Video Steganography with Perturbed Macroblock Partition

Hong Zhang
State Key Laboratory of Information Security
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, 100093, P.R.China
zhanghong@iie.ac.cn

Yun Cao*
State Key Laboratory of Information Security
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, 100093, P.R.China
caoyun@iie.ac.cn

Xianfeng Zhao
State Key Laboratory of Information Security
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, 100093, P.R.China
zhaoxianfeng@iie.ac.cn

Weiming Zhang
School of Information Science and Technology
University of Science and Technology of China
Hefei, 230026, P.R.China
zhangwm@ustc.edu.cn

Nenghai Yu
School of Information Science and Technology
University of Science and Technology of China
Hefei, 230026, P.R.China
ynh@ustc.edu.cn

ABSTRACT

In this paper, with a novel data representation named macroblock partition mode, an effective steganography integrated with H.264/AVC compression is proposed. The main principle is to improve the steganographic security in two directions. First, to embed messages, an internal process of H.264 compression, i.e., the macroblock partition, is slightly perturbed, hence the compression compliance is ensured. Second, to minimize the embedding impact, a high efficient double-layered structure is deliberately designed. In the first layer, the syndrome-trellis codes (STCs) is utilized to perform adaptive embedding, and the costs in visual quality and compression efficiency are both considered to construct the distortion model. In the second layer, facilitated by the wet paper codes (WPCs), an expected 3-bit per change gain in embedding efficiency is obtained.

Categories and Subject Descriptors

D.2.11 [**SOFTWARE ENGINEERING**]: Software Architectures—*Information hiding*; H.5.1 [**INFORMATION INTERFACES AND PRESENTATION**]: Multimedia Information Systems—*Video*

Keywords

Information hiding; video; steganography; H.264/AVC

*The corresponding author.

IH&MMSec'14, June 11–13, 2014, Salzburg, Austria.

http://dx.doi.org/10.1145/2600918.2600936.

1. INTRODUCTION

Modern steganography is the art and science of concealing the existence of the secret information into certain digital media. The hidden information should be undetectable, that is, the modified content should be perceptually and statistically (with respect to certain features) similar to its original unaltered counterpart [6].

This paper aims to design a novel steganographic methodology using digital videos as the cover media. Since digital video is one of the most influential media in our daily life, video transmission plays an ideal cloak of secret communication and provides sufficient payload capacity. The raw video is essentially a series of successive still images captured by optical devices. For the purpose of economical storage and efficient transmission, a variety of video compression technologies have been developed. It has been about 20 years since the MPEG (Motion Picture Expert Group) standard was established in 1993 [12] and MPEG-2 in 1995. Then in the pursuit of a better compression performance, H.264/AVC is developed [16] and has become one of the most commonly practiced video coding standard since 2003.

In most early video steganography, embedding is designed to take place prior to compression, and is applied directly to individual frame. However, such methodology is rarely adopted not only to avoid information lost caused by compression, but also to reduce the risk of being detected by highly-developed image-oriented steganalysis. As current video coding standards usually consist of several crucial processes, e.g., motion estimation, transformation, quantization and entropy coding, recent researches suggest to combine compression and information hiding together by directly manipulating certain coding process [15].

It is noticed that many recent high performance video steganography are inclined to utilizing the motion information, i.e., the motion vector (MV), as the data representation [10, 1, 11, 3, 4]. Although MV-based schemes have many advantages as high capacity and low quality degradation, it has a few inherent vulnerabilities which have already facilitated many targeted attacks. For instance, Zhang *et al.*

suggest that, if the embedding process can be modeled as an additive independent noise signal to the horizontal and vertical components, the statistical analysis of relative properties can be used to reveal the existence of hidden messages [19, 14]. Cao *et al.* implement video calibration for steganalysis and pointed out that, if a certain MV has been changed for embedding, the changed MV will show an inclination to revert to its prior value during re-compression [2]. Xu *et al.* make a point that in certain embedding scenarios, the mutual constraints of MVs will be destroyed [17] upon which they propose a steganalysis.

Faced with the situation stated above, we are motivated to search for other data representations to provide equivalent or even higher levels of steganographic security. Fortunately, with H.264, new opportunities for steganography can be found. As one distinguishing characteristic, H.264 allows each inter-macroblock to be further partitioned into smaller blocks of different sizes for inter-prediction. Correspondingly, an alternative data representation named "partition mode" (PM) is defined and chosen as the secret information carrier.

Definition 1. (**Partition Mode**). After partitioning macroblock (MB) into smaller blocks, the resultant partition form is defined as MB's partition mode.

The reasons for our choice are listed below. First, compression is an information-reducing process, the information required for MB partition can be exploited as the "side information" to help constructing a good distortion model for adaptive embedding. Second, other than MB partition, crucial processes, e.g., motion estimation, transformation, quantization, entropy coding, are not affected. Consequently, very limited losses of visual quality and coding efficiency would occur. Last but not least, to the best of our knowledge, no effective targeted-steganalyzer is found. The reliability of existing steganalytic models are likely to deteriorate when embedding with PM.

The prototype of PM-based data hiding schemes can be traced back to Kapotas and Skodras's work [13] which hides the scene change information by sequentially forcing the encoder to choose particular PMs. Similarly, Yang *et al.* suggest to make use of only sub-MB (with the size of 8×8) partitions [18]. Our studies show that, the existing schemes have several issues of concern. To start with, the existing schemes choose PMs arbitrarily which should be considered a serious violation of the coding principle, and the sequential embedding manner might drop the coding performance. Secondly, as analyzed in 4.3, the achieved embedding efficiency is not satisfactory. Consequently, steganalytic results in 4.4.3 demonstrated that the security level is affected to a certain degree.

In this paper, with the help of STCs [8] and WPCs [9], a ZZW-like [20] double-layered structure is designed to perform adaptive embedding during the process of MB partition. In the 1^{st} channel, each PM is assigned a distortion scalar considering the factors of visual quality and coding efficiency. For the purpose of introducing the minimal embedding impact with the given payload, syndrome-trellis coding is performed to determine the candidate set of MBs whose PM should be modified. Then the 2^{nd} channel can be built upon the coding results, and WPCs are used to embed additional messages. According to the analysis in 3.1, with the designed structure, an expected 3-bit per change gain in embedding efficiency is obtained compared to the STCs used. Moreover, by virtue of the STCs, the steganographer is free to design different distortion functions for different purposes without sharing it with the recipient. The experimental results demonstrate that, the proposed scheme can achieve satisfactory levels of coding performance and steganographic security with adequate payloads.

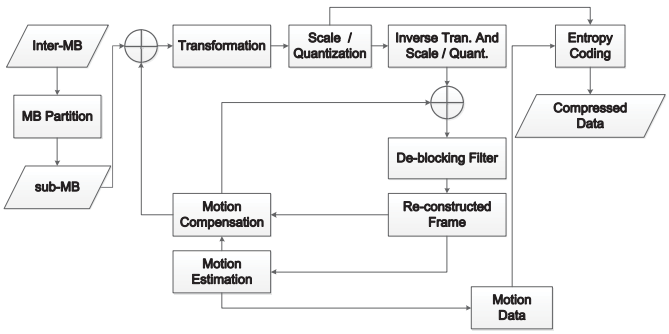

Figure 1: Structure of inter-MB coding.

The rest of the paper is structured as follows. In section 2, the basic concepts of the MB partition and the problem of distortion minimization are introduced. In section 3, the perturbed MB partition technique is presented, and we give detailed description of the double-layered embedding structure together with the analysis of embedding efficiency. In section 4, comparative experiments are conducted to show the performance of our scheme with special attention paid to the security evaluation. Finally in section 5, concluding remarks are given with some future research directions.

2. PRELIMINARIES AND NOTATIONS

2.1 MB Partition and Partition Mode

Like the other state-of-art video coding standards, H.264 reduces the temporal redundancy between frames by block based inter prediction. To be more specific, Figure 1 depicts the structure according to which an inter-MB is processed. At the very beginning, the currently coded frame is divided into non-overlapping 16×16 (in pixels) MBs. Then each MB is further partitioned into smaller blocks. After that, motion estimation is invoked for each block, and only the calculated MV along with the difference between blocks need to be further coded, e.g., DCT, quantization, and entropy coding.

As shown in Figure 2, H.264 supports seven different block sizes in inter prediction mode. As a result, there exists a two-level hierarchy inside the MB partition and the corresponding PMs can be further divided into two levels.

Definition 2. (**level-1 and level-2 PMs**). After partitioning a certain MB into smaller blocks, the resultant PM is called a **level-1 PM**, if only block sizes of 16×16, 16×8 or 8×16 are comprised, or a **level-2 PM**, if block sizes equal to or smaller than 8×8 are comprised.

Figure 3 gives examples of all level-1 PMs and some level-2 PMs. It is observed that, actually, one level-2 PM is comprised of four sub-PMs corresponding to its four 8×8 sub-MBs, and can be denoted by $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4)$.

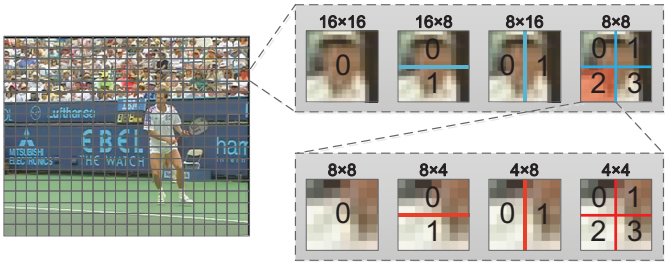

Figure 2: MB Partition.

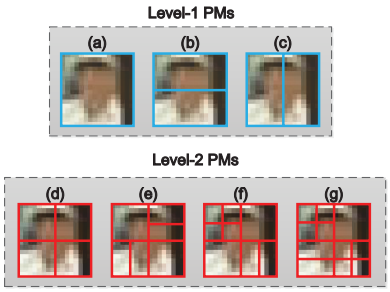

Figure 3: Examples of level-1 and level-2 PMs.

The PM decision is a trade-off between the visual quality and the coding efficiency. In this paper, we use

$$J(\mathbf{P}') = \beta SSD(\mathbf{P}') + \lambda R(\mathbf{P}') \tag{1}$$

to measure the cost of partitioning a certain MB in the form of $\mathbf{P}'$, where SSD is sum of the squared differences between the original and the reconstructed MBs, R reflects the number of bits associated with $\mathbf{P}'$, β and λ are weighting coefficients. Then a decision is made via

$$\mathbf{P} = \arg\min_{\mathbf{P}' \in \mathcal{J}} J(\mathbf{P}'), \tag{2}$$

where $\mathcal{J}$ is the set of all possible PMs.

2.2 Framework of Distortion Minimization

Without loss of generality, here we use a single inter-frame $\mathbb{F}$ with n inter-MBs as the cover. After MB partitions, the associated PMs are recorded as

$$\mathbb{P} = Partition(\mathbb{F}) = (\mathbf{P}_1, \ldots, \mathbf{P}_n). \tag{3}$$

Since the MBs' PMs are used as the data representation, $\mathbb{F}$ can be represented by $\mathbb{P}$. With a given relative payload α, a αn-bit message $\mathbf{m}$ is expected to be embedded by introducing modifications to some PMs in $\mathbb{P}$, and the resultant stego frame is expressed as $\mathbb{P}' = (\mathbf{P}'_1, \ldots, \mathbf{P}'_n)$. In this paper, the modifications are assumed to be mutually independent, and let every $\mathbf{P}_i$ be assigned a scalar γ_i expressing the distortion of replacing it with $\mathbf{P}'_i$, the overall embedding impact can be measured by the sum of per-element distortions

$$D(\mathbb{P}, \mathbb{P}') = \sum_{i=1}^{n} \gamma_i [\mathbf{P}_i \neq \mathbf{P}'_i], \tag{4}$$

here the Iverson bracket $[I]$ is defined to be 1 if the logical expression I is true and 0 otherwise.

Table 1: Binary codes of sub-PMs

sub-PM	Binary code
One 8×8 block	00
Two 8×4 blocks	01
Two 4×8 blocks	10
Four 4×4 blocks	11

In order to achieve a minimal distortion with the given payload, a flexible coding method named STCs can be leveraged to guide the embedding process. In fact, STCs are a kind of syndrome coding with which the embedding and extraction can be formulated as

$$\mathrm{Emb}(\mathbb{P}, \mathbf{m}) = \arg \min_{\mathcal{P}(\mathbb{P}') \in \mathcal{C}(\mathbf{m})} D(\mathbb{P}, \mathbb{P}'), \tag{5}$$

$$\mathrm{Ext}(\mathbb{P}') = \mathbb{H}\mathcal{P}(\mathbb{P}'). \tag{6}$$

Here, $\mathcal{P} : \mathcal{J} \to \{0,1\}$ can be any parity check function, and $\mathcal{P}(\mathbb{P}) = (\mathcal{P}(\mathbf{P}_1), \ldots, \mathcal{P}(\mathbf{P}_n))^T$. $\mathbb{H}$ is a parity-check matrix of the code $\mathcal{C}$, and $\mathcal{C}(\mathbf{m})$ is the coset corresponding to syndrome $\mathbf{m}$. In more detail, $\mathbb{H} \in \{0,1\}^{\alpha n \times n}$ is formed from a sub-matrix $\hat{\mathbb{H}} \in \{0,1\}^{h \times w}$, where h (called the *constraint height*) is a design parameter that affects the algorithm speed and efficiency and w is dictated by α [8].

3. PERTURBED MACROBLOCK PARTITION

In the proposed scheme, message embedding is implemented ultimate in the form of PM modification. We call our method perturbed macroblock partition (PMP) because during inter-frame coding the encoder (the process of MB partition) is slightly perturbed according to the coding result of the designed embedding structure.

3.1 The Double-layered Embedding Structure

Inspired by the ZZW construction [20], a double-layered structure is designed to offer two channels for embedding. With the 1^{st} channel, the STCs is used to fulfill adaptive embedding. Then with the 2^{nd} channel, WPCs is used to embed additional messages.

Under the designed structure, only level-2 PMs comprised of four sub-PMs are utilized. According to the mapping defined in Table 1, each sub-PM is assigned a 2-bit code, thus a level-2 PM can be expressed as an 8-bit vector. For example, the PM (e) in Figure 3 can be expressed as "00100100" and (g) "11001011".

Suppose the steganographer uses a cover $\mathbb{P}$ comprised of n PMs which is written as a binary matrix of the size $n \times 8$

$$\begin{array}{rcccccc} \mathbf{P}_1 & = & p_{1,1} & p_{1,2} & \ldots & p_{1,8} \\ \mathbf{P}_2 & = & p_{2,1} & p_{2,2} & \ldots & p_{2,8} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{P}_n & = & p_{n,1} & p_{n,2} & \ldots & p_{n,8}, \end{array} \tag{7}$$

and the two embedding channels is constructed as follows.

1^{st} **embedding channel**: A parity check function $\mathcal{P} : \mathcal{J}_2 \to \{0,1\}$ is used to compress $\mathbb{P}$ into the 1^{st} channel $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, where $\mathcal{P}$ is defined as

$$\mathcal{P}(\mathbf{P}) = \oplus_{i=1}^{8} p_i, \tag{8}$$

$\mathcal{J}_2$ is the set of all possible level-2 PMs and $x_i = \mathcal{P}(\mathbf{P}_i)$.

Given a relative payload α, the constructed STCs is used to embed αn message bits into the 1^{st} channel, and the number of bits flipped is recorded as r.

2^{nd} **embedding channel**: Take the first 7 bits from each PM, and write them as

$$\begin{array}{rcccccc} \tilde{\mathbf{P}}_1 & = & p_{1,1} & p_{1,2} & ... & p_{1,7} \\ \tilde{\mathbf{P}}_2 & = & p_{2,1} & p_{2,2} & ... & p_{2,7} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{P}}_n & = & p_{n,1} & p_{n,2} & ... & p_{n,7}. \end{array} \tag{9}$$

If $x_i \in \mathbf{x}$ needs to be flipped, any bit in $\mathbf{P}_i$ is allowed to be flipped. As a result, $\mathbf{P}_i$ can be mapped into any 3-bit vector by $\mathbb{H}_h \tilde{\mathbf{P}}_i^T$, where $\mathbb{H}_h$ is the parity check matrix of the $[7, 4]$ Hamming code. Then a wet paper channel can be constructed as

$$\mathbf{y} = (\tilde{\mathbf{P}}_1 \mathbb{H}_h^T, \tilde{\mathbf{P}}_2 \mathbb{H}_h^T, ..., \tilde{\mathbf{P}}_n \mathbb{H}_h^T), \tag{10}$$

and $3r$ additional message bits are expected to be embedded via wet paper coding [1].

With n level-2 PMs, totally $\alpha n + 3r$ message bits are expected to be embedded at the cost of r PM modifications, Correspondingly, the achieved embedding efficiency can be calculated as

$$e_{\text{PMP}} = \frac{\alpha n + 3r}{r} = e_{\text{STCs}} + 3. \tag{11}$$

It is noticed that, compared to the pure STCs, an expected 3-bit per change gain is obtained.

3.2 Distortion Definition

Under the framework described in 2.2, with every $\mathbf{P}_i \in \mathbb{P}$ be assigned a scalar γ_i expressing its embedding impact, the overall embedding impact can be measured by the sum of per-element distortions. Then the formulation of the scalar γ_i has become the chief problem of the adaptive steganography designing.

Suppose that after the 1^{st} channel embedding, the t^{th} bit x_t needs to be flipped. According to (8), this can be achieved by flipping any bit within $\mathbf{P}_t$. However, the steganographer is not free to choose which bit to flip since it is determined by the wet paper and Hamming coding result. In other words, it is possible for $\mathbf{P}_t$ to be changed into any PM in the set $\mathcal{K}_t = \{\mathbf{P} || w(\mathbf{P}_t) - w(\mathbf{P})| = 1\}$, where $w(\mathbf{P})$ is the Hamming weight of $\mathbf{P}$.

Since the PM modification is uncontrollable, the embedding impact of $\mathbf{P}_i$ should be measured by the maximum cost of replacing it with any PM in $\mathcal{K}_i$. Therefore γ_i is defined as

$$\gamma_i = \max\{J(\mathbf{P}_i) - J(\mathbf{P}) | \mathbf{P} \in \mathcal{K}_i\}. \tag{12}$$

3.3 Communication with Single Inter-frame

To better explain how the double-layered embedding structure is applied, this subsection gives detailed description of the communication with single inter-frame.

Suppose the steganographer has one frame $\mathbb{F}$ to be compressed in the inter-mode, and wants to communicate the message $\mathbf{m}$, then the PMP embedding process is carried out in the following 3 steps:

[1]For conciseness and without loss of generality, we assume that the capacity of the wet paper channel equals to its dry-spot number.

Pre-macroblock partition: Apply macroblock partition to $\mathbb{F}$. Meanwhile, record all the level-2 PMs $\mathbb{P} = (\mathbf{P}_1, \ldots, \mathbf{P}_n)$ and compute the associated distortion scales $\Gamma = (\gamma_1, \ldots, \gamma_n)$ using (12).

Double-layered embedding: Perform the double-layered embedding process to determine which PMs in $\mathbb{P}$ have to be changed and how the modifications should apply. With α denotes the relative payload, $\hat{\mathbb{H}}$ denotes a sub-matrix and K the seed of a pseudo-random number generator, the details are given in **Algorithm 1**.

Algorithm 1 Double-layered embedding with single inter-frame

Require: Input $\mathbb{P}$, Γ, α, $\hat{\mathbb{H}}$, K and $\mathbf{m}$
Ensure: Output $\mathbb{P}'$ and r
1: compress $\mathbb{P}$ into the 1^{st} channel buffer $\mathbf{x}$ using (8);
2: generate the STCs' parity check matrix $\mathbb{H}_s$ with α and $\hat{\mathbb{H}}$;
3: perform syndrome coding to embed αn message bits by modifying $\mathbf{x}$ to $\mathbf{x}'$;
4: record the number of flipped bits in $\mathbf{x}$ as r and the indexes of changed positions as $(I_1, \ldots, I_r)$;
5: construct the 2^{nd} channel buffer $\mathbf{y}$ with $\mathbb{P}$ and $\mathbb{H}_h$ using (10);
6: generate the WPCs' parity check matrix $\mathbb{H}_w \in \{0, 1\}^{3r \times 3n}$ with the seed K;
7: perform wet paper coding to embed $3r$ message bits by modifying $\mathbf{y}$ to $\mathbf{y}'$;
8: **for** $i = 1$ to r **do**
9: calculate the index j of the bit to be flipped in $\mathbf{P}_{I_i}$;
10: change $\mathbf{P}_{I_i}$ into $\mathbf{P}'_{I_i}$ by flipping $p_{I_i,j}$;
11: **end for**

Perturbed macroblock partition: Perform macroblock partitions to $\mathbb{F}$ according to the modified PMs.

Then further encoding processes are continued to generate the compressed stego frame $\mathbb{F}'$. Note that before $\mathbb{F}'$ is emitted, the steganographer has to share some parameters with the intended recipient as the secret key including α, $\hat{\mathbb{H}}$, K and r.

As to the recipient, he will first decompress the received stego frame $\mathbb{F}'$ to get $\mathbb{P}' = (\mathbf{P}'_1, \ldots, \mathbf{P}'_n)$, then extract the secret messages as described in **Algorithm 2**.

Algorithm 2 Extraction with single inter-frame

Require: Input $\mathbb{P}'$, α, $\hat{\mathbb{H}}$, K, and r
Ensure: Output $\mathbf{m}$
1: compress $\mathbb{P}'$ into the 1^{st} channel buffer $\mathbf{x}'$ using (8);
2: generate the STCs' parity check matrix $\mathbb{H}_s$ with α and $\hat{\mathbb{H}}$;
3: $\mathbf{m}_1 \Leftarrow \mathbb{H}_s \mathbf{x}'^T$;
4: construct the 2^{nd} channel buffer $\mathbf{y}'$ with $\mathbb{P}'$ and $\mathbb{H}_h$ using (10);
5: generate the WPCs' parity check matrix $\mathbb{H}_w \in \{0, 1\}^{3r \times 3n}$ with K;
6: $\mathbf{m}_2 \Leftarrow \mathbb{H}_w \mathbf{y}'^T$;
7: $\mathbf{m} \Leftarrow [\mathbf{m}_1\ \mathbf{m}_2]$

3.4 Communication with Video Sequence

One dominant advantage of video data as the cover object is its huge capacity. But for security reasons, each inter-

frame offers a very limited capacity. So the payloads have to be shared in practice.

In order to communicate message $\mathbf{m}$ with a relatively large size, suppose the steganographer always has sufficient covers $\mathbb{V} = (\mathbb{F}_1, \mathbb{F}_2, \ldots)$ and has shared α, $\hat{\mathbb{H}}$ and K to the recipient as the secret key. Note that in order to generate the WPCs' parity check matrix, the recipient has to be informed of the message length. As a solution to this problem, for the i^{th} frame to be compressed in the inter-mode, the number of flipped bits in its 1^{st} channel r_i is stored as a binary vector with a fixed length l and embedded with message bits alternately. For example, when embedding with $\mathbb{F}_i$, r_{i+1} is assessed in advance and then embedded into $\mathbb{F}_i$'s 2^{nd} channel with other message bits. Specific to $\mathbb{F}_1$, only a l-bit vector indicating r_2 is embedded into its 2^{nd} channel without any message bits.

4. PERFORMANCE EXPERIMENTS

4.1 Experiment Setup

Our experimental environment is based on the H.264/AVC reference encoder software JM 18.5, created by the joint video team (JVT). The baseline profile is used in compression which supports only I and P frames. To implement the PMP scheme, with the relative payload α set to 1/2 and constraint height h set to 7, a good STCs listed in [7] is used to perform the 1^{st} channel embedding. Besides, Yang *et al.*'s method is also implemented for comparison. As shown in Figure 4, 14 standard CIF sequences in the 4:2:0 YUV format are selected for tests. The frame size varies from 90 to 376 at the frame rate of 30 frame per second. All sequences are compressed by the standard encoder (referred to as STD) to produce the class of clean videos. On the other hand, for Yang's method and PMP, all sequences are subjected to compression with random messages embedded to create the class of stego videos, and the achieved embedding strength vary from 80 to 200 bits per inter-frame.

4.2 Impacts on Coding Performance

The embedding impacts on coding performance is evaluated from two aspects, i.e., the visual quality and compression efficiency, which are measured by PSNR and the average bit-rate respectively. Corresponding results are recorded in Table 2. What's more, we take a closer look at one specific sequence "stefan.yuv" and plot the dynamic changes in PSNR and the percentage of bit-rate increase compared to the STD along frames in Figure 5 and Figure 6. It is observed that, both Yang's and our PMP scheme affect the visual quality very slightly, and PMP outperforms its competitor for it introduces less bit-rate increases.

4.3 Embedding Efficiency

With PMP, as discussed in 3.1, an expected 3-bit per change gain in embedding efficiency is obtained compared to the pure STCs.

With Yang's method, the encoder is forced to partition a sub-MB choose a particular sub-PM according to the 2-bit to be embedded. Since each sub-PMs has a 1 in 4 chance of not being changed, the corresponding embedding efficiency can be calculated as

$$e_{\text{Yang's}} = \frac{2}{1/4 \times 0 + 3/4 \times 1} = \frac{8}{3}. \qquad (13)$$

Table 2: Test results. (SN (Sequence Name), FN (Frame Number), EM (Embedding Method), SP (Secret Payload (kbit)), PSNR (dB), BR (Bit-Rate (kbit/s)), EE (Embedding Efficiency)).

SN	FN	EM	SP	PSNR	BR	EE
stefan	90	STD	N/A	36.684	1415.47	N/A
		Yang's	14.42	36.713	1441.87	2.67
		PMP	14.42	36.684	1420.38	5.96
foreman	300	STD	N/A	37.166	532.08	N/A
		Yang's	24.71	37.169	541.07	2.67
		PMP	24.71	37.165	535.73	6.14
city	300	STD	N/A	35.795	477.26	N/A
		Yang's	23.90	35.809	485.27	2.67
		PMP	23.90	35.800	478.75	5.97
bus	150	STD	N/A	35.980	1443.11	N/A
		Yang's	24.87	35.985	1460.57	2.67
		PMP	24.87	35.977	1447.25	5.92
crew	300	STD	N/A	38.066	1105.52	N/A
		Yang's	44.37	38.071	1123.10	2.67
		PMP	44.37	38.068	1112.09	6.28
coastguard	300	STD	N/A	35.694	1338.14	N/A
		Yang's	43.94	35.700	1352.65	2.67
		PMP	43.94	35.693	1343.35	6.19
ice	240	STD	N/A	40.734	440.66	N/A
		Yang's	21.01	40.747	450.98	2.67
		PMP	21.01	40.737	443.67	5.92
football	260	STD	N/A	37.155	1715.63	N/A
		Yang's	42.74	37.163	1734.74	2.67
		PMP	42.74	37.160	1723.93	6.10
soccer	300	STD	N/A	36.835	816.81	N/A
		Yang's	30.83	36.848	829.31	2.67
		PMP	30.83	36.840	821.94	6.03
harbour	300	STD	N/A	35.515	1747.95	N/A
		Yang's	62.20	35.509	1765.33	2.67
		PMP	62.20	35.511	1751.91	6.11
tempete	260	STD	N/A	36.063	1502.69	N/A
		Yang's	50.01	36.068	1518.96	2.67
		PMP	50.01	36.061	1506.37	6.14
walk	376	STD	N/A	38.614	1074.30	N/A
		Yang's	47.36	38.620	1090.28	2.67
		PMP	47.36	38.610	1078.33	6.10
flower	250	STD	N/A	36.051	1947.45	N/A
		Yang's	45.24	36.049	1964.67	2.67
		PMP	45.24	36.053	1952.32	6.06
mobile	300	STD	N/A	35.227	1919.92	N/A
		Yang's	59.93	35.243	1938.90	2.67
		PMP	59.93	35.235	1925.34	6.06

Figure 4: Sequences used.

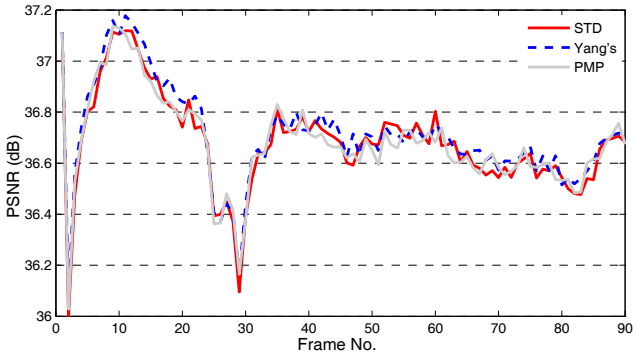

Figure 5: Dynamic changes in PSNR.

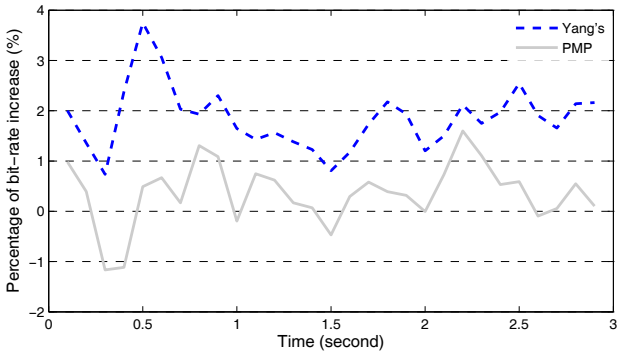

Figure 6: Dynamic changes in percentage of bit-rate increase.

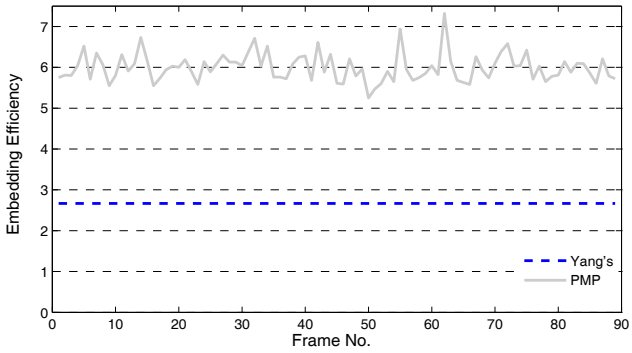

Figure 7: Dynamic changes in embedding efficiency.

After embedding with different sequences, the achieved average embedding efficiencies are recorded in Table 2, and the dynamic changes along frames of "stefan.yuv" are plotted in Figure 7.

4.4 Steganalysis

4.4.1 Steganalytic Features

To the best of our knowledge, no effective steganalysis against PM-based schemes is proposed so far. In order to test the steganographic security of the PM-based schemes, the idea of "video calibration" is adopted to design a targeted steganalytic feature set. For those MV-based schemes, it is proved that the modified MVs have the inclination to revert during recompression [2]. Analogically, we wonder whether the PMs have such inclination which can be used to reveal the fact of embedding. To test this idea, a 20-d feature vector is designed as follows:

Considering only sub-PMs are indeed modified, we pay attention to the changes in sub-PMs before and after recompression. According to Table 1, we define 4 states corresponding to the 4 different sub-PMs, i.e., s_0, s_1, s_2 and s_3. Note that, it is also possible that recompression turns some level-2 PMs into level-1 ones, so a state s_4 is defined to cover any other states. We write an imperfect transition probability matrix $\mathbb{M}$ to describe the state transitions before and after recompression as

$$\begin{array}{lllll} Pr(0,0) & Pr(0,1) & Pr(0,2) & Pr(0,3) & Pr(0,4) \\ Pr(1,0) & Pr(1,1) & Pr(1,2) & Pr(1,3) & Pr(1,4) \\ Pr(2,0) & Pr(2,1) & Pr(2,2) & Pr(2,3) & Pr(2,4) \\ Pr(3,0) & Pr(3,1) & Pr(3,2) & Pr(3,3) & Pr(3,4) \end{array} \quad (14)$$

where $Pr(i,j)$ denotes the probability of s_i to s_j state transition, and compose all the elements in $\mathbb{M}$ into a 20-d fea-

Table 3: Steganalysis Results (%).

	STMB		MVRB	
	TN	TP	TN	TP
Yang's	61.0	72.0	50.2	53.7
PMP	40.5	76.0	52.3	53.1

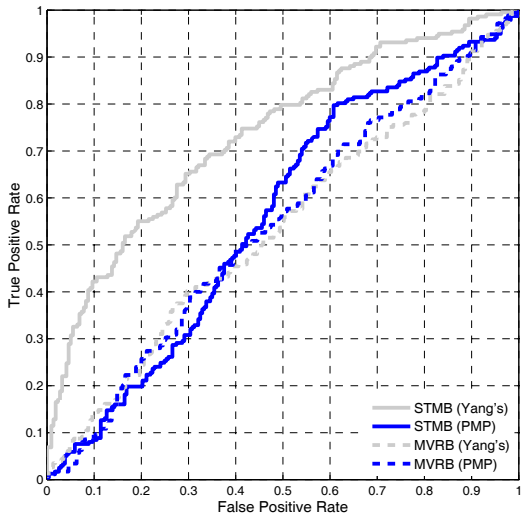

Figure 8: ROC curves of the used steganalyzers.

ture vector for steganalysis. The obtained features are then named STMB (state transition matrix-based) features.

In addition, Cao *et al*'s MVRB (motion vector reversion-based) features [2] are also leveraged to test whether detectable changes in MV domain are introduced.

4.4.2 *Training and Classification*

In our steganalysis, 9 pairs of compressed sequences (clean and stego) are randomly selected for training purposes, and the remaining 5 are left for testing. A fixed 8-frame sliding window is used to scan each sequence without overlapping, and the steganalytic features are extracted from the frames within the window. The classifier is implemented using Chang's support vector machine (SVM) [5] with the polynomial kernel.

4.4.3 *Steganalytic Results*

The true negative (TN) rates, true positive (TP) rates are computed by counting the number of detections in the test sets. The performances of the steganalyzers with two feature sets are tested, and results are recorded in Table 3. Besides, the detector receiver operating characteristic (ROC) curves of the two steganalyzers are plotted in Figure 8.

It is observed that with the considered embedding strength, the MVRB features cannot reliably detect the PM-based schemes, and PMP outperforms its competitor when attacked by the targeted steganalyzer with STMB features. We can infer that, arbitrary and sequential PM modifications may cause serious deviations from the optimal coding results, which may facilitate targeted attacks.

5. CONCLUSIONS AND FUTURE WORK

This paper presents a video steganography tightly combined with H.264 compression. A novel data representation called PM is defined and utilized to convey secret messages. To perform data hiding, optimized perturbations are introduced to the process of MB partition under a high efficient double-layered structure. Experimental results show that, satisfactory levels of coding performance and security are achieved with adequate payloads.

In the near future, the PMP scheme would be further optimized by testing on different distortion functions and embedding structures. Meanwhile, attempts of further steganalysis are to be carried out under more complicated steganalytic models to ensure security.

6. ACKNOWLEDGMENTS

The work on this paper was supported by the NSF of China under 61303259, 61170281 and 61303254, the Strategic Priority Research Program of the Chinese Academy of Sciences under XDA06030600, and the IIE's Research Project on Cryptography under Y3Z0012102.

7. REFERENCES

[1] H. Aly. Data hiding in motion vectors of compressed video based on their associated prediction error. *IEEE Transactions on Information Forensics and Security*, 6(1):14–18, 2011.

[2] Y. Cao, X. Zhao, and D. Feng. Video steganalysis exploiting motion vector reversion-based features. *IEEE Signal Processing Letters*, 19(1):35–38, 2012.

[3] Y. Cao, X. Zhao, D. Feng, and R. Sheng. Video steganography with perturbed motion estimation. In *Proc. 13th Information Hiding Conf.*, volume 6958 of *Lecture Notes in Computer Science*, pages 193–207, 2011.

[4] Y. Cao, X. Zhao, F. Li, and N. Yu. Video steganography with multi-path motion estimation. In *Proc. SPIE, Media Watermarking, Security, and Forensics*, volume 8665, pages 86650K–86650K–6, 2013.

[5] C. Chang and C. Lin. Libsvm – a library for support vector machines, 2013.

[6] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography, 2nd ed.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2008.

[7] T. Filler, J. Judas, and J. Fridrich. Minimizing embedding impact in steganography using trellis-coded quantization. In *Proceedings SPIE, Electronic Imaging, Security and Forensics of Multimedia XII*, volume 7541, pages 1–14, 2010.

[8] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *Information Forensics and Security, IEEE Transactions on*, 6(3):920–935, 2011.

[9] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography with wet paper codes. In *Proceedings of the 2004 workshop on Multimedia & security*, MM&Sec'04, pages 4–15, 2004.

[10] Y. Guo and F. Pan. Information hiding for h.264 in video stream switching application. In *Proc. IEEE Int. Conference on Inform. Theory and Inform. Security*, pages 419–421, 2010.

[11] B. Hao, L. Zhao, and W. Zhong. A novel steganography algorithm based on motion vector and matrix encoding. In *Proc. IEEE 3rd International Conference on ICCSN*, pages 406–409, 2011.

[12] ISO. *Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s-Part 1:System*. Int. Organization for Standardization, Geneva, Switzerland, 1993.

[13] S. Kapotas and A. Skodras. A new data hiding scheme for scene change detection in h.264 encoded video sequences. In *Multimedia and Expo, 2008 IEEE International Conference on*, pages 277–280, 2008.

[14] Y. Su, C. Zhang, and C. Zhang. A video steganalytic algorithm against motion-vector-based steganography. *Signal Process.*, 91(8):1901–1909, 2011.

[15] Y. Tew and K. Wong. An overview of information hiding in h.264/avc compressed video. *IEEE Trans. Circuits Syst. Video Technol.*, 2013.

[16] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):560–576, Jul. 2003.

[17] X. Xu, J. Dong, W. Wang, and T. Tan. Video steganalysis based on the constraints of motion vectors. In *Proc. IEEE International Conference on Image Processing*, 2013.

[18] X. Yang, L. Zhao, and K. Niu. An efficient video steganography algorithm based on sub-macroblock partition for h.264/avc. *Advanced Materials Research*, pages 5384–5389, 2012.

[19] C. Zhang, Y. Su, and C. Zhang. A new video steganalysis algorithm against motion vector steganography. In *Proc. 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, 2008.

[20] W. Zhang, X. Zhang, and S. Wang. Maximizing steganographic embedding efficiency by combining hamming codes and wet paper codes. In *Proc. 10th Information Hiding Conf.*, volume 5284 of *Lecture Notes in Computer Science*, pages 60–71, 2008.