Digital Video Steganalysis Based on a Spatial Temporal Detector

Yuting Su[1], Fan Yu[1] and Chengqian Zhang[1]
[1] School of Electronic Information Engineering, Tianjin University
Tianjin, 300072, China
[e-mail: zhangcqj@tju.edu.cn]
*Corresponding author: Chengqian Zhang

Received June 3, 2016; revised October 4, 2016; accepted October 24, 2016; published January 31, 2017

Abstract

This paper presents a novel digital video steganalysis scheme against the spatial domain video steganography technology based on a spatial temporal detector (ST_D) that considers both spatial and temporal redundancies of the video sequences simultaneously. Three descriptors are constructed on XY, XT and YT planes respectively to depict the spatial and temporal relationship between the current pixel and its adjacent pixels. Considering the impact of local motion intensity and texture complexity on the histogram distribution of three descriptors, each frame is segmented into non-overlapped blocks that are 8×8 in size for motion and texture analysis. Subsequently, texture and motion factors are introduced to provide reasonable weights for histograms of the three descriptors of each block. After further weighted modulation, the statistics of the histograms of the three descriptors are concatenated into a single value to build the global description of ST_D. The experimental results demonstrate the great advantage of our features relative to those of the rich model (RM), the subtractive pixel adjacency model (SPAM) and subtractive prediction error adjacency matrix (SPEAM), especially for compressed videos, which constitute most Internet videos.

Keywords: video steganalysis, spatial temporal detector, texture and motion factors

A preliminary version of this paper appeared in IEEE ICC 2009, June 14-18, Dresden, Germany. This version includes a concrete analysis and supporting implementation results on MICAz sensor nodes. This research was supported by a research grant from the IT R&D program of MKE/IITA, the Korean government [2005-Y-001-04, Development of Next Generation Security Technology]. We express our thanks to Dr. Richard Berke who checked our manuscript.

https://doi.org/10.3837/tiis.2017.01.019 **ISSN : 1976-7277**

1. Introduction

With the rapid development of digital multimedia and Internet technology, steganalysis, which is used to detect the existence of secret information with little or no knowledge of the steganography algorithm, has attracted increasing interest in recent years. Because of its strong robustness, high communication quality and high capacity, video signals constitute a better medium for steganography technolody compared to other digital media. However, compared to the significant research efforts devoted to image steganalysis, video steganalysis still remains in its infancy, especially for compressed videos, so this paper put the main emphasis on steganalysis for compressed videos.

Currently, there are many mature video steganography techniques. Some steganography methods embed secret information using spatial pixels, such as least significant bit (LSB) matching [1-3], quantization index modulation (QIM) [4,5] and the spread spectrum (SS) algorithm [6,7]. Furthermore, some steganography techniques implement the operation of embedding secret information after some type of transformation, such as discrete wavelet transformation (DWT) [8] or discrete cosine transformation (DCT) [9,10]. Motion vectors are also utilized to embed secret information [11,12,13]. Among these methods, many steganography techniques for videos adeveloped from those for images, which apply image steganography methods directly to video sequences on a frame-by-frame in the spatial domain.

Many video steganalysis techniques against those spatial domain steganography methods utilize spatial or temporal redundancy of videos. Budhia *et al*. [14] reported a steganalysis algorithm based on interframe collusion that exploited the temporal statistical visibility of hidden messages. Pankajakshan *et al*. [15] modified the model in [14] by combining spatial-temporal prediction and wavelet moments. Dong *et al*. [16] focused on exploiting and describing the inter-dependence of elements in all types of adjacency structures in video frames to obtain a universal spatial feature set for steganalysis. In [17], filtering windows with smaller dimensions were used to estimate the cover, and then, the features presented in [14] were extracted for steganalysis.

In recent works, both spatial and temporal redundancies of videos are utilized for video steganalysis. In [18], temporal and spatial redundancies were applied by averaging the frames in the sliding windows to obtain an estimation of the frame. Keren Wang *et al*. [19] used prediction error frame and differential filtering to suppress temporal and spatial redundancies and proposed the subtractive prediction error adjacency matrix (SPEAM), which is similar to the well-known subtractive pixel adjacency model (SPAM) proposed in [20].

In summary, considering only spatial or temporal redundancy of the videos cannot make sufficient use of the video content. While considering both spatial and temporal redundancies independently fails to merge the utilization of spatial and temporal redundancies. This paper presents a novel video steganalysis scheme based on a spatial temporal detector (ST_D) that utilizes the temporal and spatial redundancies of the videos simultaneously. Based on the results of motion and texture analysis, motion and texture factors are introduced to give reasonable weights for histograms of the three descriptors of each block. The experimental results demonstrate the great advantage of our method, especially for compressed videos that are the major components of the Internet videos.

In Section 2, we discusses the rationale for the proposed ST_D features and the method of constructing descriptors and implementing motion and texture analysis. In Section 3,

comparative experiments are conducted to show the performance of the proposed features. Finally, the conclusions are drawn in Section 4.

2. Proposed Method

Based on the analysis and summary presented above, this paper presents a novel spatial temporal detector (ST_D) capable of efficiently utilizing spatial and temporal redundancies of the videos simultaneously instead of independently. We establish three-dimensional orthogonal planes and stack the video sequence as a 3D signal like a cube as shown in **Fig.1** (XY denotes the frame plane and T denotes the temporal axis). For a video sequence containing K frames of size $M \times N$, three descriptors, $D_1(x,y,k)$, $D_2(x,y,k)$, $D_3(x,y,k)$ are defined on the XY, XT and YT planes, respectively, to depict the spatial and temporal relationship between a pixel in the *k*th frame and its adjacent pixels.

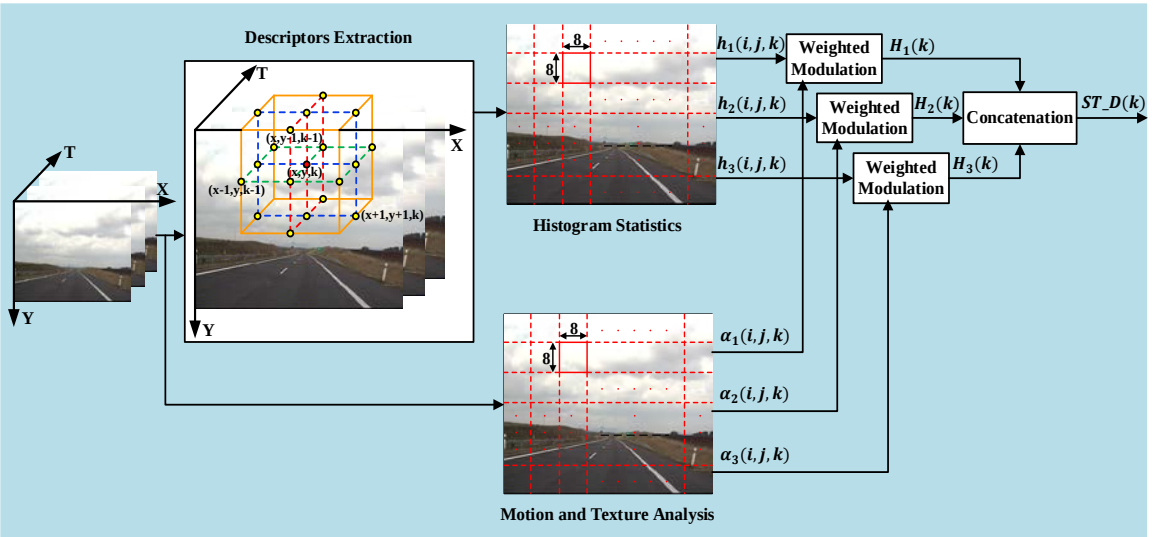


Fig. 1. The video steganalysis model based on a spatial temporal detector (ST_D).

A novel spatial temporal detector (ST_D) for each frame of the video sequences is proposed after determining the histogram statistics and weighted modulation. The comprehensive presentation of ST_D, which incorporates spatial and temporal domain information, is defined as follows:

$$ST\_D(k) = [H_1(k) \quad H_2(k) \quad H_3(k)] \tag{1}$$

Where k represents the index of frames, and $H_n(k)$ represents the weighted modulation of $\alpha_n(i,j,k)$ and $h_n(i,j,k)$ (n = 1: XY plane; 2: XT plane; 3: YT plane). The specific operation of weighted modulation is defined as follows:

$$H_1(k) = \sum_{i=1}^{M/8}\sum_{j=1}^{N/8} \alpha_1(i,j,k) h_1(i,j,k) \tag{2}$$

$$H_2(k) = \sum_{i=1}^{M/8}\sum_{j=1}^{N/8} \alpha_2(i,j,k) h_2(i,j,k) \tag{3}$$

$$H_3(k) = \sum_{i=1}^{M/8}\sum_{j=1}^{N/8} \alpha_3(i,j,k) h_3(i,j,k) \tag{4}$$

where

$$i = \left\lfloor \frac{x}{8} \right\rfloor + 1,\ j = \left\lfloor \frac{y}{8} \right\rfloor + 1 \tag{5}$$

Each frame is divided into non-overlapped blocks that are 8×8 in size for motion and texture analysis. Where i and j represent the horizontal and vertical indexes of each block respectively; "$\lfloor\ \rfloor$" represents the round-down operation; $\alpha_1(i, j, k)$ represents the texture factor of each block defined on the XY plane; $\alpha_2(i, j, k)$ and $\alpha_3(i, j, k)$ represent the motion factors of each block defined on the XT and YT planes, respectively; and $h_n(i, j, k)$ represents the histogram calculated with $D_n(x, y, k)$ for all pixels in each block. As the value of $D_n(x, y, k)$ is between 0 and 255, the histogram with 256 bins is defined as follows:

$$h_n(i, j, k) = hist(\ D_n(x, y, k)\ ) \tag{6}$$

2.1 Descriptor Construction

As shown in **Fig. 1**, three-dimensional orthogonal planes are established to utilize both spatial and temporal redundancies of the videos simultaneously. For a pixel in the kth frame with coordinates (x, y, k) (x and *y* determines the spatial position of a pixel in the *k*th frame of the video sequence), we construct three descriptors $D_1(x, y, k)$, $D_2(x, y, k)$, $D_3(x, y, k)$ on the XY, XT and YT planes, respectively, to depict the spatial and temporal relationship between this pixel and its adjacent pixels in the *k*th frame. The three descriptors with 8 adjacent pixels are defined as follows:

$$D_1(x, y, k) = \sum_{u=-1}^{1} \sum_{v=-1}^{1} s(g_{x+u, y+v, k} - g_{x, y, k}) 2^p \tag{7}$$

$$D_2(x, y, k) = \sum_{u=-1}^{1} \sum_{v=-1}^{1} s(g_{x+u, y, k+w} - g_{x, y, k}) 2^p \tag{8}$$

$$D_3(x, y, k) = \sum_{u=-1}^{1} \sum_{v=-1}^{1} s(g_{x, y+v, k+w} - g_{x, y, k}) 2^p \tag{9}$$

where

$$s(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases} \tag{10}$$

Where $g_{x,y,k}$ corresponds to the gray value of current pixel (the red circle in **Fig. 1**), and $g_{x+u,y+v,k}$, $g_{x+u,y,k+w}$, $g_{x,y+v,k+w}$ correspond to the gray values of its 8 adjacent pixels (the yellow circles in **Fig. 1**) on the XY, XT, and YT planes, respectively. Parameter p is related to the relative positions of the current pixel and its 8 adjacent pixels. The definition of p is the same as that of the local binary patterns (LBP) in [21].

2.2 Motion and Texture Factors

Because the texture complexity of video sequences is related with the histogram distribution of $D_1(x, y, k)$ and local motion intensity of the video sequences is related with the histogram distribution of $D_2(x, y, k)$ and $D_3(x, y, k)$, this will influence the detection accuracy greatly. Based on this reason, motion and texture analysis are performed. Each frame of the video sequences is segmented into non-overlapped blocks that are 8×8 in size. For motion analysis,

motion estimation is implemented with an appropriate motion searching range to find the best matching region based on video compression coding. According to the result of the motion estimation, each block in the *k*th frame is marked with two labels $l_1(k)$ and $l_2(k)$, which are defined as follows:

$$l_1(k)=\begin{cases}1 & MV_1=0, SAD_1\le T\\ 2 & MV_1\ge 0, SAD_1\le T\\ 3 & SAD_1>T\end{cases} \tag{11}$$

$$l_2(k)=\begin{cases}1 & MV_2=0, SAD_2\le T\\ 2 & MV_2\ge 0, SAD_2\le T\\ 3 & SAD_2>T\end{cases} \tag{12}$$

Where MV_1, SAD_1 represent the motion vector and the sum of the absolute differences determined using the *k*th and (*k*-1)th frames of the video sequences. MV_2, SAD_2 represent the motion vector and the sum of the absolute differences determined using the *k*th and (*k*+1)th frames of the video sequences. T is a threshold we set with value equal to 60. According to different combinations of the two labels, $l_1(k)$ and $l_2(k)$, all blocks of each frame can be divided into three types based on their motion intensity. If $l_1(k)=1$ and $l_2(k)=1$, the block is still, else if $l_1(k)=1$ or $l_2(k)=1$ or $l_1(k)=2$ and $l_2(k)=2$, the block is slightly moving, and the remaining blocks are fast moving. The specific relationship between $l_1(k)$, $l_2(k)$ and motion intensity of blocks is defined in **Table 1.**

Table 1. The relationship between $l_1(k)$, $l_2(k)$ and motion intensity of blocks

Motion intensity	$\boldsymbol{l_1(k)}$, $\boldsymbol{l_2(k)}$
Still	$l_1(k)=1 \,\&\&\, l_2(k)=1$
Slightly moving	$l_1(k)=1 \,\|\|\, l_2(k)=1 \,\|\|\, (l_1(k)=2 \,\&\&\, l_2(k)=2)$
Fast moving	others

For texture analysis, differential filtering is utilized to calculate the difference values of each blocks along four directions (horizontal, vertical, diagonal, secondary diagonal) to measure the texture complexity of the blocks. The definition of difference values along the four direction are as follows:

$$H_{x,y,k}=g_{x,y,k}-g_{x,y+1,k} \tag{13}$$

$$V_{x,y,k}=g_{x,y,k}-g_{x+1,y,k} \tag{14}$$

$$D_{x,y,k}=g_{x,y,k}-g_{x+1,y+1,k} \tag{15}$$

$$S_{x,y,k}=g_{x,y,k}-g_{x-1,y+1,k} \tag{16}$$

Where $g_{x,y,k}$, $g_{x,y+1,k}$, $g_{x+1,y,k}$, $g_{x+1,y+1,k}$, $g_{x-1,y+1,k}$ represent the gray values of a pixel and its neighborhoods in each blocks along four directions respectively.

Using various combinations of the difference values distribution, all blocks of each frame can be divided into three types based on their texture complexity. A block with all absolute

difference values of four directions, $|H_{x,y,k}|$, $|V_{x,y,k}|$, $|D_{x,y,k}|$ and $|S_{x,y,k}|$ smaller than a threshold $S1$ is flat, a block with all absolute difference values of four directions ranging from $S1$ to another threshold $S2$ is slightly complex, and the remaining block is complex. $S1$ is a boundary of flat and slightly complex region, the value must be small, so we set $S1=5$ and $S2$ is a boundary of slightly and complex region, the value cannot be larger than $S1$, so we set $S2=30$ .The specific relationship between $H_{x,y,k}$, $V_{x,y,k}$, $D_{x,y,k}$, $S_{x,y,k}$ and texture complexity of blocks is defined in **Table 2**.

Table 2. The relationship between $H_{x,y,k}$, $V_{x,y,k}$, $D_{x,y,k}$, $S_{x,y,k}$ and texture complexity of blocks

Texture complexity	$\boldsymbol{H_{x,y,k}}$, $\boldsymbol{V_{x,y,k}}$, $\boldsymbol{D_{x,y,k}}$, $\boldsymbol{S_{x,y,k}}$
Flat	$\|H_{x,y,k}\|$, $\|V_{x,y,k}\|$, $\|D_{x,y,k}\|$, $\|S_{x,y,k}\| < S1$
Slightly complex	$S1 \le \|H_{x,y,k}\|$, $\|V_{x,y,k}\|$, $\|D_{x,y,k}\|$, $\|S_{x,y,k}\| \le S2$
Complex	others

As there are $3\times 3=9$ kinds of combination of motion and texture labels, through motion and texture analysis, each frame of the video sequences can be subtly divided into 9 kinds of blocks.

2.2.1 Texture Factor

Based on the result of the texture analysis, the texture factor $\alpha_1(i,j,k)$ is introduced to give proper weight for the histogram of $D_1(x,y,k)$ for all pixels in each block; its value varies with the texture complexity of the block. For complex blocks with small spatial correlation, the weight should be small, whereas for slightly complex blocks and flat blocks with relatively weak texture complexity, the weight can be larger. In order to get the reasonable setting standard of $\alpha_1(i,j,k)$, we artificially annotate some representative flat region, slightly complex region and complex region according to the content of the video firstly. Then testing experiment is implemented to see the contribution of detection accuracy of these regions to the detection accuracy of the whole video. By the method, we get the empirical setting standard of texture factor as shown in **Table 3**.

Table 3. The setting standard of $\alpha_1(i,j,k)$ based on texture complexity of blocks

Texture complexity	$\alpha_1(\boldsymbol{i},\boldsymbol{j},\boldsymbol{k})$
Flat	0.7
Slightly complex	0.4
Complex	0

2.2.2 Motion Factor

Similar to the texture factor, $\alpha_2(i,j,k)$, $\alpha_3(i,j,k)$, which serve as motion factors, are introduced to give appropriate weights for the histograms obtained with $D_2(x,y,k)$ and $D_3(x,y,k)$ for all pixels in each block. Their values vary with the motion intensity of the block. For fast moving blocks with small temporal correlation, we set the weights to be small, whereas for still blocks and slightly moving blocks with relatively weak motion intensity, the weights can be larger. In order to get the reasonable setting standard of $\alpha_2(i,j,k)$ and $\alpha_3(i,j,k)$, we artificially annotate some representative still region, slightly moving region and fast moving region according to the content of the video. Then testing experiment is carried out to see the contribution of detection accuracy of these region to the detection accuracy of the whole video. By the method, we get the empirical setting standard of motion factors as shown in **Table 4**.

Table 4. The setting standard of $\alpha_2(i,j,k)$, $\alpha_3(i,j,k)$ based on texture complexity of blocks

Motion intensity	$\alpha_2(\boldsymbol{i},\boldsymbol{j},\boldsymbol{k})$	$\alpha_3(\boldsymbol{i},\boldsymbol{j},\boldsymbol{k})$
Still	0.7	0.7
Slightly moving	0.4	0.4
Flat moving	0	0

3. Experimental Results and Analysis

3.1 Experimental Settings

3.1.1 Test Sequences

The video database for experiments includes 14 standard YUV video sequences found at http://trace.eas.asu.edu/yuv/index.html. The parameters of the test video sequences are listed in **Table 5** and the contents of the test video sequences are presented in **Fig. 2**. For simplicity, only the first 90 frames of each video are used for the experiments. After embedding secret information on cover video sequences with steganography method to get the stego video sequences, cover and stego videos were encoded into MPEG2 and H.264 formats. We compressed the videos into MPEG2 format by using VcDemo [22] with a motion searching range of 15, GOP structure of IBBPBBPBBPBB, and bit rates of 2 Mb/s and 5 Mb/s as in [19].We also compressed the videos into H.264 format by using FFMPEG [23] with a motion searching range of 15 and bit rates of 2 Mb/s and 5Mb/s as in [19].

3.1.2 Video Steganography Method

Secret information is embedded in the Y component of each frame of the video sequences using the spread spectrum steganography technique described in [24] with a spread spectrum factor of 128 and different combinations of scaling factor α and chip-rate (cr) . For

experiments involving compressed videos, only $cr=1$ was tested, because the operation of video compressing may erase the trace of embedding secret information.

3.1.3 Classifier

The classifier is implemented using SVMs with χ^2 (chi-square) kernels. Because c serves as the penalty parameter, and γ serves as the controlling parameter balance the complexity and accuracy of the classifier. Thus, we used the multiplicative grid searching to find the optimal parameter pair (c_0, γ_0) with the highest accuracy. In this paper, all grid points of $(c=2^2, 2^3, 2^4, 2^5, \gamma=2^{-2}, 2^{-3}, 2^{-4}, 2^{-5})$ were tested.

3.1.4 Evaluation Mechanism

The sequence-level cross validation algorithm reported in [19] is used to evaluate the performance of the video steganalysis scheme. When 9 videos were chosen for training, the whole loops of i for each (c, γ) can be as large as $C_{14}^{9}=2002$. For simplicity, we set the maximum of i to be 50.

Table 5. Parameters of the test video sequences

Sequences	Class	Frames	Size
Akiyo	YUV420	300	352*288
Bus	YUV420	150	352*288
Coastguard	YUV420	300	352*288
Container	YUV420	300	352*288
Flower	YUV420	250	352*288
Hall	YUV420	300	352*288
Highway	YUV420	2000	352*288
Mobile	YUV420	300	352*288
Mother-daughter	YUV420	300	352*288
News	YUV420	300	352*288
Silent	YUV420	300	352*288
Stefan	YUV420	90	352*288
Tempete	YUV420	260	352*288
Waterfall	YUV420	260	352*288


Akiyo


Bus


Coastguard


Container


Flower


Hall


Highway


Mobile


Mother-daughter

News


Silent

Stefan


Tempete

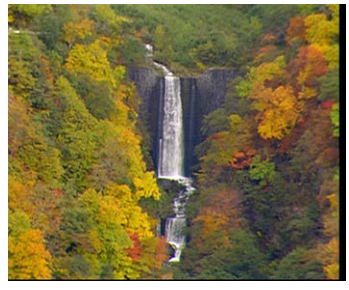
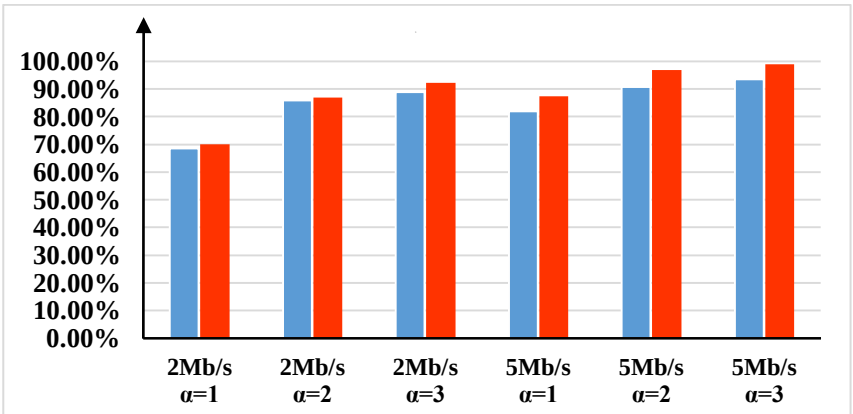

Waterfall

Fig. 2. Test video sequences.

3.2 Experiment Results

3.2.1 Effectiveness of Motion and Texture Factors

To demonstrate that the operation of introducing motion and texture factors is effective, we implemented experiments by using ST_D features and raw descriptors (RD) without motion and texture factors for compressed videos in MPEG2 format. Because this experiment was a confirmatory test, we set the maximum of i to be 20 for simplicity. **Fig. 3** presents the experiment results of the ST_D features and raw descriptors.



	2Mb/s α=1	2Mb/s α=2	2Mb/s α=3	5Mb/s α=1	5Mb/s α=2	5Mb/s α=3
RD(%)	68.55	85.89	88.96	81.98	90.68	93.57
ST_D(%)	69.93	87.02	92.31	87.46	96.91	98.89

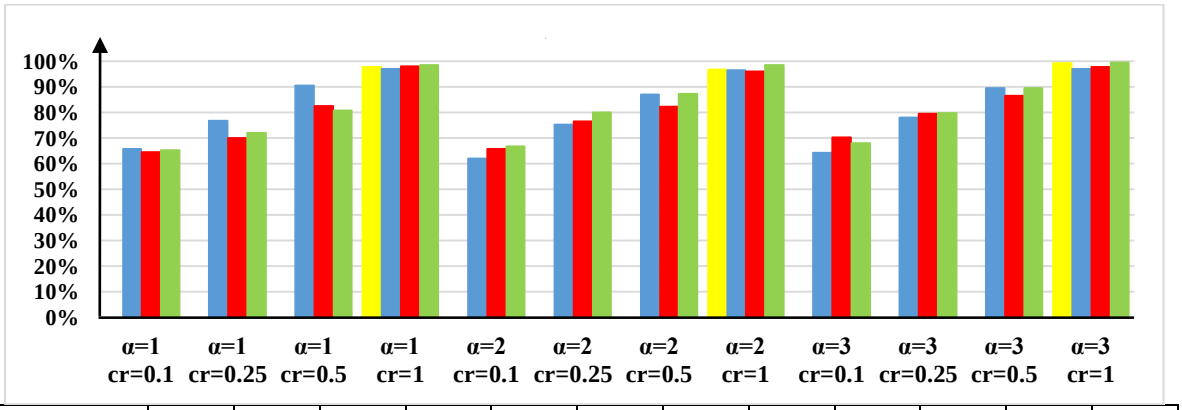
Fig. 3. Detection accuracy of raw descriptors and ST_D features for compressed video sequences (MPEG2).

Fig. 3 shows that the ST_D features perform much better than raw descriptors, implying that the operation of introducing motion and texture factors for histograms of the three descriptors of each block based on the results of motion, texture analysis is effective, and our proposed ST_D features are more sensitive to steganography.

3.2.2 Steganalysis of Uncompressed Video Sequences

To evaluate the performance of our proposed ST_D features, we conducted a comparative experiment involving rich model (RM) proposed in [25], SPEAM features with a motion searching range of 3 and the well-known SPAM features on uncompressed video sequences. **Fig. 4** shows the detection accuracy. As the dimension of RM features is up to 12753 with high computational complexity, we just test them for $cr = 1, \alpha = 1, 2, 3$.

This figure indicates that the features of SPAM and SPEAM generally have similar performance for the steganalysis of uncompressed videos, whereas those of our proposed ST_D seem superior in most cases even when compared with the RM features with high dimensionconfirming that considering spatial and temporal redundancies simultaneously may generate more useful information for video steganalysis. The exceptions of $\alpha = 1, cr = 0.1, 0.25, 0.5$ and $\alpha = 3, cr = 0.1$ occur may because for uncompressed videos with small cr values, merging the utilization of spatial and temporal redundancies may lead to interference in detection.

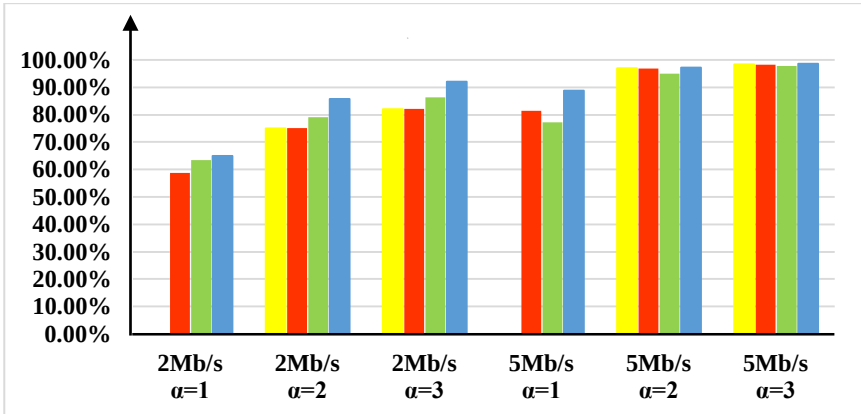


RM(%)				97.71				96.80				99.34
SPAM(%)	65.69	76.62	90.56	96.96	61.98	75.15	86.97	96.60	64.28	77.90	89.44	96.87
SPEAM(%)	64.43	69.88	82.42	97.94	65.85	76.55	82.25	95.93	70.19	79.47	86.57	97.72
ST_D(%)	63.35	71.98	80.76	98.37	66.79	79.92	87.29	98.73	68.10	79.69	89.60	99.57

Fig. 4. Detection accuracy of SPAM, SPEAM and ST_D features for uncompressed video sequences.

3.2.3 Steganalysis of Compressed Video Sequences (MPEG2)

In order to further demonstrate the effectiveness of our proposed ST_D features for real steganalytic systems, we performed a similar experiment on compressed video sequences in MPEG2 format by using ST_D, RM, SPEAM and SPAM features. Because of the high computational complexity of RM features, we just test them for $\alpha = 2, 3$. The experimental results are illustrated in **Fig. 5**.



RM (%)		76.68	84.57		90.14	96.81
SPAM(%)	58.70	75.02	82.11	81.33	96.84	98.27
SPEAM(%)	63.48	79.16	86.43	77.22	94.89	97.86
ST_D(%)	64.86	85.70	92.13	88.75	97.08	98.51

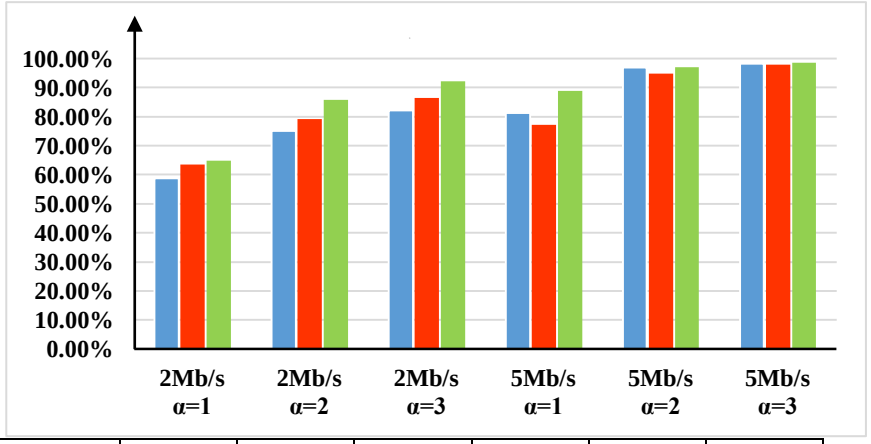
Fig. 5. Detection accuracy of SPAM, SPEAM and ST_D features for compressed video sequences (MPEG2).

Fig. 5 implies that RM, SPAM and SPEAM features exhibited similar performance, whereas the ST_D features had substantially better performance in all of the cases tested, suggesting that this device may generate more information useful for video steganalysis.

3.2.4 Steganalysis of Compressed Video Sequences (H.264)

A similar experiment was carried out on compressed video sequences in H.264 format to further confirm that our method's advantages are generalizable. The detection results are shown in **Fig. 6**.

The results are similar to those obtained for videos compressed in MPEG2 format, illustrating that the ST_D features are generally advantageous for video steganalysis compared with SPAM and SPEAM features, especially for compressed videos, which constitute most Internet videos.



	2Mb/s α=1	2Mb/s α=2	2Mb/s α=3	5Mb/s α=1	5Mb/s α=2	5Mb/s α=3
SPAM(%)	73.33	95.19	96.81	93.98	96.95	97.02
SPEAM(%)	76.60	92.74	96.26	94.17	96.35	97.83
ST_D(%)	81.98	95.22	97.91	94.19	99.50	99.64

Fig. 6. Detection accuracy of SPAM, SPEAM and ST_D features for compressed video sequences (H.264)

4. Conclusion

In this paper, a novel video steganalysis scheme is proposed against spatial domain steganography technology using ST_D, which is an efficient operator exploiting spatial and temporal redundancies of the videos simultaneously. Considering the influence of local motion intensity and texture complexity on the histogram distribution of three descriptors defined on the three planes, each frame is divided into non-overlapped blocks that are 8×8 in size for motion and texture analysis. Texture and motion factors are introduced to give reasonable weights for histograms of the three descriptors for all pixels in each block. Thus, our scheme achieves an acceptable trade-off between spatial and temporal redundancies of the videos. In this work, several comparative experiments were conducted, and the experimental results confirm that our proposed features are much more effective than RM features with high dimension, SPAM features and SPEAM features, especially for compressed videos that are the major components of the Internet videos and may, thus, be of practical and commercial value.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (61572356 and 61303208) and the Tianjin Research Program of Application Foundation and Advanced Technology (15JCQNJC41600).

References

[1] Y.C. Liao, C.H. Chen, T.K. Shih, N.C. Tang, "Data hiding in video using adaptive LSB," in *Proc. of Pervasive Computing (JCPC), 2009 Joint Conferences on*, pp.185-190, 2009. Article (CrossRef Link)

[2] P. Yadav, N. Mishra, S. Sharma, "A secure video steganography with encryption based on LSB technique," in *Proc. of IEEE International Conference on Computational Intelligence and Computing Research*, pp.1-5, 2013. Article (CrossRef Link)

[3] R. Khare, R. Mishra, I. Arya, "Video Steganography Using LSB Technique by Neural Network," in *Proc. of International Conference on Computational Intelligence and Communication Networks*, pp. 898-902, 2015. Article (CrossRef Link)

[4] J. Singh, A. Dubey, "MPEG-2 video watermarking using quantization index modulation," *in Proc. of IEEE International Conference on Internet Multimedia Services Architecture and Application*, 41(2): 108-14, 2010. Article (CrossRef Link)

[5] Y. Guan, Y. Zhu, X. Liu, G. Luo, "A digital blind watermarking scheme based on quantization index modulation in depth map for 3D video," in *Proc. of International Conference on Control Automation Robotics & Vision*, pp.346-351, 2015. Article (CrossRef Link)

[6] S. Liu, F. Shi, J. Wang, S. Zhang, "An improved spatial spread-spectrum video watermarking," in *Proc. of International Conference on Intelligent Computation Technology & Automation*, 1: 587-590, 2010. Article (CrossRef Link)

[7] L. Wei, R.Q. Hu, D.A. Pados, G. Wu, "Optimal multiuser spread-spectrum data embedding in video streams," in *Proc. of Global Communications Conference*, 2015. Article (CrossRef Link)

[8] R.J. Mstafa, K.M. Elleithy, "A high payload video steganography algorithm in DWT domain based on BCH codes (15, 11)," in *Proc. of International IEEE Wireless Telecommunications Symposium*, 2015. Article (CrossRef Link)

[9] G.R. Rajesh, A.S. Nargunam, "Steganography algorithm based on discrete cosine transform for data embedding into raw video streams," in *Proc. of Iet Chennai Fourth International Conference on Sustainable Energy and Intelligent Systems*, 2013. Article (CrossRef Link)

[10] X. Nan, Z. Pei, Z.T. Li, "A Steganography Algorithm Based on ±1 DCT Coefficients for H.264/AVC," in *Proc. of International Conference on Frontier of Computer Science & Technology*, pp.259-263, 2015. Article (CrossRef Link)

[11] P. Paulpandi, D.T. Meyyappan, "Hiding Messages Using Motion Vector Technique In Video Steganography," in *Proc. of International Journal of Engineering Trends & Technology*, 3(3), 2012. Article (CrossRef Link)

[12] H. Zhang, Y. Cao, X. Zhao, "Motion vector-based video steganography with preserved local optimality," *Multimedia Tools & Applications*, pp. 1-17, 2015. Article (CrossRef Link)

[13] K. Tasdemir, F. Kurugollu, S. Sezer, "Spatio-temporal rich model for motion vector steganalysis," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015. Article (CrossRef Link)

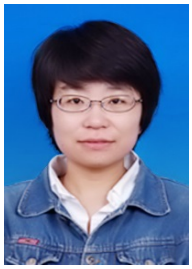
[14] U. Budhia, D. Kundur, T. Zourntos, "Digital Video Steganalysis Exploiting Statistical Visibility in the Temporal Domain," *IEEE Transactions on Information Forensics & Security*, 1(4): 502-516, 2007. Article (CrossRef Link)

[15] V. Pankajakshan, A.T.S. Ho, "Improving Video Steganalysis Using Temporal Correlation," in *Proc. of International Conference on International Information Hiding and Multimedia Signal Processing*, 1:287-290, 2007. Article (CrossRef Link)

[16] X. Xu, J. Dong, T. Tan, "Universal spatial feature set for video steganalysis," pp. 245-248, 2012. Article (CrossRef Link)
[17] V. Rana, R. Mishra, P.K. Bora, S. Kashyap, "Novel scheme of video steganalysis for detecting antipodal watermarks," in *Proc. of TENCON 2008 - 2008 IEEE Region 10 Conference*, pp. 1-5, 2008. Article (CrossRef Link)
[18] K. Kancherla, S. Mukkamala, "Video steganalysis using spatial and temporal redundancies," in *Proc. of International Conference on High PERFORMANCE Computing & Simulation*, pp. 200-207, 2009. Article (CrossRef Link)
[19] K. Wang, J. Han, H. Wang, "Digital video steganalysis by subtractive prediction error adjacency matrix," *Multimedia Tools & Applications*, 72 (1): 313-330, 2014. Article (CrossRef Link)
[20] T. Pevný, P. Bas, J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on Information Forensics & Security*, 5(2): 215-224, 2010. Article (CrossRef Link)
[21] T. Ojala, M. Pietikäinen, T. Mäenpää, "Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *Lecture Notes in Computer Science*, 1842(7):404-420, 2000. Article (CrossRef Link)
[22] Vcdemo, http://insy.ewi.tudelft.nl/content/image-and-video-compression-learning-tool-vcdemo/.
[23] Ffmpeg library, http://ffmpeg.org/.
[24] F.H. Hartung, B. Girod, "Digital Watermarking of Raw and Compressed Video," in *Proc. of SPIE - The International Society for Optical Engineering*, 2952: 205-213, 1997. Article (CrossRef Link)
[25] J. Fridrich, J. Kodovsky, "Rich Models for Steganalysis of Digital Images," *IEEE Transactions on Information Forensics & Security*, pp. 7(3):868-882, 2012. Article (CrossRef Link)

Yuting Su received the B.S.degree, the M.S. degree and the Ph.D. degree from Tianjin university, Tianjin, China, in 1995, 1998 and 2001 respectively. He is currently a Professor with the School of Electronic Information Engineering, Tianjin University, Tianjin, China. His research interests include multimedia content analysis, information security, and tensor decomposition.

Fan Yu is a graduate student in school of Electronic Information Engineering, Tianjin University, Tianjin, China. Her research interests include pattern recognition and digital video steganalysis.



Chengqian Zhang received the B.S.degree in electronic information engineering in 2003, the M.S.degree and the Ph.D.degree in signal processing in 2006 and 2009 respectively, all from Tianjin University, Tianjin, China. She is currently an associate professor with the School of Electronic Information Engineering, Tianjin University, Tianjin, China. Her research interests include multimedia security and image processing.