Multimed Tools Appl (2015) 74:11163–11186
DOI 10.1007/s11042-014-2223-8

Defining embedding distortion for motion vector-based video steganography

Yuanzhi Yao · Weiming Zhang · Nenghai Yu · Xianfeng Zhao

Received: 12 February 2014 / Revised: 9 June 2014 / Accepted: 6 August 2014 /
Published online: 3 September 2014

Abstract This paper presents an effective methodology for motion vector-based video steganography. The main principle is to design a suitable distortion function expressing the embedding impact on motion vectors by exploiting the spatial-temporal correlation based on the framework of minimal-distortion steganography. Two factors are considered in the proposed distortion function, which are the statistical distribution change (SDC) of motion vectors in spatial-temporal domain and the prediction error change (PEC) caused by modifying the motion vectors. The practical embedding algorithm is implemented using syndrome-trellis codes (STCs). Experimental results show that the proposed method can enhance the security performance significantly compared with other existing motion vector-based video steganographic approaches, while obtaining the higher video coding quality as well.

Keywords Steganography · Steganalysis · Motion vector · Embedding distortion · Prediction error

Y. Yao · W. Zhang (✉) · N. Yu
Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China
e-mail: zhangwm@ustc.edu.cn

Y. Yao
e-mail: yaoyz@mail.ustc.edu.cn

N. Yu
e-mail: ynh@ustc.edu.cn

X. Zhao
State Key Laboratory of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing 100093, China
e-mail: zhaoxianfeng@iie.ac.cn

1 Introduction

Steganography is the art and science of data hiding which embeds secret data into a digital cover media, such as digital audio, image, video, etc., without arousing suspicion. It aims to set up a covert communication path between two parties such that a possible attacker in the middle cannot detect its existence. On the other side, steganalysis aims to detect the presence of the hidden secret data in those stego media exploiting the statistical evidence. The steganographic system is considered broken if there exists a steganalytic algorithm which can decide whether a given media contains the hidden secret data or not with a high probability than random guessing. Therefore, the undetectable performance should be carefully considered in designing the steganographic system.

To resist detection, a secure steganoraphic scheme should minimize the embedding impact on the cover. The embedding impact is usually formulated as a distortion function that is defined according to some characters of the media, and then some stego-coding techniques can be used to minimize distortion for the embedding process.

The early stego-coding methods only deal with constant distortion (e.g. matrix embedding codes [5]), for which the modifications on each cover element are equally risky, so the purpose of coding is to reduce the number of modifications. However, the embedding impact on different parts of the cover are significantly different for steganography. For instance, modifications on less textured areas of image will be easily detected by steganalysis. To solve this problem, Fridrich et al. proposed wet paper codes [12], in which the cover elements are divided into two parts, wet elements and dry elements. By wet paper codes, the sender can embed messages by only modifying dry elements and the receiver can extract the message without any knowledge on the positions of dry elements. In the model of wet paper codes, the embedding distortion on dry element are still equal, which can not reflect the difference of embedding impact on the cover elements. So coding methods for more general distortion model are desired. Syndrome-trellis codes (STCs), proposed by Filler et al. in [8] and [10], are codes for such general distortion model. STCs can minimize various kinds of distortions and the receiver can extract the embedded messages without any knowledge about the definition of distortion. So far, STCs serve as the most powerful coding method for steganography.

With the mature coding techniques, the rest problem for steganography is how to define the distortion function to reasonably reflect the embedding impact. In fact, a non-reasonable distortion function with even a best coding method can not ensure the security of steganography. A good distortion function should grasp factors that influence the ability of resisting various detections. Recently, some methods on defining distortion for spatial images [9, 15, 20] and JPEG images [9, 14, 17, 21] have been proposed, which show that an effective distortion function combing with an efficient coding method can greatly improve the security of steganography. However, as pointed by Ker et al. [18], how to define the distortion for video steganography is still an open problem (the Open Problem 4 presented in [18]).

In fact, as the popular and important media for today's media-centric society, video is an ideal cover for steganography, because video contains more adequate information and more complicated coding modes than image and audio. Many steganographic methods have been applied for compressed video. In [28], data hiding is implemented by modifying the partition modes of the sub-macroblocks according to the message bits to be embedded. The other two data hiding methods are proposed in [22]. The first method embeds message bits by modulating the quantization step of a constant bitrate video. Predefined macroblock level features are extracted from each encoded marcoblock to construct the expanded feature matrix. And the embedded message bits are predicted using a second-order multivariate regression. The second method in [22] assigns macroblocks to arbitrary slice groups according to the message bits to be embedded using the flexible macroblock ordering (FMO) feature of H.264/AVC. Xu et al. proposed to embed message bits in horizontal components or vertical components of motion vectors using LSB (Least Significant Bit) replacement [27]. The larger component of each motion vector is used to carry one single bit. Fang and Chang [6] used a pair of motion vectors as the cover unit. For each pair, if the phase angle difference does not satisfy the embedding condition, one of them should be modified until the condition is satisfied. The methods in [27] and [6] choose the candidate motion vector whose magnitude is larger than a predefined threshold for data embedding to introduce less distortion.

However, Aly pointed to embed data in the candidate motion vectors based on their associated macroblock prediction errors [1], which can achieve a minimum distortion to the reconstructed video and minimize the coded bit rate increment. The motion vectors associated with larger reconstructed prediction errors are chosen and both horizontal and vertical components are embedded using LSB replacement. The methods in [1, 6, 27] share one feature in common in which they decide the candidate motion vectors for data embedding during motion estimation following the predefined selection rule. However, if the selection rule is public, the steganalyzer can easily build the attacking methods. Moreover, the asymmetry artifact caused by LSB replacement leaves the obvious statistical evidence for steganaysis in [27] and [1]. Cao et al. introduced perturbed motion estimation (PME) to perform data embedding [3], in which some motion vectors are defined to be defective, i.e., modifications on these motion vectors are forbidden. And then wet paper codes [12] are applied to motion vectors, which enables to embed messages into a cover with defective cells.

The above mentioned video steganographic schemes [1, 3, 6, 27] can not reach a high level of security, because they neither elaborated a reasonable distortion function nor adopted a powerful stego-coding method to minimize the distortion. Although the stego-coding method was used in [3], only a simple distortion model (wet/dry model) and wet paper codes were considered. Note that the state-of-the-art steganographic schemes for both uncompressed image (e.g., HUGO [20] and WOW [15]) and compressed image (e.g., UNIWARD [16]) are based on STCs. So we expect that a reasonable distortion function combining the STCs can greatly improve the security of video steganography.

In this paper, we study how to define distortion for motion vector-based steganography. We pay attention to motion vectors for the following three reasons. First, motion vectors are the key information expressing the video content during the encoding process and the decoding process which constitute an important part of the coded bit stream and they are losslessly coded, so the high embedding capacity can be achieved using motion vectors as the cover. Secondly, the visual quality degradation introduced by embedding data in motion vectors is relatively limited through the motion compensation and the residue coding. Finally but not the least, image steganographic techniques for spatial domain [9, 15, 20] and DCT (Discrete Cosine Transform) domain [9, 14, 17, 21] can also be applied to video, but the motion vector is one special character of video that should be further exploited as steganographic covers.

Thus, the design of distortion function on embedding messages into motion vectors has become the core problem in this paper. The statistical distribution change (SDC) introduced by modifying motion vectors is considered in proposed distortion function because the change of cover source statistical distribution is the main evidence exploited by steganalysis. To effectively express the statistical distribution change, the co-occurrence matrices computed from motion vector component differences are applied using the spatial correlation and the temporal correlation. On the other hand, as the vital character of motion estimation, there is a direct connection between the block prediction error and the motion vector. Therefore, the prediction error change (PEC) should be also considered, which is revisited in Section 3 in detail. The two factors are nonlinearly combined together to form a distortion function to help the coder choose the motion vector which may introduce the minimal embedding impact for data embedding.

The rest of this paper is organized as follows. In Section 2, we describe the framework of minimal-distortion steganography. The design of distortion function with analysis is elaborated in Section 3. We show the implementation of the practical video steganographic method in Section 4 followed by the experimental results in Section 5. Finally, the conclusion is drawn in Section 6.

2 Framework of minimal-distortion steganography

In this paper, matrices and sets of variables will be written in boldface using capital letters. Vectors will be always typeset in boldface lower case.

The cover sequence is denoted by $\mathbf{x} = (x_1, x_2, ..., x_n)$, where the signal x_i is an integer, such as the gray value of a pixel. The embedding operation on x_i is formulated by the range I_i. An embedding operation is called binary if $|I_i| = 2$ and ternary if $|I_i| = 3$ for all i. For example, the embedding operation of decreasing the absolute values of the quantized DCT coefficients can be represented by $I_i = \{x_i, x_i - sign(x_i)\}$ and the ±1 embedding operation can be represented by $I_i = \{x_i - 1, x_i, x_i + 1\}$.

We assume that the cover $\mathbf{x}$ to be fixed and the embedding operations on x_i are independent mutually, so the distortion introduced by changing $\mathbf{x}$ to $\mathbf{y} = (y_1, y_2, ..., y_n)$ can be simply denoted by $D(\mathbf{y}) = D(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \rho_i(x_i, y_i)$, where $\rho_i(x_i, y_i) \in \mathbb{R}$ is the cost of changing the ith cover element x_i to y_i ($y_i \in I_i, i = 1, 2, \ldots, n$).

Assume that the embedding algorithm changes $\mathbf{x}$ to $\mathbf{y} \in \mathcal{Y}$ with probability $\pi(\mathbf{y}) = P(Y = \mathbf{y})$, and then as proved in [7, 10], the sender can send up to $H(\pi)$ bits on average with average distortion $E_\pi(D)$ such that

$$H(\pi) = -\sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) \log \pi(\mathbf{y}), \quad E_\pi(D) = \sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) D(\mathbf{y}) \tag{1}$$

After defining the distortion function, the sender will consider the following optimization problems:

- Minimizing the average distortion for a fixed average payload of m bits,

$$\min_{\pi} E_\pi(D), \text{ subject to } H(\pi) = m. \tag{2}$$

- Maximizing the average payload for a fixed average distortion D_ε,

$$\max_{\pi} H(\pi), \text{ subject to } E_\pi(D) = D_\varepsilon. \tag{3}$$

As pointed out in [10], problems (2) and (3) are dual to each other, that is, the optimal distribution for problem (3) for some value of D_ε is also optimal for problem (2). Following the maximum entropy principle, the solution has the form of a Gibbs distribution [7]:

$$\pi_\lambda(y_i) = \frac{\exp(-\lambda\rho_i(x_i, y_i))}{\sum_{y_i\in I_i} \exp(-\lambda\rho_i(x_i, y_i))}, \quad i = 1, 2, \ldots, n \tag{4}$$

where the parameter λ is obtained from the corresponding constraints (2) or (3).

The minimal distortion can be approached by embedding messages with STCs (Syndrome-Trellis Codes) [10]. STCs are a kind of syndrome coding, which can be formulated as

$$Emb(\mathbf{x}, \mathbf{m}) = \arg \min_{\mathbf{y} \in C(\mathbf{m})} D(\mathbf{x}, \mathbf{y}), \text{ and } \mathbf{Hy} = \mathbf{m} \tag{5}$$

Here, $\mathbf{x}$ is a binary vector of covers (e.g., the LSBs of cover elements), $\mathbf{y}$ is the corresponding stego vector, and $\mathbf{m}$ is the message. $\mathbf{H}$ is the parity-check matrix and $C(\mathbf{m})$ is the coset corresponding to syndrome $\mathbf{m}$. The syndrome coding process is to find the stego $\mathbf{y}$, satisfying $\mathbf{Hy} = \mathbf{m}$ and having minimal distortion. After receiving the stego, the receiver can easily extract messages by computing $\mathbf{Hy}$.

The STCs are based on syndrome coding using linear convolutional codes with the optimal binary quantizer implemented using the Viterbi algorithm run in the dual domain. STCs can be viewed as a generalized matrix embedding, and the parity check matrix $\mathbf{H}$ used in STCs is constructed by a $h \times w$ sub-matrix $\hat{\mathbf{H}}$. As traditional matrix embedding, we want to find the optimal solution of $\mathbf{Hy} = \mathbf{m}$. For STCs, each solution can be represented as a path through the syndrome trellis of $\mathbf{H}$. The length of the path is defined by number of modified pixels and corresponding distortions. The relative payload is equal to $1/w$ that is determined by the width w of the sub-matrix. The height h of the sub-matrix determines the number of paths, and there are 2^h choices for each grid of the trellis. Therefore, larger h means more powerful capacity to minimize distortion but also higher computational complexity.

The STCs can be applied into binary cover directly such as the LSB layer of cover element. However, current studies make clear that ± 1 embedding is safer than binary embedding operation. Zhang et al. [30, 31] presented an efficient double-layered embedding scheme where ± 1 embedding is decomposed into two binary embedding operations, namely embedding messages into LSB layer and second LSB layer respectively. Motivated by [30, 31], Filler et al. proposed STCs-based ± 1 embedding [10], which first embeds messages in the second LSB layer and then embeds messages in the LSB layer. The numbers of bits hidden in each layer need to be transmitted to the receiver, so the receiver can determine the parity-check matrix and further extract messages from each layer.

Next, we define the distortion function for motion vectors and design a video steganographic method using two-layered STCs-based ± 1 embedding.

3 Proposed distortion function

3.1 From motion estimation to distortion definition

In order to economically store digital videos on the storage-constrained devices or efficiently transmit video over the bandwidth-limited networks, video coding should be widely employed to compress raw videos to the coded bit streams. The spatial redundancy of videos is reduced using the intra prediction coding. Because the raw video is essentially a series of highly content-correlated images, the temporal correlation can be employed more efficiently to reduce the redundancy in video coding.

State-of-the-art video coding standards remove the temporal redundancy via motion estimation applied to $b \times b$ pixels block. The coding structure for an inter-coded macroblock [25] is showed in Fig. 1 .For encoding the current block, the encoder searches for the best matching block within the previous coded reference frame based on the Lagrangian cost function. Suppose the raw and uncompressed video is available to the sender. Let $e_{i,j,t}$ denotes the prediction error of the (i, j)th block in the tth frame, $f_{x,y,t}$ denotes the luma

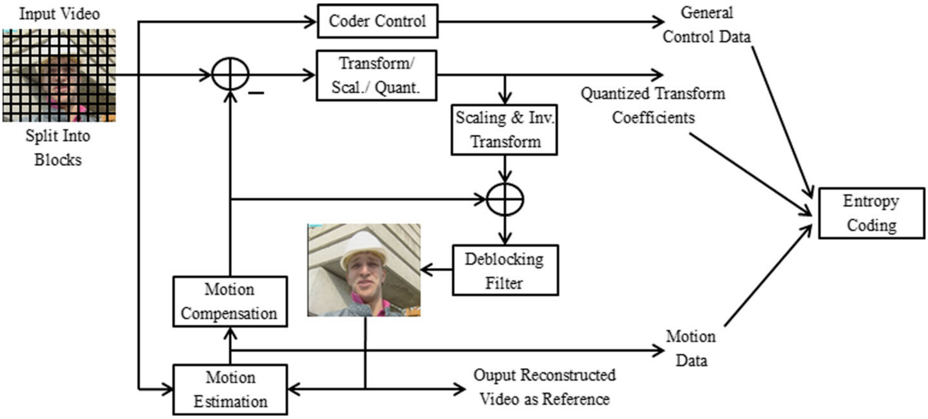


Fig. 1 Coding structure for an inter-coded macroblock

pixel value of the (x, y)th pixel in the tth current frame and $\tilde{f}_{x,y,t}$ denotes the luma pixel value of the (x, y)th pixel in the tth reference frame with $i = \lceil x/b \rceil$ and $j = \lceil y/b \rceil$. The prediction error is measured as the SAD (Sum of Absolute Difference) of the pixel value difference:

$$e_{i,j,t} = \sum_{x=x_0+1}^{x_0+b} \sum_{y=y_0+1}^{y_0+b} |f_{x,y,t} - \tilde{f}_{x+mvx,y+mvy,t-1}| \quad (6)$$

where $x_0 = (i-1) \times b$ and $y_0 = (j-1) \times b$. Given the Lagrange parameter λ_{Motion} and the decoded reference picture, rate-constrained motion estimation for the (i, j)th block is performed by minimizing the Lagrangian cost function [26], then the optimal motion vector is determined as

$$\mathbf{mv}_{i,j,t} = (mvx_{i,j,t}, mvy_{i,j,t}) = \arg\min_{\mathbf{mv}}\{e_{i,j,t} + \lambda_{Motion} R_{Motion}(\mathbf{mv})\} \quad (7)$$

where $mvx_{i,j,t}$ and $mvy_{i,j,t}$ are the horizontal component and the vertical component of the motion vector respectively. In (7), $R_{Motion}(\mathbf{mv})$ is the number of bits used to code both components of the motion vector $(mvx_{i,j,t}, mvy_{i,j,t})$. Consequently, the motion vector $\mathbf{mv}$ expressing the motion information of the current block $\mathbf{C}$ and the pixel differential signal $\mathbf{D} = \mathbf{C} - \tilde{\mathbf{P}}$ between the current block $\mathbf{C}$ and the best reference block $\tilde{\mathbf{P}}$ should be further coded.

3.2 Statistical distribution change of motion vectors

The original motion vectors obtained through motion estimation represent the spatial correlation of blocks within the frame and the temporal correlation between the consecutive frames. Thus, there exists the original statistical distribution of motion vectors which matches the video content. Figure 2 shows the distribution of difference array value $mvx_{i,j,t} - mvx_{i,j+1,t}$, $mvy_{i,j,t} - mvy_{i,j+1,t}$, $mvx_{i,j,t} - mvx_{i,j,t-1}$ and $mvy_{i,j,t} - mvy_{i,j,t-1}$ calculated from approximately 7200 frames from 18 video sequences [29]. The ridge profile in Fig. 2 suggests that the strong correlation exists among the motion vector components in spatial domain and temporal domain.

However, the data embedding will break the spatial-temporal correlation leaving statistical evidence that may be utilized by steganalyzers. As a result, the profile in Fig. 1 will

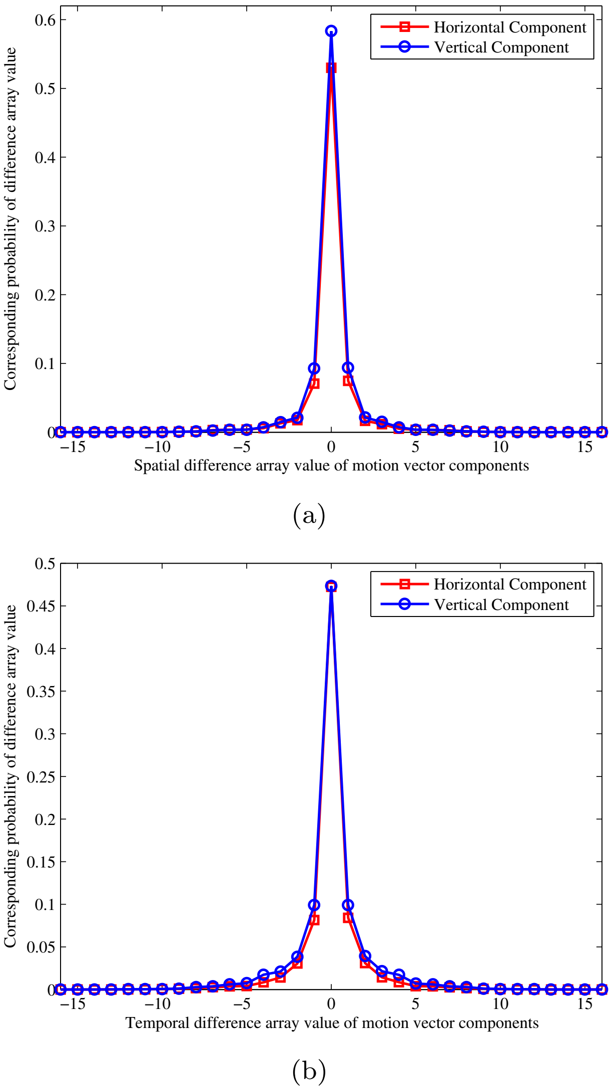


Fig. 2 Distribution of difference array value of motion vector components. **a** Spatial difference array; **b** temporal difference array

expand into outer region. When steganalyzers model the horizontal components and the vertical components of one frame as the matrix respectively, most spatial domain steganalysis algorithms in image can be exploited for video steganalysis through combining the temporal correlation. For example, the center of mass (COM) of the histogram characteristic function (HCF) has been used in [23] for motion vector-based video steganalysis. Therefore, we can confirm that the statistical distribution change (SDC) should be considered in designing the distortion function. The less statistical distribution change is introduced by the motion vector modification, the less embedding impact can be achieved.

Considering the spatial correlation, pixels within adjacent blocks have a high possibility of belonging to the same object, which can be reflected by the phenomenon that adjacent blocks have similar motion trends. We base on this fact that changes made in motion vectors of blocks with similar motion trends often tend to be highly detectable by blind steganalysis which should lead to high embedding distortion values and motion vectors of blocks with different motion trends can be changed more often.

Inspired by [10], we consider the model built from a set of all straight 3-element lines in four different orientations containing the (i, j)th motion vector component which we call cliques as shown in Fig. 3. The set of four-direction cliques is used to calculate the statistical distribution change (SDC) of motion vectors under the spatial correlation. We separate the horizontal components and the vertical components of motion vectors in one frame to construct two component matrices $\mathbf{MVX}_t$ and $\mathbf{MVY}_t$. Generally speaking, let the component matrix $\mathbf{MVX}_t$ and $\mathbf{MVY}_t$ be the $H \times W$ matrices. H and W are the height and the width of the frame in the unit of the block respectively. The calculation starts by computing the second-order difference array of the horizontal components which is for a horizontal left-to-right direction

$$dx^{\rightarrow}_{i,j,t}(\mathbf{MVX}_t) = mvx_{i,j,t} - mvx_{i,j+1,t} \tag{8}$$

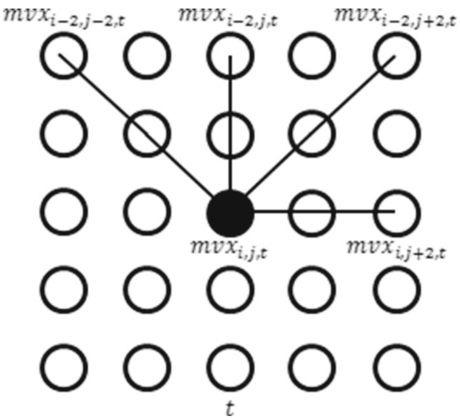
$$dx^{\rightarrow}_{i,j+1,t}(\mathbf{MVX}_t) = mvx_{i,j+1,t} - mvx_{i,j+2,t} \tag{9}$$

for $i = 1, ..., H$, $j = 1, ..., W-2$. We consider the co-occurrence matrix computed from the second-order difference array of the horizontal components in (8) and (9). Then we define the count of neighboring triples of horizontal components $\{mvx_{i,j,t}, mvx_{i,j+1,t}, mvx_{i,j+2,t}\}$ with differences $mvx_{i,j,t} - mvx_{i,j+1,t} = p$ and $mvx_{i,j+1,t} - mvx_{i,j+2,t} = q$ as

$$A^{\rightarrow}_{p,q}(\mathbf{MVX}_t) = \sum_{i=1}^{H} \sum_{j=1}^{W-2} [dx^{\rightarrow}_{i,j,t} = p, dx^{\rightarrow}_{i,j+1,t} = q] \tag{10}$$

where the function $[I]$ is defined to be 1 if the logical expression I is true and 0 otherwise. The superscript arrow "→" denotes the fact that the differences are computed by subtracting the right component from the left one.

Fig. 3 Difference array used for calculating the statistical distribution change using the spatial correlation



By analogy, we can calculate the other second-order difference arrays of the horizontal components as

$$dx^{\nearrow}_{i,j,t}(\mathbf{MVX}_t) = mvx_{i,j,t} - mvx_{i-1,j+1,t} \tag{11}$$

$$dx^{\nearrow}_{i-1,j+1,t}(\mathbf{MVX}_t) = mvx_{i-1,j+1,t} - mvx_{i-2,j+2,t} \tag{12}$$

$$dx^{\uparrow}_{i,j,t}(\mathbf{MVX}_t) = mvx_{i,j,t} - mvx_{i-1,j,t} \tag{13}$$

$$dx^{\uparrow}_{i-1,j,t}(\mathbf{MVX}_t) = mvx_{i-1,j,t} - mvx_{i-2,j,t} \tag{14}$$

$$dx^{\nwarrow}_{i,j,t}(\mathbf{MVX}_t) = mvx_{i,j,t} - mvx_{i-1,j-1,t} \tag{15}$$

$$dx^{\nwarrow}_{i-1,j-1,t}(\mathbf{MVX}_t) = mvx_{i-1,j-1,t} - mvx_{i-2,j-2,t} \tag{16}$$

which are for the diagonal left-to-right direction, the vertical down-to-up direction and the diagonal right-to-left direction respectively. Similarly, we can define the counts of neighboring triples of horizontal components with differences computed from other three directions as $A^{\nearrow}_{p,q}(\mathbf{MVX}_t)$, $A^{\uparrow}_{p,q}(\mathbf{MVX}_t)$ and $A^{\nwarrow}_{p,q}(\mathbf{MVX}_t)$.

Applying the same operators in the vertical components of motion vectors, we have other four counts of neighboring triples, which are $A^{\rightarrow}_{p,q}(\mathbf{MVY}_t)$,$A^{\nearrow}_{p,q}(\mathbf{MVY}_t)$, $A^{\uparrow}_{p,q}(\mathbf{MVY}_t)$ and $A^{\nwarrow}_{p,q}(\mathbf{MVY}_t)$ respectively.

Considering the temporal correlation, which can be proved by the observation that the same object among adjacent frames is usually threaded by the same motion trajectory, we define the second-order difference array of the horizontal components for a temporal direction as shown in Fig. 4.

$$dx^{\bullet}_{i,j,t}(\mathbf{MVX}_t) = mvx_{i,j,t} - mvx_{i,j,t-1} \tag{17}$$

$$dx^{\bullet}_{i,j,t-1}(\mathbf{MVX}_{t-1}) = mvx_{i,j,t-1} - mvx_{i,j,t-2} \tag{18}$$

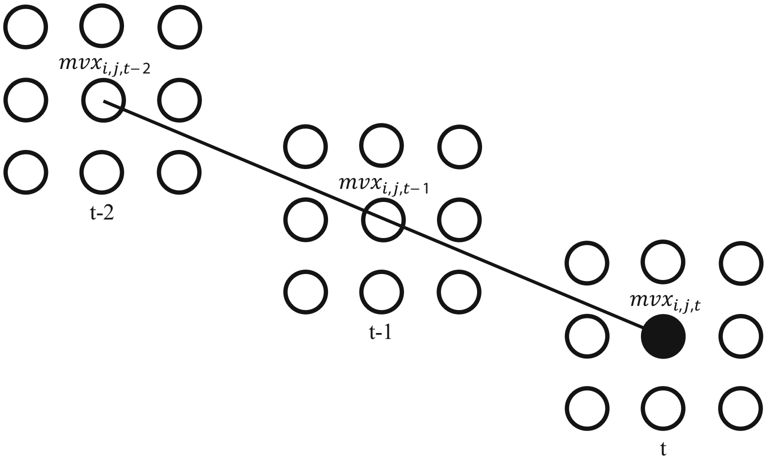


Fig. 4 Difference array used for calculating the statistical distribution change using the temporal correlation

Then we define the count of neighboring triples of horizontal components $\{mvx_{i,j,t}, mvx_{i,j,t-1}, mvx_{i,j,t-2}\}$ with differences $mvx_{i,j,t} - mvx_{i,j,t-1} = p$ and $mvx_{i,j,t-1} - mvx_{i,j,t-2} = q$ as

$$A^{\bullet}_{p,q}(\mathbf{MVX}_t) = \sum_{i=1}^{H}\sum_{j=1}^{W}\left[dx^{\bullet}_{i,j,t} = p, dx^{\bullet}_{i,j,t-1} = q\right] \tag{19}$$

Applying the second-order difference array for a temporal direction in vertical components, $A^{\bullet}_{p,q}(\mathbf{MVY}_t)$ can also be computed. For decreasing computing complexity, we limit the both changes of horizontal components and vertical components at the same time during data embedding, so the modified motion vector $\mathbf{mv}'_{i,j,t}$ is chosen from the candidate modified motion vectors set $\mathbf{CMV}_{i,j,t}$ using STCs-based ±1 embedding:

$$\begin{aligned}\mathbf{mv}'_{i,j,t} \in \mathbf{CMV}_{i,j,t} = \{&(mvx_{i,j,t}, mvy_{i,j,t} - 1),\ (mvx_{i,j,t} - 1, mvy_{i,j,t}),\\ &(mvx_{i,j,t}, mvy_{i,j,t} + 1),\ (mvx_{i,j,t} + 1, mvy_{i,j,t})\}\end{aligned} \tag{20}$$

Therefore, the count can be written in a simple way as $A^{d}_{p,q}(\mathbf{MV}_t)$, in which $d \in \{\rightarrow, \nearrow, \uparrow, \nwarrow, \bullet\}$. Let $\mathbf{mv}'_{i,j,t}\mathbf{MV}_{\sim i,j,t}$ be the motion vector field obtained from the tth frame whose the (i, j)th motion vector $\mathbf{mv}_{i,j,t}$ has been replaced with $\mathbf{mv}'_{i,j,t}$. We define the statistical distribution change (SDC) of the (i, j)th motion vector modification in the tth frame by

$$\mathrm{SDC}_{i,j,t} = \sum_{\substack{p,q \in \{-128, ..., 128\} \\ d \in \{\rightarrow, \nearrow, \uparrow, \nwarrow, \bullet\}}} \omega_{p,q} \left| A^{d}_{p,q}(\mathbf{MV}_t) - A^{d}_{p,q}(\mathbf{mv}'_{i,j,t}\mathbf{MV}_{\sim i,j,t}) \right| \tag{21}$$

where $\omega_{p,q} = 1/(1 + \sqrt{p^2 + q^2})$ is heuristically defined weights and the search range of motion vectors is set as 16 using the quarter-pixel precision. In (21), the smaller p^2 and q^2 corresponds to the larger weight because the changes made in motion vectors of blocks with similar motion trends tend to be highly detectable.

3.3 Prediction error change of motion vectors

In video coding, the modification of motion vector will make the associated prediction error changed. Steganalyzers can utilize the motion vector reversion phenomenon during video recompression in [4].

After motion estimation is performed on the current coded block $\mathbf{C}$ with size $b \times b$, the differential signal $\mathbf{D}$ between the current coded block $\mathbf{C}$ and the corresponding best reference block $\tilde{\mathbf{P}}$ is denoted as $\mathbf{D} = \mathbf{C} - \tilde{\mathbf{P}}$. Subsequently, DCT, quantization and entropy coding are performed on $\mathbf{D}$. During the first compression, we denote the reconstructed block as $\tilde{\mathbf{C}} = \tilde{\mathbf{P}} + \tilde{\mathbf{D}}$, where $\tilde{\mathbf{D}}$ is the reconstruction of $\mathbf{D}$ after inverse quantization and IDCT.

We suppose the original motion vector $\mathbf{mv}$ of the current block $\mathbf{C}$ has been modified to $\mathbf{mv}'$ and the modified differential signal is calculated as $\mathbf{D}' = \mathbf{C} - \tilde{\mathbf{P}}'$ instead of $\mathbf{D}$, which $\tilde{\mathbf{P}}'$ is the corresponding reference block after modifying the original motion vector. Thus, we denote the reconstructed block as $\tilde{\mathbf{C}}' = \tilde{\mathbf{P}}' + \tilde{\mathbf{D}}'$.

When we perform video recompression without data embedding, motion estimation is conducted on the reconstructed block using the measure SAD. Therefore, two prediction errors $\hat{e}_1 = \mathrm{SAD}(\tilde{\mathbf{C}}, \tilde{\mathbf{P}})$ and $\hat{e}_2 = \mathrm{SAD}(\tilde{\mathbf{C}}', \tilde{\mathbf{P}}')$ will be calculated for comparison as

$$\begin{aligned}\hat{e}_1 &= \textstyle\sum_{1\le i,j\le b} |\tilde{c}_{i,j} - \tilde{p}_{i,j}| \\ &= \textstyle\sum_{1\le i,j\le b} |(\tilde{p}_{i,j} + \tilde{d}_{i,j}) - \tilde{p}_{i,j}| \\ &= \textstyle\sum_{1\le i,j\le b} |(\tilde{p}_{i,j} + d_{i,j}) - \tilde{p}_{i,j} + (\tilde{d}_{i,j} - d_{i,j})| \\ &= \textstyle\sum_{1\le i,j\le b} |c_{i,j} - \tilde{p}_{i,j} + \triangle d_{i,j}|\end{aligned} \tag{22}$$

and similarly

$$\begin{aligned}\hat{e}_2 &= \textstyle\sum_{1\le i,j\le b} |\tilde{c}'_{i,j} - \tilde{p}'_{i,j}| \\ &= \textstyle\sum_{1\le i,j\le b} |(\tilde{p}'_{i,j} + \tilde{d}'_{i,j}) - \tilde{p}'_{i,j}| \\ &= \textstyle\sum_{1\le i,j\le b} |(\tilde{p}'_{i,j} + d'_{i,j}) - \tilde{p}'_{i,j} + (\tilde{d}'_{i,j} - d'_{i,j})| \\ &= \textstyle\sum_{1\le i,j\le b} |c_{i,j} - \tilde{p}'_{i,j} + \triangle d'_{i,j}|\end{aligned} \tag{23}$$

Bellifemine et al. [2] had pointed out that the 2D-DCT cofficients of the differential signal tend to be less correlated if the motion estimation is used. Thus, the distribution of the cofficients $\mathbf{Y} = \mathrm{DCT}(\mathbf{D})$ can be well modeled with the Laplacian probability distribution [13]. The difference between the original value and its reconstruction has zero mean, i.e., $E[\triangle d] = E[\tilde{d} - d] = 0,\ \tilde{d} \in \tilde{\mathbf{D}},\ d \in \mathbf{D}$. Therefore, the expectations of $\hat{e}_1$ and $\hat{e}_2$ can be estimated as

$$\begin{aligned}E[\hat{e}_1] &= \textstyle\sum_{1\le i,j\le b} E[|c_{i,j} - \tilde{p}_{i,j} + \triangle d_{i,j}|] \\ &= \textstyle\sum_{1\le i,j\le b} |c_{i,j} - \tilde{p}_{i,j}| \\ &= \mathrm{SAD}(\mathbf{C}, \tilde{\mathbf{P}})\end{aligned} \tag{24}$$

and similarly

$$\begin{aligned}E[\hat{e}_2] &= \textstyle\sum_{1\le i,j\le b} E[|c_{i,j} - \tilde{p}'_{i,j} + \triangle d'_{i,j}|] \\ &= \textstyle\sum_{1\le i,j\le b} |c_{i,j} - \tilde{p}'_{i,j}| \\ &= \mathrm{SAD}(\mathbf{C}, \tilde{\mathbf{P}}')\end{aligned} \tag{25}$$

Since the inequation $\mathrm{SAD}(\mathbf{C}, \tilde{\mathbf{P}}) < \mathrm{SAD}(\mathbf{C}, \tilde{\mathbf{P}}')$ holds during the first compression, we have $E[\hat{e}_1] < E[\hat{e}_2]$. It means that during recompression the current block will choose $\tilde{\mathbf{P}}$ for better prediction instead of $\tilde{\mathbf{P}}'$. For a block whose motion vector has been modified, its motion vector has an inclination to revert to the original motion vector during recompression.

Based on the motion vector reversion phenomenon, larger values in motion vector shift distance and larger changes in associated prediction errors indicate a larger probability that the motion vector has been once modified. Therefore, we should consider the prediction error change (PEC) as the other important factor in designing the distortion function. The prediction error change (PEC) can be denoted as

$$\mathrm{PEC}_{i,j,t} = |e_{i,j,t}(\mathbf{mv}_{i,j,t}) - e_{i,j,t}(\mathbf{mv}'_{i,j,t})| \tag{26}$$

where $e_{i,j,t}(\mathbf{mv}_{i,j,t})$ is the prediction error associated with the (i, j)th motion vector in the tth frame.

From this point of view, any modifications of motion vectors with large prediction error changes should lead to larger embedding impact because there are more obvious reversion samples the warden can use for training the steganalyzer. Meanwhile, according to (7), the better coding quality can be obtained if we keep smaller prediction error changes after data embedding. Considering the above analysis, in order to reduce the embedding impact, we need to embed the message bits into those motion vectors which are associated with small values of the prediction error change.

3.4 Distortion function definition

According to the framework of minimal-distortion steganography in Section 2, the embedding impact is captured by a non-negative additive distortion measure. In this paper, we define a distortion sum D of all motion vectors in the tth frame in the form

$$D(\mathbf{MV}_t, \mathbf{MV}'_t) = \sum_{i=1}^{H} \sum_{j=1}^{W} \rho_{i,j,t}(\mathbf{mv}_{i,j,t}, \mathbf{mv}'_{i,j,t}) \tag{27}$$

where $\mathbf{MV}_t$ is the cover motion vector field in the tth frame, and $\mathbf{MV}'_t$ is the corresponding stego motion vector field. H and W are the height and the width of the frame in the unit of the block respectively. The variable $0 \leq \rho_{i,j,t} \leq \infty$ is our designed distortion function expressing the multi-level embedding impact of the (i, j)th motion vector modification in the tth frame in the unit of the block. Combining the above two considered factors, the distortion function is defined as

$$\rho_{i,j,t}(\mathbf{mv}_{i,j,t}, \mathbf{mv}'_{i,j,t}) = \begin{cases} \infty & \mathbf{mv}_{i,j,t} = \mathbf{0} \\ \mathrm{SDC}_{i,j,t} \times (\mathrm{PEC}_{i,j,t} + \alpha)^{\beta} & \text{otherwise} \end{cases} \tag{28}$$

As seen from (28), the embedding distortion is controlled by SDC and PEC simultaneously except for the zero motion vector. The zero motion vector is regarded as "wet" element, which is prevented from modification by defining distortion as ∞. Additionally, the motion vector should also be prohibited for data embedding by defining distortion as ∞ if the associated modified motion vector is out of the motion search range. The experimental parameter $0 \leq \beta \leq 1$ is used to control the embedding impact caused by PEC since it is rather difficult to distinguish the contributions of SDC and PEC by theoretical analysis, which is revisited in the following experiment section. Meanwhile, the parameter α can be selected as a relatively small positive constant to ensure the embedding distortion keeps positive when modifying the motion vector, e.g., the value 1 in case $\mathrm{PEC}_{i,j,t}$ is zero.

4 Implementation of steganographic method

As the distortion function is given, let us introduce the practical implementation of proposed steganographic method using two-layered STCs-based ±1 embedding.

Suppose the total frame number of the input video sequence[1] is N. I-frames are coded as usual without data embedding. In our proposed steganographic method, there are three stages for completing the video coding and data embedding scheme.

Firstly, we execute the motion estimation step for all blocks in the tth frame to obtain the total motion vectors constituting the motion vector field $\mathbf{MV}_t$ and the associated prediction error matrix $\mathbf{E}_t$. Then we use the designed distortion function to define the embedding distortion for every motion vector in the tth frame and store defined distortion. We execute the step for the next $(t + 1)$th frame to define the distortions for remaining motion vectors

[1] The video sequence can be a series of raw images or a series of decompressed images obtaining from the coded bit stream.

and repeat this step until distortions of the total motion vectors in N frames have been defined. Thus, the distortion definition stage has finished.

Secondly, we accumulate the motion vectors of N frames together and apply the ±1 embedding with two-layered STCs [10] for data embedding to generate the modified motion vector field $\mathbf{MV}'_t$ for $1 \le t \le N$ which is the data embedding stage.

Thirdly, we use the modified motion vector field $\mathbf{MV}'_t$ to compute the differential signals for all blocks again in the tth frame. Then all blocks are coded using the modified motion vectors to complete the coding for the tth frame and repeat this step for the next $(t+1)$th frame until all frames of the video sequence have been coded into the bit stream.

Algorithm 1 Embedding Distortion Definition

```
Input the video sequence $\mathbf{F}_t, 1 \le t \le N$
Output the original motion vector field $\mathbf{MV}_t, 1 \le t \le N$ and the defined distortion matrix
    $\mathbf{D}_t = \{\rho_{i,j,t}\}, 1 \le t \le N$
initialize $t = 1$;
for each frame in the $N$ frames do
    if $\mathbf{F}_t$ is an I-frmae then
        set $\mathbf{MV}_t$ and $\mathbf{D}_t$ as NULL;
    else
        obtain the motion vector field $\mathbf{MV}_t$ and the prediction error matrix $\mathbf{E}_t$ through
        motion estimation;
        for each block $(i, j)$ in frame $\mathbf{F}_t$ do
            traversal each $\mathbf{mv}'_{i,j,t} \in \mathbf{CMV}_{i,j,t}$ in Eq. (20) and define distortions using
            $\rho_{i,j,t}(\mathbf{mv}_{i,j,t}, \mathbf{mv}'_{i,j,t})$ in Eq. (28);
        end for
        store the defined distortions in $\mathbf{D}_t$;
    end if
end for
```

Algorithm 2 Data Embedding

```
Input the binary message sequence $\mathbf{m}$ with length $l$, the original motion vector field $\mathbf{MV}_t, 1 \le$
    $t \le N$, the defined distortion matrix $\mathbf{D}_t = \{\rho_{i,j,t}\}, 1 \le t \le N$ and the number of P-frames
    $N_P$
Output the modified motion vector field $\mathbf{MV}'_t, 1 \le t \le N$
calculate the cover length $n = H \times W \times N_P \times 2$;
calculate the embedding rate $r = l/n$;
construct the cover vector $\mathbf{x}$ and $\mathbf{y}$ containing horizontal components and vertical compo-
    nents respectively form $\mathbf{MV}_t, 1 \le t \le N$;
construct the distortion vector $\mathbf{Dx}$ and $\mathbf{Dy}$ containing defined distortions caused by hor-
    izontal component modification and vertical component modification respectively from
    $\mathbf{D}_t, 1 \le t \le N$;
with $\mathbf{m}$, $\mathbf{x}$ and its corresponding distortion vector $\mathbf{Dx}$ as input parameters, embed data in
    $\mathbf{x}$ using ±1 two-layered STCs with embedding rate $r/2$;
for each $x_i$ in $\mathbf{x}$ do
    if $x_i$ is modified during data embedding then
        set the corresponding elememt $Dy_i$ in distortion vector $\mathbf{Dy}$ as $\infty$;
    end if
end for
with $\mathbf{m}$, $\mathbf{y}$ and its corresponding distortion vector $\mathbf{Dy}$ as input parameters, embed data
    in $\mathbf{y}$ using ±1 two-layered STCs with embedding rate $r/2$;
construct the modified motion vector field $\mathbf{MV}'_t, 1 \le t \le N$;
```

The modified coding scheme is compliant with the coding standard because we do not change the standard coding algorithms but only adjust the order in a reasonable way. Algorithm 1 describes the necessary steps for defining the embedding distortion on the motion vectors. The distortion matrix $\mathbf{D}_t = \{\rho_{i,j,t}\}, 1 \leq t \leq N$ contains the defined embedding distortion of each motion vector in motion vector field $\mathbf{MV}_t$, $1 \leq t \leq N$.

Algorithm 2 describes the necessary data embedding step, in which the embedding rate r is measured by the average embedded bits per motion vector (*bpmv*). In Step 4, **Dx** containing defined distortions caused by horizontal component modification and **Dy** containing defined distortions caused by vertical component modification are obtained from $\mathbf{D}_t = \{\rho_{i,j,t}\}, 1 \leq t \leq N$. In Step 5 and Step 11, the two-layered STCs are implemented by using the STCs toolbox with $h = 10$ [11]. The Step 8 is conducted to limit the both changes of motion vector components simultaneously.

Algorithm 3 Video Coding Using Modified Motion Vectors

```
Input the modified motion vector field $\mathbf{MV}'_t, 1 \leq t \leq N$
Output the coded bit stream of $N$ frames.
initialize $t = 1$;
for each frame in the $N$ frames do
    if $\mathbf{F}_t$ is an I-frame then
        encode $\mathbf{F}_t$ as usual using intra prediction;
    else
        encode $\mathbf{F}_t$ using $\mathbf{MV}'_t$;
    end if
end for
```

After data embedding, we use the modified motion vectors to finish video coding and generate the coded bit stream as described in Algorithm 3.

Algorithm 4 Data Extraction

```
Input the coded bit stream of $N$ frames
Output the extracted binary message sequence $\mathbf{m}$
decode the coded bit stream to obtain the motion vector field $\mathbf{MV}'_t, 1 \leq t \leq N$
STCs decoding is used to extract the binary message sequence $\mathbf{m}$
```

The message receiver can easily extract messages by matrix computing using STCs in the proposed steganographic method. Algorithm 4 describes the data extraction processing when the message receiver has obtained the video coded bit stream of video sent by the steganographer.

5 Experimental results and analysis

5.1 Experimental setup

5.1.1 Test sequences

As shown in Fig. 5, a video database containing 18 commonly used test sequences [29] in 4:2:0 YUV format is used in our experiments. Two of them have the resolution of 176 × 144 the remains have the resolution of 352 × 288. The detailed description of all test sequences is given in Table 1 in which each sequence is divided into 60-frame subsequences without overlapping and the total number of subsequences sums up to 116. There are two object motion types in the test sequences, in which the class A means that the objects are moving but the camera is relatively still and the class B means that the objects and the camera are both moving.

Fig. 5 Video sequences used in our experiments

5.1.2 Training and classification

The proposed video steganographic method has been integrated into the Joint Model (JM) version 10.2 [24] of H.264/AVC. The quantization parameter (QP) 28 is used to encode each subsequence. Besides, the motion vector magnitude-based steganographic method proposed by Xu et al. [27], the motion vector prediction error-based steganographic method proposed by Aly [1] and the steganographic method using perturbed motion estimation proposed by Cao et al. [3] are also implemented for performance comparison.

Two typical steganalysis algorithms are used to compare the security performance of the above four steganographic methods. One is Su et al.'s [23] algorithm (denoted by S1)

Table 1 Description of video sequences in detail

Sequence	Number of frames	Resolution	Object motion
Bus	150	352 × 288	B
Carphone	381	176 × 144	A
City	300	352 × 288	B
Coastguard	300	352 × 288	B
Crew	300	352 × 288	B
Flower	250	352 × 288	B
Football	260	352 × 288	B
Foreman	300	352 × 288	A
Harbour	300	352 × 288	A
Highway	2000	352 × 288	B
Ice	240	352 × 288	A
Mobile	300	352 × 288	B
Paris	1065	352 × 288	A
Soccer	300	352 × 288	B
Stefan	90	352 × 288	B
Suzie	150	176 × 144	A
Templete	260	352 × 288	B
Waterfall	260	352 × 288	B

using features derived from the histogram of motion vector component difference and the other one is Cao et al.'s [4] algorithm (denoted by S2) exploiting the motion vector reversion phenomenon during recompression. As described in these staganalysis algorithms, we use a fixed-size sliding window to scan each subsequence without overlapping, and the steganalytic features representing the clean or stego compressed videos are extracted from the frames within the window. The length of the sliding window in our experiments is set as 6.

All subsequences are compressed by JM 10.2 with same setting to produce the class of clean videos. On the other hand, for a given steganographic method, all subsequences are compressed with random messages embedding to create the class of stego videos. The video database consisting of all subsequences is evenly divided into a training set and a testing set respectively and the ensemble classifier [19] is employed to the steganalytic features because it enables fast training with comparable performance to the much more complex support vector machine (SVM). As the primary objective in steganography, the security performance is evaluated using the minimum average classification error probability such that:

$$P_E = \min_{P_{FA}}(P_{FA} + P_{MD}(P_{FA}))/2, \tag{29}$$

where P_{FA} and P_{MD} are the false-alarm probability and the missed-detection probability respectively.

Note that one steganographic method is broken as long as there exists one steganalytic algorithm that can detect it with a high accuracy rate. Therefore, we introduce another measurement $Global\_P_E$ to depict the comprehensive undetectable ability of the steganographic method.

$$Global\_P_E = \min_{i \in \mathbf{S}} P_{Ei} \tag{30}$$

where **S** represents the set of used steganalysis algorithms. P_{Ei} is the value of P_E under the attack of the ith steganalysis algorithm.

5.2 Testing contributions of SDC and PEC

In this section, we will test the contributions of SDC and PEC in our distortion function. A series of experiments corresponding to different values of β are conducted. For an illustration, β is selected in the range of [0, 1] with a step of 0.1 with $\alpha = 1$. The detection result against S1 and S2 using different parameter values is illustrated in Fig. 6. The embedding rates of 0.2 $bpmv$ and 0.3 $bpmv$ are selected respectively. With the increasing of parameter β, P_E under the attack of S1 decreases monotonously at the given embedding rates. However, P_E under the attack of S2 does not exhibit the monotonous trend from which we can see that PEC can help to resist the motion vector reversion phenomenon-based steganalysis. We can conclude that the contributions of the two factors against different staganalysis algorithms are distinguishing. For improving the comprehensive undetectable performance, we select β as 0.4 for our proposed distortion function in the following experiment.

5.3 Steganalysis results

The steganalysis results at five embedding rates are listed in Table 2, in which the bold digits denote the values of $Global\_P_E$. As known, the higher value of P_E is achieved, the better security performance is obtained, and P_E equal to 0.5 means perfect undetectability. For

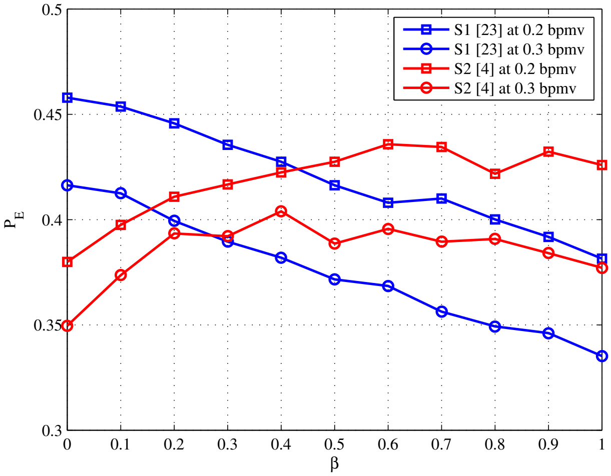


Fig. 6 Detection result against S1 [23] and S2 [4] corresponding to different parameter values

better depicting the security performance comparision, we illustrate the steganalysis result as follows.

Figure 7 shows the detection result against the steganalysis algorithm S1 [23]. S1 extracts a 12-dimension feature from one P-frame utilizing the center of mass (COM) of histogram of motion vector component difference. S1 is designed based on the basic hypothesis for steganalysis that some statistical characteristics of the cover object will change in data embedding process. It can been seen that the proposed method outperforms other three steganographic methods in security performance across all embedding rates. In motion vectors-based steganography, the spatial-temporal correlation of motion vectors should be jointly considered to design the steganographic rule. However, Xu et al.'s and Aly's methods do not combine the correlation in spatial-temporal domain so that the undetectable performance is relatively weaker than our proposed method. Cao et al.'s method does not perform well against S1 for it pays more attention to prediction error change in the distortion model but neglects preserving the correlation of motion vectors despite that wet paper codes are used.

Table 2 Minimum average classification error probability (%) of Su et al.'s [23] steganalysis algorithm (S1) and Cao et al.'s [4] steganalysis algorithm (S2) with different embedding rates (ER) and values in boldface denote the $Global\_P_E$

	Xu et al. [27]		Aly [1]		Cao et al. [3]		Proposed	
ER (*bpmv*)	S1	S2	S1	S2	S1	S2	S1	S2
0.1	42.98	**38.47**	43.81	**13.35**	**38.51**	38.95	47.09	**43.77**
0.2	37.93	**30.68**	36.78	**6.74**	**28.38**	32.63	43.30	**42.56**
0.3	35.09	**28.03**	30.59	**4.41**	**23.18**	29.21	**38.09**	40.20
0.4	30.87	**17.46**	28.54	**3.90**	**14.34**	23.12	**34.36**	35.82
0.5	27.97	**14.91**	24.23	**3.19**	**9.55**	19.76	**31.55**	32.79

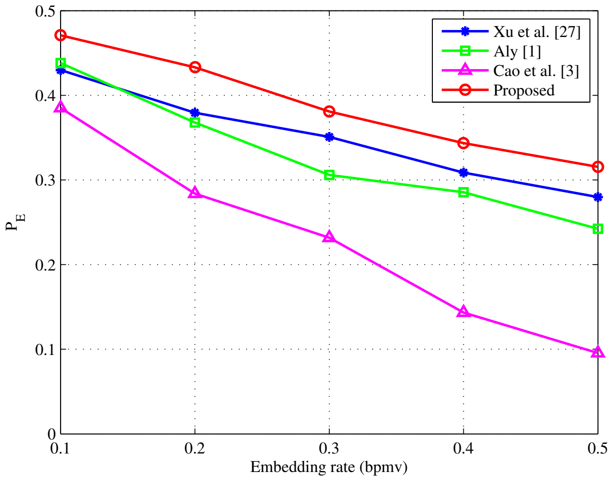


Fig. 7 Detection result against Su et al.'s [23] steganalysis algorithm S1

The detection result on resisting the steganalysis algorithm S2 [4] is shown in Fig. 8. In steganalysis algorithm (S2), a 15-dimension feature is extracted from one P-frame utilizing the motion vector reversion phenomenon. The shift distances in the prediction error change before and after recompression are calculated to construct features. The security performances of the other three methods fluctuate obviously compared with the result on resisting the steganalysis algorithm S1, while the proposed method still performs best. Cao et al.'s method can achieve a similar security performance compared with our method at the low embedding rate, say 0.1 $bpmv$, but the gap gets larger when the embedding rate increases.

The comprehensive security performance on resisting both S1 and S2 is shown in Fig. 9. It can be seen that the proposed method significantly outperforms the other three

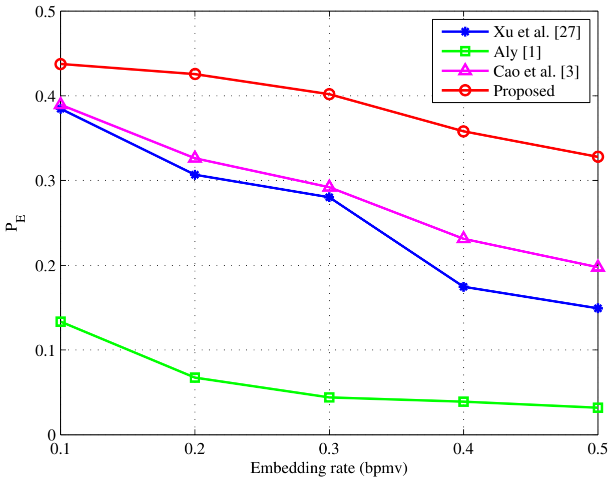


Fig. 8 Detection result against Cao et al.'s [4] steganalysis algorithm S2

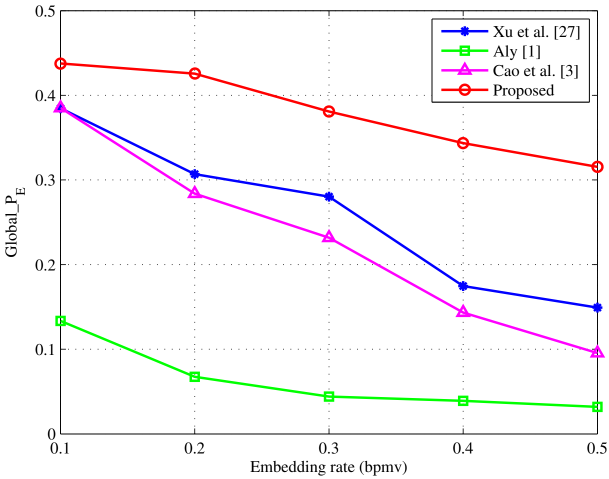


Fig. 9 Comprehensive undetectable performance against steganalysis algorithms S1 [23] and S2 [4]

steganographic methods. The $Global\_P_E$ of the proposed method at five embedding rates are all above 0.3. Steganalysis algorithm S1 and S2 attack the steganographic methods based on two different respects. S1 is designed using the statistical characteristics change in spatial-temporal domain and S2 utilizes the prediction error change from the motion vector reversion phenomenon. In our proposed method, the distortion function aims to preserve the spatial-temporal correlation of motion vectors and suppress the associated prediction error change during data embedding. Therefore, the proposed method can obtain the better comprehensive security performance than the other three methods. Usually we think the steganographic method is relatively secure if the minimum average classification error probability (PE) under steganalysis is above 40 %. As illustrated in Fig. 9, the secure embedding rate of our proposed steganographic method is about 2.5 *bpmv*. The secure embedding rates of other three steganographic methods are all less than 0.1 *bpmv*.

5.4 Impacts on visual quality and compression efficiency

Data embedding in motion vectors is associated with the process of video coding, so the video coding quality in terms of visual quality and compression efficiency should be considered. Moreover, it is a latent requirement that the video steganographic method should not cause severe visual quality degradation or coded bit rate increment.

In our emperiment, we use commonly adopted measurement PSNR to evaluate the visual quality of stego sequences. The PSNR (dB) is calculated by comparing the uncompressed video sequence before data embedding and the decoded reconstructed video sequence after data embedding. The visual quality comparison for all test sequences using four steganographic methods at 0.5 *bpmv* is listed in Table 3, in which the visual quality of the decoded reconstructed clean video sequence without data embedding is also given for reference. We can draw the conclusion that the visual quality degradation after data embedding is very slight because the differential signal between reference block and current coded block can be computed again using corresponding modified motion vector, which is an advantage of motion vector-based steganography.

Table 3 Comparison of average luma PSNR values (dB) using four steganographic methods at embedding rate 0.5 *bpmv*

Test Sequence	Cover	Xu et al. [27]	Aly [1]	Cao et al. [3]	Proposed
Bus	34.425	34.185	34.235	34.265	**34.295**
Carphone	36.295	36.053	36.022	36.090	**36.123**
City	34.676	34.412	34.432	34.470	**34.498**
Coastguard	34.410	34.212	34.234	34.260	**34.274**
Crew	36.948	36.798	36.790	36.812	**36.840**
Flower	34.520	34.270	34.363	34.437	**34.442**
Football	35.938	35.842	35.848	35.840	**35.860**
Foreman	36.080	35.814	35.794	35.862	**35.898**
Harbour	34.046	33.864	33.854	33.900	**33.908**
Highway	37.937	37.723	37.759	37.757	**37.769**
Ice	39.558	39.283	39.270	39.380	**39.382**
Mobile	33.616	33.454	33.430	33.508	**33.522**
Paris	35.518	35.375	35.243	**35.436**	35.435
Soccer	35.664	35.496	35.504	35.502	**35.540**
Stefan	35.340	35.070	35.030	35.210	**35.240**
Suzie	36.955	36.625	36.610	36.710	**36.735**
Templete	34.520	34.357	34.320	34.388	**34.410**
Waterfall	34.688	34.250	34.337	34.347	**34.395**

The average luma PSNR value (dB) in boldface means that the corresponding steganographic method can obtain the best visual quality of stego sequences

During data embedding, the original motion vector is manually modified so that the prediction precision of block will decrease, which causes the increment of the SAD of corresponding differential signal. Thus, the more message bits are embedded in motion vectors, the more coded bits will cost for encoding the differential signals. As a result, the number of coded bits used in the stego compressed video will increase compared with the clean compressed video. It should be noted that steganography is a covert communication process so the severe bit rate increment will deteriorate the communication efficiency, which is not appreciated. Therefore, we compare the average coded bit rate increment of all stego sequences using four steganographic methods at five embedding rates in Table 4, in which can been seen that the coded bit rate increase obviously with increasing of the embedding

Table 4 Comparison of average coded bit rate increment (%) using four steganographic methods at five embedding rates

Embedding Rate (*bpmv*)	0.1	0.2	0.3	0.4	0.5
Xu et al. [27]	2.79	4.64	6.50	8.85	10.16
Aly [1]	4.52	7.49	9.99	11.82	13.71
Cao et al. [3]	2.12	2.49	2.77	3.26	3.87
Proposed	**1.72**	**2.14**	**2.58**	**3.10**	**3.66**

The average coded bit rate increment (%) in boldface means that the corresponding steganographic method can keep the smallest coded bit rate increment rate, especially in Xu et al.'s method and Aly's method. Through comparison, our proposed method can keep the relatively less bit rate increment because the distortion function considering PEC helps STCs to choose modified motion vectors with less Lagrangian cost increment in (7) to some extent.

6 Conclusion and future work

In this paper, an effective video steganographic method exploiting the spatial-temporal correlation relying on the principle of minimizing the embedding impact has been proposed. We grasp two important factors which are the statistical distribution change (SDC) and the prediction error change (PEC) of motion vectors during data embedding respectively. Considering these two factors simultaneously, we design a distortion function for motion vectors to express the embedding impact effectively. Two-layered syndrome-trellis codes (STCs) are utilized to implement the practical embedding method. Experiments show that the proposed method achieves a significant improvement on resisting two typical steganalysis algorithms. Meanwhile, the performance in terms of the visual quality and the compression efficiency in our method also outperforms other existing motion vector-based steganographic methods.

It can be predicted that the more effective distortion function can be obtained by exploiting the spatial-temporal correlation more deeply. As the possible future work, we seek for the optimization for our method to express the embedding impact more accurately. Meanwhile, the research against the further steganalysis using high-dimension features which are extracted from the co-occurrence matrices of motion vectors and differential signals can be also attempted.

Acknowledgments This work was supported in part by the Natural Science Foundation of China under Grant 61170234 and Grant 60803155, and in part by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant XDA06030601.

References

1. Aly H (2011) Data hiding in motion vectors of compressed video based on their associated prediction error. IEEE Trans Inf Forensic Secur 6(1):14–18
2. Bellifemine F, Capellino A, Chimienti A et al (1992) Statistical analysis of the 2D-DCT coefficients of the differential signal for images. Signal Process Image Commun 4(6):477–488
3. Cao Y, Zhao X, Feng D, Sheng R (2011) Video steganography with perturbed motion estimation. In: Proceedings of 13th international workshop information hiding, lecture notes in computer science, vol 6958, pp 193–207
4. Cao Y, Zhao X, Feng D (2012) Video steganalysis exploiting motion vector reversion-based features. IEEE Signal Process Lett 19(1):35–38
5. Crandall R (1998) Some notes on steganography. Steganography mailing list. http://os.inf.tu-dresden.de/westfeld/crandall
6. Fang D-Y, Chang L-W (2006) Data hiding for digital video with phase of motion vector. In: Proceedings of international symposium circuits and systems (ISCAS), pp 1422–1425
7. Filler T, Fridrich J (2010) Gibbs construction in steganography. IEEE Trans Inf Forensic Secur 5(4):705–720
8. Filler T, Judas J, Fridrich J (2010) Minimizing embedding impact in steganography using trellis-coded quantization. In: Proceedings of SPIE, media forensics and security II, vol 7541, pp 05-01–05-14
9. Filler T, Fridrich J (2011) Design of adaptive steganographic schemes for digital images. In: Proceedings of SPIE, media watermarking, security and forensics III, vol 7880, pp 0F-01–0F-14

10. Filler T, Judas J, Fridrich J (2011) Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Trans Inf Forensic Secur 6(3):920–934
11. Filler T, Judas J, Fridrich J Syndrome-trellis codes toolbox, ver. 1.0, http://dde.binghamton.edu/download/syndrome/
12. Fridrich J, Goljan M, Lisonekm P, Soukal D (2005) Writing on wet paper. IEEE Trans Signal Process 53(10):3923–3935
13. Gormish MJ, Gill JT (1993) Computation-rate-distortion in transform coders for image compression. SPIE Vis Commun Image Process:146–152
14. Guo L, Ni J, Shi YQ (2012) An efficient JPEG steganographic scheme using uniform embedding. In: Proceedings of international workshop infornamtion forensics security (WIFS), pp 169–174
15. Holub V, Fridrich J (2012) Designing steganographic distortion using directional filters. In: Proceedings of international workshop information forensics security (WIFS), pp 234–239
16. Holub V, Fridrich J (2013) Digital image steganography using universal distortion. In: Proceedings of first international workshop information hiding and multimedia security, pp 59–68
17. Huang F, Huang J, Shi YQ (2012) New channel selection rule for JPEG steganography. IEEE Trans Inf Forensic Secur 7(4):1181–1191
18. Ker AD, Bas P, Bohme R et al (2013) Moving steganography and steganalysis from the laboratory into the real world. In: Proceedings of ftrst internatonal workshop information hiding and multimedia security, pp 45–58
19. Kodovsky J, Fridrich J, Holub V (2012) Ensemble classifiers for steganalysis of digital media. IEEE Trans Inf Forensic Secur 7(2):432–444
20. Pevný T, Filler T, Bas P (2010) Using high-dimensional image models to perform highly undetectable steganography. In: Proceedings of 12th international workshop information hiding, lecture notes in computer science, pp 161–177
21. Sachnev V, Kim HJ (2012) Modified BCH data hiding scheme for JPEG steganography. EURASIP J Adv Signal Proc 89:1–10
22. Shanableh T (2012) Data hiding in MPEG video files using multivariate regression and flexible macroblock ordering. IEEE Trans Inf Forensic Secur 7(2):455–464
23. Su Y, Zhang C, Zhang C (2011) A video steganalytic algorithm against motion-vector-based steganography. Signal Process 91(8):1901–1909
24. The H.264/AVC joint model JM), ver. 10.2, http://iphome.hhi.de/suehring/tml/download/old_jm
25. Wiegand T, Sullivan GJ, Bjontegaard G, Luthra A (2003) Overview of the H.264/AVC video coding standard. IEEE Trans Circuits Syst Video Technol 13(7):560–576
26. Wiegand T, Schwarz H, Joch A et al (2003) Rate-constrained coder control and comparison of video coding standards. IEEE Trans Circuits Syst Video Technol 13(7):688–703
27. Xu C, Ping X, Zhang T (2006) Steganography in compressed video stream. In: Proceedings of international conference innovative computing, information and control, pp 803–806
28. Yang X, Zhao L, Niu K (2012) An efficient video steganography algorithm based on sub-macroblock partition for H.264/AVC. Adv Mater Res 433–440:5384–5389
29. YUV Video Sequences, http://trace.eas.asu.edu/yuv/index.html
30. Zhang X, Zhang W, Wang S (2007) Efficient double-layered steganographic embedding. Electron Lett 43:482–483
31. Zhang W, Zhang X, Wang S (2007) A double layered “plus-minus one” data embedding scheme. IEEE Signal Process Lett 14(11):848–851

Yuanzhi Yao received his B.S. degree in electronic engineering from Hefei University of Technology in 2011. He is currently pursuing the Ph.D. degree in electronic engineering in University of Science and Technology of China. His research interests include information hiding and video coding.

Weiming Zhang received his M.S. degree and Ph.D. degree in 2002 and 2005 respectively from the Zhengzhou Information Science and Technology Institute, Zhengzhou, China. Currently, he is an associate professor with the School of Information Science and Technology, University of Science and Technology of China. His research interests include information hiding and cryptography.

Nenghai Yu received his B.S. degree in 1987 from Nanjing University of Posts and Telecommunications, M.E. degree in 1992 from Tsinghua University and Ph.D. degree in 2004 from University of Science and Technology of China, where he is currently a professor. His research interests include multimedia security, multimedia information retrieval, video processing and information hiding.

Xianfeng Zhao received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2003. From 2003 to 2005, he was a postdoctoral fellow with the Data Assurance and Communication Security Center, Chinese Academy of Sciences (CAS), Beijing. From 2006 to 2011, he was an associate professor with the State Key Laboratory of Information Security (SKLOIS), Institute of Software, CAS, Beijing. Since 2012, he has been a professor with SKLOIS, which was moved to Institute of Information Engineering, CAS, Beijing, in 2012. He is a member of IEEE, China Computer Federation, and Chinese Association for Cryptologic Research. His research interests include information hiding and multimedia security.