Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 16(5): 4559–4580
DOI: 10.3934/mbe.2019228
Received: 25 January 2019
Accepted: 09 May 2019
Published: 23 May 2019

Research article

Data-Loss resilience video steganography using frame reference and data ensemble reconstruction

Fengyong Li[1,*], Jiang Yu[2] and Yanli Ren[3]

[1] College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, P.R.China
[2] School of Information and Computer, Shanghai Business School, Shanghai, P.R.China
[3] School of Communication and Information Engineering, Shanghai University, Shanghai, P.R.China

* **Correspondence:** Email: fyli@shiep.edu.cn.

Abstract: In this paper, we propose a robust video steganographic method, which can efficiently hide confidential messages in video sequences, and ensure that these messages are perfectly reconstructed by recipient. To apply proposed scheme to video sequences, we must be faced with two nontrivial problems: (a) how to effectively minimize the total steganographic distortion for each video frame? (b) how to recover the hidden messages if some frames are lost or damaged? We tackle the first question by designing a new distortion function, which employs two continuous adjacent frames with the same scene as side-information. The second question is addressed by data sharing. In this mechanism, the original data is expanded and split into multiple shares by using multi-ary Vandermonde matrix. Since these shares contain a lot of data redundancy, the recipient can recover the hidden data even if some frames are damaged or lost during delivery. Extensive experiments show that proposed scheme outperforms the state-of-the-arts in terms of robustness and diverse attacks.

Keywords: Information hiding; video steganography; side-information; data reconstruction; robustness

1. Introduction

Steganography is an efficient privacy communication measurement in which secret messages are embedded into digital media, such as digital images, video or audio files, to implement the information delivery [1–4]. As a countermeasure to steganography, steganalysis [5–8] is mainly used to detect the presence of hidden data in a digital media. Traditional steganography mainly involves *single* digital image, and always combines the side-information based distortion function and Syndrome-Trellis Codes (STC) [4] to implement steganography. For example, data hider uses the non-round coefficients

from one uncompressed image to measure the steganography costs of each DCT coefficient and then embed messages when saving them as compressed images. This strategy is consistently feasible in the context of modern steganography, because the side-information is not available to the recipient (or steganalyst). However, when the size of secret messages is too big, the steganography for single image does not work. One feasible way is to hide the messages among a batch of images. We usually name this scheme as batch steganography [9–11]. Nevertheless, batch steganography is not easy, at least inconvenient, to apply in real world due to the following two reasons. First, traditional distortion function mainly focuses on single image, the distortion definition for batch steganography is not straightforward. Second, batch steganography needs a huge of homogeneous covers, which are hard to be obtained due to diverse social networks.

One natural question arises: Is there a kind of media that can avoid the above two problems to facilitate data hiding? The answer is positive. Video sequences provide this possibility because they usually consist of a number of homogeneous image frames and thus have a higher capacity. Nevertheless, some ones may wonder if video steganography is as successful as traditional side-information based image steganography, because most of videos from different acquisition devices, e.g. cell phones or digital cameras, are always saved as JPEG format, not the uncompressed format. Accordingly, if we use the video to hide the secret messages, it has to answer two key questions: (1) How to design optimal distortion function by using continuous video frames with same (or approximate) scenes? (2) How to design embedding strategy to ensure that stego videos can resist diverse network attacks, such as usual noise attacks, video frame attacks and video compression attacks?

Aiming at the first question, since video always contains compressed image frames, the existing distortion function based on side-information cannot be transplanted directly. Nevertheless, we can get some inspiration from different definition of side-information, for example, designing the steganography cost by using multiple image with the same scene. Actually, several works has been developed in this direction. In [12], the authors proposed a new view to model the differences between the printed image and its scan version. Unfortunately, this scheme is inconvenient due to two pitfalls: (1) this scheme is rather labor-consuming due to requiring a large number of scan versions. (2) the difference among scan images maybe lead to the complication increasing. To remove this weakness, in [13], the authors designed a different type of side-information by multiple compressed images with the same scene. This scheme avoided time-consuming and formed a more secure method even if only two images are used. Although a quite significant increasing for anti-detection can be obtained easily with respect to the case of single image, this scheme is rather difficult to be practical because the image database is hard to build.

Regarding the second question, existing video steganographic methods [14–19] can be divided into two categories according to the information embedding domain. One is spatial domain based video steganography, in which the data is embedded directly into raw pixel values, and they usually refer to the processing of image steganography, such as Least Significant Bit (LSB) Matching method [1], Spread Spectrum (SS) method [20], and BCH code [15] et al.. Although spatial methods can embed high capacity messages, it is inevitable to loss the hidden messages once the stego videos are damaged by unexpected network interference, such as noise, compression, or frame losing. Another type is the joint-compression domain video steganography. In this category, most of methods embed data into different types of compressed video, e.g. motion vectors (MVs) methods [17, 18], inter/intra prediction methods [19], quantized DCT coefficients methods [14, 16] et al.. These compression domain

based methods have a similarity, that is, lower embedding capacity. Moreover, although some compression domain based methods can effectively resist double compression attack, the secret messages are very hard to be recovered once the video frames are lost or damaged during delivery. Therefore, the robustness for video steganography needs to be further improved. This paper tries to fill this gap.

Facing the aforementioned problems, we make the following novel contributions in video steganography:

- We propose a robust video steganography scheme by a new distortion function and ensemble reconstruction mechanism. The proposed solution can not only improve the security performance of stego video, but also ensure the completeness of original data even if some video frames are damaged in delivery.
- Proposed scheme investigates another form of side-information by referring the adjacent image frames with the same scene, and then employs the side-information to design distortion function. This distortion function is effective because the message senders do not need to access the uncompressed image frames.
- Vandermonde matrix is used to expand and divide original data to multiple shares, which are embedded in the continuous frames by combining the designed distortion function and STC algorithm. Subsequently, ensemble reconstruction mechanism is designed to ensure the completeness and correctness of original data, even if partial data is damaged during delivery.
- Comprehensive experiments are performed with classical video sequences. The experimental results demonstrate that proposed scheme can significantly improve the overall performance on visual quality, robustness and anti-steganalysis, leading to a superiority for existing video steganographic methods.

The rest of this paper is organized as follows. Section 2 provides the details of proposed scheme by introducing the procedure of distortion function and data ensemble reconstruction. Subsequently, comprehensive experiments are performed to evaluate the performance of proposed scheme. The experimental results and corresponding discussions are presented in Section 3. Finally, Section 4 concludes the paper.

2. Data-loss resilience video steganography

2.1. The framework of proposed scheme

The framework of our proposed robust video steganography scheme is shown in Figure 1. The proposed scheme is mainly comprised of two parts: data embedding and data extraction. In the data embedding stage, we firstly use a multi-ary Vandermonde matrix to expand original data and then divide them into multiple small shares, which are considered as "actual embedding data". Secondly, we consider the continuous adjacent frames with same scene as pre-cover to provide the side-information, and then design an efficient distortion function by referring to the adjacent frames. Finally, the data shares are embedded into each frame by an existing cost-based embedding scheme. In the data extraction stage, we extract the undamaged shares from the received video frames, and then recover the original data by an ensemble reconstruction mechanism, although the video frames might be damaged or intercepted during the delivery. We claim that the original data can be recovered perfectly as long as the recipient can obtain enough undamaged shares.

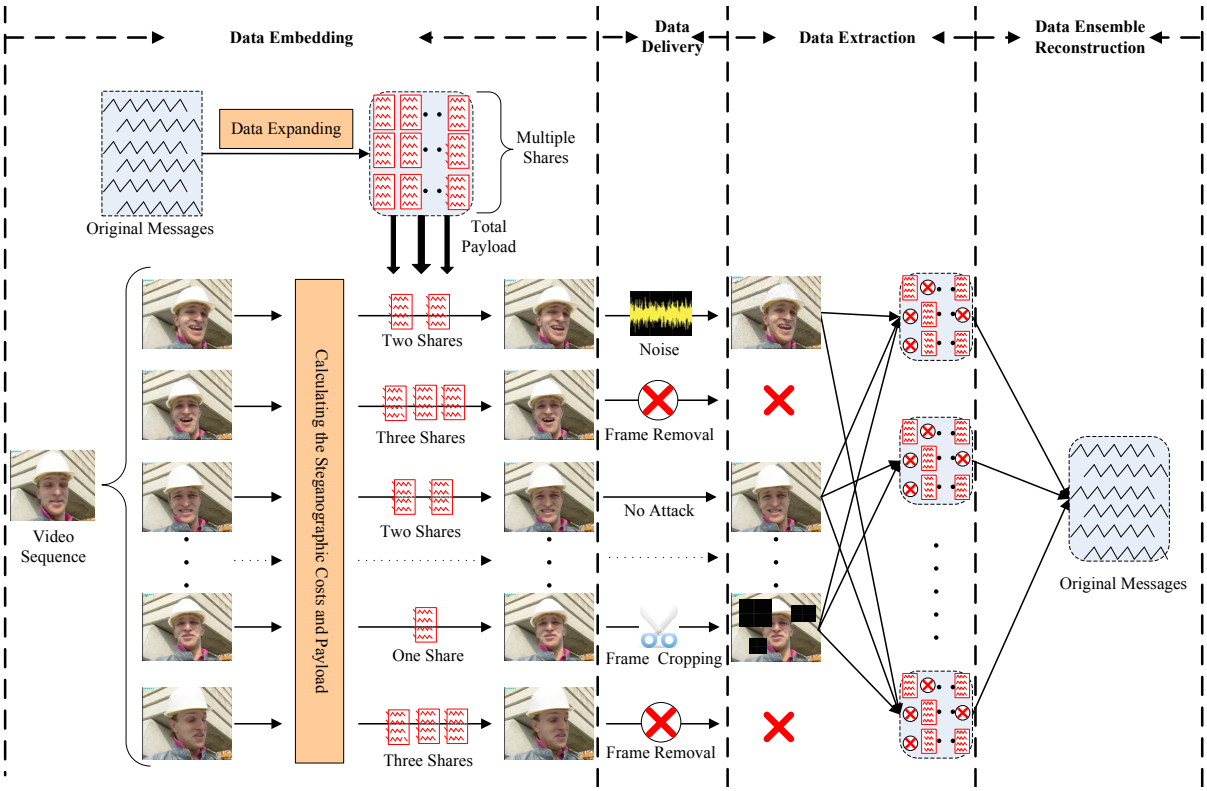


Figure 1. Framework of proposed robust batch steganography scheme.

2.2. Data decomposition and reconstruction

A video sequence usually contains a lot of image frames with (approximately) same scene, if message sender hopes to deliver secret messages by video sequence, he can spread the messages into continuous image frames. At the receiving end, the receiver extracts the secret messages from these image frames according to a fixed order. Unfortunately, video sequences may be attacked/damaged during transmission, such as the network noise or the warden who might try to remove the video frames. In this case, it is unreasonable to assume that the recipient can receive the information completely and accurately. To improve the robustness of video steganography, in this section, we try to use matrix decomposition mechanism [11,21] to divide the original messages into multiple shares. Since each share only carry a small portion of valid information, partial loss for these shares do not affect the recovery for original messages. The corresponding details can be explained by Figure 2.

Assume that the given secret messages are a binary stream. To expand the original data, we first present the original data into q-ary symbol system, where q is an odd prime. Actually, this procedure is rather simple. The messages are segmented into multiple pieces. Each piece contains L_1 bits, which can be converted to L_2 q-ary digits according to the following equation.

$$L_1 = \lfloor L_2 \cdot \log_2 q \rfloor . \tag{2.1}$$

We can provide a simple sample to explain it graphically. Assume that $L_1 = 4$, $L_2 = 2$ and the

original data is converted into 5-ary notational system. Three binary pieces, (1101 0110 1001), can be converted to six 5-ary digits (23 11 14). Notably, the size reduction from L_1 to L_2 can be calculated by the following equation.

$$r = 1 - \frac{L_1}{L_2 \cdot \log_2 q} \tag{2.2}$$

Clearly, when L_1 and L_2 are very large, r is close to 0.

When the original data (binary stream) is converted completely, we integrate all q-ary digits as a sequence and then expand them into multiple shares by the following steps.

Step 1 : Segment the q-ary digit sequence into K small blocks. Denote each of them as $\{d_{k,1}, d_{k,2}, \cdots, d_{k,m}\}$, where m represents the length of a digital block and $k \in [1, K]$.

Step 2 : Build Vandermonde matrix **A**

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ a_1 & a_2 & \cdots & a_n \\ a_1^2 & a_2^2 & \cdots & a_n^2 \\ \vdots & \vdots & \cdots & \vdots \\ a_1^{m-1} & a_2^{m-1} & \cdots & a_n^{m-1} \end{bmatrix} \bmod q \tag{2.3}$$

where $a_1, a_2, \cdots, a_n \in [0, q-1]$ are named as the indices of **A** and they are different with each other. In addition, m, n, and q must satisfy $m \leq n \leq q$.

Step 3 : With the following equation, each digital block $\{d_{k,1}, d_{k,2}, \cdots, d_{k,m}\}$ can be expanded to n shares.

$$\begin{bmatrix} t_{k,1} & t_{k,2} & t_{k,3} & \cdots & t_{k,n} \end{bmatrix} = \begin{bmatrix} d_{k,1} & d_{k,2} & d_{k,3} & \cdots & d_{k,m} \end{bmatrix} \cdot \mathbf{A}, \tag{2.4}$$

where the symbol "$\cdot$" in Equation (2.4) presents the multiplication operator in q-ary notational system.

In order to understand data decomposition mechanism easily, we provide an actual example. Assume that $q = 7$, $n = 6$, $m = 3$, and the original data are three 7-ary digits [2 1 4]. We set the indices $a_1, a_2, \cdots, a_n$ of Vandermonde matrix as [5 3 1 0 2 4]. So, the Vandermonde matrix can be built easily by Equation (2.3) and the original digit vector [2 1 4] can be expanded as [2 6 0 2 6 0] according to Equation (2.4).

According to the above steps, m q-ary digits from the original data can be expanded easily to n q-ary digits. Obviously, there is some redundancy in these n q-ary digits. We denote the redundancy rate as R_e, which can be calculated easily as follows.

$$R_e = 1 - \frac{m}{n}. \tag{2.5}$$

Obviously, as long as the loss (or damaged) rate for $t_{k,1}, t_{k,2}, t_{k,3}, \cdots, t_{k,n}$ is not more than R_e, the original digits $d_{k,1}, d_{k,2}, d_{k,3}, \cdots, d_{k,m}$ can be just reconstructed by Equation (2.6).

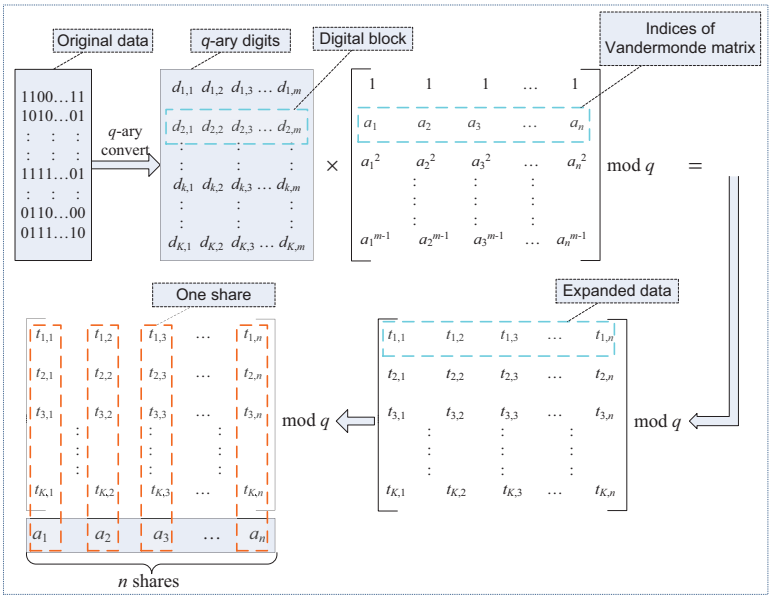


Figure 2. The procedure of original data decomposition using Vandermonde matrix.

$$\left[\begin{array}{ccccc} d_{k,1} & d_{k,2} & d_{k,3} & \cdots & d_{k,m} \end{array}\right] = \left[\begin{array}{ccccc} t'_{k,1} & t'_{k,2} & t'_{k,3} & \cdots & t'_{k,m} \end{array}\right] \cdot (\mathbf{A}')^{-1} \tag{2.6}$$

where $t'_{k,1}, t'_{k,2}, t'_{k,3}, \cdots, t'_{k,m}$ are m undamaged digits, which are selected from the received digits (they maybe contain the wrong digits). $\mathbf{A}'$ is $m \times m$ Vandermonde matrix built by the indices $a'_1, a'_2, \cdots, a'_m$ (referring to Equation (2.3)). $\mathbf{A}'^{-1}$ is the inversion matrix of $\mathbf{A}'$ in q-ary notational system, whose calculation details can be found in [11].

2.3. Distortion function based on frame reference

In this section, we describe the design details of a new distortion function when the sender possesses more than two continuous frames with the (approximately) same scene. Since the continuous video frames have strong correlation, when multiple continuous frames are used to carry the given messages, we can consider the adjacent frames as pre-covers, and calculate steganographic cost (distortion) of current cover to provide a better guidance for video steganography.

In the following, we describe the detailed designing procedure of distortion function when the continuous three cover frames are available. These three frames are considered as JPEG version and denoted $\mathbf{F^{(1)}}$, $\mathbf{F^{(2)}}$ and $\mathbf{F^{(3)}}$. We denote the quantized DCT coefficients in three frames as $x_{ij}^{(1)}$, $x_{ij}^{(2)}$ and $x_{ij}^{(3)}$, respectively. We then pronounce $x_{ij}^{(2)}$ as cover frame, $x_{ij}^{(1)}$ and $x_{ij}^{(3)}$ as side-information.

When $x_{ij}^{(1)}$ and $x_{ij}^{(3)}$ are considered as side-information, the message sender can calculate the steganographic cost of modifying coefficient $x_{ij}^{(2)}$ by -1 and +1. The corresponding costs are denoted as $\rho_{ij}^{(2)}(-1)$ and $\rho_{ij}^{(2)}(+1)$. In order to ensure that the proposed distortion function can reflect the cost of changing the cover coefficient $x_{ij}^{(2)}$ more accurately, we select the classical embedding schemes, such as J-UNIWARD [3], as the basis of calculating steganographic cost. Since the side-information is used

to improve further the accuracy of distortion function, we keep the original costs (the costs calculating by classical steganographic schemes) when $x_{ij}^{(1)} = x_{ij}^{(2)}$ and $x_{ij}^{(2)} = x_{ij}^{(3)}$, otherwise, re-modulate them. In other words, the values $x_{ij}^{(1)}$ and $x_{ij}^{(3)}$ are only useful when $x_{ij}^{(1)} \neq x_{ij}^{(2)}$ OR $x_{ij}^{(2)} \neq x_{ij}^{(3)}$. Since proposed distortion function refers to two adjacent frames, the new cost $\rho_{ij}^{'}(\pm 1)$ can be explained by the following four-cases procedures:

Case1 : When $x_{ij}^{(1)} = x_{ij}^{(2)}$ and $x_{ij}^{(2)} = x_{ij}^{(3)}$,

$$\rho_{ij}^{'}(\pm 1) = \rho_{ij}^{(2)}(\pm 1). \tag{2.7}$$

Case2 : When $x_{ij}^{(1)} \neq x_{ij}^{(2)}$ and $x_{ij}^{(2)} = x_{ij}^{(3)}$,

$$\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\alpha(Q)\rho_{ij}^{(2)}\left(s_{ij}\right) + \rho_{ij}^{(2)}\left(s_{ij}\right)}{2}, \tag{2.8}$$

where $s_{ij} = \text{sign}\left(x_{ij}^{(1)} - x_{ij}^{(2)}\right)$.

Case3 : When $x_{ij}^{(1)} = x_{ij}^{(2)}$ and $x_{ij}^{(2)} \neq x_{ij}^{(3)}$,

$$\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\rho_{ij}^{(2)}\left(s_{ij}\right) + \beta(Q)\rho_{ij}^{(2)}\left(s_{ij}\right)}{2}, \tag{2.9}$$

where $s_{ij} = \text{sign}\left(x_{ij}^{(3)} - x_{ij}^{(2)}\right)$.

Case4 : When $x_{ij}^{(1)} \neq x_{ij}^{(2)}$ and $x_{ij}^{(2)} \neq x_{ij}^{(3)}$

$$\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\alpha(Q)\rho_{ij}^{(2)}\left(s_{ij}\right) + \beta(Q)\rho_{ij}^{(2)}\left(s_{ij}\right)}{2}, \tag{2.10}$$

where $s_{ij} = \text{sign}\left(x_{ij}^{(1)} - x_{ij}^{(2)} + x_{ij}^{(3)} - x_{ij}^{(2)}\right)$.

Clearly, $\alpha(Q)$ and $\beta(Q)$ are two modulation factors referring to the compression factor Q, where $\alpha(Q), \beta(Q) \in [0, 1]$ and $Q \in [1, 100]$. They are utilized to control the actual cost values calculated from the side-information and will be discussed later.

Notably, in this section, we only design an new distortion function, which is considered to provide a better steganographic cost measurement. An actual steganographic method can be formed by combining the new distortion function and STC algorithm [4]. The corresponding details can be found in following section.

2.4. Data embedding and ensemble reconstruction

2.4.1. Data embedding

Following data decomposition mechanism and distortion measurement, we design a robust video steganographic scheme.

Algorithm 1: Data Embedding in Video Sequence

Input: Video frames $v_1, v_2, \ldots, v_S$, $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_n\}$, modulation parameters $\alpha(Q)$ and $\beta(Q)$, multi-ary parameter q.

Output: Stego video sequence V'.

for $t \leftarrow 1$ **to** $S-2$ **do**
  Compute steganographic costs $\rho_{ij}^{(v_{t+1})}(-1)$ and $\rho_{ij}^{(v_{t+1})}(+1)$ from the cover frame $x_{ij}^{(v_{t+1})}$;
  for $i \leftarrow 1$ **to** M **do**
    for $j \leftarrow 1$ **to** N **do**
      if $x_{ij}^{(v_t)} = x_{ij}^{(v_{t+1})}$ *and* $x_{ij}^{(v_{t+1})} = x_{ij}^{(v_{t+2})}$ **then**
        $\rho_{ij}^{'}(\pm 1) = \rho_{ij}^{(v_{t+1})}(\pm 1)$;
      end
      if $x_{ij}^{(v_t)} \neq x_{ij}^{(v_{t+1})}$ *and* $x_{ij}^{(v_{t+1})} = x_{ij}^{(v_{t+2})}$ **then**
        $s_{ij} = \operatorname{sign}\left(x_{ij}^{(v_t)} - x_{ij}^{(v_{t+1})}\right)$;
        $\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\alpha(Q)\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)+\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)}{2}$;
      end
      if $x_{ij}^{(v_t)} = x_{ij}^{(v_{t+1})}$ *and* $x_{ij}^{(v_{t+1})} \neq x_{ij}^{(v_{t+2})}$ **then**
        $s_{ij} = \operatorname{sign}\left(x_{ij}^{(v_{t+2})} - x_{ij}^{(v_{t+1})}\right)$;
        $\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)+\beta(Q)\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)}{2}$;
      end
      if $x_{ij}^{(v_t)} \neq x_{ij}^{(v_{t+1})}$ *and* $x_{ij}^{(v_{t+1})} \neq x_{ij}^{(v_{t+2})}$ **then**
        $s_{ij} = \operatorname{sign}\left(x_{ij}^{(v_t)} - x_{ij}^{(v_{t+1})} + x_{ij}^{(v_{t+2})} - x_{ij}^{(v_{t+1})}\right)$;
        $\rho_{ij}^{'}\left(s_{ij}\right) = \frac{\alpha(Q)\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)+\beta(Q)\rho_{ij}^{(v_{t+1})}\left(s_{ij}\right)}{2}$;
      end
    end
  end
  Computing the total shares for the current frame v_{t+1} and then embed the corresponding messages using costs $\rho_{ij}^{'}$ and STC algorithm to obtain stego frame v'_{t+1};
end
Integrate all stego frames $v'_1, v'_2, \ldots, v'_S$ to build stego video sequence V';

Denote the given secret messages M_o and video sequence as V. Proposed scheme tries to spread M_o into video frames* and ensure that the recipient can get the complete messages even if the video sequence is damaged during delivery. Therefore, proposed scheme is believed to be able to resist the diverse video attacks, such as noise, frame cropping or removal. The specific embedding procedure can be implemented as follows.

Step1 : Decompose the video sequence V to a batch of frames $v_1, v_2, \ldots, v_S$, which are ensured to

*We do not consider that how the sender informs the recipient of the length of each share, or how many shares correspond to a frame or a video sequence, because it could be solved by hiding the information in the frame or video header or by other secret channel.

have the (approximate) same scenes[†]. We denote the quality factor of video frames as Q, and then get two parameters $\alpha(Q)$ and $\beta(Q)$ by referring to the modulation parameters table, which will be discussed in the next section.

Step2 : Convert the original data M_o to binary stream, which is subsequently converted to a $K \times m$ q-ary digital matrix $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_m\}$.

Step3 : Build Vandermonde matrix $\mathbf{A}$ by $a_1, a_2, \cdots, a_n$ and calculate the expanded data by the following equation.

$$\mathbf{T} = \mathbf{D} \times \mathbf{A} \mod q \tag{2.11}$$

where $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_n\}$ is a $K \times n$ matrix, $\mathbf{t}_i = (t_{1,i}, t_{2,i}, \cdots, t_{K,i})^{\mathrm{T}}$ is a data vector, $i \in [1, n]$. Since the indices of Vandermonde matrix must be delivered with the expanded data, we denote $(t_{1,i}, t_{2,i}, \cdots, t_{K,i}, a_i)^{\mathrm{T}}$ as a complete share.

Step4 : With the expanded data $\mathbf{T}$ and the cover frames $v_1, v_2, \ldots, v_S$. We can embed these shares into each frame by combining costs ρ'_{ij} and STC algorithm. The detailed procedure can be found in Algorithm 1.

2.4.2. Data ensemble reconstruction

Once stego video is delivered through insecure network channel, it might face to diverse network attacks. According to proposed data reconstruction procedure, the recipient can reconstruct the original information by ensemble decision, even if partial stego frames are removed or intercepted during delivery. Assume that the remaining stego frames can extract m' complete shares (maybe contain some modified digits), $n \geq m' \geq m$. We select m shares from the remaining stego frames, and then extract their data vectors $\mathbf{t'}_1$, $\mathbf{t'}_2$, $\mathbf{t'}_3$, $\cdots$, $\mathbf{t'}_m$ and the corresponding indices a'_1, a'_2, $\cdots$, a'_m. The original information can be recovered correctly by an ensemble reconstruction mechanism whose detailed procedure is shown in Algorithm 2.

We also provide an actual example to explain our ensemble mechanism. According to the example in Section 2.2 and results of data decomposition, the complete expanded data should be [2 6 0 2 6 0]. We assume two digits are modified in delivery, that is, the last digit is lost and the second digit is changed (assuming '6' to '4'). Thus, the digits that are received by recipient are [2 **4** 0 2 6]. Since the original data have three digits, we can randomly select three digits from [2 **4** 0 2 6] and repeat four times (corresponding to the parameter $en = 4$ in Algorithm 2). Assume that these four selections are [0 2 6], [**4** 2 6], [2 0 6] and [2 **4** 2], respectively, we then calculate their Vandermonde inverse matrix according the method in [11] and obtain four "suspicious" original data, [2 1 4], [2 0 1], [2 1 4] and [2 4 2]. Finally, the majority voting (corresponding to the MaxVoting function in Algorithm 2) is used to give the final decision [2 1 4]. An actual example can be found in Figure 3. Although the recipient does not know which digits are modified in video transmission, the ensemble mechanism probably makes a correct decision by majority voting.

[†]In fact, the scene cuts are very common in video and always produce the frames with diversity contents. However, we do not discuss this special case lonely, because when new scene is cut, video frames can be re-extracted from the new scene to ensure they have the (approximate) same contents.

Algorithm 2: Data Ensemble Reconstruction From Attacked Video

Input: m' data vectors $\mathbf{t'}_1, \mathbf{t'}_2, \mathbf{t'}_3, \cdots, \mathbf{t'}_{m'}$, the indices $a'_1, a'_2, \cdots, a'_{m'}$ for m' data vectors, multi-ary parameter q, ensemble rounds en.

Output: $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_m\}$.

for $i \leftarrow 1$ **to** en **do**
  Randomly select m data vectors as $\mathbf{T'} = \{\mathbf{t'}_1, \mathbf{t'}_2, \mathbf{t'}_3, \cdots, \mathbf{t'}_m\}$, and then denote their corresponding indices as $\{a'_1, a'_2, \cdots, a'_m\}$.
  Construct Vandermonde matrix $\mathbf{A'}$ by the indice vector $\{a'_1, a'_2, \cdots, a'_m\}$.
  Calculate the inverse matrix $\mathbf{A'}^{-1}$ by $\mathbf{A'}$.
  $\mathbf{D}_i = \mathbf{T'} \times \mathbf{A'}^{-1} \bmod q$.
end
$\mathbf{D} = \text{MaxVoting}(\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \cdots, \mathbf{D}_{en})$.

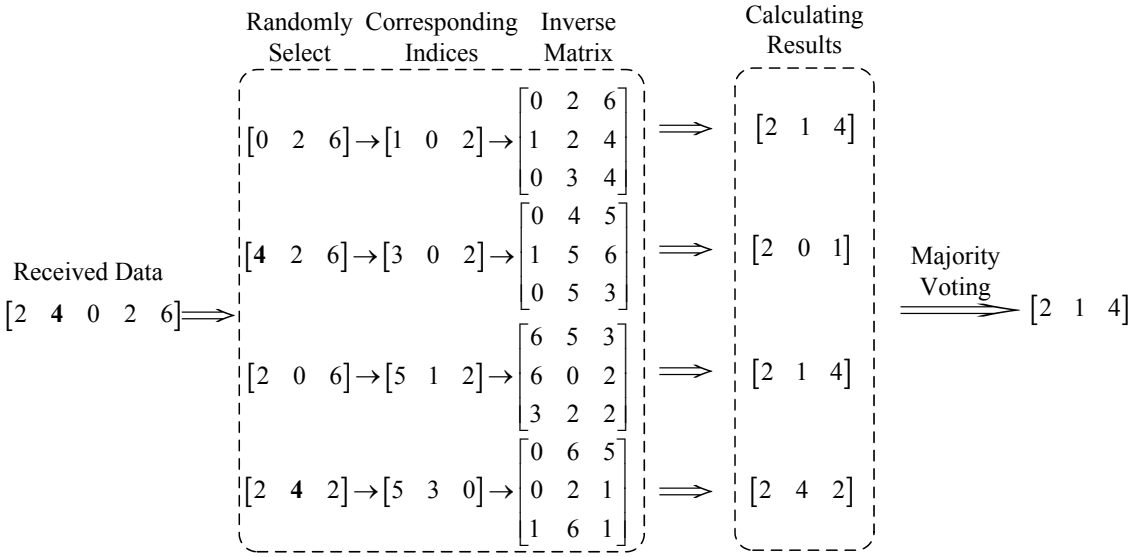


Figure 3. An actual example for ensemble reconstruction mechanism.

2.5. Several important notes

We would like to raise the attention to readers that if too many shares are lost, for example, when $m' < m$, proposed scheme is not able to recover the original data. Actually, the main principle has been explained by Equation (2.5).

In this algorithm, we use ensemble voting strategy (the function *MaxVoting* in Algorithm 2) to decide the correct original data. Although the received shares are complete, they maybe contain some modified digits, e.g. digit 3 may be changed to 7 due to the noise interference. Thus, the calculated original data matrices $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \cdots, \mathbf{D}_{en}$ in Algorithm 2 might be different. We can give the correct decision by counting the maximum same occurrences for $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \cdots, \mathbf{D}_{en}$. Also, we do not set $m' = m$ in data reconstruction, because if $m' = m$, $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \cdots, \mathbf{D}_{en}$ might be different each other, this result is invalid in the ensemble strategy.

Figure 4. The classical video sequences in our experiments.

Table 1. Detailed descriptions of video sequences.

Sequence	Resolution	Number of frames
Bus	352×288	150
City	352×288	300
Coastguard	352×288	300
Crew	352×288	300
Flower	352×288	250
Football	352×288	260
Foreman	352×288	300
Harbour	352×288	300
Highway	352×288	2000
Ice	352×288	240
Mobile	352×288	300
Paris	352×288	1065
Soccer	352×288	300
Tempete	352×288	260
Waterfall	352×288	260

3. Experimental results and analysis

3.1. Experimental setup

We carry out our experiments on a classical video database [22], e.g. Figure 4, which contains 15 test sequences with 4 : 2 : 0 YUV format. These video sequences have the same resolution of 352×288, and belong to diverse categories, including people, architecture, landscape, flowers, and so on. The detailed descriptions are given in Table 1.

In addition, to verify our proposed method, each video is separated to a number of image frames.

Table 2. Optimal modulation parameter combinations $(\alpha(Q), \beta(Q))$ for different relative payloads (bpf).

Payload r	Quality factor Q					
	70	75	80	85	90	95
0.1K	(0.087, 0.086)	(0.091, 0.092)	(0.192, 0.201)	(0.284, 0.291)	(0.479, 0.464)	(0.654, 0.662)
0.2K	(0.095, 0.093)	(0.114, 0.107)	(0.190, 0.181)	(0.286, 0.274)	(0.457, 0.461)	(0.623, 0.619)
0.3K	(0.094, 0.099)	(0.104, 0.117)	(0.167, 0.163)	(0.243, 0.252)	(0.437, 0.431)	(0.581, 0.597)
0.4K	(0.085, 0.083)	(0.090, 0.103)	(0.150, 0.144)	(0.248, 0.224)	(0.407, 0.389)	(0.563, 0.542)
0.5K	(0.076, 0.079)	(0.084, 0.091)	(0.137, 0.129)	(0.226, 0.224)	(0.375, 0.370)	(0.503, 0.489)

Then, all frames are compressed with same quality factor to avoid the influence of different quantization matrices for steganalysis. Since only the luminance contains a lot of non-zero coefficients for video frames, we just hide the messages into Y component. On the other hand, for a given steganographic algorithm, all frames are embedded with random messages embedding and then create new stego videos. Moreover, in order to test the performance of proposed scheme, we select some experimental video to train the corresponding parameters, e.g. $\alpha(Q)$ and $\beta(Q)$. The ensemble classifier is employed to show the comparable results.

As the main concern of steganography, embedding capacity and anti-steganalysis performance are two important focuses. In our experiments, we measure the embedding capacity by bit per frame (*bpf* for short), which is explained as follows.

$$r = \frac{\text{The total number of embedding bits}}{\text{The number of frames}} \tag{3.1}$$

Similarly, the anti-steganalysis performance is evaluated by minimum average classification probability error (P_E in short).

$$P_E = \min_{P_{FA}} \left(P_{FA} + P_{MD}\right)/2 \tag{3.2}$$

where P_{FA} and P_{MD} are the false-alarm and the missing detection rates of a detector, respectively.

3.2. Test optimal modulation parameters $\alpha(Q)$ and $\beta(Q)$

We design a series of experiments to test the contribution of two modulation parameters $\alpha(Q)$ and $\beta(Q)$ and give their optimal values, which are determined when the P_E has the minimum.

Two video sequences, *Highway* and *Paris*, are used to perform this experiment because they contain more frames ($2000 + 1065 = 3065$). We save these frames with different quality factors and then divide them equally into training and testing sets. Gabor Filter Residual (GFR) feature set [7] and ensemble classifier [23] are used to provide the experimental results because they can effectively detect modern steganography, e.g. J-UNIWARD [3]. We determine the optimal modulation factors experimentally by getting the minimal P_E. Six quality factors, 70, 75, 80, 85, 90, 95, are tested to obtain the optimal

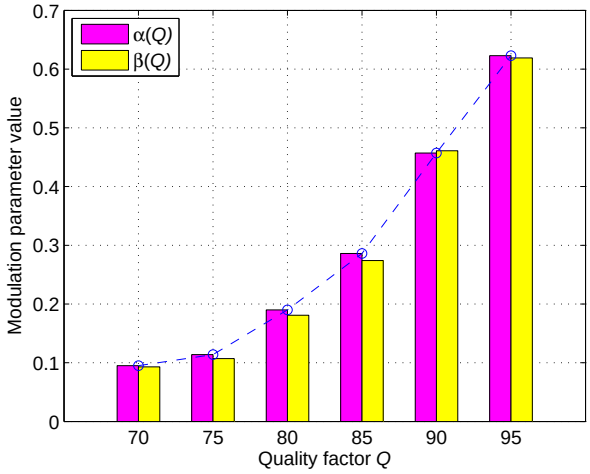


Figure 5. The modulation parameters $\alpha(Q)$ and $\beta(Q)$ with JPEG quality factor Q increasing. The relative payload is $r = 0.2$ Kbpf.

modulation parameters. In Figure 5, we give the changing trend of two parameters under relative payloads $r = 0.2$Kbpf. As can be seen in this figure, there is only a slight difference between $\alpha(Q)$ and $\beta(Q)$. This is because the adjacent frames have the same scene (same content). When the continuous adjacent frames are used as the reference (pre-cover) of current frame, there might have a (approximate) same steganographic costs. In addition, we can see that the modulation value becomes significantly bigger with the quality factors increasing. In fact, this interesting phenomenon is mainly related to the calculation procedure of original distortion function.

Table 2 shows the optimal modulation parameters for different payloads by carrying out a series of experiments. It can be observed that the optimal modulation parameter values gradually decrease with the payload increasing. Actually, since the new distortion depends on the original distortion (e.g. $\rho_{ij}(\pm 1)$ in Section 2.3), with respect to embedding in single frame, the new distortion function referring to the continuous adjacent frames will significantly increase empirical security, especially for the large payloads and small quality factors.

3.3. Robustness analysis for proposed scheme

In our proposed scheme, we introduce Vandermonde matrix to divide the original data into multiple shares (e.g. Figure 2), and then hide these shares in a series of video frames by combining new distortion function and STC algorithm. In this section, we analyze the robustness of proposed scheme.

With the Equation (2.4) and Figure 2, we know that m q-ary digits can be expanded to n q-ary digits by Vandermonde matrix. Obviously, n q-ary digits carry m original digits. In other words, for n expanded digits, each of them only carries $\frac{m}{n}$ valid original digits. Therefore, there is some redundancy in n expanded digits, which can be calculated by $\frac{n-m}{n}$ (referring to Equation (2.5)). As such, as long as the recipient can receive no less than m digits from n expanded digits, he can recover the m original digits. Actually, this procedure can be deduced easily from the Figure 2. For example, we assume that the recipient has received m' shares, $m \leq m' \leq n$. He selects m ones from m' shares, and then extract m expanded data vectors $\mathbf{t'}_1, \mathbf{t'}_2, \mathbf{t'}_3, \cdots, \mathbf{t'}_m$ and their corresponding indices $a'_1, a'_2, \cdots, a'_m$. According to

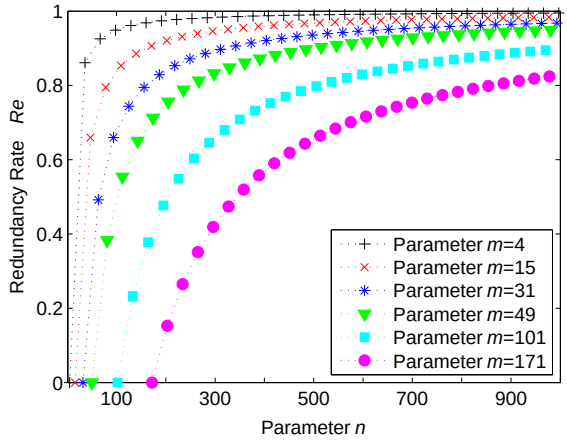


Figure 6. The relationship between redundancy rate R_e and two parameters m and n. Six different values, $m = 4$, $m = 15$, $m = 31$, $m = 49$, $m = 101$, $m = 171$, are tested and satisfy the condition $m \leq n \leq q$.

Equation (2.3), a Vandermonde matrix $\mathbf{A}'$ can be built according to m indices.

$$\mathbf{A}' = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ a'_1 & a'_2 & \cdots & a'_m \\ (a'_1)^2 & (a'_2)^2 & \cdots & (a'_m)^2 \\ \vdots & \vdots & \cdots & \vdots \\ (a'_1)^{m-1} & (a'_2)^{m-1} & \cdots & (a'_m)^{m-1} \end{bmatrix} \bmod q. \tag{3.3}$$

We can prove that matrix $\mathbf{A}'$ has an inverse matrix in q-ary notational system because it is a full-rank matrix. We omit the actual proof [11] here due to space limitations. Denote $\mathbf{A'^{-1}}$ as the inverse matrix of $\mathbf{A}'$, the original data $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_m\}$ can be calculated easily by the following equation.

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \mathbf{d}_3 & \cdots & \mathbf{d}_m \end{bmatrix} = \begin{bmatrix} \mathbf{t'}_1 & \mathbf{t'}_2 & \mathbf{t'}_3 & \cdots & \mathbf{t'}_m \end{bmatrix} \cdot (\mathbf{A}')^{-1} \tag{3.4}$$

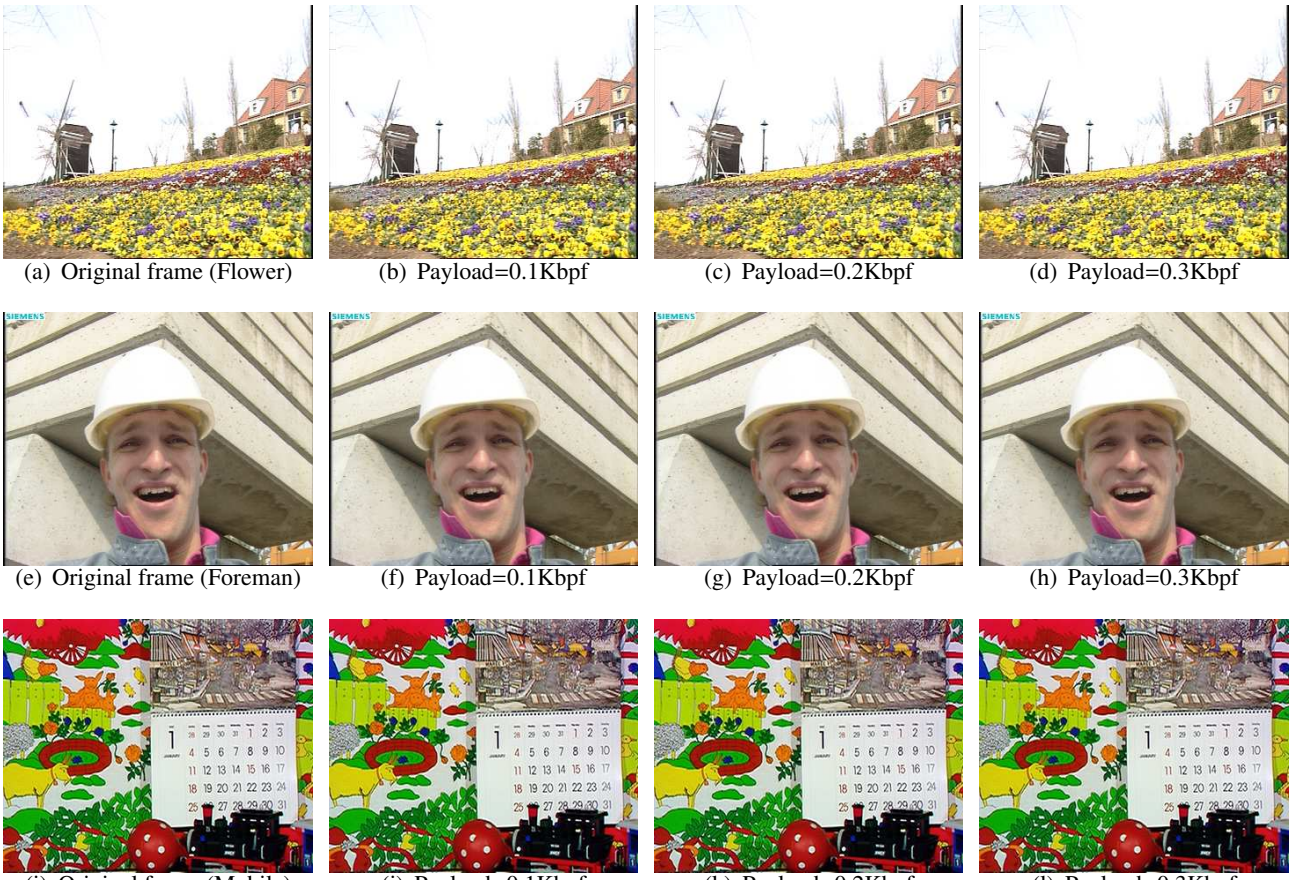
In general, the redundancy rate R_e depends on the parameters m and n. It can be up very high if we set an extreme gap between m and n. Table 3 shows the relationship between redundancy rate R_e and parameters m and n. As can been in this table, when the parameter m is fixed, R_e will become higher with n increasing. This conclusion can be also validated theoretically by the trend of lines in Figure 6.

3.4. Comparison with the state of the arts

In this section, we compare the proposed video steganographic scheme with other state-of-the-arts. The performance comparison mainly focuses on three aspects: visual quality, robustness, and anti-steganalysis. We should raise the readers' attention that we do not use the video sequences *Highway*

Table 3. Different parameter combinations (m, n) and different redundancy rate R_e.

m	n	q	R_e
5	9	11	44.44%
5	13	17	61.54%
5	29	31	82.76%
5	71	73	92.96%
5	251	251	98.01%



(a) Original frame (Flower) (b) Payload=0.1Kbpf (c) Payload=0.2Kbpf (d) Payload=0.3Kbpf

(e) Original frame (Foreman) (f) Payload=0.1Kbpf (g) Payload=0.2Kbpf (h) Payload=0.3Kbpf

(i) Original frame (Mobile) (j) Payload=0.1Kbpf (k) Payload=0.2Kbpf (l) Payload=0.3Kbpf

Figure 7. Visual quality of original and stego frames for Flower (frame 48), Foreman (frame 30) and Mobile (frame 63). The relative payloads are 0.1 Kbpf, 0.2 Kbpf, and 0.3 Kbpf.

and *Paris* in the following experiments, because they are used in Section 3.2 to find the optimal modulation parameters. If these two video sequences are re-used, the corresponding experiments might encounter the over-fitting.

Table 4. PSNR(dB) comparison by using three steganographic methods with payload r = 0.1K bpf. All testing video sequences are used in this experiment to give the average results.

Sequence	Cover	Chang et al. [14]	Liu et al. [15]	Mstafa et al. [16]	Proposed Scheme
Bus	33.378	32.963	33.084	33.190	33.212
City	34.898	34.657	34.775	34.782	34.804
Coastguard	34.562	34.157	34.321	34.379	34.483
Crew	37.012	36.792	36.801	36.887	36.901
Flower	34.501	34.125	34.301	34.374	34.423
Football	35.928	35.645	35.709	35.756	35.811
Foreman	36.068	35.798	35.887	35.892	35.975
Harbour	34.040	33.724	33.882	33.910	33.922
Ice	39.556	39.084	39.163	39.192	39.221
Mobile	33.476	33.003	33.098	33.104	33.192
Soccer	35.518	35.241	35.287	35.296	35.332
Tempete	34.531	34.302	34.339	34.387	34.401
Waterfall	34.668	34.012	34.206	34.213	34.279

3.4.1. Visual quality comparison

The imperceptibility is very important for video steganography. It is always required that the steganographic method should not cause severe visual quality degradation.

In our experiment, we use J-UNIWARD algorithm to calculate original distortion and then employ proposed new distortion function to further improve the original distortion. Peak signal-to-noise ratio (PSNR for short) is used to evaluate the visual quality of stego video sequences. Since the video frames are compressed format, we calculate the PSNR (dB) by comparing the uncompressed video sequence before data embedding and the decompression reconstructed video sequence after data embedding. We test all video sequences with three payloads 0.1Kbpf, 0.2Kbpf, and 0.3Kbpf. Figure 7 shows the visual quality of proposed scheme with three payloads. As can be seen in this figure, the stego frames and original frames are apparently difficult to distinguish. This demonstrates that proposed steganographic scheme has a high visual quality. Additionally, Table 4 lists the PSNR values of all video sequences by comparing three steganographic methods, Chang's method [14], Liu's method [15], and Mstafa's method [16]. The payload is fixed to 0.1K bpf. We can get the conclusion from this table that proposed scheme has a slight visual quality degradation after data hiding, but, comparison with other methods, it still has a significant superior performance.

3.4.2. Robustness testing

We further test the robustness of proposed scheme with a series of experiments. The corresponding experiments can be performed by three attack forms: usual attack, frame attack and video compression attack.

We firstly test the robustness of proposed scheme for usual attack forms. Three usual attack forms, including Salt&Peppers noise, Gaussian noise and Median filtering, are used to provide testing results.

In these experiments, we set data decomposition parameter combination as ($m = 5$, $n = 13$, $q = 17$), the redundancy rate is thus $R_e = 61.54\%$. In other words, as long as the expanded data is lost (or damaged) no more than 61.54%, the original data can be recovered perfectly. In data embedding procedure, we evenly hide the expanded data (multiple shares) in each frame with payload $0.2K$ bpf. For Salt&Peppers noise attack, the noise intention is fixed $I = 0.01$ and 0.05, and for Gaussian noise attack, we also fix the parameters $V = 0.01$ and 0.05, while for Median filtering, the size of block is set to 3×3. We verify proposed scheme by using and un-using ensemble mechanism, where the ensemble rounds are 11 (corresponding to $en = 11$ in Algorithm 2). Each experiment is run 100 times. Each time, we randomly extract data from the remaining (complete) frames. The results are from average calculation as the times that the original data can be recovered correctly over the total testing times. Table 5 shows the experimental results. It can be observed that, for three usual attack forms, when the ensemble mechanism is not used, the ratio that original data can be recovered perfectly is rather low, only 67% for Salt&Peppers noise with $I = 0.01$, 63% for Gaussian noise with $V = 0.01$ and 79% for Median filtering. This is because the noise might modify the embedding information so that the recovered original data is wrong even if they seem to be complete.

In addition, we also test two video frame attack forms, frame cropping and frame removal. For frame cropping, we set two cases, only one frame cropping and all frames cropping. The cropping scales are fixed 20% and 40%. For frame removal, five removal ratios are tested for all frames, 20%, 40%, 50%, 60% and 70%. The corresponding experimental results are shown in Table 6. As can be seen from this table, when the cropping for all frames is more than 40%, the data reconstruction ability for non-ensemble scheme becomes inferior. When the removal ratio is up to 70%, the original data cannot be recovered because the lost ratio has exceeded the redundancy rate $R_e = 61.54\%$. Overall, proposed scheme can implement a robust recovery for the original data even if they are lost or damaged during delivery.

Moreover, in order to gain more insight, we also test the proposed scheme by using the H.264 video compression with different quantization parameters (QP for short). In this attack from, compression is applied to every macroblock of video frame and QP is used to control the level of compression. Table 7 presents the experimental results. We can observe from this table that the data reconstruction ability significantly becomes inferior with QP value increasing, because lower QP value maybe cause an inferior video quality, leading to a larger data modification. Also, for compression testing with different QP values, $QP = 10, 20, 30, 40$, we further test PSNR for three video steganographic schemes, Chang's scheme [14], Dalal's scheme [24] and proposed scheme. Three video sequences, Bus, Flower and Foreman, are used to give the experimental results, which are shown in Table 8. As can be seen that increasing of QP value results in more compression and leads to a lower video quality.

3.4.3. Anti-steganalysis testing

As a good video steganographic method, the anti-steganalysis capability is also an important focus. Since video steganographic schemes consistently hide the messages in a batch of video frame, we use three batch steganographic detection schemes (also named steganographer detection methods), hierarchical clustering scheme (HC) [25], local outlier factor detection (LOF) [26], and ensemble clustering scheme (EC) [27], to demonstrate the security performance of different video steganographic schemes against steganalysis.

Table 9 shows the comparison results between proposed scheme and three state-of-the-art schemes

Table 5. Robustness testing for three usual attack forms, Salt&Peppers noise, Gaussian noise and Median filtering. We test proposed scheme with ensemble mechanism (Ensemble) and without ensemble mechanism (Non-Ensemble), respectively. All testing video sequences are used in this experiment to give the average results.

Attacking Forms	Attacking Parameters	Non-Ensemble	Ensemble
Salt&Peppers	$I = 0.01$	67%	100%
	$I = 0.05$	46%	94%
Gaussian noise	$V = 0.01$	63%	98%
	$V = 0.05$	42%	92%
Median filtering	3×3 block	79%	100%

Table 6. Robustness testing for two video frame attack forms, frame cropping and frame removal. We test proposed scheme with ensemble mechanism (Ensemble) and without ensemble mechanism (Non-Ensemble), respectively. All testing video sequences are used in this experiment to give the average results.

Attacking Forms	Attacking Parameters	Non-Ensemble	Ensemble
Frame Cropping	20% (One frame)	100%	100%
	40% (One frame)	100%	100%
	20% (All frames)	82%	98%
	40% (All frames)	53%	90%
Frame Removal	20% frames	100%	100%
	40% frames	100%	100%
	50% frames	68%	100%
	60% frames	15%	62%
	70% frames	0%	0%

Table 7. Robustness testing for H.264 video compression attack with different quantization parameter (QP). We test proposed scheme with ensemble mechanism (Ensemble) and without ensemble mechanism (Non-Ensemble), respectively.

Attacking Forms	Attacking Parameters	Non-Ensemble	Ensemble
H.264 Compression	$QP = 40$	31%	78%
	$QP = 30$	44%	84%
	$QP = 20$	52%	95%
	$QP = 10$	73%	100%

Table 8. When H.264 compression attacks with different quantization parameter (QP) are used, PSNR(dB) comparison for three video steganographic schemes, Chang's scheme [14], Dalal's scheme [24] and proposed scheme. Three video sequences, Bus, Flower and Foreman, are used in this experiment.

Schemes	Video	Quantization parameter QP			
		$QP = 10$	$QP = 20$	$QP = 30$	$QP = 40$
Chang et al. [14]	Bus	33.544	33.012	32.910	32.874
	Flower	34.890	34.576	34.250	34.011
	Foreman	35.169	35.083	34.980	34.911
Dalal et al. [24]	Bus	33.788	33.654	33.540	33.217
	Flower	35.014	34.983	34.756	34.669
	Foreman	36.701	36.542	36.102	35.818
Proposed	Bus	34.095	34.013	33.953	33.881
	Flower	35.870	35.704	35.556	35.231
	Foreman	37.017	36.809	36.544	36.223

with three payloads r = 0.01K, 0.02K, 0.03K bpf, respectively. In this experiment, we regard each video sequence as a cluster (total 13 clusters). Each experiment, we randomly choose 7 clusters (video sequences), and select 50 frames from each one. Then, one cluster is randomly chosen as the guilty who uses three payloads mentioned above to hide messages, respectively. Each experiment is repeated 100 times and the overall identification accuracy rate is used to evaluate anti-steganalysis that is denoted by the number of correctly identification over the total testing number. From Table 9, we can observe that proposed scheme has a lower accuracy rate than that of other schemes. This illustrates that proposed scheme is more secure. In addition, we find that the EC method is conclusively more efficient comparing with HC and LOF methods. This is because EC method uses the ensemble clustering mechanism containing a number of sub-clustering, it can experimentally give a superior detection performance. Actually, this interesting phenomenon has been conclusively verified in [27].

4. Conclusions and future works

In this paper, we proposed a robust video steganographic scheme. We first expand original data to multiple shares. This mechanism can ensure that the recipient recover the original data successfully even if they only obtain a part of data. Then, a new distortion function is designed by using continuous adjacent video frames as side-information, which can further improve the security performance of steganography. Proposed scheme is robust in the sense that the recipient can recover the hidden data even if some frames are damaged or lost during delivery. Extensive experiments are performed to show that our proposed schemes outperform existing video steganographic schemes in terms of visual quality, robustness and anti-steganalysis.

While proposed scheme has shown a good performance in the diverse tests, we should note that it has an obvious short on the utilization of video sequences because proposed scheme consistently expands the original data to multiple shares, this makes that the actual hidden data become more and

Table 9. Overall identification accuracy rate for four video steganographic schemes. Three steganalysis methods, HC, LOF and EC, are used.

Schemes	Payload r	HC method	LOF method	EC method
Chang et al. [14]	0.1K	82%	88%	90%
	0.2K	84%	89%	92%
	0.3K	88%	94%	97%
Liu et al. [15]	0.1K	79%	82%	87%
	0.2K	82%	90%	91%
	0.3K	86%	90%	93%
Mstafa et al. [16]	0.1K	77%	80%	84%
	0.2K	80%	85%	90%
	0.3K	85%	86%	89%
Proposed	0.1K	73%	78%	79%
	0.2K	78%	82%	86%
	0.3K	81%	84%	88%

more, leading to a significant cover-consuming. Nevertheless, we believe that this is only a small problem because we can easily obtain massive video sequences from the Internet. Moreover, although our method is robust in social networks, if too many shares are damaged or lost, proposed scheme will do not work.

Finally, we believe that there may be some room for further improvement. For example, the distortion function can be designed by involving more adjacent frames, although the complexity may rise sharply. In addition, we should consider to further reduce the computational complexity for inverse matrix in q-ary notation system. The above two issues are left as our future works.

Acknowledgments

This work was supported by Natural Science Foundation of China under Grants (No.61602295, No.U1736120) and Natural Science Foundation of Shanghai (No.16ZR1413100, No.18ZR1427500) and the Foreign Visiting Scholar Program of Shanghai Municipal Education Commission.

Conflict of interest

The authors declare that there is no conflict of interests regarding the publication of this article.

References

1. J. Mielikainen, LSB matching revisited, *IEEE. SPL.*, **13**(2006), 285–287.
2. F. Li, X. Zhang, J. Yu, et al., Adaptive JPEG steganography with new distortion function, *Ann. Telecommun.*, **69**(2014), 431–440.

3. V. Holub, J. Fridrich and T. Denemark, Universal distortion function for steganography in an arbitrary domain, *EURASIP JIS.*, Special Issue on Revised Selected Papers of the 1st ACM IH and MMS Workshop, (2014), 2014:1.
4. T. Pevný, T. Filler and P. Bas, Using high-dimensional image models to perform highly undetectable steganography, *Proc. IH2010*, Calgary, AB, Canada, June 28–30, **6387**(2010), 161–177.
5. F. Li, X. Zhang, H. Cheng, et al., Digital image steganalysis based on local textural features and double dimensionality reduction, *SCN*, **9**(2016), 729–736.
6. B. Chen, G. Feng, X. Zhang, et al., Mixing high-dimensional features for JPEG steganalysis with ensemble classifier, *SIVP*, **8**(2014), 1475–1482.
7. X. Song, F. Liu, C. Yang, et al., Steganalysis of adaptive JPEG steganography using 2D Gabor filters, *Proc. IH&MMSec2015* , Portland, OR, USA, June 17-19, (2015), 15–23.
8. F. Li, X. Zhang, B. Chen, et al., JPEG steganalysis with high- dimensional features and Bayesian ensemble classifier, *IEEE SPL*, **20**(2013), 233–236.
9. Z. Zhao, Q. Guan, X. Zhao, et al., Universal embedding strategy for batch adaptive steganography in both spatial and JPEG domain, *MTAP*, **77**(2018), 14093–14113.
10. R. Cogranne, V. Sedighi and J. Fridrich, Practical strategies for content-adaptive batch steganography and pooled steganalysis, *Proc. ICASSP2017*, New Orleans, USA, March 5–9, (2017), 2122–2126.
11. F. Li, K. Wu, X. Zhang, et al., Robust batch steganography in social networks with non-uniform payload and data decomposition, *IEEE Access*, **6**(2018), 29912–29925.
12. E. Franz, Embedding considering dependencies between pixels, *Proc. SPIE, Electron. Imag., Secur., Forens., Steg., and Watermark. of Multimedia Contents X*, **6819**(2008), D 1–12.
13. T. Denemark and J. Fridrich, Steganography with multiple JPEG images of the same scene, *IEEE TIFS*, **12**(2017), 2308–2319.
14. P. Chang, K. Chung, J. Chen, et al., A DCT/DST-based error propagation-free data hiding algorithm for HEVC intra-coded frames, *JVCIR*, **25**(2014), 239–253.
15. Y. Liu, Z. Li, X. Ma, et al., A robust data hiding algorithm for H.264/AVC video streams. *J. Sys. Sof.*, **86**(2013), 2174–2183.
16. R. Mstafa, K. Elleithy and E. Abdelfattah, A robust and secure video steganography method in DWT-DCT domains based on multiple object tracking and ECC, *IEEE Access*, **5**(2017), 5354–5367.
17. H. Aly, Data hiding in motion vectors of compressed video based on their associated prediction error, *IEEE TIFS*, **6**(2011), 14–18.
18. H. Zhang, Y. Cao and X. Zhao, Motion vector-based video steganography with preserved local optimality, *MTAP*, **75**(2016), 13503–13519.
19. H. Zhang, Y. Cao, X. Zhao, et al., Video steganography with perturbed macroblock partition, *Proc. IH&MMSec2014*, Salzburg, Austria, Jun. 11–13, (2014), 115–122.
20. I. Cox, J. Kilian, F. Leighton, et al., Secure spread spectrum watermarking for multimedia, *IEEE TIP*, **6**(1997), 1673–1687.

21. V. Neagoe, Inversion of the Van der Monde matrix, *IEEE SPL*, **3**(1996), 119–120.
22. YUV Video Sequences, Available from: http://trace.eas.asu.edu/yuv/index.html, [Accessed on 2018].
23. J. Kodovský, J. Fridrich and V. Holub, Ensemble classifier for steganalysis of digital media, *IEEE TIFS*, **7**(2012), 432–444.
24. M. Dalal and M. Juneja, A robust and imperceptible steganography technique for SD and HD videos, *MTAP*, 2018, to appear. https://doi.org/10.1007/s11042-018-6093-3.
25. A. Ker and T. Pevný, A new paradigm for steganalysis via clustering, *Proc. SPIE, Media Watermark., Secur., Forens. III*, **7880**(2011), 78800U.
26. A. Ker and T. Pevný, The steganographer is the outlier: realistic large-scale steganalysis, *IEEE TIFS*, **9**(2014), 1424–1435.
27. F. Li, K. Wu, J. Lei, et al., Steganalysis over large-scale social networks with high-order joint features and clustering ensembles, *IEEE TIFS*, **11**(2016), 344–357.

AIMS Press