *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 814–525, 2014.

[13] R. Mstafa and K. Elleithy, A highly secure video steganography using Hamming code (7, 4), in *IEEE Long Island Systems, Applications and Technology Conf.*, New York, NY, USA, 2014, pp. 1–6.

[14] M. Sadek, A. Khalifa, and M. Mostafa, Video steganography: A comprehensive review, *Multimedia Tools & Applications*, vol. 74, no. 17, pp. 1–32, 2014.

[15] K. Divya and K. Mahesh, Random image embedded in videos using LSB insertion algorithm, *International Journal of Engineering Trends & Technology*, vol. 13, no. 8, pp. 381–385, 2014.

[16] K. Churin, J. Preechasuk, and C. Chantrapornchai, Exploring video steganography for hiding images based on similar lifting wavelet coefficients, in *Advances in Information Technology*. Springer, 2013.

[17] I. Richardson, *H. 264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia*. John Wiley & Sons, 2004.

[18] J. Zhang, A. Ho, and G. Qiu, Robust video watermarking of H. 264/AVC, *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 2, pp. 205–209, 2007

[19] S. Singh and T. Siddiqui, A security enhanced robust steganography algorithm for data hiding, *International Journal of Computer Science Issues*, vol. 9, no. 1, pp. 131–139, 2012.

[20] Y. Liu, Z. Li, and X. Ma, A robust without intra-frame distortion drift data hiding algorithm based on H. 264/AVC, *Multimedia Tools and Applications*, vol. 72, no. 1, pp. 613–636, 2014.

[21] Y. Zhang, M. Zhang, and J. Wang, A novel information hiding algorithm based on grey relational analysis for H. 264/AVC, in *2015 International Conference on Intelligent Networking and Collaborative Systems*, Taipei, China, 2015 pp. 365–369.

[22] S. Guizani and N. Nasser, An audio/video crypto—Adaptive optical steganography technique, in *2012 8th International Wireless Communications and Mobile Computing Conference*, Limassol, Cyprus, 2012, pp. 1057–1062.

[23] C. Geetha and C. Puttamadappa, Enhanced stego-crypto techniques of data hiding through geometrical figures in an image, in *2015 2nd International Conference on Electronics and Communication Systems*, Cairo, Egypt, 2015, pp. 116–122.

[24] Y. Chai, An improved secret image sharing scheme with steganography, in *2011 International Conference on Mechatronic Science, Electric Engineering and Computer*, Jilin, China, 2011, pp. 1335–1338.

[25] L. Li, A. El-Latif, and X. Yan, A lossless secret image sharing scheme based on steganography, in *2012 2nd International Conference on Instrumentation, Measurement, Computer, Communication and Control*, Hangzhou, China, 2012, pp. 1247–1250.

[26] X. Li, K. Hu, and G. Zhang, An adaptive video watermarking based on secret image sharing, in *2012 5th International Symposium on Computational Intelligence and Design*, Hangzhou, China, 2012, pp. 359–362.

[27] B. Surekha and G. Swamy, A semi-blind image watermarking based on discrete wavelet transform and secret sharing, in *2012 International Conference on Communication, Information & Computing Technology*, Mumbai, India, 2012, pp. 1–5.

[28] S. Patil and P. Deshmukh, Verifiable image secret sharing in matrix projection using watermarking, in *2014 International Conference on Circuits, Systems, Communication and Information Technology Applications*, 2014, pp. 225–229.

[29] G. Sullivan and T. Wiegand, Rate-distortion optimization for video compression, *Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[30] A. Shamir, How to share a secret, *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[31] A. Bjorck and V. Pereyra, Solution of Vandermonde systems of equations, *Mathematics of Computation*, vol. 24, no. 112, pp. 893–903, 1970.

[32] A. Ben-Israel, A Cramer rule for least-norm solutions of consistent linear equations, *Linear Algebra and Its Applications*, vol. 43, pp. 223–226, 1982.

[33] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Wiley, 2005.

[34] T. Kasami, Weight distributions of Bose-Chaudhuri-Hocquenghem codes, Report no. R-317, Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, USA, Aug. 1966.

[35] A. Liveris, Z. Xiong, and C. Georghiades, Compression of binary sources with side information at the decoder using LDPC codes, *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, 2002.

[36] P. Gaborit, G. Murat, O. Ruatta, G. Zémor, Low rank parity check codes and their application to cryptography, in *the International Workshop on Coding and Cryptography*, Bergen, Norway, 2013, pp. 168–180.

[37] E. Berlekamp, Goppa codes, *IEEE Transactions on Information Theory*, vol. 19, no. 5, pp. 590–592, 1973.

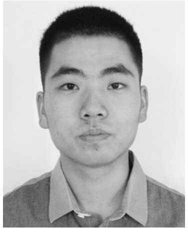
[38] R. Hamming, Error detecting and error correcting codes, *Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, 1950.

[39] G. Hu, H. Zhang, and L. Wang, A class of the hamming weight hierarchy of linear codes with dimension 5, *Tsinghua Science and Technology*, vol. 19, no. 5, pp. 442–451, 2014.

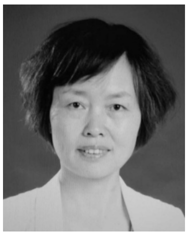
[40] J. Deng, *Basic Method of Grey System Theory*, (in Chinese). Wuhan, China: Huazhong University of Science and Technology Press, 2005.

[41] M. Heidari and S. Gaemmaghami, Universal image steganalysis using singular values of dct coefficients, in *2013 10th International ISC Conference on Information Security and Cryptology*, Yazd, Iran, 2013, pp. 1–5.

[42] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, (in Chinese), Beijing, China: Publishing House of Electronics Industry, 2000.



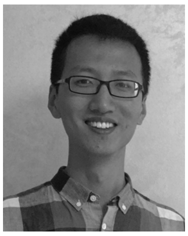
Yingnan Zhang received the MS degree from Engineering University of PAP in 2014. He is currently a PhD candidate at Department of Electronic Technology in Engineering University of PAP. His research interests are information hiding and video processing.



Minqing Zhang is currently a professor, PhD advisor at Engineering University of PAP, China. She received the PhD degree from North Western Polytechnical University in 2016. Prof. Zhang is currently engaged in the research on information security and image processing.

Xiaoyuan Yang received the MS degree from Xidian University in 1991. He is currently a professor, PhD advisor at Engineering University of PAP. His research interests include cryptography and information security.

Duntao Guo received the MS degree from Northwestern Polytechnical University in 2014. He is currently a lecturer at Engineering University of PAP, China. His current interest is video processing.



Longfei Liu received the MS degree in 2013 from Engineering University of the PAP. He is currently a teaching assistant at Engineering University of PAP. His research interests include network security and stream cipher.