A Steganalytic Algorithm to Detect DCT-based Data Hiding Methods for H.264/AVC Videos

Peipei Wang
State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China 100093
School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China 100093
wangpeipei@iie.ac.cn

Yun Cao
State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China 100093
School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China 100093
caoyun@iie.ac.cn

Xianfeng Zhao*
State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China 100093
School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China 100093
zhaoxianfeng@iie.ac.cn

Meineng Zhu
Beijing Institute of Electronics Technology and Application
No. 15 Xinjiangongmen Rd
Beijing, China 100091
zmneng@163.com

ABSTRACT

This paper presents an effective steganalytic algorithm to detect Discrete Cosine Transform (DCT) based data hiding methods for H.264/AVC videos. These methods hide covert information into compressed video streams by manipulating quantized DCT coefficients, and usually achieve high payload and low computational complexity, which is suitable for applications with hard real-time requirements. In contrast to considerable literature grown up in JPEG domain steganalysis, so far there is few work found against DCT-based methods for compressed videos. In this paper, the embedding impacts on both spatial and temporal correlations are carefully analyzed, based on which two feature sets are designed for steganalysis. The first feature set is engineered as the histograms of noise residuals from the decompressed frames using 16 DCT kernels, in which a quantity measuring residual distortion is accumulated. The second feature set is designed as the residual histograms from the similar blocks linked by motion vectors between inter-frames. The experimental results have demonstrated that our method can effectively distinguish stego videos undergone DCT manipulations from clean ones, especially for those of high qualities.

*Corresponding author

IH&MMSec '17, June 20–22, 2017, Philadelphia, PA, USA

DOI: http://dx.doi.org/10.1145/3082031.3083245

KEYWORDS

Steganalysis; data hiding; video; DCT; H.264/AVC

ACM Reference format:
Peipei Wang, Yun Cao, Xianfeng Zhao, and Meineng Zhu. 2017. A Steganalytic Algorithm to Detect DCT-based Data Hiding Methods for H.264/AVC Videos. In *Proceedings of IH&MMSec '17, June 20–22, 2017, Philadelphia, PA, USA, ,* 11 pages.
DOI: http://dx.doi.org/10.1145/3082031.3083245

1 INTRODUCTION

The goal of steganalysis is to reveal the existence of secret information embedded into innocent-looking media. Relevant techniques are developed to cope with the abuse of steganography. Nowadays, facilitated by the advanced video compression and network technology, the compressed digital video has become one of the most influential media and also one of the most suitable cover for data hiding. Hence video targeted steganalysis has attracted much attention recently.

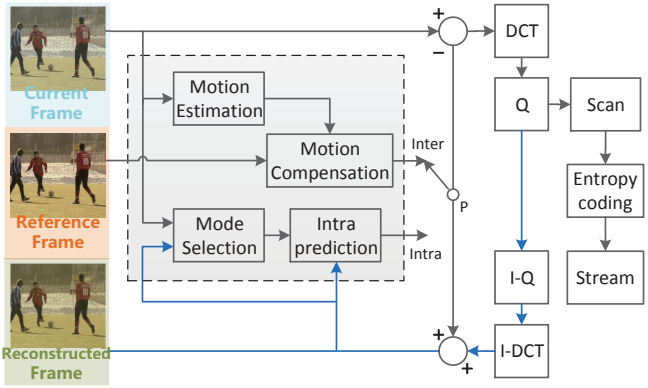


Figure 1: The compression process of H.264/AVC.

H.264/AVC [1] is developed in the pursuit of better compression performance and has become one of the most commonly practiced video coding standards. Up to date data hiding methods are usually integrated into the video compression process (Figure 1). Such approaches hide information by modifying certain output coefficients during compression procedure, such as motion vectors (MV) [3, 17, 20], inter prediction modes [8, 21], quantized DCT coefficients [11–14, 18] and variable length codes [9, 10].

This paper focuses on attacking DCT-based data hiding methods which are designed to satisfy hard real-time requirements. By directly modifying the quantized DCT coefficients of compressed video, the computationally expensive of full decoding and re-compression processes can be avoided. Several typical DCT-based methods [11–14, 18] have been proposed in video data hiding. By embedding information into the non-zero quantized DCT coefficients after zigzag scanning, Nakajima *et al.* [14] proposed the high capacity data hiding method. And in [18], both of the Mquant and quantized DCT coefficients are simultaneously manipulated with the attempt of preserving video quality. However, above methods all suffer from distortion drift of intra-frame. In order to avoid this problem, Ma *et al.* [12] utilized several paired-coefficients of each 4×4 DCT block for data embedding and the intra-frame distortion compensation was also implemented. And in [13], they further accumulated the distortion introduce by paired-coefficients' modifications and employed the directions of intra-frame prediction to avert the distortion drift. Recently, Lin *et al.* [11] proposed an improved method by perturbing the coefficient-pairs defined by a new set of sifted 4×4 blocks. In contrast to [13], this algorithm preserved the stego video's visual quality as well as increased the embedding capacity greatly. The problem of above DCT-based data hiding methods is that, the modifications of quantized DCT coefficients will lead to certain pixels' change in uncompressed stream, which leaves the opportunity of being detected by steganalysis.

A series of steganalytic methods [5, 15, 19] have been developed by exploiting the spatial or temporal correlation in video stream. Pankajakshan *et al.* [15] investigated the temporal correlation by extracting features from residual frames after temporal prediction. And Zarmehi *et al.* [19] estimated the cover frames and computed features both from video frames and residual matrix. In [5], in order to combine spatial correlation with temporal correlation among frames, the gray-level co-occurrence matrix between blocks was utilized to establish markov model of inter-frames. However, all above steganalytic approaches are not specialized in detecting DCT-based data hiding methods. Although they perform poorly in DCT-based steganalysis, these methods guide our approach against video data hiding. Moreover, the research of data hiding using quantized DCT coefficients has been well studied in image steganography. The effective schemes in JPEG image steganalysis [6, 7] also inspire the design of the proposed steganalytic feature.

The main contribution of this paper is the proposal of the first targeted steganalytic approach for data hiding in H.264/AVC videos by directly manipulating quantized DCT coefficients. By considering both the embedding impact on both spatial and temporal correlation, two feature sets are designed for training and classification. The intra-frame feature set is engineered as the residual histograms using DCT kernels and a quantity measuring embedding distortion is accumulated in the histograms. Considering the correlation of blocks between inter-frames, the second feature set is computed from the similar blocks linked by MVs.

The rest of this paper is organized as follows. The preliminaries are introduced in Section 2 and the methods of DCT-based data hiding are modeled is Section 3. In Section 4, the spatial and temporal correlations are elaborated and the design of final feature is also presented. Section 5 shows the experimental results and followed by the conclusions and future works given in Section 6.

2 PRELIMINARIES

2.1 Intra-Frame Coding

As the integral part of H.264/AVC compression [1], intra-frame prediction is employed to reduce the spatial redundancy in video frames. Intra-frame prediction algorithm used in H.264/AVC provides two sizes of luminance block: 4×4 and 16×16. Because 16×16 luminance block usually varies smoothly and human eyes are very sensitive to the change of luminance values, the existing data hiding methods only consider the 4×4 blocks in I frames. There are nine prediction modes for each 4×4 luminance block, which is shown in Figure 2. During intra-frame prediction, each current block is predicted based on the previously encoded adjacent block. As shown in Figure 3, $B_{i,j}$ is the current 4×4 block with pixels labeled as a to p. By using an optimal prediction mode of the nine modes, the prediction block $B^p_{i,j}$ is obtained. The reference pixels (A to M) located in the reconstructed neighboring blocks are utilized for prediction. The difference matrix between prediction block and original block constructs the residual block $R_{i,j}$, which undergoes the following compression process.

The integer DCT and quantization operation applied to $R_{i,j}$ can be formulized as follows.

$$\boldsymbol{R}^{QDCT}_{i,j} = (\boldsymbol{C}_f \boldsymbol{R}_{i,j} \boldsymbol{C}_f^T) \otimes \frac{\boldsymbol{E}_f}{Q_{step}} = \begin{bmatrix} r_{00} & r_{01} & r_{02} & r_{03} \\ r_{10} & r_{11} & r_{12} & r_{13} \\ r_{20} & r_{21} & r_{22} & r_{23} \\ r_{30} & r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{1}$$

where

$$\boldsymbol{C}_f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

$$\boldsymbol{E}_f = \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix}$$

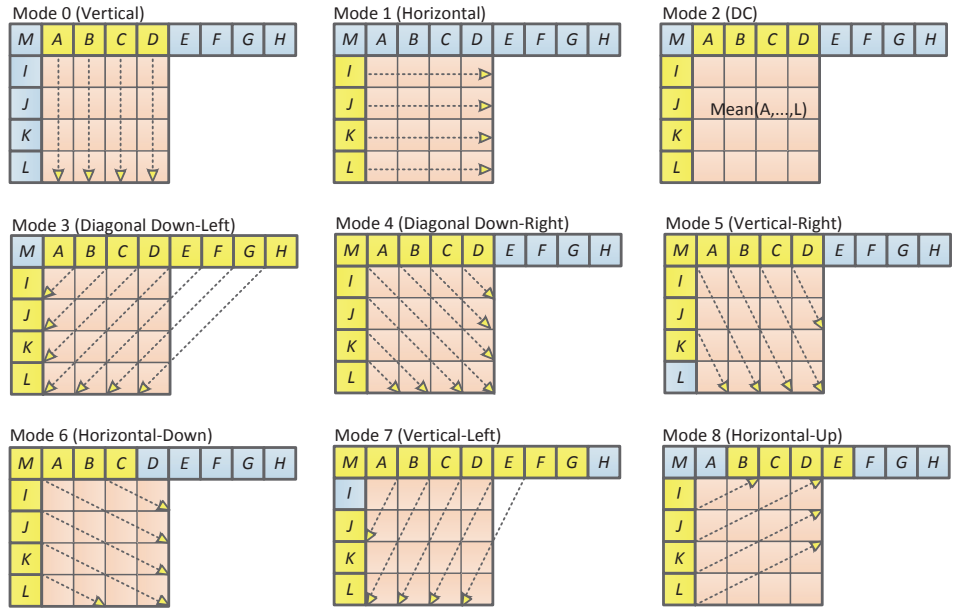


Figure 2: Nine prediction modes for 4×4 luma block.

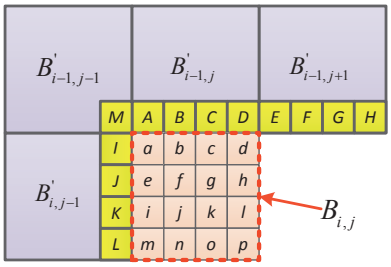


Figure 3: The predicted 4×4 luma block and the reference pixels in neighboring blocks.

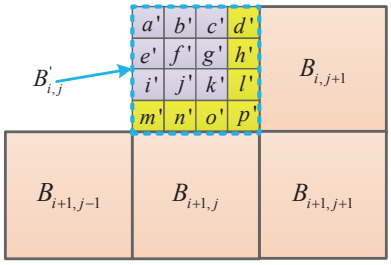


Figure 4: The reconstructed 4×4 luma block and neighboring blocks affected by intra-frame prediction.

and $\otimes$ denotes the product operation of the corresponding elements, Q_{step} is the quantizer step size determined by quantization parameter (QP), a and b are real numbers assigned by $a = 1/2$ and $b = \sqrt{2/5}$ respectively.

2.2 Distortion Drift

Current DCT-based data hiding methods [11–14, 18] embed information by perturbing $R_{i,j}^{QDCT}$ parameter, which is the quantized DCT coefficient of $R_{i,j}$. The reconstructed block is derived by $B'_{i,j} = B^p_{i,j} + R'_{i,j}$. $R'_{i,j}$ denotes the reconstructed residual block, which is obtained by inverse DCT transforming and dequantizing $R_{i,j}^{QDCT}$. As depicted in Figure 4, four neighboring blocks are affected by current reconstructed block because the pixels in bottom row and right column of $B'_{i,j}$ are utilized for their intra-frame predictions. After embedding information into $R_{i,j}^{QDCT}$, $R_{i,j}^{QDCTs}$ is obtained and the corresponding reconstructed block is $B'^{s}_{i,j}$. If $\{m', n', o', p'\}$ of $B'^{s}_{i,j}$ are changed and the prediction mode of neighboring block $B_{i+1,j}$ is Mode 0, the distortion will drift to $B'_{i+1,j}$.

3 MODELING OF DCT-BASED DATA HIDING METHODS

In order to avoid the distortion drift, the existing data hiding methods [11–13] classify the 4×4 blocks according to their neighboring blocks' prediction modes and then provide the qualified paired-coefficients under different conditions. The distortion drift occurs only if $\{d', h', l', p', m', n', o', p'\}$ are reference pixels in neighboring blocks' prediction. If the prediction mode of $B_{i,j+1}$ satisfies $PM_{i,j+1} = \{1, 2, 4, 5, 6, 8\}$, $\{d, h, l, p\}$ are used in prediction process. $\{m, n, o, p\}$ are reference pixels if $PM_{i+1,j} = \{0, 2, 3, 4, 5, 6, 7\}$ and $PM_{i+1,j-1} = \{3, 7\}$. And $PM_{i+1,j+1} = \{4, 5, 6\}$ indicates that pixel p is referred for the prediction of $B_{i+1,j+1}$. The currently optimal approach [11] of categorizing the 4×4 blocks is recorded

Table 1: Five categories of 4×4 block used in DCT-based data hiding

Category	$PM_{i,j+1}$	$PM_{i+1,j}$	$PM_{i+1,j-1}$	$PM_{i+1,j+1}$
1st	{1,2,4,5,6,8}	{1,8}	{0,1,2,4,5,6,8}	ANY
2st	{0,3,7}	{0,2,3,4,5,6,7}	{3,7}	ANY
3st	{0,3,7}	{1,8}	{0,1,2,4,5,6,8}	{4,5,6}
4st	{1,2,4,5,6,8}	{0,2,3,4,5,6,7}	{3,7}	ANY
5st	{0,3,7}	{1,8}	{0,1,2,4,5,6,8}	{0,1,2,3,7,8}

in Table 1. Based on whether the seven pixels are used in intra-prediction of neighboring blocks, every block to be embedded is classified to one of the five categories.

In DCT-based data hiding methods, different coefficient-pairs of matrix $R_{i,j}^{QDCT}$ are modified in different categories. For the first category, the $\{d', h', l', p'\}$ of $B'_{i,j}$ are reference pixels in prediction. In order to prevent distortion drift, the modifications of $R_{i,j}^{QDCT}$'s coefficients should not perturb the right column pixels. The paired-coefficients of $\{(r_{00}, r_{02}), (r_{10}, r_{12}), (r_{20}, r_{22}), (r_{30}, r_{32})\}$ are qualified for embedding. Similarly, if $B_{i,j}$ belongs to second category, $\{m', n', o', p'\}$ will be employed in prediction and $(r_{01}, r_{21}), (r_{02}, r_{22}), (r_{03}, r_{23}), (r_{00}, r_{20})$ are modified to embed information. Because only p' is referred in prediction of the block in category three, the coefficient-pairs can also be obtained through the above method. For category four, restricted by the circumstance that all seven related pixels will be used in prediction, one bit can only be embedded through modifying $(r_{00}, r_{02}, r_{20}, r_{22})$ to $(r_{00}+1, r_{02}-1, r_{20}-1, r_{22}+1)$. In the fifth category, it is unnecessary to consider distortion drift because none of $\{d', h', l', m', n', o', p'\}$ is reference pixel. And all of the parameters of $R_{i,j}^{QDCT}$ can be used to embed sixteen bits. Thus the information can be embedded through above manipulations, and the distortion drift is averted in DCT-based data hiding.

However, the modifications of quantized DCT coefficients will reflect on the pixels' values of decompressed stream. The dequantization and integer inverse DCT are operated to $R_{i,j}^{QDCT}$ during decompression process, which can be demonstrated by following formulate.

$$\boldsymbol{R}'_{i,j} = \boldsymbol{C}_r^T (\boldsymbol{R}_{i,j}^{QDCT} \times Q_{step} \otimes \boldsymbol{E}_r) \boldsymbol{C}_r \tag{2}$$

where

$$\boldsymbol{C}_r = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1/2 & -1/2 & -1 \\ 1 & -1 & -1 & 1 \\ 1/2 & -1 & 1 & -1/2 \end{bmatrix}$$

$$\boldsymbol{E}_r = \begin{bmatrix} a^2 & ab & a^2 & ab/2 \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab/2 \\ ab & b^2 & ab & b^2 \end{bmatrix}$$

For example, one bit is embedded by modifying (Y_{00}, Y_{02}) to $(Y_{00}+1, Y_{02}-1)$ in the first category. The difference matrix between $R_{i,j}^{QDCT}$ and $R_{i,j}^{QDCTs}$ is

$$\triangle \boldsymbol{R}_{i,j}^{QDCT} = \boldsymbol{R}_{i,j}^{QDCTs} - \boldsymbol{R}_{i,j}^{QDCTc} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{3}$$

And the corresponding difference between $R'_{i,j}$ and $R'^{s}_{i,j}$ equals

$$\begin{aligned} \triangle \boldsymbol{R}'_{i,j} &= \boldsymbol{R}'^{s}_{i,j} - \boldsymbol{R}'^{c}_{i,j} = \boldsymbol{C}_r^T (\triangle \boldsymbol{R}_{i,j}^{QDCT} \times Q_{step} \otimes \boldsymbol{E}_r) \boldsymbol{C}_r \\ &= Q_{step} \times \begin{bmatrix} 0 & 2a^2 & 2a^2 & 0 \\ 0 & 2a^2 & 2a^2 & 0 \\ 0 & 2a^2 & 2a^2 & 0 \\ 0 & 2a^2 & 2a^2 & 0 \end{bmatrix} \end{aligned} \tag{4}$$

The reconstructed block after embedding is calculated by $B'^{s}_{i,j} = B'_{i,j} + R'_{i,j} + \triangle R'_{i,j}$. Although the right column of $B'^{s}_{i,j}$ is not perturbed by modifications, the pixels' values in the middle two columns are all changed due to DCT-based data hiding.

4 PROPOSED STEGANALYSTIC FEATURE SETS

4.1 Spatial Correlation in Intra-Frame

Video stream is actually composed of continuous frames and every frame can be viewed as an image in spatial domain. In this section, we analyze the spatial correlation in intra-frame and propose the intra-frame feature with residual distortion measure.

4.1.1 Feature based on H.264 phase. In H.264/AVC[1], in order to reduce computation complexity during video compression, the 4×4 integer DCT and quantization is reformed from the following transformation formula

$$\boldsymbol{R}^{QDCT} = \boldsymbol{A}\boldsymbol{R}\boldsymbol{A}^T \tag{5}$$

where $\boldsymbol{A}$ is the identity orthogonal matrix defined by

$$\boldsymbol{A} = \begin{bmatrix} \frac{1}{2}cos(0) & \frac{1}{2}cos(0) & \frac{1}{2}cos(0) & \frac{1}{2}cos(0) \\ \sqrt{\frac{1}{2}}cos(\frac{\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{5\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{7\pi}{8}) \\ \sqrt{\frac{1}{2}}cos(\frac{2\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{6\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{10\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{14\pi}{8}) \\ \sqrt{\frac{1}{2}}cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{9\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{15\pi}{8}) & \sqrt{\frac{1}{2}}cos(\frac{21\pi}{8}) \end{bmatrix}$$

And the inverse quantization and inverse integer DCT is calculated by

$$\boldsymbol{R} = \boldsymbol{A}^T \boldsymbol{R}^{QDCT} \boldsymbol{A} \tag{6}$$

These formulas are actually derived from 2-D DCT and inverse DCT transformation. For every frame with the size

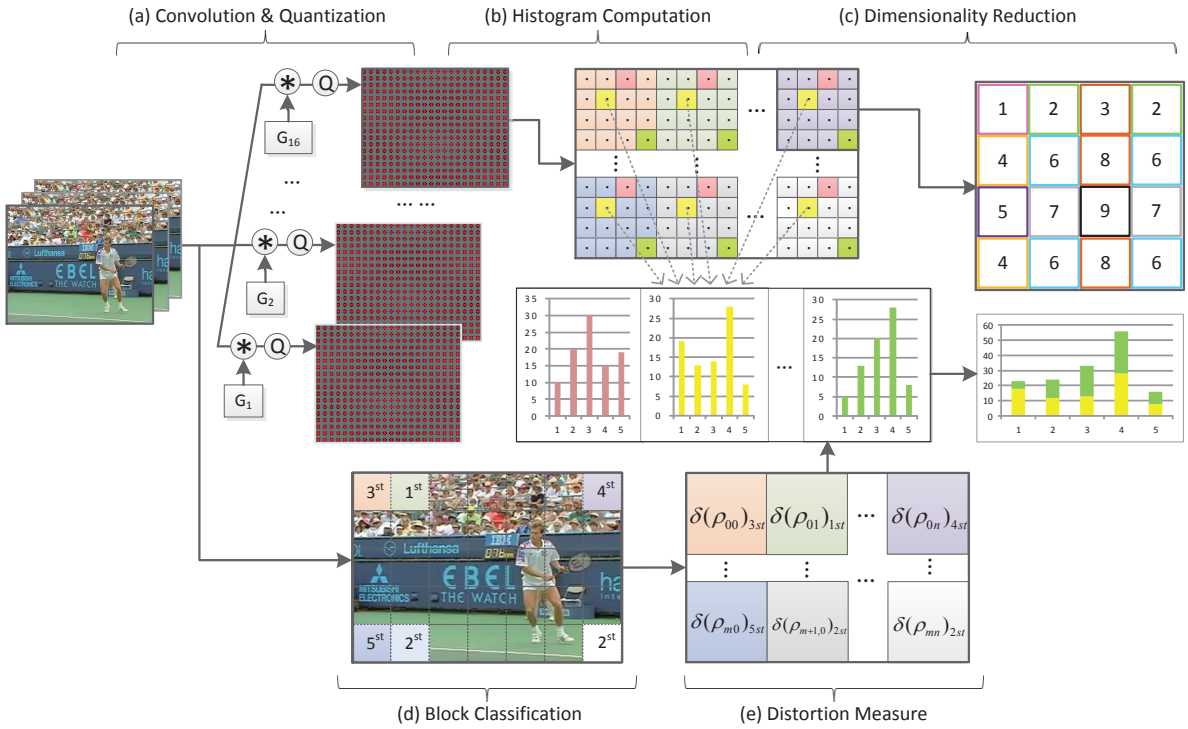


Figure 5: The extraction process of intra-frame feature based on H.264 phase with distortion measure.

of $M \times N$, the (i,j)th 4×4 residual block, $1 \le i \le M/4$, $1 \le j \le N/4$, is formed by $R_{ij}(m,n)$, which is restricted to $0 \le m \le 3$ and $0 \le n \le 3$. The DCT coefficient matrix of residual block is composed of $R_{ij}^{DCT}(u,v)$, where $0 \le u \le 3$ and $0 \le v \le 3$. The 2-D DCT and IDCT transformation can be formulated by

$$R_{ij}^{DCT}(u,v) = \frac{w_u w_v}{2} \sum_{m=0}^{3} \sum_{n=0}^{3} R_{ij}(m,n) \cdot cos\frac{(2m+1)u\pi}{8} \cdot cos\frac{(2n+1)v\pi}{8} \tag{7}$$

$$R_{ij}(m,n) = \sum_{u=0}^{3} \sum_{v=0}^{3} \frac{w_u w_v}{2} R_{ij}^{DCT}(u,v) \cdot cos\frac{(2m+1)u\pi}{8} \cdot cos\frac{(2n+1)v\pi}{8} \tag{8}$$

where $w_0 = 1/\sqrt{2}$ and $w_u = 1, w_v = 1$ if $u > 0, v > 0$.

As described in Section 3, the modifications of residual blocks' quantized DCT coefficients will perturb the pixels in reconstructed blocks, which inspires us to detect the DCT-based data hiding methods in our algorithm. In image steganalysis, JPEG phase based methods [6, 7] perform effectively to detect JPEG stegnanography. Derived from the principle of JPEG phase feature, video steganalysis based on H.264 phase is proposed, in which the H.264 phase is defined as the locations of the 4×4 pixel grid in every block.

In order to compute the steganalytic feature based on H.264 phase, the compressed video is first decompressed to spatial domain. And as the result, a series of frames $\boldsymbol{F}$ which consist of pixels are obtained. Then the noise residuals are calculated by convolving $\boldsymbol{F}$ with DCT kernels.

$$\boldsymbol{U}(\boldsymbol{F}, \boldsymbol{G}) = \{\boldsymbol{U}^{(u,v)} | 0 \le u, v \le 3\} \tag{9}$$

$$\boldsymbol{U}^{(u,v)} = \boldsymbol{F} * \boldsymbol{G}^{(u,v)}$$

where $\boldsymbol{U}^{(u,v)} \in \mathbb{R}^{(M-3)\times(N-3)}$ and the operator $*$ denotes the convolution without padding. The DCT kernels are formed as follows.

$$\boldsymbol{G}^{(u,v)} = \{G_{mn}^{(u,v)} | 0 \le m, n \le 3\} \tag{10}$$

$$G_{mn}^{(u,v)} = \frac{w_u w_v}{2} \cdot cos\frac{(2m+1)u\pi}{8} \cdot cos\frac{(2n+1)v\pi}{8} \tag{11}$$

where every DCT basis $G_{uv}^{(m,n)}$ is 4×4 matrix and there are 16 DCT bases in this operation. Consequently, a set of 16 residual matrices are formed after convolutions. Subsequently the residual matrix is quantized by

$$\boldsymbol{U}(\boldsymbol{F}, \boldsymbol{G}, Q) = Q(\boldsymbol{U}(F, G)/q) \tag{12}$$

where q is a fixed quantization step, Q is a quantizer with $\{0, 1, 2, ...T_r\}$ as the centroids and T_r is the truncation threshold. The convolution and quantization process is depicted in Figure 5(a). After these two step, each of the 16 quantized residual matrices is used to compute histograms as follows.

$$h_\tau^{(u,v)}(\boldsymbol{F}, \boldsymbol{G}, Q) = \sum_{i=1}^{\lfloor M/4 \rfloor} \sum_{j=0}^{\lfloor N/4 \rfloor} [U_{ij}^{(u,v)}(F, G, Q) = \tau] \tag{13}$$

As indicated in Figure 5(b), there are 16 histograms that can be obtained for each residual matrix and every histogram consists T_r+1 values. Based on the symmetries of the kernel and DCT basis, the number of histograms can be reduced by merging certain bins of the histograms. As shown in Figure 5(c), the histograms of the phases labeled with the same number can be merged and the number of histograms is reduced from 16 to 9. Therefore, for each frame, the dimensionality of the H.264 phase-based feature can be calculated by $16 \times 9 \times (T_r+1) = 144 \times (T_r+1)$.

4.1.2 Residual distortion measure. In order to better detect the current DCT-based data hiding methods, the residual distortion is measured and incorporated into the design of the proposed feature.

For the frames of cover videos, the quantized DCT coefficients in the (i,j) residual block is denoted by $R_{ij}^{QDCTc}(u,v)$. Deduced from Eq.2, Eq.8 and Eq.11, the decompressed residuals can be calculated by $R_{ij}^{'c}(m,n) = \sum_{u,v=0}^{3} G_{mn}^{(u,v)} \cdot q_{mn} \cdot R_{ij}^{QDCTc}(u,v)$, where q_{mn} is the quantization step. The quantized DCT residuals after embedding $s_{i,j}(u,v)$ are $R_{ij}^{QDCTs}(u,v) = R_{ij}^{QDCTc}(u,v) + s_{i,j}(u,v)$. And the corresponding decompressed residual block is $R_{ij}^{'s}(m,n) = \sum_{u,v=0}^{3} G_{mn}^{(u,v)} \cdot q_{mn} \cdot R_{ij}^{QDCTs}(u,v)$. Thus the difference between decompressed stego block and cover block is computed by

$$\begin{aligned} B_{i,j}^{'s}(m,n) &= B'_{i,j}(m,n) + R_{ij}^{'s}(m,n) \\ B_{i,j}^{'c}(m,n) &= B'_{i,j}(m,n) + R_{ij}^{'c}(m,n) \\ \triangle B'_{i,j}(m,n) = \triangle R'_{i,j}(m,n) &= R_{ij}^{'s}(m,n) - R_{ij}^{'c}(m,n) \\ &= \sum_{u,v=0}^{3} G_{mn}^{(u,v)} \cdot q_{mn} \cdot s_{i,j}(u,v) \end{aligned} \tag{14}$$

Based on the linearity of convolution, the difference between $R_{ij}^{'s}$ and $R_{ij}^{'c}$ is used as the accumulated quantity in the feature calculation. Thus the residual distortion is defined by

$$\rho(\boldsymbol{s}) = \boldsymbol{R}^s * \boldsymbol{G} - \boldsymbol{R}^c * \boldsymbol{G} = \boldsymbol{s} * \boldsymbol{G} \tag{15}$$

where $\rho(\boldsymbol{s}) = \{\rho_{ij}(\boldsymbol{s}) | 1 \le i \le M/4, 1 \le j \le N/4\}$. After modifying the quantized DCT coefficients, the difference between the stego and cover decompressed residual is denoted as

$$\triangle R'_{i,j} = \begin{bmatrix} \triangle R'_{0,0} & \triangle R'_{0,1} & \triangle R'_{0,2} & \triangle R'_{0,3} \\ \triangle R'_{1,0} & \triangle R'_{1,1} & \triangle R'_{1,2} & \triangle R'_{1,3} \\ \triangle R'_{2,0} & \triangle R'_{2,1} & \triangle R'_{2,2} & \triangle R'_{2,3} \\ \triangle R'_{3,0} & \triangle R'_{3,1} & \triangle R'_{3,2} & \triangle R'_{3,3} \end{bmatrix} \tag{16}$$

Based on the theorem in [16], the sum of square errors is utilized as the measure of distortion

$$\delta(\rho_{ij}) = \sum_{m=0}^{3} \sum_{n=0}^{3} \triangle R_{m,n}^{'2} \tag{17}$$

As described in Section 3, the secret information is embedded to quantized DCT paired-coefficients according to the 4×4 blocks' classification. Take the first category as example, the four vertical coefficient-pairs in $\{(r_{00}, r_{02}), (r_{10}, r_{12}), (r_{20}, r_{22}), (r_{30}, r_{32})\}$ are the suggested parameters for modifications. If the pair (r_{10}, r_{12}) is perturbed to $(r_{10}+1, r_{12}-1)$, the difference between R_{ij}^{QDCTs} and R_{ij}^{QDCTc} is computed by

$$\triangle R_{ij}^{QDCT} = R_{ij}^{QDCTs} - R_{ij}^{QDCTc} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{18}$$

And difference between decompressed cover block $R_{ij}^{'c}$ and stego block $R_{ij}^{'s}$ can be derived from Eq. 2 as follows.

$$\begin{aligned} \triangle \boldsymbol{R}'_{i,j} &= \boldsymbol{R}_{i,j}^{'s} - \boldsymbol{R}_{i,j}^{'c} = \boldsymbol{C}_r^T (\triangle \boldsymbol{R}_{i,j}^{QDCT} \times Q_{step} \otimes \boldsymbol{E}_r) \boldsymbol{C}_r \\ &= Q_{step} \times \begin{bmatrix} 0 & 2ab & 2ab & 0 \\ 0 & ab & ab & 0 \\ 0 & -ab & -ab & 0 \\ 0 & -2ab & -2ab & 0 \end{bmatrix} \end{aligned} \tag{19}$$

where $a = 1/2, b = \sqrt{2/5}$ and Q is the step size of a quantizer determined by QP. Then the distortion induced by perturbing (r_{10}, r_{12}) is measured by

$$\begin{aligned} \delta(\rho_{ij})(r_{10}, r_{12}) &= \sum_{m=0}^{3} \sum_{n=0}^{3} \triangle R_{m,n}^{'2} \\ &= (4 \times (2ab)^2 + 4 \times (ab)^2) \times Q^2 \\ &= 20 \times (ab)^2 \times Q^2 = 2Q^2 \end{aligned} \tag{20}$$

Similarly the residual distortions of other pairs are obtained $\delta(\rho_{ij})(r_{00}, r_{02}) = \delta(\rho_{ij})(r_{20}, r_{22}) = \delta(\rho_{ij})(r_{30}, r_{32}) = 2Q^2$. And the total distortion of first category equals $8Q^2$, which is the sum of these four sub-distortions. By the same derivation, the distortion of second and third category is also $8Q^2$. In category four, $(r_{00}, r_{02}, r_{20}, r_{22})$ is modified to $(r_{00}+1, r_{02}-1, r_{20}-1, r_{22}+1)$ and the corresponding distortion is

$$\begin{aligned} \delta(\rho_{ij})(r_{00}, r_{02}, r_{20}, r_{22}) &= \sum_{m=0}^{3} \sum_{n=0}^{3} \triangle R_{m,n}^{'2} \\ &= 4 \times (4a^2)^2 \times Q^2 = 64a^4 \times Q^2 \\ &= 4Q^2 \end{aligned} \tag{21}$$

Because all of the sixteen bits can be utilized for embedding in the fifth category, the overall distortion is $16Q^2$. Thus the distortion measure of DCT-based data hiding can be presented by $\delta(\rho_{ij}) = \{\delta(\rho_{ij})_{1st}, \delta(\rho_{ij})_{2st}, \delta(\rho_{ij})_{3st}, \delta(\rho_{ij})_{4st}, \delta(\rho_{ij})_{5st}\} = \{8Q^2, 8Q^2, 8Q^2, 4Q^2, 16Q^2\}$.And the H.264 phase based histogram with this distortion measure is formulized as

$$\dot{h}_{\tau}^{(u,v)}(\boldsymbol{F}, \boldsymbol{G}, Q) = \sum_{i=1}^{\lfloor M/4 \rfloor} \sum_{j=0}^{\lfloor N/4 \rfloor} [U_{ij}^{(u,v)}(\boldsymbol{F}, \boldsymbol{G}, Q) = \tau] \cdot \delta(\rho_{ij}) \tag{22}$$

As shown in 5(e)(d), the residual distortion induced by modifications is measured based on the block classification in DCT-based data hiding methods. Then it can cooperate with the histogram computation to extract the intra-frame feature.

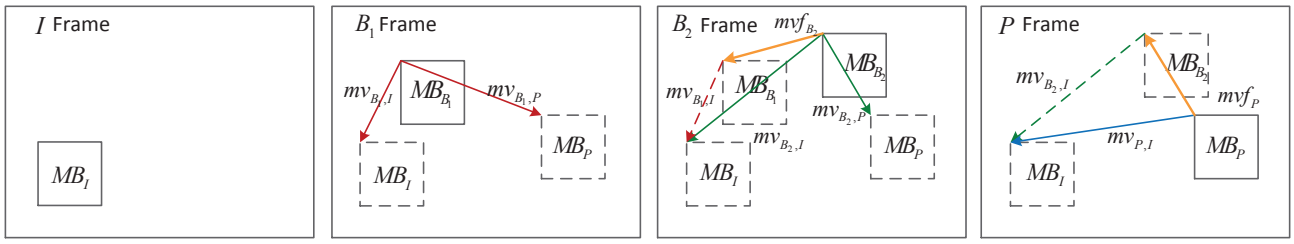


Figure 6: Compute SMV based on Inter-frame prediction in GOP.

4.2 Temporal Correlation between Inter-Frames

Due to the continuity in time, strong correlation usually exists between the blocks of adjacent frames, which is probable to be disturbed by embedding manipulation. Thus we exploit the correlation of similar blocks between adjacent frames, from which the inter-frame feature is extracted.

4.2.1 4×4 *block link using motion vectors.* Motion estimation is designed to reduce the temporal redundancy between video frames. This is achieved by allowing blocks of pixels from currently coded frame to be matched with those from reference frame(s). As the result of motion estimation, MV represents the spatial displacement offset between a block and its prediction.

Therefore, the similar blocks along the time-axis can be linked by the MVs. However, MVs' values are greatly determined by the performance of motion estimation. In order to decide whether the block pointed by MV is the similar one of current block, the MVs are screened by

$$|SAD - \frac{SAD_1 + \cdots + SAD_{\mathbb{N}^2-1}}{\mathbb{N}^2 - 1}| < \mu \qquad (23)$$

Here, SAD is the sum of absolute difference between the block and its prediction, μ is the standard deviation of SAD. This formula computes the difference between the current SAD and the average SAD of its neighbors. Because 4×4 is the size of the investigated blocks, the motion of neighboring blocks is homogeneous. If the SAD of this MB is distinctly large compared with its neighbors, it is a singularity in prediction and the reference block is not the similar block. All of the inaccurate MVs would be screened out after this procedure, leaving those pointing to similar blocks.

There are three types of frames including I,P and B frame in group of pictures (GOP). A GOP usually starts with an I frame and consists of several P and B frames. In the inter-prediction, I frame conducts as the reference frame, one frame is used to predict P frame and two frames are referred when predicting B frame. Therefore, in video compression, the MVs point to the reference blocks in disorder. In order to link the similar blocks in chronological order along the time axis, we need to calculate the sequential MVs (SMV). The set of SMVs in frame F_t is denoted by $\{smv_t\}$, in which every SMV points to the prediction block in $(t-1)$th frame.

As shown in Figure 6, in every frame, the smv_t can be computed from their original MVs. In frame F_{B_1}, smv_{B_1} equals $mv_{B_1,I}$,which is the MV obtained by referring to the corresponding MB_I in frame I. In frame F_{B_2}, according to the operation rule of vector (the Triangle Rule), we can get $smv_{B_2} = mv_{B_2,I} - mv_{B_1,I}$. And similarly $smv_P = mv_{P,I} - mv_{B_2,I}$ can be obtained in frame F_P by analyzing the constraints of the original MVs.

For a given video sequence $\{F_t\}_{t=1}^{T}$, the SMV set $\{smv_t\}$ of every frame can be constructed by above computation and smv_t points to the location of corresponding similar block in frame F_{t-1}. All of the similar blocks in GOP can be linked by smv_t, which is diagrammatically expressed in Figure 7. If the location of predicted block crosses the boundary in reference frame, the one with largest overlapping area is selected as the linked block. Besides 4×4 luma block, 16×16 luma block also exists in H.264/AVC. In order to link the associated blocks which are similar to the 4×4 block in I frame, the block of larger size is divided into 4×4 sub-blocks. Consequently, all of the similar 4×4 blocks are linked in sequence with the head of 4×4 block in I frame.

4.2.2 Feature based on block link . Besides the intra-frame feature which is extracted based on the spatial correlation, we also extract feature by exploiting the correlation between inter-frames.

After above manipulation, the temporal correlation is exploited by linking the similar blocks in video GOP. In order to extract feature based on such correlation, we construct the temporal pattern by splicing the linked blocks, which is illustrated in Figure 8. Every row of the pattern consists of T linked 4×4 blocks which are similar to the same one in I frame, where T is the frame number of this GOP. Supposing

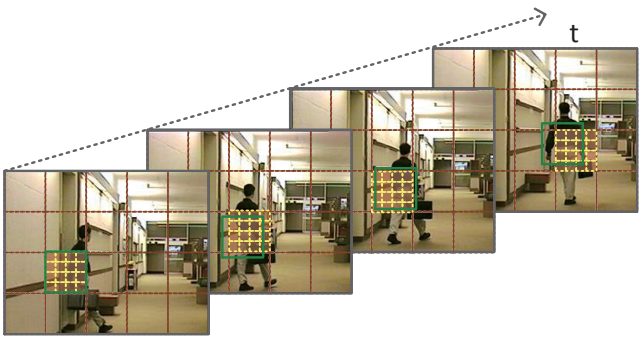


Figure 7: 4×4 **block link by SMVs.**

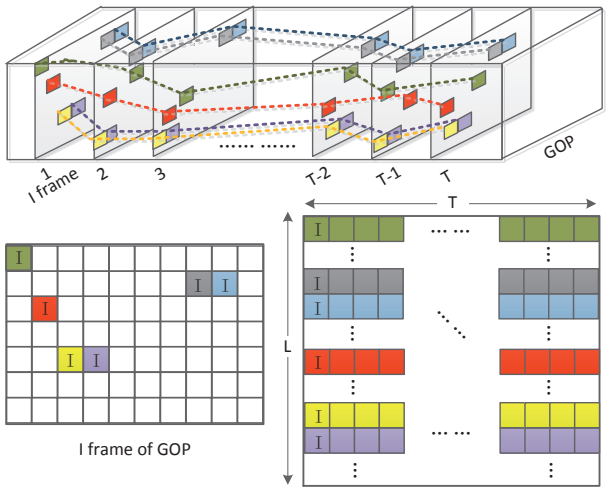


Figure 8: Construct pattern using linked blocks.

L 4×4 blocks in I frames can be linked, the number of links in this GOP is L. Thus L rows of the linked 4×4 blocks compose the temporal pattern of this GOP.

Then features are extracted from the pattern using the same method described in Subsection 4.1. The temporal histogram is calculated by

$$\ddot{h}_{\tau}^{(u,v)}(\boldsymbol{P}, \boldsymbol{G}, Q) = \sum_{i=1}^{T}\sum_{j=0}^{L}[U_{ij}^{(u,v)}(\boldsymbol{P}, \boldsymbol{G}, Q) = \tau] \cdot \delta(\rho_i) \quad (24)$$

where T and L is the width and height of the constructed pattern in 4×4 block unit, $\boldsymbol{P}$ is the decompressed pattern, $\delta(\rho_i)$ is the distortion of the block in ith row. It is noticeable that the linked block's distortion in one row is identical, which equals to the distortion of the similar block in I frame. And for each constructed pattern, the dimensionality of temporal feature can also be calculated by $16 \times 9 \times (T_r + 1) = 144 \times (T_r + 1)$.

4.3 Final Feature Design

As described in Subsection 4.1 and 4.2, the intra-frame and inter-frame correlations of video sequence are exploited to detect DCT-based data hiding. In order to improve the steganalytic performance, we utilize the distortion measure to optimize the feature extraction in spatial and temporal domain. Based on above analysis, we design the final feature and summarize the implementation in pseudo-code. In the below pseudo-code, the kernels of convolution in $\mathbf{G}$ are formed by 16 DCT bases, and Q is the quantizer step size determined by QP.

1. Divide the compressed video into K GOPs and the head of every GOP is I frame.
2. For every GOP,
   1) decompress the I frame under investigation to the spatial domain. For every spatial frame $\boldsymbol{F}_{\kappa}$,
      a) compute the noise residual $\boldsymbol{U}(\boldsymbol{F}_{\kappa}, \boldsymbol{G})$ and quantize it to get $\boldsymbol{U}(\boldsymbol{F}_{\kappa}, \boldsymbol{G}, Q)$;
      b) evaluate $\delta(\rho_{ij}) \in \{8Q^2, 8Q^2, 8Q^2, 4Q^2, 16Q^2\}$ for every (i,j)th 4×4 block by classifying it to one of the five categories;
      c) calculate the histogram $\dot{h}_{\tau}^{(u,v)}(\boldsymbol{F}_{\kappa}, \boldsymbol{G}, Q)$ by $\sum_{i=1}^{\lfloor M/4 \rfloor}\sum_{j=0}^{\lfloor N/4 \rfloor}[U_{ij}^{(u,v)}(\boldsymbol{F}_{\kappa}, \boldsymbol{G}, Q) = \tau] \cdot \delta(\rho_{ij})$, where $0 \le \tau \le T_r, 0 \le u, v \le 3$.
   2) decompress the P and B frames in this GOP, and during the decompression,
      a) construct the temporal pattern $\boldsymbol{P}_{\kappa}$ by linking the spatial blocks similar to the same one in $\boldsymbol{F}_{\kappa}$;
      b) compute the noise residual $\boldsymbol{U}(\boldsymbol{P}_{\kappa}, \boldsymbol{G})$ and quantize it to get $\boldsymbol{U}(\boldsymbol{P}_{\kappa}, \boldsymbol{G}, Q)$;
      c) compute the histogram values in $\ddot{h}_{\tau}^{(u,v)}(\boldsymbol{P}_{\kappa}, \boldsymbol{G}, Q)$ $\sum_{i=1}^{T}\sum_{j=0}^{L}[U_{ij}^{(u,v)}(\boldsymbol{P}_{\kappa}, \boldsymbol{G}, Q) = \tau] \cdot \delta(\rho_i)$, here $\delta(\rho_i)$ is the block's distortion in $\boldsymbol{F}_{\kappa}$.
   3) merge the bins of $\dot{h}_{\tau}^{(u,v)}(\boldsymbol{F}_{\kappa}, \boldsymbol{G}, Q)$ and $\ddot{h}_{\tau}^{(u,v)}(\boldsymbol{P}_{\kappa}, \boldsymbol{G}, Q)$ using symmetrization rules, then concatenate two merg -ed histograms to construct the final feature set.
3. manipulate every $GOP_{\kappa}, 1 \le \kappa \le K$ as Step 2 until the end of the compressed video.

After the above process, K features are extracted from the video. Because the feature dimensionality of intra-frame or inter-frame correlation equals $144 \times (T_r + 1)$, the concatenated feature's dimension is supposed to be $2 \times 144 \times (T_r + 1) = 288 \times (T_r + 1)$. And there are 1440 values in the final feature set with T_r set as 4. In the next section, the extracted features will be further subjected to classifier for experimental evaluation.

5 EXPERIMENTS

5.1 Experimental Setup

As the well-known H.264/AVC codec, x264 [2] is utilized to implement the data hiding and steganalytic algorithms in our experiments. The video database consists of 100 standard CIF sequences in YUV 4:2:0 format. These sequences vary from 100 to 2000 frames in length and are all coded at the frame rate of 30*fps*.

In order to evaluate steganalytic method against existing DCT-based data hiding schemes, Ma's [13] and Lin's [11] methods are realized to embed information. The embedding

Table 2: Percentage of 4×4 blocks in each category under different QPs

QP	1st	2st	3st	4st	5st
20	24.72%	26.63%	1.60%	43.91%	3.14%
24	23.15%	26.17%	1.10%	45.78%	3.80%
28	23.03%	24.59%	1.20%	47.18%	4.00%
32	22.18%	23.31%	0.90%	50.11%	3.50%
36	18.26%	19.91%	1.30%	57.73%	2.80%
40	15.62%	17.13%	1.10%	63.95%	2.20%

Table 3: Detection accuracies (%) of Da's method, DCTR feature and proposed steganalytic algorithm against DCT-based data hiding methods

DCT-based Data Hiding	QP	ER (*bpnc*)	Da's method		DCTR feature		Proposed algorithm	
			Ensemble Classifier	SVM	Ensemble Classifier	SVM	Ensemble Classifier	SVM
Ma's method	28	0.05	53.30	62.79	64.27	67.70	66.59	76.90
		0.1	58.16	75.23	74.19	76.93	84.72	95.81
	32	0.05	53.01	61.19	58.64	63.64	60.78	70.68
		0.1	58.09	73.54	70.61	74.53	80.29	86.70
	36	0.05	52.04	58.74	54.34	59.42	57.20	64.24
		0.1	56.87	71.51	65.93	68.98	73.02	79.13
Lin's method	28	0.05	57.03	74.20	74.82	81.70	89.38	95.71
		0.1	66.87	87.58	83.07	85.67	94.29	98.41
		0.15	69.75	91.11	81.80	86.04	95.81	99.13
	32	0.05	53.99	68.53	70.32	76.77	82.64	89.12
		0.1	64.15	84.02	87.84	90.32	96.13	98.37
		0.15	69.49	90.53	89.91	90.98	96.96	98.69
	36	0.05	50.68	50.25	56.68	59.99	67.46	64.97
		0.1	59.80	78.62	79.45	81.81	92.38	97.85
		0.15	68.36	88.79	87.44	88.97	95.54	98.62

rate (ER) is measured by the number of bits embedded per non-zero coefficient (*bpnc*). We test the percentage distribution of blocks in each categories and calculate the maximum embedding rate that can be achieved under different QPs. As shown in Table 2 and Figure 9, the maximum embedding rate reduces with the increase of QP and the embedding capacity of Lin's method is larger than that of Ma's. In our experiments, the videos are compressed with common QP set (28, 32, 36) and the different embedding rates are respectively considered for Ma's and Lin's methods (0.05 *bpnc*, 0.1 *bpnc* for Ma's method and 0.05 *bpnc*, 0.1 *bpnc*, 0.15 *bpnc* for Lin's method).

Because there are not any approaches which are specifically designed to detect DCT-based data hiding methods, Da's video steganalytic approach [5] and the DCTR feature [6] in image steganalysis are leveraged for comparisons. In order to compute DCTR feature using video stream, every YUV frame decompressed from cover or stego video is treated as an image in spatial domain and the kernels used in feature computation are 4×4 DCT bases.

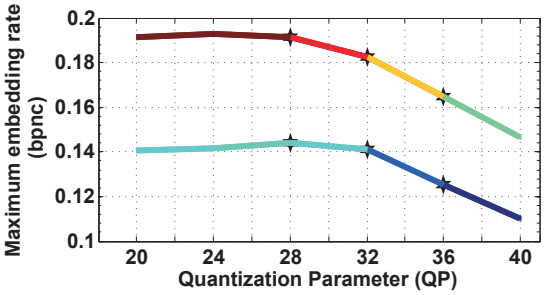


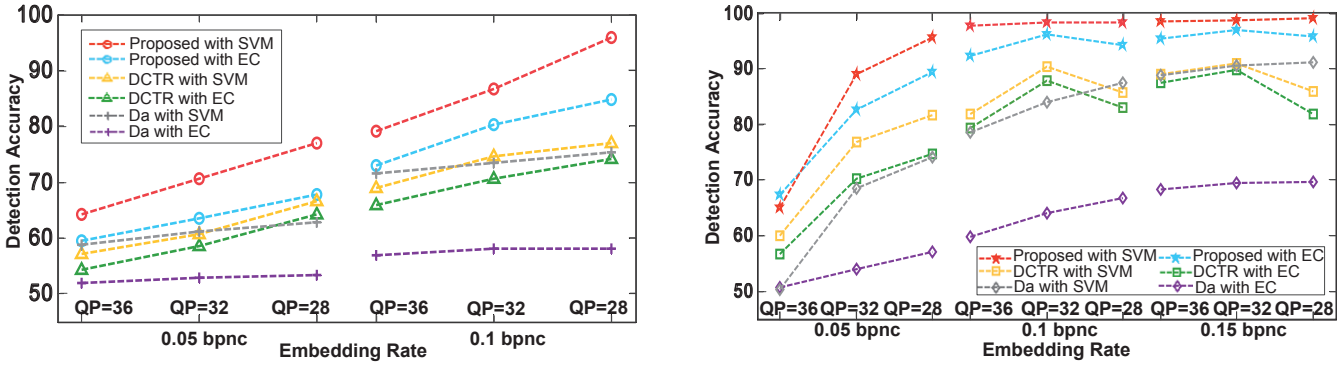
Figure 9: Maximum embedding rate of DCT-based data hiding methods under different QPs.

Both of Ensemble Classifier and LibSVM toolbox (SVM) [4] are utilized as classifiers. 50 percent cover-stego pairs are randomly selected for training and the remaining ones for testing. The true negative (TN) rate and the true positive (TP) rate are computed by counting the number of detections in the test sets. By averaging TN and TP, the detection accuracy is obtained. After repeating 20 times, the average detection accuracies are used to evaluate the final performance.

5.2 Results and Discuss

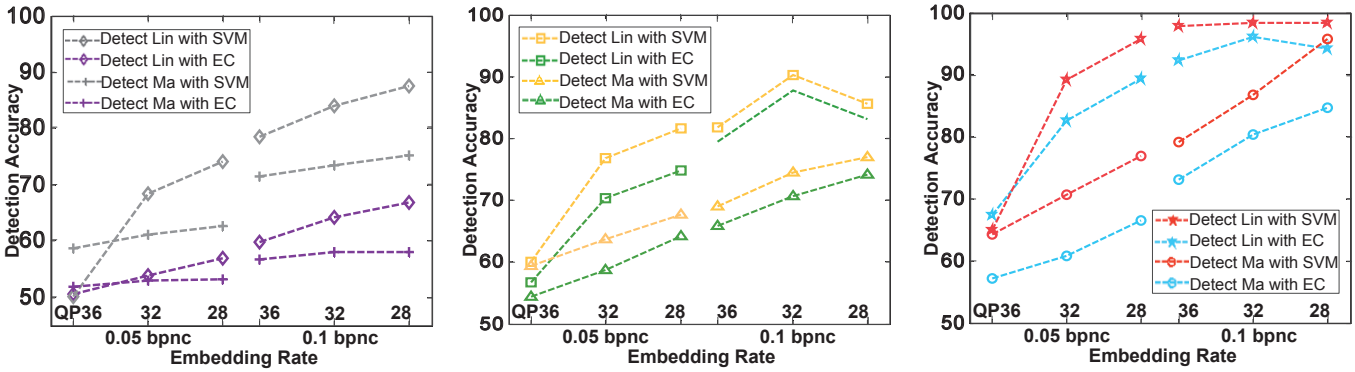
In the experiments, Da's method, DCTR feature and the proposed algorithm are utilized to detect data hiding methods under different cases. The detection accuracies against DCT-based data hiding methods are recorded in Table 3. Figure 10 depicts the comparison of their performances in detecting Ma's and Lin's methods respectively. It can be observed that the proposed approach performs best among the three steganalytic approaches. And in most cases, the steganalytic scheme using DCTR feature performs better than Da's method. The steganalytic performance meliorates with the decrease of QP. It is because less loss is induced in higher quality videos which are compressed with lower QP values, and the features extracted from these streams are more effective. Due to the low dimension of extracted features, the SVM outperforms Ensemble Classifier for all of the three steganalytic methods in classification.

In order to compare the detection accuracies against Ma's and Lin's methods, Figure 11 illustrates the steganalytic performances of the three steganalytic algorithms respectively. As the expense of large embedding capacity, the security



(a) (b)

Figure 10: Detection accuracies (%) against (a) Ma's and (b) Lin's DCT-based data hiding methods.



(a) (b) (c)

Figure 11: Steganalytic performances (%) of (a) Da's (b) DCTR and (c) proposed algorithms.

level of Lin's is inferior to Ma's method. Although the Da's approach and DCTR feature can detect the DCT-based data hiding to some extent, the proposed approach improves the detection accuracy significantly. Our method achieves the detection accuracy of 99.13% when detecting Lin's scheme with 0.15 *bpnc* and QP=28. And the detection accuracy against Ma's method is up to 95.81% with 0.1 *bpnc* and QP = 28. Overall, our proposed method can effectively detect the existing DCT-based data hiding methods.

6 CONCLUSIONS

In this paper, an effective steganalytic approach is proposed for H.264/AVC videos to detect covert information hidden in the quantized DCT coefficients. Both the intra-frame and inter-frame correlations are considered in the design of steganalytic features. Considering the temporal correlation of blocks in intra-frame, the first feature set using 16 DCT kernels is proposed. Moreover, the residual distortion incorporated into the feature design is defined by analyzing the state-of art embedding strategy. In order to exploit the temporal correlation, the similar blocks between inter-frames are linked using MVs, from which the second feature set is extracted. The experimental results have demonstrated that our method can effectively distinguish stego H.264 videos undergone DCT manipulations from clean ones, especially for those compressed using low QP values.

As part of our future work, the steganalytic approach to detect data hiding methods in low quality videos is to be exploited. Moreover, the steganalysis with light computation is also to be further studied in practical applications.

ACKNOWLEDGMENTS

This work was supported by the NSFC under U1636102 and U1536105, and National Key Technology R&D Program under 2014BAH41B01, 2016YFB0801003 and 2016QY15Z2500.

REFERENCES

[1] 2003. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification, document ITU-T Rec. H.264/ISO/IEC 14496-10 AVC, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG,JVTG050. (May 2003).

[2] 2014. VideoLAN. x264. (2014). Available: http://www.videolan.org/developers/x264.html.

[3] Y. Cao, H. Zhang, X. Zhao, and H. Yu. 2015. Video Steganography Based on Optimized Motion Estimation Perturbation. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '15)*. ACM, New York, NY, USA, 25–31. DOI: http://dx.doi.org/10.1145/2756601.2756609

[4] C.Chang and C.Lin. 2001 [online]. LIBSVM: A Library for Support Vector Machines. (2001 [online]). Available: http://www.csie.ntu.edu.tw/cjlin/libsvm.

[5] Ting Da, ZhiTang Li, and Bing Feng. 2015. *A Video Steganalysis Algorithm for H.264/AVC Based on the Markov Features.* Springer International Publishing, Cham, 47–59. DOI: http://dx.doi.org/10.1007/978-3-319-22186-1_5

[6] V. Holub and J. Fridrich. 2015. Low-Complexity Features for JPEG Steganalysis Using Undecimated DCT. *IEEE Transactions on Information Forensics and Security* 10, 2 (Feb 2015), 219–228. DOI:http://dx.doi.org/10.1109/TIFS.2014.2364918

[7] V. Holub and J. Fridrich. 2015. Phase-aware projection model for steganalysis of JPEG images. (2015). DOI: http://dx.doi.org/10.1117/12.2075239

[8] S. Kapotas and A. Skodras. 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*. 277–280. DOI:http://dx.doi.org/10.1109/ICME.2008.4607425

[9] S. Kim, S. Kim, Y. Hong, and C. Won. 2007. *Data Hiding on H.264/AVC Compressed Video.* Springer Berlin Heidelberg, Berlin, Heidelberg, 698–707. DOI: http://dx.doi.org/10.1007/978-3-540-74260-9_62

[10] K. Liao, S. Lian, Z. Guo, and J. Wang. 2012. Efficient Information Hiding in H.264/AVC Video Coding. *Telecommun. Syst.* 49, 2 (Feb. 2012), 261–269. DOI: http://dx.doi.org/10.1007/s11235-010-9372-5

[11] T. Lin, K. Chung, P. Chang, Y. Huang, H. Liao, and C. Fang. 2013. An improved DCT-based perturbation scheme for high capacity data hiding in H.264/AVC intra frames. *Journal of Systems and Software* 86, 3 (2013), 604–614. DOI: http://dx.doi.org/10.1016/j.jss.2012.10.922

[12] X. Ma, Z. Li, J. Lv, and W. Wang. 2009. Data Hiding in H.264/AVC Streams with Limited Intra-Frame Distortion Drift. In *2009 International Symposium on Computer Network and Multimedia Technology*. 1–5. DOI: http://dx.doi.org/10.1109/CNMT.2009.5374766

[13] X. Ma, Z. Li, H. Tu, and B. Zhang. 2010. A Data Hiding Algorithm for H.264/AVC Video Streams Without Intra-Frame Distortion Drift. *IEEE Transactions on Circuits and Systems for Video Technology* 20, 10 (Oct 2010), 1320–1330. DOI: http://dx.doi.org/10.1109/TCSVT.2010.2070950

[14] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima. 2005. Rewritable Data Embedding on MPEG Coded Data Domain. In *2005 IEEE International Conference on Multimedia and Expo*. 682–685. DOI:http://dx.doi.org/10.1109/ICME.2005.1521515

[15] V. Pankajakshan and A. Ho. 2007. Improving Video Steganalysis Using Temporal Correlation. In *Proceedings of the Third International Conference on International Information Hiding and Multimedia Signal Processing (IIH-MSP 2007) - Volume 01 (IIH-MSP '07)*. IEEE Computer Society, Washington, DC, USA, 287–290. http://dl.acm.org/citation.cfm?id=1336956.1337487

[16] Athanasios Papoulis and A. A. Maradudin. 1962. The Fourier Integral and Its Applications. 51, 1 (1962), 159–161.

[17] P. Wang, Hong Zhang, Yun Cao, and Xianfeng Zhao. 2016. A Novel Embedding Distortion for Motion Vector-Based Steganography Considering Motion Characteristic, Local Optimality and Statistical Distribution. In *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '16)*. ACM, New York, NY, USA, 127–137. DOI: http://dx.doi.org/10.1145/2909827.2930801

[18] K. Wong, K. Tanaka, K. Takagi, and Y. Nakajima. 2009. Complete Video Quality-Preserving Data Hiding. *IEEE Transactions on Circuits and Systems for Video Technology* 19, 10 (Oct 2009), 1499–1512. DOI: http://dx.doi.org/10.1109/TCSVT.2009.2022781

[19] N. Zarmehi and M. Akhaee. 2016. Digital video steganalysis toward spread spectrum data hiding. *IET Image Processing* 10, 1 (2016), 1–8. DOI:http://dx.doi.org/10.1049/iet-ipr.2014.1019

[20] H. Zhang, Y. Cao, and X. Zhao. 2017. A Steganalytic Approach to Detect Motion Vector Modification Using Near-Perfect Estimation for Local Optimality. *IEEE Transactions on Information Forensics and Security* 12, 2 (Feb 2017), 465–478. DOI: http://dx.doi.org/10.1109/TIFS.2016.2623587

[21] H. Zhang, Y. Cao, X. Zhao, W. Zhang, and N. Yu. 2014. Video Steganography with Perturbed Macroblock Partition. In *Proceedings of the 2Nd ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '14)*. ACM, New York, NY, USA, 115–122. DOI:http://dx.doi.org/10.1145/2600918.2600936