Undetectable video steganography by considering spatio-temporal steganalytic features in the embedding cost function

Negin Ghamsarian & Morteza Khademi

Multimedia Tools and Applications
An International Journal

ISSN 1380-7501

Multimed Tools Appl
DOI 10.1007/s11042-020-08617-y

Springer

Multimedia Tools and Applications
https://doi.org/10.1007/s11042-020-08617-y

Undetectable video steganography by considering spatio-temporal steganalytic features in the embedding cost function

Negin Ghamsarian[1] · **Morteza Khademi**[1]

Received: 17 January 2019 / Revised: 2 November 2019 / Accepted: 2 January 2020 /
Published online: 14 March 2020

Abstract

The basic requirement of a steganography approach is security against steganalysis attacks. In other words, a steganography method is reliable as long as it withstands all of the known steganalysis approaches. In order to preserve the security of a steganography method, the statistical features of the embedded and the original media must be as close as possible. To achieve this goal, in this paper steganography is applied by introducing the following contributions. Firstly, a new method is suggested to find the most impalpable embedded motion vector (MV). Also, a novel modification cost function with respect to the MVs' intra-frame and inter-frame statistical differences before and after embedding is proposed for the syndrome-trellis coder. Furthermore, a pseudo-random generator is introduced for altering the arrangement of motion vectors which are used by the syndrome-trellis coder to improve its efficiency. Experimental results show that the proposed method is the most secure MV-based steganography scheme against the state-of-the-art video steganalysis methods as well as preserving other steganography measurements including imperceptibility and compression ratio. Moreover, the computational cost of the proposed scheme is far less than its main rival.

Keywords Video steganography · Information hiding · Motion vector · Motion estimation · H.264/AVC · Security · Syndrom-trellis code · Video compression · Blind steganalysis

1 Introduction

Since the rise of the Internet and wireless communications, protection of sensitive information against unwanted access or manipulation has been of prime concern. Cryptography

✉ Morteza Khademi
khademi@um.ac.ir

Negin Ghamsarian
negin@itec.aau.at

[1] Department of Electrical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

methods are employed to maintain information security. In these methods, the message content is converted to a meaningless one using a key and then transmitted through the communications channel. The main issue in cryptography schemes is that the eavesdropper may easily suspect the existence of the confidential message as soon as meaningless content is observed. In other words, cryptography does not hide the communication; whereas, in some cases, we aim to conceal the communication from the eavesdropper. The technique employed to achieve this is named steganography. Indeed, contrary to cryptography, steganography is the science of concealing the confidential information inside a cover media so that only the transmitter and the legitimate recipient have been aware of that.

The quality of a steganography method is evaluated by applying its measurements. The main steganography measurements are imperceptibility, capacity, robustness, and security. Imperceptibility evaluates the ability of a steganography approach to avoid visual degradation in the host media during embedding. Capacity means the number of bits that can be embedded in a cover media using a steganography technique[1]. Security and robustness of a steganography method are the measurements of its ability to resist the operations of passive and active wardens respectively. Security is calculated employing steganalysis attacks, and steganalysis is the procedure of detecting the presence of a hidden message in a suspicious media. Security against passive wardens' operations is the primary requirement of steganography which is usually known as undetectability. Robustness refers to the capability of a steganography method to preserve secret information despite information eliminating efforts of the warden [3, 13, 31].

Besides the above measurements, there are other important criteria such as compression ratio and computational cost [2]. Nowadays, almost all types of digital media are compressed to reduce the cost of transmission and the space required for storage. The compression ratio evaluates the effect of embedding with a steganography method on increasing the output bitrate of the cover media.

Almost all types of digital media such as text, image, audio, video and network protocol packets can be used as steganography hosts. Due to the high capacity of video and redundancy of its information, this media is a suitable host for embedding high volume secret messages.

Video steganography approaches lie in the following two categories: 2D methods and 3D methods [41]. The methods which embed a secret message in video frames independently and regardless of correlations among frames such as embedding on intra-prediction modes [14, 24, 25, 32, 37, 50] are considered as 2D video steganography. These techniques have no advantage over image steganography techniques except capacity, and can be detected by image steganalysis attacks. Any method which uses the third dimension of video to embed confidential information belongs to the 3D category. These methods are more secure against steganalysis techniques compared to 2D methods. The information hidden by 3D methods cannot be detected by image steganography methods due to using inter-frame components of video [4, 8, 41, 44]. The 3D methods include but not limited to embedding on DCT coefficients [26, 30, 33, 57], variable-length codes [33], quantization parameters [40, 46], inter prediction modes [27, 55], and motion vectors [1, 7, 9, 10, 16, 36, 48, 51, 54, 56]. Among all of 3D video steganography strategies, video steganography based on motion vectors is set to be the most secure technique against steganalysis schemes for the following reasons:

[1]There is, however, another definition of capacity from the information-theoretic viewpoint [34]

1) By changing motion vectors, the intra-frame and inter-frame statistics of video change indirectly and randomly.
2) There exist much lower correlations among neighboring MVs than neighboring pixels (For more explanation, please refer to Fig. 1 and Fig. 2 in [44]). As a result, a video containing manipulated MVs is highly misleading.
3) A majority of video steganalysis methods consider the embedding procedure as an additive noise; these attacks cannot detect MV-based steganography schemes [7].

Moreover, manipulating motion vectors leads to negligible distortion in the visual characteristics of the output video thanks to the motion compensation phase in the video codec algorithm. Hence, much attention nowadays has been focused on motion-vector based video steganography.

In this study, we propose an information hiding scheme on motion vectors of video. The main objective of our research is increasing the security of MV-based video steganography against the most powerful steganalysis algorithms. We also aim to achieve a high level of imperceptibility, computational cost, and compression ratio at the same time. Our main contributions and paper organization are as follows. Section 2 includes the related work on video steganography and steganalysis. Section 3 presents the basic concepts which will be later used in our proposed method. Section 4 contains the proposed method and the following contributions:

1) For the first time, a cost function is proposed based on the original and modified MVs' statistical differences in the side of steganalyzer and MVs' spatial correlations.
2) A novel method for finding proper MVs for manipulation is introduced based on a comparison between posterior statistics of original and altered MVs.
3) A pseudo-randomization scheme is proposed to increase the efficiency of the syndrome-trellis coder.
4) The proposed method is the most secure video steganography method against the outstanding recent video steganalysis approaches.

In Section 5, the experimental setup is introduced and the experimental results are demonstrated. Finally, discussion and conclusion are presented in Sections 6 and 7.

2 Related work on video steganography and steganalysis

MV-based video steganography involves two major problems: how to find the appropriate MVs for embedding, and how to alter the selected MVs so as to leave as less embedding evidence as possible. Up to this point, the evolution of MV-based video steganography techniques can be separated into three fundamental stages [56]. In the early generation, appropriate motion vectors for embedding were chosen using a predetermined threshold and a particular criterion such as MVs magnitude [16, 48, 54], or MVs' corresponding block prediction error [1]. Then, confidential information was embedded in the magnitude or the phase of the selected motion vectors. Because of improper MV selection and modification methods, these approaches failed to preserve secrecy against initial steganalysis techniques [8].

In order to enhance embedding efficiency (the number of confidential message bits embedded per unit modification [13]) and consequently acheive higher level of security, second-generation methods have applied error-correcting codes such as BCH codes [35],

Wet Paper Codes (WPC) [7, 10, 19, 20] or Syndrome-Trellis Codes (STC) [9, 10, 17, 51]. The mentioned approaches consider the embedding procedure as a source coding problem. In wet paper coding, we assume the source is like a memory with some damaged cells [29]. From the steganography point of view, defective cells correspond to the cover elements which lead to easier detectability in the side of steganalyzer after modification. In [7], the cost of modifying each motion vector is defined as the lowest mean square error (MSE) correspond to a motion vector in the whole search window, which has different Least Significant Bit (LSB) from the original MV. Regarding the number of message bits that must be embedded, a wet vector is then formed using the evaluated costs. Afterward, wet paper coding is applied to specify the MVs to be modified, and selected MVs are replaced by MVs with the lowest costs.

Unlike wet paper codes, all of the MVs can be employed in syndrome-trellis codes, although, they differ greatly in their level of reliability. The more reliable a cover element is, the lower cost its manipulation imposes. These codes aim to minimize the total cost of embedding for a particular payload. Because of its efficiency in distortion minimization and consequently security improvement, syndrome-trellis codes caused scientists to turn their attention from uniform embedding to content-adaptive steganography. Therefore, designing the state-of-the-art steganography methods in these days is limited to two problems: (1) finding the most accurate manipulation cost (distortion function) with respect to statistical detectability (2) finding the best method to manipulate selected cover elements so as to seem as innocent-looking as possible [21].

It is claimed that altering motion vectors turn them from locally optimal to non-optimal [8, 45]. Accordingly, reversion based features in [8] and AoSO feature set in [45] are exploited for classification. Also, in [43, 44, 47], detectors are designed based on the correlation between each MV and its neighbors. In order to increase security against these attacks, Cao et al. [9] proposed a cost function for altering MVs based on their associated uncertainty. In [51], the authors proposed a distortion function for STC by defining Statistical Distribution Change (SDC) and Prediction Error Change (PEC) of MVs.

Attempting to preserve local optimality during embedding procedure, [10] and [56] have formed a new generation of MV-based video steganography methods. It is demonstrated in [10] that tampered MVs can be locally optimal in SAD (Sum of Absolute Differences) sense at the receiver's side. So, the associated distortion scale of MVs is calculated according to their one-distant-locally-optimal neighbors. Next, a N_1-bits message is embedded using STC. Following that, a wet paper channel is evaluated with modified MVs in the STC stage being regarded as wet cells; and a N_2-bits message is embedded using WPC. In [56], the search area of MVs' locally-optimal neighbors depends on the maximum acceptable computational cost.

Although [7, 9, 10, 51, 56] are the most successful MV-based steganography methods until now, they still have some weaknesses. In [9], the cost of altering MVs is evaluated using various Lagrangian multipliers for compressing each block. Since just one quantization scale is applied in the MV selection stage of video codec which is available on the steganalyser's side, using other quantization scales in cost function seems to be useless. Moreover, the MV with minimum SAD is selected as modified MV, without taking the Lagrangian multiplier into consideration. In [51], due to lossy compression of associated prediction error block and using rate-distortion optimization in advanced video coding standards, there can be a nonlinear relationship between PEC and changes in steganalysis features. Therefore, a distortion function based on PEC is not rigorous. Furthermore, SDC cannot be computed in the case of variable-block-size video coding. On top of that, authors suggest accumulating the MVs of N successive

Table 1 Detector reliability of MVRB against our proposed method and rival steganography approaches (SA) with different motion estimation algorithms (ME), embedding rates (ER), and quantization parameters (QP).

Sequence	ME	FULL			HEX		
	QP	17	27	32	17	27	32
Akiyo		23	10	2	16	7	2
Carphone		26	15	7	16	10	4
Deadline		25	18	10	20	14	7
Foreman		29	16	8	19	11	5
Highway		10	2	1	6	1	1
Mobile		11	9	8	7	6	4
News		30	21	10	23	15	8
Silent		45	23	9	35	18	5

frames and then perform embedding on them altogether, while this strategy has two major flaws[2]. The main issue of [10] is the fact that we have to transmit either the rate N_1/N_2 or one of the numbers $N1$ or $N2$ for each P-frame as side information so that the message can be extracted in receiver's side. Therefore, we should embed this information in a contractual location in the video which leads to a substantial increase in embedding rate. Otherwise, we have to send the information through a secure channel. In [56], even though the idea of finding locally-optimal neighbors is admirable, no alternative way is proposed when there is not any locally-optimal neighbor in the SAD sense for an MV. In Table 1, the percentage of MVs without a locally-optimal neighbor is represented that confirms the aforementioned approach is not always implementable. Besides, Since [7, 9, 56] use the whole search window of video compression algorithm to find the best modified MV, the computational costs of these methods are relatively high. In addition to taking more computational cost, selecting modified MVs from a big search range may negatively affect the compression ratio. Moreover, because of information loss stages such as DCT quantization, an originally compressed video can contain some non-locally-optimal MV. As shown in Table 2, usually with decreasing the quantization scale, the percentage of non-locally-optimal MVs in a clear video increases from the steganalyser's perspective. Thus a locally-optimal MV is not always the most undetectable cloak.

3 General theory

3.1 Motion estimation and compensation

One part of the H.264 encoder for inter-frames is the selection of subdivisions using a cost function. In this stage, four motion estimation modes including full ME (one motion vector

[2]For two reasons, embedding should be conducted in each P-frame or B-frame separately and in the order of occurrence. Firstly, the cost of altering MVs in each frame depends on the coded reference frames. Therefore, it is obvious that embedding on MVs of N consecutive frames multiples the time and space complexity. On top of that, due to the fact that MVs of each frame are extremely dependent on the pixels' values of its reference frames, embedding in the reference frame not only lead to some changes in clean MVs of subsequent frames, but also may result in great changes in their corresponding embedding costs. Hence, the calculated costs in such schemes are not trustworthy. Instead, we should calculate embedding costs and conduct the embedding procedure in one frame, and then move on to the next frame.

Table 2 The average percentage of non-locally-optimal motion vectors according to the lagrangian multiplier from the receiver's point of view using H.264/AVC codec and different motion estimation algorithms (ME) and quantization parameters (QP) for QCIF sequences

Sequence	ME	FULL			HEX		
	QP	17	27	32	17	27	32
Akiyo		5	4	2	25	21	17
Carphone		16	11	8	65	46	36
Deadline		8	7	4	37	32	28
Foreman		19	16	12	75	61	49
Highway		20	19	8	75	56	30
Mobile		9	14	15	72	71	69
News		7	7	5	33	28	23
Silent		12	11	9	42	34	25

for the entire 16 × 16 macroblock), horizontal (two MVs, one per each 8 × 16 block), vertical (two MVs, one per each 16 × 8 block), and quaternary (four MVs, one per each 8 × 8 block) are considered for each 16 × 16 macroblock. The macroblock is divided based on the mode. Then for each block, the best MV (MV_{b_k}) is calculated applying (1) based on the rate-distortion-optimization (λ is Lagrangian multiplier, $\lambda_{ME} = \sqrt{\lambda_{mode}}$, $\lambda_{mode} = 0.85 \times 2^{(QP-12)/3}$, and R_{mv} is the number of bits required for transmitting the candidate MV). Finally, the Lagrangian cost of each mode is calculated and the best partitioning mode is selected using (2) (cfr_{mode} is the final bitrate of coefficients using the existing $mode$). If the quaternary ME is selected, selection of subdivisions is performed again for each of the four blocks (Fig. 1).

$$MV_{b_k} = \arg\min_{mv}[J_{b_k,mv}] = \arg\min_{mv}[SAD_{b_k,mv} + \lambda_{ME} \times R_{mv}] \tag{1}$$

$$Mode = \arg\min_{mode}[SSD_{b_k,MV,mode} + \lambda_{mode} \times cfr_{mode}] \tag{2}$$

$$SAD_{b_k,mv} = \sum_{x=X(b_k)}^{\substack{X(b_k)\\+BS_x(b_k)}} \sum_{y=Y(b_k)}^{\substack{Y(b_k)\\+BS_y(b_k)}} |F^{Org}_{x,y,t} - F^{Rec}_{x+mv_x,y+mv_y,t-1}| \tag{3}$$

$$SSD_{b_k,mv} = \sum_{x=X(b_k)}^{\substack{X(b_k)\\+BS_x(b_k)}} \sum_{y=Y(b_k)}^{\substack{Y(b_k)\\+BS_y(b_k)}} (F^{Org}_{x,y,t} - F^{Rec}_{x,y,t})^2 \tag{4}$$

$F^{Org}_{x,y,t}$ and $F^{Rec}_{x,y,t}$ in (3) and (4) refer to quantity of the pixel in location (x, y) in the tth original and recounstructed P-frame respectively. $X(b_k)$ and $Y(b_k)$ are the corresponding row and colomn of the pixel in top-left corner of kth block, and $BS_x(b_k)$ and $BS_y(b_k)$ are the width and height of kth block (Fig. 1).

3.2 Distortion minimization using syndrome-trellis codes

Assuming that we have N blocks in a P-frame, the MVs of these blocks are subjected to a LSB function to obtain $p = \{LSB_{mv_1}, LSB_{mv_2}, ..., LSB_{mv_N}\}$ (LSB_{mv_N} is the least

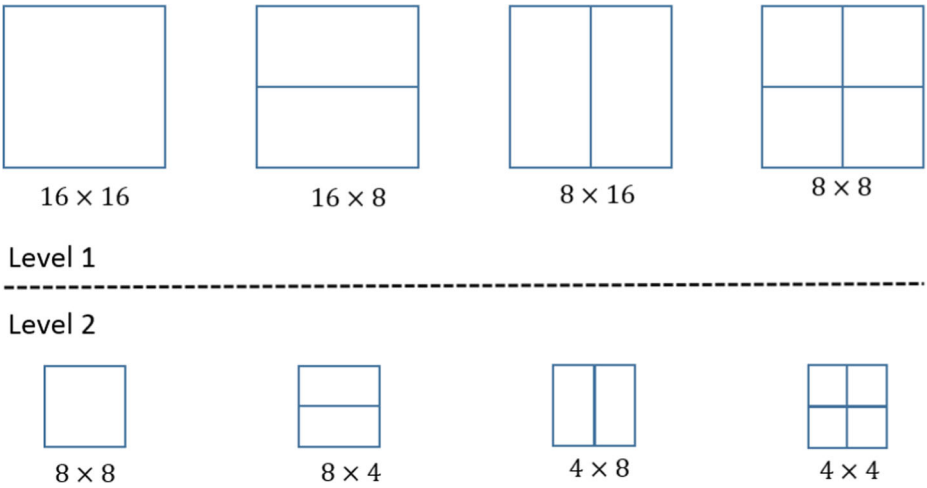


Fig. 1 Levels of motion estimation in H.264/AVC codec [38]

significant bit of sum of horizontal and vertical components of Nth MV). The syndrome-trellis coding begins by having the LSB vector (p), cost of modifying the MV of each block ($COST = \{Cost(mv_1), Cost(mv_2), ..., Cost(mv_N)\}$), the secret message bit stream (m), and a parity check matrix $H \in \{0, 1\}^{rN \times N}$ being previously shared between the transmitter and the legitimate recipient (r is the embedding rate). First, all p' vectors which satisfy (5) are obtained.

$$P'(m) = \{p' \in \{0, 1\}^N | p'H^T = m\} \tag{5}$$

Among all possible vectors p', the one that imposes the lowest cost of embedding is chosen as follows.

$$\widetilde{p} = \underset{p' \in P'(m)}{\arg\min} [\sum_{k=1}^{N} COST(k).(p(k) \oplus p'(k))] \tag{6}$$

Where $\oplus$ is the XOR (exclusive OR) function. With a simple comparison between p and $\widetilde{p}$, we will realize which MVs must be altered. In the recipient side, after obtaining MVs and forming the LSB vector $\widetilde{p} = \{LSB_{m\tilde{v}_1}, LSB_{m\tilde{v}_2}, ..., LSB_{m\tilde{v}_N}\}$, the message can be extracted by (7).

$$m = \widetilde{p}H^T \tag{7}$$

The syndrome trellis coder can access utmost $h \times w$ motion vectors for embedding one bit of confidential message. Usually, $6 \leq h \leq 15$, the quantity of which affects the speed and efficiency of the algorithm (The larger the h is, the slower the algorithm works); Because the time-and-space complexity for solving the trellis is $\mathcal{O}(2^h n)$ (n is the number of MVs in each frame). Also, w is determined by Embedding Rate ($1/(w + 1) \leq ER \leq 1/w$). For example if $h = 8$ and embedding rate equals $4/10$, so $w = 2$ and thus trellis coder has access to utmost 16 MVs (for more explanation, the reader is requested to refer to [17]). Since MVs are organized as the raster scan, accessible MVs for embedding each secret bit are close together. As it is depicted in Fig. 2, in overwhelming majority types of videos including videos with a fixed camera or a fixed background, or videos containing rigid moving objects (such as a moving car), MVs represent specific patterns. They are equal (in place of rigid objects), or equivalent to zero in large parts (in place of background).

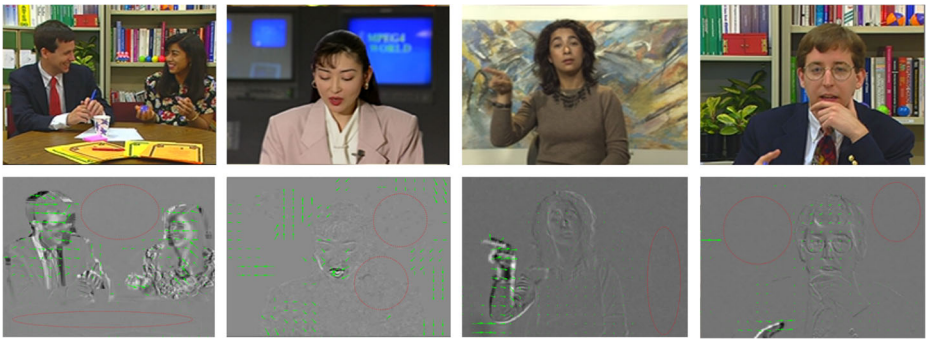


Fig. 2 MVs' patterns in background and rigid objects (red circles are representatives of the big areas in which all the motion vectors are equal to zero)

Therefore, their statistics including embedding costs are the same. So in these places, the performance of the Viterbi algorithm is likely to degrade.

Considering the above explanations, a novel arrangement of MVs is needed so as to increase the efficiency of syndrome-trellis coder.

3.3 Near-perfect steganalytic features

In [53], A set of steganalytic features is proposed supposing the original MVs of the received video should be locally optimal with respect to the Lagrangian multiplier. Since it can be evaluated by the steganalyzer, the Lagrangian multiplier can be used for feature extraction. After decompression, each reconstructed block is subjected to further motion estimation utilizing the obtained Lagrangian multiplier and eight MVs centering around the received MV. Then, four types of features are calculated based on the cost of the best MVs (MVs with the lowest cost) and received MVs. Type 1 feature set is based on the percentage of locally-optimal MVs in each of 9 positions; type 2 feature set is based on the cost difference between the received and the best MV in each position; type 3 and type 4 feature sets are such as type 1 and type 2 respectively, with one difference. The cost function in the first two types is based on SAD (Sum of Absolute Differences), whereas the cost function in the next two types is based on SATD (Sum of Absolute Transformed Differences).

Considering this successful strategy for MV-based steganography detection, the relationship between the Lagrangian costs of the original and altered MV should be as close as possible in the receiver's side in order that less convincing evidence of steganography can be obtained by attackers.

4 Propose method

Overview We aim to conceal the confidential information in MVs which cause the most similar spatio-temporal statistics of the resulting P-frame with that of the original one. In order to achieve the lowest possible output bitrate, the alternative MV is selected from the set of four nearest MVs to the original one. Because video coding is a lossy compression method, statistics of MVs differ before and after encoding. Steganalyzers can just access the encoded information available on the receiver's side. This suggests that the modified MV should be the one with the closest statistics to that of the original MV after encoding. Thus

in contrary to the state-of-the-art steganography methods, we exploit the posterior statistics of MVs in cost function and alternative MV selection procedure. Inspired by [53], we use steganalytic features partly based on local optimality to compute the most undetectable modified MVs and their corresponding modification costs. A syndrome-trellis coder is then used to improve embedding efficiency. The modification cost of each MV is composed of two costs: temporal and spatial cost. The temporal cost measures the difference between the steganalytic features of the alternative and original MV based on local optimality. This cost enforces the Syndrome-trellis-coder to select alternative MVs with the closest steganalytic features to that of the original ones. The spatial cost is used to ensure that the MVs belonging to the rigid objects or fixed backgrounds that have less indeterministic statistics will not be modified as far as possible. Figure 3 represents the block diagram of the proposed MV-based video steganography algorithm.

Embedding Procedure The process of embedding the secret message (encrypted information) and encoding is performed on each P-frame separately and in the order of occurrence.

Step 1. Original MVs Computing: At this stage, MVs in each frame, the type of blocks, and motion compensation blocks are obtained during video coding.

Step 2. Alternative MVs Computing: In this step, the best alternative MV is obtained using the following method:

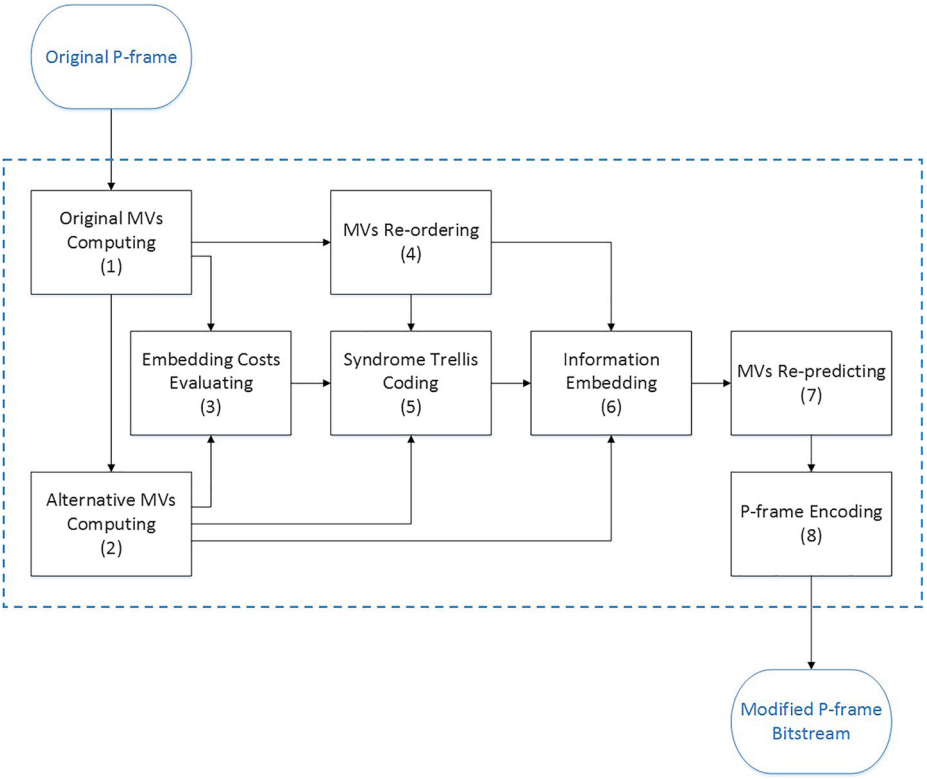


Fig. 3 Block diagram of the proposed steganography method

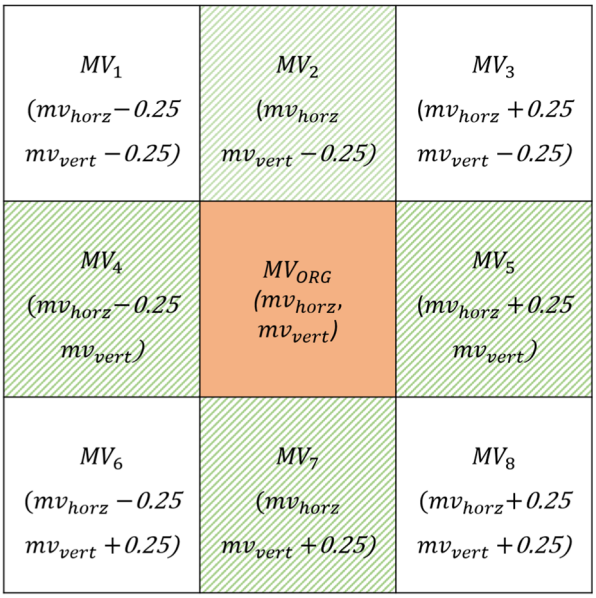


Fig. 4 Candidate alternative motion vectors

MVs which cause the best possible bitrate. In video codec standards, each motion vector is predicted by utilizing its previous encoded neighboring MVs; and the residual block is coded. Thus altering each MV affects the prediction of subsequent MVs and consequently bitrate. In order to prevent increasing the output bitrate, it is necessary to choose the embedded motion vector from a set of nearest possible MVs (according to the employed video coding standard, the smallest acceptable unit for MV alteration is 0.25). According to Fig. 4, among eight neighborhoods centered around the original MV, there are 4 MVs (shaded blocks) that can be used as the altered MV due to having different LSBs (LSBs of the sum of their horizontal and vertical components) from that of the original MV.

MVs which have the closest temporal statistics to that of the original MV from the steganalyzer's perspective. First, the block is compressed using each of these four MVs. Then, we put ourselves in the eavesdropper's shoes; so in each of these five cases (original and altered MVs), we decode the block's bitstream and compute the locally-optimal MV by having eight neighbors of the received MV, the Lagrangian cost function, and the reconstructed block. Afterwards, we compute the difference of Lagrangian costs (DoLC) of the received and locally-optimal MVs, and difference of that MVs (DoMV) from the encoded one applying (8) and (9). Among 4 candidate MVs, the MVs which have the closest $DoMV$ to $DoMV_{org}$ lie in the set S_1 using (10). These are the MVs which have nearest statistics to the original MV from the steganalyzer's point of view in case of local optimality.

$$DoLC_i = (J_{MV_i^R} - J_{MV_i^{Opt}})/J_{MV_i^R} \tag{8}$$

$$DoMV_i = |MV_{ih}^{Opt} - MV_{ih}^R| + |MV_{iv}^{Opt} - MV_{iv}^R| \tag{9}$$

$$S_1 = \{MV_j || DoMV_{org} - DoMV_j| = M_1\} \tag{10}$$

$$M_1 = \min\{|DoMV_{org} - DoMV_j| | j = 2, 4, 5, 7\} \tag{11}$$

In (8), J_x is the Lagrangian cost function associated with the motion vector x. In (9), MV_{ih}^{R} and MV_{iv}^{R} stand for the horizontal and vertical components of the decoded MV. Also, MV_{ih}^{Opt} and MV_{iv}^{Opt} stand for the horizontal and vertical components of locally-optimal MVs obtained by further motion estimation respectively.

The most locally optimal MVs. Among all of MVs existing in the set S_1, the MVs that seem to be more optimal in the receiver's side (with respect to DoMV) are selected by applying (12).

MV which imposes the nearest Lagrangian cost to that of the original MV. Finally, among MVs belonging to set S_2, the MV which has the closest $DoLC$ to $DoLC_{org}$ is selected as the alternative MV (14).

$$S_2 = \{MV_j \in S_1 | (\frac{DoMV_{org} - DoMV_j}{|DoMV_{org} - DoMV_j|}) = M_2\} \tag{12}$$

$$M_2 = \max\{(\frac{DoMV_{org} - DoMV_j}{|DoMV_{org} - DoMV_j|}) | MV_j \in S_1\} \tag{13}$$

$$MV_{Alternative} = \min_{MV_i}\{|DoLC_j - DoLC_{org}| | MV_i \in S_2\} \tag{14}$$

Step 3. Embedding Costs Evaluating: In this step, temporal and spatial costs are evaluated which subsequently form the overall cost of modification for different MVs.

Temporal cost function. For each block (b_k), temporal cost of MV modification is obtained by (15). Indeed, the temporal cost of modifying each MV is equal to the difference between the original and alternative motion vector plus the absolute difference between exponentials of "difference of Lagrangian costs" of original and alternative MVs from the receiver's perspective. In (15), exp stands for "exponential" and $exp(x)$ equals e^x. Exponential function is used to enlarge the difference between $DoLC_{MV_{org}}(b_k)$ and $DoLC_{MV_{Modified}}(b_k)$.

$$\begin{aligned} Cost_{Temporal}(b_k) = {} & DoMV_{MV_{Modified}}(b_k) \\ & + |exp\{DoLC_{MV_{org}}(b_k)\} - exp\{DoLC_{MV_{Modified}}(b_k)\}| \end{aligned} \tag{15}$$

Spatial cost function. As mentioned in [51], neighboring MVs may belong to a same rigid objects or fixed backgrounds. These MVs are not suitable cloaks for confidential information as their manipulation can leave statistical clues about embedding. Accordingly, we should consider the spatial correlations of neighboring MVs in the embedding cost function. The more similar the neighboring MVs of a MV are, the more spatial cost its modification imposes. Inspired by picture processing grammar [11] and XPG modeling [12], the spatial cost function is calculated as follows.

First, the MVs of eight neighbors of each block are obtained. Assuming the situation of the top-left pixel of existing block is equal to $(X(b_k), Y(b_k))$, and the

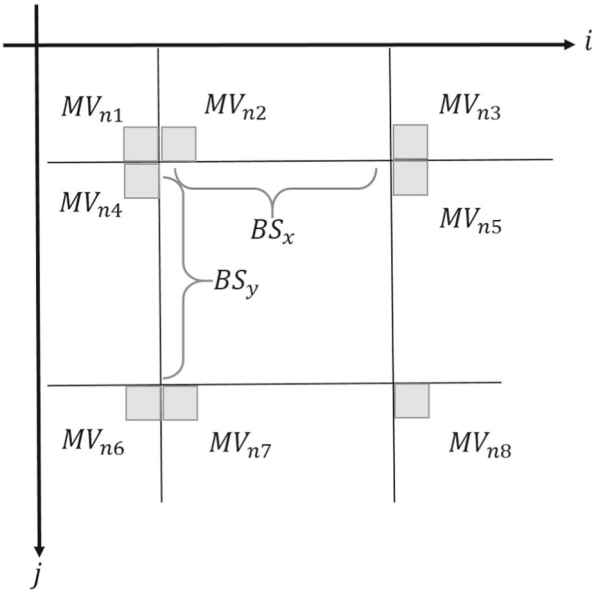


Fig. 5 Motion vectors of the neighboring blocks

block's size is $(BS_x(b_k), BS_y(b_k))$, the neighboring MVs are calculated as follows (Fig. 5).

$$\begin{aligned}
MV_{n1}(b_k) &= mv(X(b_k)-1, Y(b_k)-1) \\
MV_{n2}(b_k) &= mv(X(b_k), Y(b_k)-1) \\
MV_{n3}(b_k) &= mv(X(b_k)+BS_x(b_k), Y(b_k)-1) \\
MV_{n4}(b_k) &= mv(X(b_k)-1, Y(b_k)) \\
MV_{n5}(b_k) &= mv(X(b_k)+BS_x(b_k), Y(b_k)) \\
MV_{n6}(b_k) &= mv(X(b_k)-1, Y(b_k)+BS_y(b_k)) \\
MV_{n7}(b_k) &= mv(X(b_k), Y(b_k)+BS_y(b_k)) \\
MV_{n8}(b_k) &= mv(X(b_k)+BS_x(b_k), Y(b_k)+BS_y(b_k))
\end{aligned} \tag{16}$$

Next, the spatial cost is calculated employing (17). In (18), the function $Unique(x)$ counts the unique observations in the set x. $S^h_{MV}(b_k)$ and $S^v_{MV}(b_k)$ are the sets of horizontal and vertical components of eight neighboring MVs of the original MV in kth block respectively. Figure 6 shows three examples of neighboring MVs of a block and their corresponding spatial costs.

$$Cost_{spatial}(b_k) = Cost^h_{spatial}(b_k) + Cost^v_{spatial}(b_k) \tag{17}$$

$$\begin{aligned}
Cost^h_{spatial}(b_k) &= \frac{9 - Unique(S^h_{MV}(b_k))}{8} \\
Cost^v_{spatial}(b_k) &= \frac{9 - Unique(S^v_{MV}(b_k))}{8}
\end{aligned} \tag{18}$$

$$\begin{aligned}
S^h_{MV}(b_k) &= \{MV^h_{ni}(b_k) | i = 1, 2, ..., 8\} \\
S^v_{MV}(b_k) &= \{MV^v_{ni}(b_k) | i = 1, 2, ..., 8\}
\end{aligned} \tag{19}$$

$MV_1 =$ (0, 0.25)	$MV_2 =$ (0, 0.25)	$MV_3 =$ (0, 0.25)
$MV_4 =$ (0, 0.25)		$MV_5 =$ (0, 0.25)
$MV_6 =$ (0, 0.25)	$MV_7 =$ (0, 0.25)	$MV_8 =$ (0, 0.25)

$Cost_{spatial}(b_1) = 2$

$MV_1 =$ (0, 0.25)	$MV_2 =$ (0, 0.25)	$MV_3 =$ (0, 0.25)
$MV_4 =$ (0, 0.25)		$MV_5 =$ (0, 0.25)
$MV_6 =$ (0, 0.75)	$MV_7 =$ (0, 0.75)	$MV_8 =$ (0, 0.75)

$Cost_{spatial}(b_2) = 1.87$

$MV_1 =$ (0, -0.25)	$MV_2 =$ (0.25, 0.25)	$MV_3 =$ (0.5, 0.5)
$MV_4 =$ (0, 0)		$MV_5 =$ (0.5, 0. 5)
$MV_6 =$ (0, 0.5)	$MV_7 =$ (0, 0.25)	$MV_8 =$ (0.5, 0.75)

$Cost_{spatial}(b_3) = 1.25$

Fig. 6 Examples of a block's neighboring MVs and spatial cost of modifying its corresponding MV

1) The main cost function: Finally, the cost of altering the kth MV of the existing P-frame is evaluated employing following function (α and β are adaptable parameters. In experiments, we have set $\alpha = 4$ and $\beta = 2$). We have added 1 to each of the two cost functions in order that the value of both parentheses is not lower than 1 and consequently, multiplication of them is always greater than both values.

$$Cost(b_k) = (\alpha Cost_{emporal}(b_k) + 1)^{\beta} \times (Cost_{spatial}(b_k) + 1) \quad (20)$$

Step 4. MVs Re-ordering: In videos with slow movements, fixed background or fixed camera, the order of scanning MVs must be pseudo-random. This can be performed by sharing a key between the transmitter and the legitimate receiver. This pseudo-random selection is not to improve the cryptographic security, but to improve security against steganalysis attacks. Without employing a pseudo-randomizer, all available motion vectors for embedding one bit are likely to be a

part of the background or improper for embedding. Also, the MVs related to suitable regions of video such as deformable parts (e.g., a moving arm) having more appropriate statistical properties for altering will not be completely utilized.

In this step, the order of MVs is modified by a pseudorandom-generator key which is obtained using (21) (In (21), n is the number of MVs in the frame, ER is the embedding rate, and h is the parameter of syndrome-trellis coder). As mentioned in Section 2, the syndrome-trellis coder has access to $h \times w$ motion vectors for embedding each bit. Hence, the pseudorandom-generator key is optimized with respect to the embedding rate and h. Actually, we partition all MVs of the frame into the maximum number of accessible MVs for Syndrome-trellis coder. This way, the syndrome-trellis coder can access different parts of the frame during embedding each bit. The transformation matrix is then formed as (22). Finally, the order of MVs is altered by applying (23) (MV and RMV are the vector of primary MVs and re-ordered MVs respectively).

$$K = \lfloor \frac{n \times ER}{h} \rfloor \tag{21}$$

$$\mathbf{X} = \begin{pmatrix} 0 \times K + 1 & 1 \times K + 1 & \dots & ((h \times w) - 1) \times K + 1 \\ 0 \times K + 2 & 1 \times K + 2 & \dots & ((h \times w) - 1) \times K + 2 \\ \vdots & \vdots & \ddots & \vdots \\ 0 \times K + K & 1 \times K + K & \dots & ((h \times w) - 1) \times K + K \end{pmatrix} \tag{22}$$

$$RMV(:, i) = \begin{cases} MV(:, f_X(i)) & \text{if} \quad 1 \le i \le (h \times w) \times K \\ MV(:, i) & \text{else} \end{cases} \tag{23}$$

$$f_X(i) = X(\lceil \frac{i}{h \times w} \rceil, i - (\lceil \frac{i}{h \times w} \rceil - 1)(h \times w)) \tag{24}$$

$$\mathbf{MV} = \begin{pmatrix} MV_h(1) & MV_h(2) & \dots & MV_h(n) \\ MV_v(1) & MV_v(2) & \dots & MV_v(n) \end{pmatrix} \tag{25}$$

Figure 7 shows an example of MVs re-ordering and its effect on the arrangement of MVs that are fed into the Syndrome-trellis-coder. In this example, $h \times w$ is assumed to be equal to 4.

Step 5. Syndrome-trellis coding: The binary vector p is formed using parity check function and RMV (26). The syndrome-trellis-coder takes vector p along with MVs' modification costs and payload, calculates optimal MVs for embedding, and returns $\widetilde{p}$ (6).

$$p(i) = LSB(RMV(1, i) + RMV(2, i)) \tag{26}$$

Step 6. Information Embedding: For each element of $\widetilde{P}$ that is unequal to P, we replace its corresponding MV with the alternative MV obtained in the step 2 (14).

Step 7. MVs Re-predicting: In order to transmit correct information to the decoder, it is necessary to perform MV prediction after embedding on the MVs of each frame (Please refer to the Appendix for more explanation).

Step 8. P-frame Encoding: In this stage, the rest of the typical video compression algorithm (including variable-length coding) for the existing frame is applied using the information obtained in the previous stages.

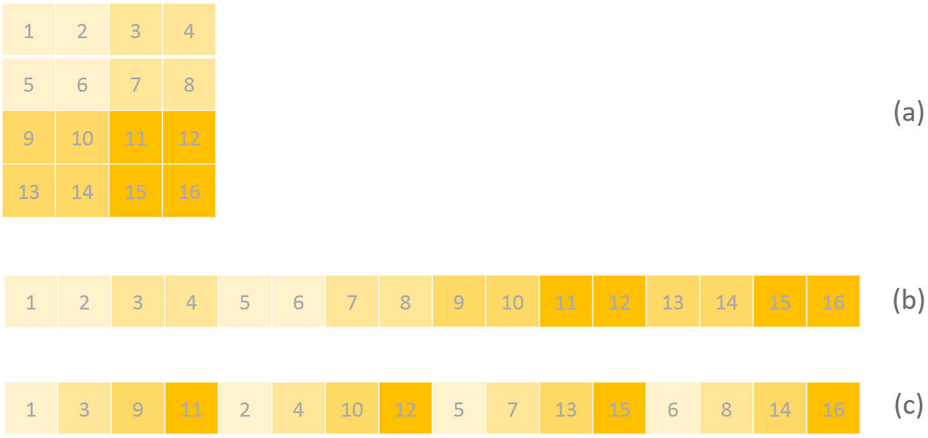


Fig. 7 Effect of MVs reordering on the arrangement of MVs that are fed into the Syndrome-trellis-coder. **a** blocks and their numbers in a P-frame. **b** the usual arrangement of MVs that are fed into the Syndrome-trellis-coder using raster scanning. **c** the arrangement of MVs after re-ordering

4.1 Extraction procedure

The extraction algorithm is also applied to P-frames one after another. The confidential message is then obtained by accumulating extracted information and subsequently decrypting it.

Step 1. Reordering MVs: After decoding the bitstream of MVs [42], their order is altered by employing the common key and (21)-(23).
Step 2. Syndrome-trellis decoding: Having embedding rate, the vector of reordered MVs and h, we apply syndrome-trellis decoding to obtain the confidential message.

5 Experiments

5.1 Experimental settings

- **Database:** Figure (8) shows the first frame of 22 video sequences with PAL QCIF resolution and without prior lossy compression[3] (192 × 144 pixels), being downloaded from [49] and used to construct the database. The database consists of a wide range of videos with respect to diversity in camera and objects movement. Since the video sequences contain a different number of frames, all sequences are trimmed into non-overlapping 60-frame subsequences, and utmost five 60-frame subsequences are applied for experiments. Totally 84 subsequences are used for experiments.
- **Video Compression Method:** Without loss of generality, H.264's baseline profile is employed for video compression [38]. Motion estimation methods are exhaustive (full) search and HEX search [52]; motion estimation resolution is equivalent to quarter-pixel; the search range is 8 pixels, and quantization parameter is equal to 17, 27 and 32.

[3]Prior compression can affect the performance of steganalysis [18].

Fig. 8 tree structured motion compensation for H.264 [38]

- **Competitor Steganography Approaches:** In order to evaluate the performance of the proposed method, we compare its results (denoted by Prop*) with the results of methods [9] (denoted by ALG1), [56] (denoted by ALG2), [10] (denoted by ALG3), and [7] (denoted by ALG4) [4].
- **Steganalysis Manners:** In order to illustrate the security of the proposed method in comparison with rivals, we apply MVRB features [8], AoSO features [45], and NPELO features [53], which are the best video steganalysis methods against MV-based video steganography to date (The method [39] which exploits the entropy of blocks for feature extraction has proved to be a reliable detector, and its results are close to that of [53]). Meanwhile, features in [8] are extracted using macroblocks; while in H.264 video compression algorithm, each macroblock may contain some blocks. Therefore, in order to improve the performance of this steganalyzer, the features are extracted using blocks.
- **Training and Classification:** For each steganography method, a random sequence with uniform distribution[5] is produced and embedded in all subsequences with rates 0.1, 0.2 and 0.3 of total MVs per P-frame. All embedded and compressed videos are separated into non-overlapping 12-frame subsequences, and a feature vector is extracted from each subsequence. Indeed, we have 420 embedded twelve-frame-subsequences and 420 clean twelve-frame-subsequences per each training and testing round (each number in Tables 3, 4 and 5). Afterwards, 80% of feature vectors are randomly chosen for training, and the remaining feature vectors are exploited for testing the classifiers. We use MATLAB's SVM toolbox for training each steganalyzer and classification. Also, Gaussian and Polynomial kernel functions are applied and the best answers are listed.
- **Evaluation Criteria:** We compare the detector reliability (AUC) of steganalyzer approaches against proposed and rival methods to illustrate the fact that our scheme outperforms the best existing MV-based video steganography approaches in case of undetectability [5, 6, 15]. In order to clarify the effect of applying a pseudorandom-generator key on security improvement, we have shown the results of the proposed method without reordering MVs (denoted by Prop) and with reordering MVs (denoted by Prop*). Imperceptibility is measured by PSNR which can show the perception of quality degradation by human visual system better than other criteria [13, 31]. Also,

[4] As it is mentioned before, since there is not always a candidate embedded MV which is locally optimal in receiver's side, ALG2 and ALG3 are not implementable regarding their settings. Hence, we have to revisit these algorithms by adding an alternative plan: in case of lack of locally optimal candidate MV, we select the MV contributing the smallest Lagrangian cost as embedded MV.

[5] Distribution of the message could affect the steganalysis performance [23].

Table 3 Detector reliability of MVRB against our proposed method and rival steganography approaches (SA) with different motion estimation algorithms (ME), embedding rates (ER), and quantization parameters (QP)

ER	QP	ME	FULL						HEX					
		SA	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4
0.1	17		**0.52**	0.54	0.96	0.52	0.56	0.73	**0.51**	0.52	0.85	0.52	0.53	0.53
	27		**0.53**	0.53	0.97	0.53	0.53	0.54	**0.50**	0.52	0.87	0.52	0.51	0.52
	32		**0.52**	0.52	0.99	0.52	0.54	0.54	**0.50**	0.51	0.88	0.51	0.52	0.52
0.2	17		**0.54**	0.65	0.99	0.61	0.63	0.82	**0.53**	0.54	0.92	0.54	0.58	0.58
	27		**0.52**	0.54	0.99	0.55	0.56	0.59	**0.51**	0.52	0.92	0.52	0.53	0.56
	32		**0.53**	0.54	1.00	0.54	0.56	0.59	**0.51**	0.51	0.93	0.51	0.52	0.54
0.3	17		**0.69**	0.76	1.00	0.73	0.71	0.89	**0.56**	0.57	0.95	0.57	0.63	0.65
	27		**0.54**	0.58	1.00	0.61	0.60	0.66	**0.53**	0.54	0.95	0.54	0.55	0.60
	32		0.55	**0.54**	1.00	0.56	0.58	0.64	0.52	**0.51**	0.95	0.51	0.52	0.56

Best results are highlighted in bold

Table 4 Detector reliability of AoSO against our proposed method and rival steganography approaches (SA) with different motion estimation algorithms (ME), embedding rates (ER), and quantization parameters (QP)

ER	QP	ME	FULL						HEX					
		SA	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4
0.1	17		**0.51**	0.51	0.86	0.52	0.53	0.56	**0.50**	0.50	0.60	0.53	0.52	0.53
	27		**0.50**	0.51	0.80	0.51	0.53	0.51	0.51	**0.50**	0.60	0.52	0.51	0.52
	32		**0.51**	0.51	0.84	0.51	0.57	0.54	0.51	**0.50**	0.61	0.51	0.53	0.52
0.2	17		0.54	0.55	0.97	**0.53**	0.58	0.68	**0.51**	0.51	0.76	0.56	0.55	0.56
	27		**0.51**	0.51	0.95	0.51	0.54	0.56	**0.50**	0.51	0.75	0.54	0.52	0.54
	32		**0.51**	0.51	0.94	0.52	0.56	0.55	**0.50**	0.51	0.79	0.53	0.52	0.54
0.3	17		0.61	0.65	0.99	**0.59**	0.67	0.79	**0.51**	0.51	0.86	0.59	0.58	0.62
	27		**0.52**	0.53	0.99	0.52	0.57	0.62	**0.50**	0.51	0.87	0.57	0.53	0.56
	32		**0.52**	0.52	0.98	0.53	0.58	0.55	**0.51**	0.51	0.89	0.55	0.53	0.56

Best results are highlighted in bold

Table 5 Detector reliability of NPELO against our proposed method and rival steganography approaches (SA) with different motion estimation algorithms (ME), embedding rates (ER), and quantization parameters (QP)

ER	QP	ME	FULL						HEX					
		SA	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4	Prop.*	Prop	ALG1	ALG2	ALG3	ALG4
0.1	17		**0.52**	0.52	0.94	0.52	0.70	0.76	**0.51**	0.51	0.67	0.53	0.54	0.55
	27		**0.51**	0.52	0.94	0.51	0.72	0.68	**0.51**	0.51	0.68	0.52	0.57	0.52
	32		**0.51**	0.51	1.00	0.52	0.84	0.72	**0.51**	0.51	0.71	0.51	0.67	0.56
0.2	17		**0.60**	0.63	0.99	0.66	0.85	0.92	**0.52**	0.52	0.83	0.58	0.56	0.60
	27		**0.53**	0.53	1.00	0.58	0.76	0.79	0.52	**0.51**	0.87	0.54	0.57	0.57
	32		**0.53**	0.53	1.00	0.56	0.84	0.81	0.53	**0.52**	0.89	0.53	0.66	0.59
0.3	17		**0.74**	0.77	1.00	0.81	0.93	0.96	**0.53**	0.53	0.91	0.64	0.60	0.66
	27		**0.57**	0.59	1.00	0.64	0.82	0.86	**0.52**	0.52	0.93	0.59	0.59	0.59
	32		**0.55**	0.56	1.00	0.62	0.88	0.86	**0.53**	0.53	0.95	0.56	0.68	0.62

Best results are highlighted in bold

mean output bitrate per P-frame is measured to compare the compression ratio (The results correspond to standard video compression are denoted by STD). Additionally, the percentage of increase in compression time has been shown to demonstrate the difference in the computational cost of the proposed method and rivals.

- **Other settings:** In the competitor approaches [7, 9, 56], embedding is applied by employing full search. So the search range for finding embedded MVs is the same as the search range for motion estimation and equal to 8 pixels. On the contrary, embedded MVs in the proposed method are selected from the set of only four MVs around the original MVs. Also in [9], The set of Lagrangian multipliers which are used for MVs' embedding cost function is $\lambda = [0, 2, 4, 6, 8]$ and we set $b = -2$ and $\alpha = 0.5$ for the distortion function. The toolbox for syndrome-trellis code which is used to implement the proposed and rival methods is downloaded from [22].

  We set $h = 8$ for the syndrome-trellis coder used in the proposed and competitor algorithms. Also, we just use $ER = 0.2$ and $QP = 27$ for all criteria exclusively of security.

5.2 Experimental results

1) **Security:** The security of the proposed method and ALG1-4 against three detectors are compared in Tables 3–5. Table 3 shows that ALG1 cannot resist against MVRB and the classes in some cases are completely distinctive by this feature set. Other algorithms have shown a fair level of security against this detector, specifically when it comes to HEX motion estimation algorithm. The figures for the proposed method, however, are much closer to 0.5 than that of rivals.

   According to Table 4, the detector AoSO is not reliable against the proposed method and competitors, except for ALG1. The resistance of competitors has increased when the fast search algorithm has been applied in video compression. The proposed method has surpassed all rivals in security against AoSO, the figures of which hover around 0.51.

   Only the proposed method is secure against NPELO (Table 5). ALG2 which showed acceptable resistance against previous detectors fails to resist against NPELO in high embedding rates in both HEX and FULL ME conditions. Furthermore, a sharp decrease in security with increasing the embedding rate can be observed in all rivals, whereas these changes in the performance of the proposed scheme are quite negligible.

   The positive impact of MVs reordering in the performance of the syndrome-trellis coder can be observed in Tables 3-5. In the case of FULL search ME algorithm and smaller quantization parameters where MVs are overwhelmingly locally-optimal and less information is lost during compression, the influence of reordering is more prominent.

   In summation, the records of various detectors' reliability against the proposed method and rivals have proved that not only our method has surpassed all of the competitors in undetectability, but also it can highly resist against the best detectors so that their results are close to random guessing.

2) **Imperceptibility:** Table 6 compares PSNR between the original and embedded sequences. Looking first at the results of FULL search, it can be perceived that there is a slight difference between the PSNR correspond to the output of the standard compression algorithm and embedding algorithms, exclusively of ALG1. The reason why we see the average reduction of $0.18db$ in PSNR of ALG1 is improper cost function which does not reflect the characteristics of the alternative MV. Although the replaced

Table 6 PSNR of video sequences produced by standard compression, our proposed method, and rival steganography approaches (SA) with different motion estimation algorithms (ME). Embedding rate for all steganography methods is set to 0.2, and the quantization parameter is equal to 27

Sequence	ME	FULL						HEX					
	SA	STD	Prop.*	ALG1	ALG2	ALG3	ALG4	STD	Prop.*	ALG1	ALG2	ALG3	ALG4
Akiyo		40.26	40.31	39.88	**40.34**	40.32	40.33	39.92	39.97	39.52	**40.03**	39.94	40.03
Carphone		39.11	39.11	38.94	39.10	39.12	**39.14**	38.58	38.57	38.42	38.60	38.58	**38.63**
Deadline		37.83	37.83	37.59	37.82	37.80	**37.84**	37.65	37.63	37.40	37.62	37.63	**37.67**
Foreman		38.38	38.35	38.20	38.37	38.38	**38.39**	37.65	37.64	37.54	**37.69**	37.65	37.69
Highway		38.46	38.46	38.41	38.46	38.46	**38.48**	38.23	38.23	38.16	38.25	38.24	**38.28**
Mobile		36.35	**36.34**	36.31	36.34	36.34	36.34	36.19	**36.19**	36.16	36.19	36.18	36.19
News		39.01	39.02	38.71	39.03	39.03	**39.05**	38.85	38.88	38.49	38.87	38.84	**38.90**
Silent		38.03	**38.03**	37.81	38.03	38.03	38.03	37.74	37.77	37.57	**37.78**	37.77	37.77

Best results are highlighted in bold

Table 7 Average output bitrate of video sequences produced by standard compression, our proposed method, and algorithms 1-4 (motion estimation: FULL, embedding rate: 0.2, Quantization Parameter: 27)

Sequence	Average Output Bitrate						Average Number of Embedded Bits				
	STD	Prop.*	ALG1	ALG2	ALG3	ALG4	Prop.*	ALG1	ALG2	ALG3	ALG4
Akiyo	0.87	**0.87**	1.10	0.87	0.87	0.87	27.29	**28.59**	27.39	27.51	27.51
Carphone	1.94	1.95	2.23	**1.94**	1.94	1.96	38.46	**39.03**	38.46	38.46	38.29
Deadline	1.77	1.79	2.16	1.78	1.80	**1.77**	33.36	**35.47**	33.90	33.53	33.46
Foreman	2.43	2.46	2.82	2.45	**2.44**	2.46	**43.73**	43.37	43.24	42.81	43.00
Highway	2.54	**2.55**	2.72	2.55	2.55	2.57	42.68	41.92	43.03	**43.44**	42.95
Mobile	6.54	6.57	6.89	6.58	6.58	**6.56**	36.76	36.44	36.80	**36.83**	36.53
News	1.34	1.35	1.63	1.35	**1.34**	1.35	30.44	**32.00**	30.42	30.66	30.75
Silent	1.75	1.77	2.09	**1.76**	1.76	1.76	35.17	**36.78**	35.51	35.08	35.20

Best results are highlighted in bold

Table 8 Average output bitrate of video sequences produced by standard compression, our proposed method, and algorithms 1-4 (motion estimation: HEX, embedding rate: 0.2, Quantization Parameter: 27)

Sequence	Average Output Bitrate						Average Number of Embedded Bits				
	STD	Prop.*	ALG1	ALG2	ALG3	ALG4	Prop.*	ALG1	ALG2	ALG3	ALG4
Akiyo	1.13	**1.13**	1.37	1.13	1.14	1.13	30.49	**31.37**	29.97	30.41	30.44
Carphone	2.64	2.65	2.92	**2.64**	2.65	2.65	38.05	**38.20**	37.53	37.64	38.15
Deadline	2.42	2.44	2.81	2.43	2.46	**2.40**	35.41	**36.37**	35.58	36.05	36.07
Foreman	3.64	3.65	3.93	3.63	3.65	**3.62**	39.98	40.20	**40.37**	40.10	39.75
Highway	2.89	2.90	3.05	2.90	**2.89**	2.90	36.83	**37.39**	37.03	37.02	36.73
Mobile	8.05	8.06	8.33	8.03	8.06	**8.01**	38.31	38.27	**38.37**	37.88	38.22
News	1.78	1.78	2.12	1.79	1.79	**1.77**	32.63	**34.41**	33.69	32.51	32.78
Silent	2.15	2.15	2.50	2.15	2.15	**2.14**	35.00	**35.90**	34.75	34.93	34.66

Best results are highlighted in bold

Table 9 Computational cost of our proposed method and rivals (embedding rate: 0.2, quantization parameter: 27)

Sequence	ME	FULL					HEX				
	EA	Prop.*	ALG1	ALG2	ALG3	ALG4	Prop.*	ALG1	ALG2	ALG3	ALG4
Akiyo		**11.87**	18.88	86.38	13.31	22.13	**13.85**	19.88	90.37	15.72	25.85
Carphone		**11.97**	19.31	82.50	13.48	22.86	**13.41**	19.17	88.51	15.34	25.12
Deadline		**11.55**	18.10	84.41	13.05	21.57	**13.33**	18.62	88.86	14.91	25.68
Foreman		**11.45**	19.12	80.32	13.16	22.20	**13.01**	18.35	87.43	14.70	25.32
Highway		**12.04**	19.31	81.35	13.51	22.66	**13.69**	19.44	88.83	15.13	25.93
Mobile		**10.56**	16.40	81.14	11.73	19.88	**11.46**	16.01	86.44	12.67	21.26
News		**11.62**	18.32	85.10	13.03	21.65	**13.65**	19.58	89.68	15.87	25.85
Silent		**11.29**	18.19	83.26	12.97	20.97	**13.65**	19.42	89.27	15.57	26.79

Best results are highlighted in bold

MV is obtained using $\lambda_{ME} = 0$ and consequently we may expect a better PSNR in comparison to the original video, the impact of this phase is negated by the inaccurate cost function. The average differences in PSNR for other algorithms is roughly equal to $0.001db$ that is quite acceptable.

Moving on to the results of HEX search, a more perceptible average increase in PSNR of ALG2 and ALG4 ($0.02db$ and $0.04db$) can be seen. This phenomenon is because of their cost function which uses the whole allowed search window to find the replaced MV. However, an increase in PSNR cannot be regarded as a merit if it negatively affects the computational cost or output bitrate. There is a small increase in PSNR of the proposed method, being equal to $0.01db$ on average. Also, the results of ALG1 are similar to that in the FULL search.

3) **Compression Ratio:** Tables 7 and 8 display the average output bitrate (Kilobytes per P-frame) and the size of the embedded message (Bits per P-frame). Since the size of the confidential information in each sequence is approximately equal in the proposed and rival approaches, we can expect the output bitrates to be similar as well. The only method which has demonstrated weak performance in the compression ratio is ALG1, and the output bitrates of other methods including ours are close to that of the standard compression.
4) **Computational Cost:** In Table 9, the ratio of embedding time to standard compression time is indicated. As can be seen, the computational cost of the proposed method is far less than that of competitor schemes, being roughly equal to 11% and 12% in the case of FULL and HEX ME algorithms respectively. In sharp contrast with the proposed algorithm, ALG2, the security of which is the best among all rivals, has the highest computational cost. Indeed, the average computational cost of ALG2 is more than 7 times that of the proposed method. The computational cost that ALG3 imposes is the closest one to the proposed method, owing to the fact that both schemes consider only 4 MVs around each original MV as candidates of embedded MV.

6 Discussion

The idea behind our proposed method is rooted in the fact that the outputs of all of the lossy compression algorithms are regarded as innocent media, notwithstanding the lost information which is not retrievable. Thus we can increase the security of a steganography method if the information is hidden in media so that statistical clues of manipulation can be lost through the procedure of lossy compression. In order to design a reliable steganography method, the following points should be noted:

1) With increasing the size of video frames, the necessity of reordering MVs and its impact on security improvement will increase. For example, in a video sequence with CIF resolution, because the size of the background is larger than that in QCIF resolution, more successive MVs sets can have improper statistics for altering. Therefore, the impact of pseudo randomization in undetectability would be remarkable.
2) Since the trellis encoder has access to just $h \times w$ motion vectors, with increasing the embedding rate, the number of MVs which are accessible to embed one bit would decrease; So the impact of reordering MVs would be more palpable. A simple comparison between the security degradation of the proposed method and rivals with increasing the embedding rate (Table 3-5) can prove the mentioned logical statement.

3) By reordering MVs, even using a smaller h in the trellis coder, we can reach higher security. Actually, the role of the pseudo-randomizer key is to disperse undetectable MVs.
4) A smaller change in motion vectors leads to a smaller change in the statistical characteristics of the video. Hence, efficient usage of compression techniques which support a smaller sub-pixel motion estimation can substantially contribute to improving the steganography performance.
5) Purposive selection of video compression settings can enhance steganographic security. As a concrete example, video compression settings which lead to more non-locally-optimal MVs with respect to the Lagrangian multiplier from the eavesdropper's point of view increase the resistance of MV-based steganography against steganalysis attacks. Therefore, fast search algorithms producing more indeterministic components and consequently reducing the percentage of locally optimal MVs can provide more appropriate content for embedding.
6) It should not be left unmentioned that the obtained results are the average of results for different covers. It is amply clear that the selection of host media plays a prominent role in the security of steganography methods against steganalysis attacks. The existence of more deformable objects, less fixed background, faster-moving objects, and fast-moving cameras will result in a better performance of steganography.

7 Conclusion

This paper suggests an adaptive approach for hiding information in MVs of the video. According to the results, a dramatic increase in the security of the proposed method compared to current outstanding MV-based steganography methods is observed. Moreover, the proposed approach defeats the prominent steganalysis attacks. Indeed, due to unreliable evidence of modification, steganalyzers cannot distinguish a clear video from a stego one produced by the proposed algorithm. This improvement originates from three contributions:

- *Preserving temporal statistics of MVs by applying the statistical differences of the original and the best alternative MV:* The most important MV-based temporal steganalytic features are extracted based on local optimality of the received MVs. MV-based steganography usually causes the MVs to shift from optimal to non-optimal. This problem can be addressed by selecting alternative MVs which have the closest local-optimality-related steganalytic features at the receiver's side. All the information which are available in the steganalyzer's side can be utilized in the steganographer side, whereas some information being available in the transmitter side may vanish during the lossy compression procedure. This superiority should be utilized against eavesdroppers. We have proposed a method to select the MVs with closest local-optimality-related steganalytic features to that of the original MVs. Moreover, the embedding cost function is designed so that alternative MVs impose lower embedding costs.
- *Preserving spatial statistics of MVs by applying their correlation to the cost function:* MVs may have specific patterns in some areas of the video. The less the entropy of MVs around a particular MV is, the more perceptible changes its altering will impose on the statistical characteristics of the output video. For instance, if all of the neighbors of an MV are equal to zero, changing this MV may result in moving some feature vectors of the detector to a particular area. As a consequence, the margin between the two classes of clean and dirty media will increase, and the detector can reach more

distinctive classes. By contrast, if all of the neighboring MVs are different together, manipulating the central MV leads to scattering the feature vectors to different areas and this sparse shift will cause an increase in security against steganalysis attacks. The designed embedding cost has an indirect relationship with the entropy of neighboring MVs. Thus the syndrome-trellis-coder is encouraged to choose MVs from deformable dynamic regions rather than static regions.

- *Improving the availability of proportionate MVs for syndrome-trellis coder using an adjustable pseudorandomization key:* An optimized embedding cost function is the necessary condition for a secure steganography method, but not the sufficient condition. In fact, the distribution of proper hosts to be selected by the syndrome-trellis coder is as important as a perfect cost function. The syndrome-trellis coder sometimes does not have access to low-cost MVs for manipulation. The mentioned problem is supposed to be alleviated to a great extent by re-ordering MVs.

It is expected that the detector's reliability will drop further in real-world conditions where the exact information about the embedding rate and ME algorithm are not available [28]. As a result, secure steganography with higher embedding rates is possible.

The proposed approach also causes a slight improvement in compression ratio and visual quality, as well as achieving the smallest computational cost in comparison with rivals.

Appendix

Herein, we aim to clarify the necessity of MVs further prediction before coding MVs. In the general video codec procedure, each MV is computed applying a motion estimation algorithm (1). Then, the selected MV of the block k (denoted by MV_{b_k}) undergoes the coding process. Since neighboring MVs are often highly correlated, the entropy of MV differences is far less than that of MVs. Therefore, transmitting the MV differences instead of MVs can have a significant effect on minimizing the cost of transmission. Hence, the difference between each MV_{b_k} and the predicted MV of its corresponding block (denoted

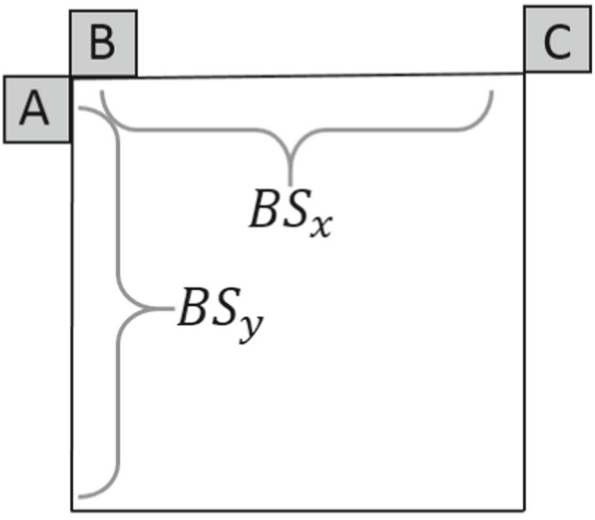


Fig. 9 Neighboring pixels, MV of which is used for MV prediction[38]

by $MV_{b_k}^P$) is calculated (27) and consequently coded using Exp-Golomb code which is a kind of variable-length codes (VLCs). The predicted MV ($MV_{b_k}^P$) is a function of MVs of three neighboring pixels in previously coded blocks and size and position of the block (Fig. 9). The prediction method is based on block size and availability of neighboring MVs. At the receiver side, blocks are decoded and arranged in the raster scan. The received MV for each block is added to its prediction which can be calculated using previously decoded MVs. After MVs manipulation, the predicted MVs of some blocks may change. If we do not perform a further prediction and instead transmit the difference between manipulated MVs and their first predictions, the extracted MV in the receiver side may not be correct. Therefore, not only the confidential information may be lost, but also a wrong block is used as a prediction block which can detrimentally affect the visual quality of the output video.

$$MV_{b_k}^D = MV_{b_k} - MV_{b_k}^P = MV_{b_k} - f(b_k, A_{b_k}, B_{b_k}, C_{b_k}) \tag{27}$$

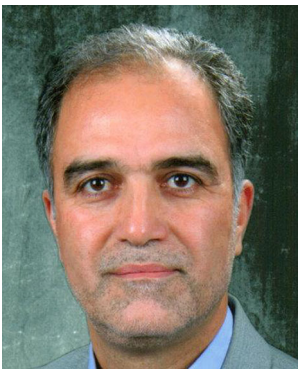
References

1. Aly HA (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. Altaay AAJ, Sahib SB, Zamani M (2012) An Introduction to Image Steganography Techniques, 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), pp 122–126
3. Bilal M, Imtiaz S, Abdul W, Ghouzali S, Asif S (2013) An Information-Theoretic model for steganography. Multimed Tools Appl 72:306–318
4. Budhia U, Kundur D, Zourntos T (2006) Digital video steganalysis exploiting statistical visibility in the temporal domain. IEEE Trans Inf Forensic Secur 1:502–516
5. Bòhme R (2010) Principles of Modern Steganography and Steganalysis. Springer, Berlin, pp 11–77
6. Bòhme R (2010) Advanced Statistical Steganalysis, vol. 0 of Information Security and Cryptography. Springer, Berlin
7. Cao Y, Zhao X, Feng D, Sheng R (2011) Video steganography with perturbed motion estimation. In: Proceedings of the 13th International Conference on Information Hiding, IH'11. Springer, Berlin, pp 193–207
8. Cao Y, Zhao X, Feng D, Features R-B (2012) Video steganalysis exploiting motion vector Reversion-Based features, signal processing letters. IEEE 19(1):35–38
9. Cao Y, Zhang H, Zhao X, Yu H (2015) Covert communication by compressed videos exploiting the uncertainty of motion estimation. IEEE Commun Lett 19(2):203–206
10. Cao Y, Zhang H, Zhao X, Yu H (2015) Video steganography based on optimized motion estimation perturbation. In: Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security. ACM, New York, pp 25–31
11. Chang S-K (1971) Picture processing grammar and its applications. Inf Sci 3(2):121–148
12. Costagliola G, Deufemia V, Polese G (2007) Visual language implementation through standard compiler-compiler techniques. J Vis Lang Comput 18:165–226, 04
13. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2008) Digital Watermarking and Steganography, 2nd edn. Morgan Kaufmann Publishers Inc., San Francisco
14. Dalal M, Juneja M (2018) Video steganography techniques in spatial domain—a survey. In: Mandal JK, Saha G, Kandar D, Maji AK (eds) Proceedings of the International Conference on Computing and Communication Systems. Springer, Singapore, pp 705–711
15. Fawcett T (2006) An Introduction to ROC Analysis. Pattern Recogn Lett 27:861–874
16. Fang DY, Chang LW (2006) Data hiding for digital video with phase of motion vector. 2006 IEEE International Symposium on Circuits And Systems, Vols 1-11. Proceedings, pp 1422–1425
17. Filler T, Judas J, Fridrich J (2011) Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Trans Inf Forensic Secur 6(3 Part 2):920–935
18. Fridrich J, Goljan M, Du R (2001) Steganalysis based on jpeg compatibility 4518:11
19. Fridrich J, Goljan M, Soukal D (2004) Perturbed quantization steganography with wet paper codes. In: Proceedings of the 2004 Workshop on Multimedia and Security, MM&Sec'04. ACM, New York, pp 4–15

20. Fridrich J, Goljan M, Lisonek P, Soukal D (2005) Writing on wet paper. IEEE Trans Signal Process 53:3923–3935
21. Fridrich J, Kodovsky J (2013) Multivariate gaussian model for designing additive distortion for steganography, ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp 2949–2953
22. Fridrich J (1999) http://dde.binghamton.edu/download/syndrome/
23. Ghasemzadeh H (2017) Multi-layer architecture for efficient steganalysis of undermp3cover in multi-encoder scenario. CoRR, vol. arXiv:1710.01230
24. Hu Y, Zhang C, Su Y (2007) Information hiding based on intra prediction modes for h.264/avc. In: 2007 IEEE International conference on multimedia and expo, pp 1231–1234
25. Hu SD, U KT (2011) A Novel Video Steganography Based on Non-uniform Rectangular Partition, 2011 14th IEEE International Conference on Computational Science and Engineering, pp 57–61
26. Idbeaa TF, Samad SA, Husain H (2015) An adaptive compressed video steganography based on pixel-value differencing schemes. In: 2015 International conference on advanced technologies for communications (ATC), pp 50–55
27. Kapotas SK, Skodras AN (2008) A new data hiding scheme for scene change detection in h.264 encoded video sequences. In: 2008 IEEE International conference on multimedia and expo, pp 277–280
28. Ker AD, Bas P, Böhme R, Cogranne R, Craver S, Filler T, Fridrich J, Pevný T (2013) Moving steganography and steganalysis from the laboratory into the real world. In: Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security. ACM, New York, pp 45–58
29. Kuznetsov AV, Tsybakov BS (1974) Coding in a memory with defective cells, problems inform. Transmission 7:132–138
30. Li Y, Chen HX, Zhao Y (2010) A new method of data hiding based on H.264 encoded video sequences. International Conference on Signal Processing Proceedings, ICSP, pp 1833–1836
31. Li B, J H, J H, Y QS (2011) A survey on image steganography and steganalysis. J Inf Hiding Multimed Signal Process 2(4):142–172
32. Liu B, Liu F, Yang C, Sun Y (2008) Secure steganography in compressed video bitstreams. In: 2008 Third international conference on availability, reliability and security. IEEE, pp 1382–1387
33. Liao K, Lian S, Guo Z, Wang J (2012) Efficient information hiding in h.264/avc video coding. Telecommun Syst 49(2):261–269
34. Moulin P, O'Sullivan JA (2003) Information-theoretic analysis of information hiding. IEEE Trans Inf Theory 49:563–593
35. Mstafa RJ, Elleithy KM (2015) A novel video steganography algorithm in the wavelet domain based on the klt tracking algorithm and bch codes. In: 2015 Long island systems, applications and technology, pp 1–7
36. Pan F, Xiang L, Yang XY, Guo Y (July 2010) Video steganography using motion vector and linear block codes. In: 2010 IEEE International conference on software engineering and service sciences, pp 592–595
37. Rana S, Bhogal RK (2018) A highly secure video steganography inside dwt domain hinged on bcd codes. In: Singh R, Choudhury S, Gehlot A (eds) Intelligent communication, control and devices. Springer, Singapore, pp 719–729
38. Richardson IE (2003) H.264 and MPEG-4 video compression: Video Coding for Next-generation Multimedia. No. 1. Wiley, New York
39. Sadat ES, Faez K, Saffari Pour M (2018) Entropy-Based Video Steganalysis of Motion Vectors. Entropy 20:244
40. 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
41. Sharp A, Qi Q, Yang Y, Peng D, Sharif H (2013) A video steganography attack using multi-dimensional discrete spring transform. In: 2013 IEEE International conference on signal and image processing applications, pp 182–186
42. Schöffmann K, Fauster M, Lampl O, Böszörmenyi L (2007) An evaluation of parallelization concepts for baseline-profile compliant h.264/avc decoders. In: Kermarrec AM, Bougé L, Priol T (eds) Euro-par 2007 parallel processing, Springer, pp 782–791
43. Su Y, Zhang C, Zhang C (2011) A video steganalytic algorithm against motion-vector-based steganography. Signal Process 91(8):1901–1909
44. Tasdemir K, Kurugollu F, Sezer S (2016) Spatio-temporal rich model-based video steganalysis on cross sections of motion vector planes. IEEE Trans Image Process 25:3316–3328
45. Wang K, Zhao H, Wang H (2014) Video steganalysis against motion vector-based steganography by adding or subtracting one motion vector value. IEEE Trans Inf Forensic Secur 9(5):741–751
46. Wong K, Tanaka K, Takagi K, Nakajima Y (2009) Complete video quality-preserving data hiding. IEEE Trans Circ Syst Video Technol 19(10):1499–1512

47. Wu HT, Liu Y, Huang J, Yang XY (2014) Improved steganalysis algorithm against motion vector based video steganography. In: 2014 IEEE International conference on image processing (ICIP), pp 5512–5516
48. Xu C, Ping X, Zhang T (2006) Steganography in compressed video stream. In: Proceedings of the First International Conference on Innovative Computing, Information and Control - Volume 1, ICICIC '06. IEEE Computer Society, Washington, pp 0–3
49. Xiph.org (1999) https://media.xiph.org/video/derf/
50. Yang G, Li J, He Y, Kang Z (2011) An information hiding algorithm based on intra-prediction modes and matrix coding for H.264/AVC video stream, {AEU}. Int J Electron Commun 65(4):331–337
51. Yao Y, Zhang W, Yu N, Zhao X (2015) Defining embedding distortion for motion vector-based video steganography. Multimed Tools Appl 74(24):11163–11186
52. Zhu C, Lin X, Chau L-P (2002) Hexagon-based search pattern for fast block motion estimation. IEEE Trans Circ Syst Video Technol 12:349–355
53. Zhang H, Cao Y, Zhao X (2017) A steganalytic approach to detect motion vector modification using near-perfect estimation for local optimality. IEEE Trans Inf Forensic Secur 12:465–478
54. Zhang M, Guo Y (2014) Video steganography algorithm with motion search cost minimized. In: 2014 9Th IEEE conference on industrial electronics and applications, pp 940–943
55. Zhang H, Cao Y, Zhao X, Zhang W, Yu N (2014) Video steganography with perturbed macroblock partition. In: Proceedings of the 2Nd ACM Workshop on Information Hiding and Multimedia Security. ACM, New York, pp 115–122
56. Zhang H, Cao Y, Zhao X (2016) Motion vector-based video steganography with preserved local optimality. Multimed Tools Appl 75:13503–13519
57. Zhang Y, Zhang M, Yang X, Guo D, Liu L (2017) Novel video steganography algorithm based on secret sharing and error-correcting code for h.264/avc. Tsinghua Sci Technol 22:198–209

Negin Ghamsarian received her M.Sc. degree in Electrical Engineering- communication systems from Ferdowsi University of Mashhad in Iran in 2016. She is currently Ph.D. student of Computer Science and project assistant in University of Klagenfurt in Austia. Her research interests include image and video processing, Information Security, machine learning, and deep learning.



Morteza Khademi received his B.Sc. and M.S. degrees in Electrical Engineering from Isfahan University of Technology, Isfahan, Iran, in 1985 and 1987, respectively, and Ph.D. degree in Electrical Engineering from the University of Wollongong, Wollongong, Australia, in 1995. He joined Ferdowsi University of Mashhad, Iran in 1987. He is currently a professor at the Department of Electrical Engineering, Ferdowsi University of Mashhad. During the past, he has co-chaired conferences such as "Electrical Engineering (ICEE2004)" and "Machine Vision and Image Processing (MVIP2006)" in Iran. He has received multiple awards including Outstanding Graduate Student Award in 1999, and The Best Translation Award for the translation of "Digital Image Processing by Gonzales" from AmirKabir University, Iran, in 2005. His current research interests include video communications, biomedical signal processing, and data analysis.