Signal Processing: *Image Communication* 25 (2010) 577–587

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Reliable JPEG steganalysis based on multi-directional correlations

Yong Wang [a,*], Jiufen Liu [a], Weiming Zhang [a], Shiguo Lian [b]

[a] *Department of Applied Mathematics, Information Science and Technology Institute, Zhengzhou, China*
[b] *France Telecom R&D (Orange Labs), Beijing, China*

ARTICLE INFO

Article history:
Received 29 December 2009
Accepted 30 June 2010

Keywords:
Steganography
Steganalysis
Calibration
Multi-directions
Analysis of variance
Support vector machine

ABSTRACT

With the wide application of JPEG images, JPEG steganography attracts more and more researchers, and accordingly, the detection of JPEG steganography becomes also important. There exist some blind JPEG steganalysis methods, while most of them are either unreliable or time-consuming. This paper presents a reliable and efficient steganalysis scheme to detect the popular JPEG steganography algorithms. First, a novel kind of transition probability matrix is constructed to describe correlations of the quantized DCT coefficients in multi-directions. Then, by merging two different calibrations, a 96-dimensional feature vector is extracted. Additionally, the SVM (Support Vector Machine) is trained to build the steganalyzer. Finally, the proposed feature is evaluated, and a series of experiments are performed on 4 kinds of typical steganography in different embedding ratios, showing that for these steganography algorithms, the new method is more reliable than the best effective blind detection methods existed.

1. Introduction

With the rapid development of network technology, the Internet has been widely used for information exchange. During the information exchange, security is an important issue [1–3]. To solve the problem, two kinds of techniques are often used, i.e., cryptography and steganography. In the former, the sender encrypts the secret information into an unintelligible form with a cipher, and the receiver could decrypt the information to get the original one. The meaningless encrypted information would raise the concerns of code breakers who could intercept the suspicious data. Differently, in the latter, i.e., steganography, the sender embeds messages in the innocuous-looking cover media, such as e-mails [4], videos [5] or digital images [6]. Then, the modified cover object, which is called the stego object, is sent to the receiver over the Internet. The receiver could attain the secret messages from the stego objects. The imperceptibility of the stego objects would cheat the visual observer to ensure the information transmission security.

The main requirement of steganography is the statistical undetectability, which means that the attacker could not judge whether an object is the stego or cover based on the statistics of the object. To accomplish it, many steganographic techniques for images have been developed. They could be classified into two classes, i.e., spatial domain techniques like Least Significant Bit (LSB) [7] and LSB Matching [8], and transform domain techniques like F5 [9], OutGuess [10], JpHide [11], StegHide [12] and YASS [13]. During embedding, all the steganographic methods would bring distortions to the cover image inevitably. To reduce the distortions, a general technique called steganographic codes is usually used [14–17]. In other words, steganographic codes would improve the embedding efficiency. Additionally, steganographic codes are independent of the cover object's types, which make it utilized in either spatial domain or transform domain steganographic techniques.

* Corresponding author. Tel.: +86 15938771080.
E-mail addresses: creasynec@hotmail.com (Y. Wang), jiufenliu@163.com (J. Liu), zwmshu@gmail.com (W. Zhang), shiguo.lian@ieee.org (S. Lian).

0923-5965/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2010.06.003

The counterpart of steganography is steganalysis, which aims to detect the presence of secret messages and even to extract them. Generally, the steganography is regarded as being broken when the secret messages are discovered [18]. There are two kinds of steganalytic methods, i.e., the targeted steganalysis and the blind steganalysis. The former could reveal the secret messages or even estimate the embedding ratio with the knowledge of the specific steganographic algorithm. For example, RS [19], WAM [20] and its improved version [21] can detect the spatial steganography reliably, and Li et al. [22] and Zhang and Zhang [23] could discover YASS and LSB matching steganography. While the latter, blind steganalysis, aims at discovering the presence of the hidden messages from unknown steganographic programs. Generally, blind steganalysis extracts sensitive features at first, such as the Markov transition probability matrix [24], statistical moments of characteristic function of subband histograms [25], and the merging feature set [26]. And then, a classifier such as SVM (Support Vector Machine) [27] or artificial neural network [28], is trained to distinguish the stego image from the cover. Because of the universality and flexibility, blind steganalysis is more attractive in many practical applications.

Due to the popularity of JPEG image on the Internet, we constrain our discussion to the JPEG format in this paper. During the past decade, many JPEG steganographic algorithms [9–13] have been reported, and some works have been carried out on blind JPEG steganalysis. Fridrich [29] first investigated a blind steganalyzer targeted on JPEG steganography, where a set of distinguishable features are extracted from the DCT (Discrete Cosine Transform) domain and the cropped calibration is proposed to estimate the statistics of the original image. The scheme could successfully break four kinds of JPEG steganography to some extent, but it is not satisfactory because the rather limited number of features could not fully exhibit the image's characteristics. Shi et al. [24] regarded the Markov transition probability matrix as the raw representation of the statistical characteristics of the image, based on which a 324-dimensional feature vector is obtained along four directions, i.e., horizontal, vertical, diagonal, and minor diagonal. By exploring the correlations of coefficients in different directions, the scheme achieves superior performance than the method in [29]. But the extracted features could not exhibit the coefficient correlations on inter-block. Fu et al. [30] proposed a Markov-based scheme to break the JPEG steganographic algorithm. Different from Shi's scheme, Fu et al. focused on capturing correlations on both intra-block and inter-block of the quantized DCT coefficients. This scheme's detection results are more convincing than the ones of [24], but it is still not satisfactory especially when detecting stego images with low embedding ratio. Pevny and Fridrich [26] merged the existing DCT feature set and the Markov-based feature set to improve the performance of steganalysis markedly. The steganalyzer could detect six kinds of JPEG steganography reliably, outperforming all the previous schemes [24,29,30] mentioned above, which is regarded as the best effective steganalyzer having been presented, but it costs much time to extract those complicated features.

Since the Markov transition probability matrix can capture the second order characteristics, it has been used in blind JPEG steganalysis widely and effectively [24,30]. However, the Markov approach [24] considers the coefficient correlation in only one direction at each time, so it cannot exhibit the correlations of the neighboring DCT coefficients completely. Thus, the multi-directional based feature should be considered. Motivated by this idea, we construct a new transition probability matrix in this paper to describe the multi-directional correlations of the quantized DCT coefficients. Through theoretical analysis, we show that the new transition probability matrix would collect more information of the correlations than the Markov transition probability matrix. Based on the proposed matrix, we extract features from both intra-blocks and inter-blocks., whose sensitivity will be evaluated and compared with the Markov features. Moreover, two kinds of calibrations are combined to minimize the impact of the cover image's energy. And finally, a blind classifier is set up to distinguish the stego images from the covers. A set of experiments are performed on 4 popular JPEG steganographic algorithms: F5, OutGuess, JpHide and StegHide, to evaluate the performance of the proposed blind steganalysis method. In experiments, we will compare the proposed method with the state of the art blind JPEG steganalysis methods, especially the best effective JPEG steganalyzer in [26].

The remainder of this paper is organized as follows. In Section 2, we briefly introduce the most related algorithms. The proposed blind JPEG steganalysis method, including the feature extraction and SVM-based classifier, is presented in detail in Section 3. In Section 4, we evaluate the proposed features, and compare the proposed classifier with existing ones by detecting various steganographic schemes in different embedding ratios. Finally, in Section 5, the paper is concluded and future work is given.

2. Related work

This section gives a review of the steganalysis schemes based on Markov approach, followed by an introduction of two existing calibrations techniques which are most related to our proposed scheme.

2.1. JPEG steganalysis based on Markov approach

There exists some JPEG steganalysis [24,30] based on Markov approach. Markov transition probability matrix (MTPM) describes the correlation of two elements whose distance is r at the angle θ, as shown in Fig. 1. For a given matrix I whose elements vary in [1, n], the $n \times n$ MTPM is defined as

$$p_{r,\theta}(x,y) = p(F(i_1,j_1) = x | F(i_2,j_2) = y), \quad (1)$$

where $i_2 = i_1 + r\cos\theta$, $j_2 = j_1 + r\sin\theta$ and $F(i, j)$ the value of matrix element at location (i, j). When $r=n$, it is called n-step MTPM, which refers to the transition probabilities separated by $n-1$ elements.

According to the correlation of the neighboring coefficients, the image would be considered as the Markov

process. Based on this view, Shi et al. [24] characterized the different JPEG coefficient matrix by using MTPM. To enlarge the disorder caused by message embedding, different DCT coefficients are calculated along four directions: horizontal, vertical, diagonal and minor diagonal, which would be denoted as follows:

$$F_h(u,v) = F(u,v) - F(u+1,v) \tag{2}$$

$$F_v(u,v) = F(u,v) - F(u,v+1) \tag{3}$$

$$F_d(u,v) = F(u,v) - F(u+1,v+1) \tag{4}$$

$$F_m(u,v) = F(u+1,v) - F(u,v+1) \tag{5}$$

Shi et al. [24] used one-step Markov transition probability matrix to extract features from different DCT coefficients, which could be denoted by

$$M_h(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(F_h(u,v)=i, F_h(u+1,v)=j)}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v}\delta(F_h(u,v)=i)} \tag{6}$$

$$M_v(i,j) = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(F_v(u,v)=i, F_v(u,v+1)=j)}{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-1}\delta(F_v(u,v)=i)} \tag{7}$$

$$M_d(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v-2}\delta(F_d(u,v)=i, F_d(u+1,v)=j)}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_d(u,v)=i)} \tag{8}$$

$$M_m(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v-2}\delta(F_m(u,v)=i, F_m(u+1,v)=j)}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_m(u,v)=i)} \tag{9}$$

where S_u and S_v represent the width and height of the image, respectively, and

$$\delta(m,n) = \begin{cases} 1 & if \quad m=n \\ 0 & if \quad m\neq n \end{cases} \tag{10}$$

According to the experimental results in [24], most of the DCT difference coefficients fall into the range $[-4, 4]$. To reduce the computational complexity, the element whose absolute value is larger than 4 is reassigned a new absolute value 4 without changing the sign. That is, $9\times 9=81$ dimensional features are extracted for each MTPM, and totally $4\times 9\times 9=324$ dimensional features are constructed for future steganalysis.

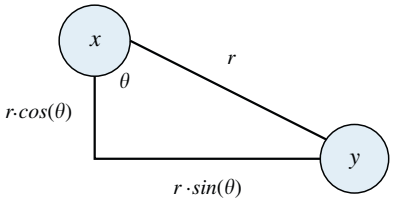

Fig. 1. The formation of the Markov transition probability matrix.

2.2. Calibrations

The concept of calibration is first introduced in [31] to attack F5 [9], and further used in [29] to improve the detection accuracy of the feature-based blind JPEG steganalysis. The technique used in [29], named as *image cropped calibration* in this study, could recover the statistics of the cover and remove the image-to-image variations. That reduces the effect of the diversity of images and makes the extracted features more sensitive.

(1) *Image cropped calibration*: As Fig. 2 shows, decompress the test image I to spatial domain, then crop it by first 4 rows and 4 columns, finally recompress the cropped image using the same quantization table as the cover image to get the calibrated image I_c.
Then the calibrated feature would be caculated as follows:

$$f_1 = f(I) - f(I_c) \tag{11}$$

where $f(I)$ represents the extracted feature of the image I.
Huang and Huang [32] generalized the concept of the calibration as macroscopic and microscopic situation. The image cropped calibration, which could estimate the global histogram of the cover image, is regarded as macroscopic calibration. Huang and Huang perform mean filtering on spatial domain of the JPEG image for the purpose of getting the same gross representation of the cover and stego image. That would be considered as a kind of microscopic calibration, which would be merged with the macroscopic one to improve the performance of steganalysis. In this study, we replace the mean filtering as prediction error, which is expected to be more effective.
The prediction error [33] is another technique used in steganalysis, which would find a raw representation of images and make it easier to explore the minor change between cover and stego images. The technique, which we call prediction error calibration in this study, would be used to reduce the effect of the carrier huge energy. It is expected that the prediction error calibration could erase image content and enhance the distortion caused by embedding.

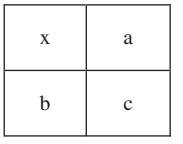

Fig. 3. The neighboring pixels.

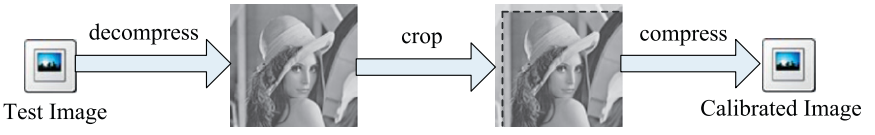

Fig. 2. Image cropped calibration [29].

(2) Prediction error calibration: Let x be a pixel of the test image I, the neighboring pixels are a, b and c, as Fig. 3 shows

The predicted value of x could be expressed as

$$\hat{x}=\begin{cases}\max(a,b) & \text{if} \quad c\le\min(a,b)\\ \min(a,b) & \text{if} \quad c\ge\max(a,b)\\ a+b-c & \text{otherwise}\end{cases} \tag{12}$$

Then the calibrated feature would be calculated by

$$f_2=f(I-I_p) \tag{13}$$

where I_p represents the corresponding predicted image of the image I.

3. The proposed steganalysis method

In this section, we describe the architecture of the proposed method, the feature extraction method based on a new multi-directional transition probability matrix and calibrations, and the summarized steps of the proposed detection method.

3.1. Architecture of the proposed steganalysis method

As Fig. 4 shows, the proposed method is composed of four main phases:

(1) Trained sets construction: Collect various kinds of images, embed secret messages into them with different JPEG steganography, such as F5, OutGuess, etc., and then merge the covers and stegos to construct the training sets.
(2) Calibrations: Perform two kinds of calibrations to get the macroscopic calibrated images and microscopic calibrated images, respectively.
(3) Feature extraction: Utilize the proposed multi-directional probability transition matrix to extract the 96-dimensional features from the calibrated images' intra-blocks and inter-blocks.
(4) Detection: Train the classifier and obtain some optimal parameters, which would be used in the following detection. Calculate the features of the test images as described in Phases 2 and 3, and then send them to the classifier to determine whether they are cover or stego images.

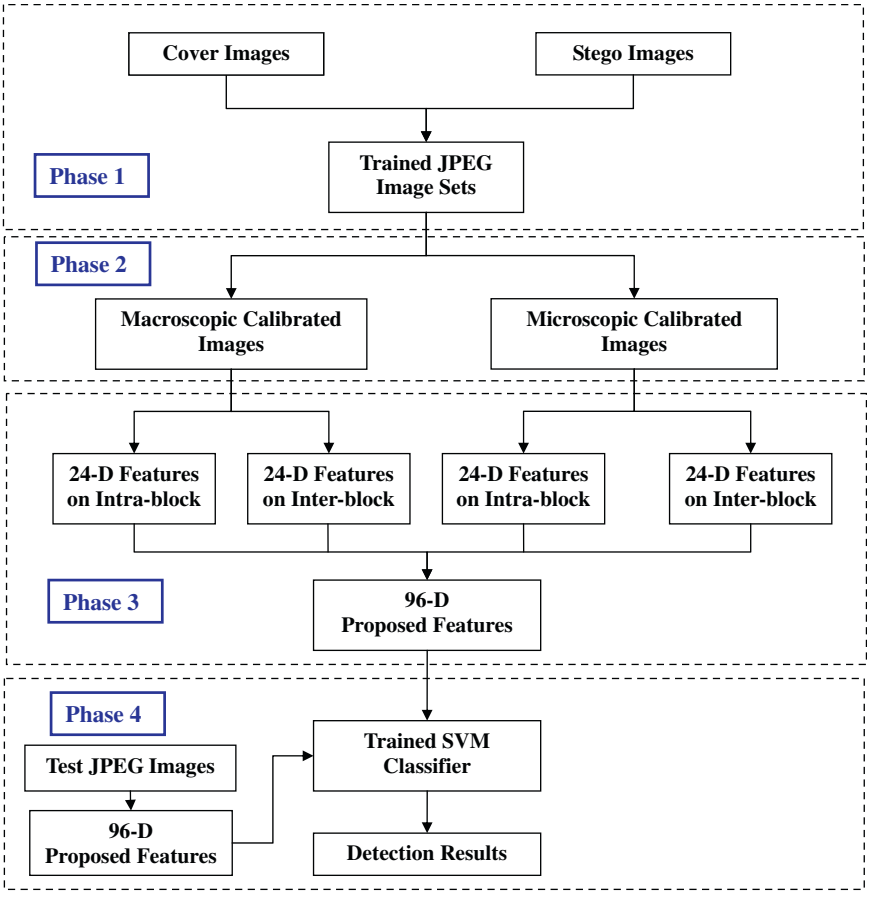

Fig. 4. Architecture of the proposed steganalysis method.

In these phases, the key point is the feature extraction. In the following content, we will focus on the proposed feature extraction method.

3.2. Feature extraction

3.2.1. Transition probability matrix based on correlations in multi-directions

Suppose the matrix $\mathbf{M}$ has the dimension $h \times w$, where its element $\mathbf{M}(i,j)$ is integer belonging to $[1, N]$. If $\mathbf{M}(i,j)$ is not on the boundary, it would have 8 neighboring elements, which are denoted as

$$\mathbf{M}(i+r,j+s), \quad |r| \le 1, \quad |s| \le 1, \quad r^2+s^2 \neq 0. \tag{14}$$

For the neighboring region,

$$l = \sum_{|r| \le 1, |s| \le 1, r^2+s^2 \neq 0} \delta(|\mathbf{M}(i,j)-\mathbf{M}(i+r,j+s)|,k) \tag{15}$$

In Eq. (15), l represents how many times $|\mathbf{M}(i,j)-\mathbf{M}(i+r,j+s)|$ equals k for $0 \le k \le N-1$. If $l=0$, it means that the probability of $|\mathbf{M}(i,j)-\mathbf{M}(i+r,j+s)| = k$ is 0. In other words, the absolute value of difference would not equal k in the neighboring region. This case is not under our consideration, so l belongs to [1,8].

Scan all the elements within the matrix boundaries, the probability of that k appears l times in the matrix could be calculated as follows:

$$p(k,l) = \frac{\sum_{i,j}\delta(\sum_{r,s}\delta(|\mathbf{M}(i,j)-\mathbf{M}(i+r,j+s)|,k),l)}{\sum_{l=1}^{8}\sum_{k=0}^{N-1}\sum_{i,j}\delta(\sum_{r,s}\delta(|\mathbf{M}(i,j)-\mathbf{M}(i+r,j+s)|,k),l)} \tag{16}$$

where $1 \le i \le w$, $1 \le j \le h$, $|r| \le 1$, $|s| \le 1$, $r^2+s^2 \neq 0$, $0 \le k \le N-1$ and $1 \le l \le 8$.

Regarding $p(k, l)$ as an element of the new transition probability matrix in the position (k, l), an $N \times 8$ matrix could be attained to capture the multi-directional correlations of the neighboring elements. For convenience, we define $f(\mathbf{M})$ as the multi-directional transition probability matrix of the matrix $\mathbf{M}$

$$f(\mathbf{M}) = \begin{pmatrix} p(0,1), & p(0,2), & \cdots, & p(0,8) \\ p(1,1), & p(1,2), & \cdots, & p(1,8) \\ \vdots & \vdots & \ddots & \vdots \\ p(N-1,1), & p(N-1,2), & \cdots, & p(N-1,8) \end{pmatrix} \tag{17}$$

As a simple example, a 3×3 matrix whose elements vary in [1,4] has only one 8-neighboring region with a central element 2 as follows:

$$\mathbf{M} = \begin{pmatrix} 2 & 1 & 2 \\ 3 & 2 & 4 \\ 3 & 2 & 1 \end{pmatrix} \tag{18}$$

Then, the proposed transition probability matrix of $\mathbf{M}$ is

$$f(\mathbf{M}) = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{bmatrix} 0 & 0 & 1/3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \tag{19}$$

where the row indexes in {0, 1, 2, 3} represent the absolute values of the difference, and the column indexes in {1, 2, …, 8} represent the frequency of their appearance.

Define $m, n \in [1, N]$, and it could be observed that when $r=0$, $s=1$, $k=|m-n|$, and $l=1$, the numerator of Eq. (16) would be rewritten as

$$p(|\mathbf{M}(i,j)-\mathbf{M}(i,j+1)| = |m-n|) = \sum_{i=1}^{w}\sum_{j=1}^{h}\delta(|\mathbf{M}(i,j)-\mathbf{M}(i,j+1)|,|m-n|) \tag{20}$$

Eq. (20) is very similar to one-step MTPM in the horizontal direction which is expressed as

$$p(\mathbf{M}(i,j+1)=n|\mathbf{M}(i,j)=m) = \frac{\sum_{i=1}^{w}\sum_{j=1}^{h}\delta(\mathbf{M}(i,j)=m,\mathbf{M}(i,j+1)=n)}{\sum_{i=1}^{w}\sum_{j=1}^{h}\delta(\mathbf{M}(i,j)=m)} \tag{21}$$

It is observed that the situation described by Eq. (21) is included in Eq. (16). Therefore the proposed transition probability would be considered as a kind of joint probability with partial probability being the Markov transition probability.

In fact, in order to cover all the information of the neighboring region, we should build an 8-dimensional matrix whose each dimension equals N. However, this will bring enormous computational complexity, so some information is sacrificed in the proposed matrix. Even so this novel transition probability matrix contains more information than the Markov approach. It is also expected that the feature would be more sensitive and effective.

3.2.2. Feature construction based on the proposed matrix

In the following content, the JPEG feature on intra-block and inter-block will be extracted based on the proposed matrix.

JPEG feature extraction from intra-blocks: Consider the matrix consisting of all the JPEG quantized DCT coefficients. All 8×8 blocks of the matrix are denoted by $\mathbf{M}_t$, where $t = 1,2,\ldots,K$, and K is the total number of blocks. In each 8×8 block, most of the energy is concentrated in the low frequency coefficients, and the DC (direct current) coefficients are not touched by modern JPEG steganography. Therefore, we only focus on the 10 low frequency AC coefficients above the minor diagonal, which all have 4 neighbors. Because Fu et al. [30] pointed out 96.59% of the AC coefficients fall into $[-7, 7]$, the coefficients in this range will be utilized to reduce the size of proposed transition probability matrix. If the coefficient is either larger than 7 or smaller than -7, it will be set to 7 or -7. Then, the proposed transition probability matrix for $\mathbf{M}_t$ will be calculated by $f(\mathbf{M}_t)$, and the intra-block average transition probability matrix will be computed by $H_{intra} = 1/K(\sum_{r=1}^{K} f(\mathbf{M}_t))$, where H_{intra} is a 15×8 matrix.

JPEG feature extraction from inter-blocks: Scan all of the 8×8 blocks by rows and columns to arrange inter-block matrix in each of the pre-mentioned 27 AC coefficients' positions. Then we will get 27 new rearranged inter-block matrices $\mathbf{M}_{(w,h)}, w < h$, $w^2+h^2 \neq 0$, $w,h = 0,1,\ldots,7$. The inter-block matrix in position (0, 1) is shown in Fig. 5. Similarly, the proposed transition probability matrix $f(\mathbf{M}_{(w,h)})$ could be calculated, and the inter-block average

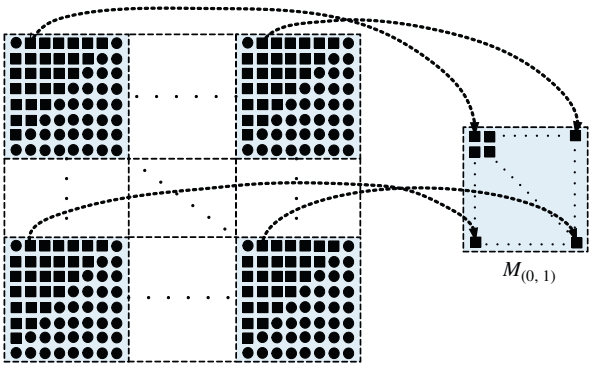

Fig. 5. JPEG DCT coefficient matrix and new rearranged inter-block matrix in (0, 1).

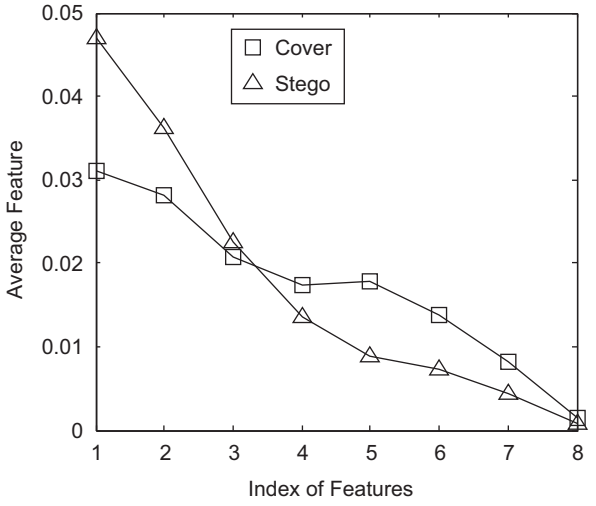

Fig. 6. Average values of the first line of H_{inter} before and after embedding using F5 steganography for 1000 images.

transition probability matrix could be computed by $H_{inter}=(1/27)\sum f(\mathbf{M}_{(w,h)})$, where H_{inter} is a 15×8 matrix.

In order to observe the change of the proposed feature before and after embedding, we embed 4000 bits of messages into 1000 JPEG images, which are converted from TIFF images in the NRCS Photo Gallery [34]. Then, we calculate the inter-block matrix of each image to get $H^j_{inter} j=1,2,\ldots 1000$, and the average value of the proposed features in the first line of the inter-block matrix would be denoted as $f_i=1/1000\sum_{j=1}^{1000} H^j_{inter}(1,i)$. The mean values of features are shown in Fig. 6 which is indexed from 1 to 8. It is observed that f_1 and f_2 increase after embedding, but the values from f_4 to f_7 decrease individually. It is because that the embedding weakens the correlation of the AC coefficients and neighboring coefficients on inter-blocks. The probability of the absolute different value appearing frequently would decrease, and the probability of the absolute different value appearing rarely would decrease. In other words, the probability of the proposed matrix on inter-block will transfer from the back to the front after embedding using F5. Extensive experiments convince us that the same conclusions does exist for other steganographic algorithms and intra-blocks. The obvious change of the proposed features illuminates that they are suitable to be used for classifying the stego image from cover image.

It seems that f_8 is almost the same before and after embedding. In fact, f_8 represents that the absolute difference value 0 appears 8 times, that is, the central coefficient equals each of the neighboring 8 coefficients. This situation rarely occurs relative to the high appearing times such as f_1 and f_2. On the other hand, f_8 is very sensitive to the embedding. As long as one of the 8 neighboring coefficients changes, the value of f_8 would decrease immediately. As Fig. 7 shows, points representing the cover image feature $H_{inter}(0,8)$ scatter at the higher part, but they decrease nearly to zero after embedding which is denoted by the squares. In addition, it is also observed that part of the points overlap with the squares, which will make trouble for future classifications. In the next subsection, the feature calibration would be used to solve the problem.

3.2.3. Feature calibration

To improve the performance of the proposed features, we fused the image cropped calibration and prediction error calibration as macroscopic and microscopic calibrations. All features in the proposed method were obtained from the quantized DCT coefficients of the test image and its corresponding macroscopic calibrated image, or from the test image and its microscopic calibrated image, respectively. Next we show how to calibrate the extracted features.

Let J denote the quantized DCT coefficient matrix of the test image I and J_c denote the cropped version of J. Calculate the proposed intra-block feature $H_{intra}(J)$, $H_{intra}(J_c)$ and inter-block feature $H_{inter}(J)$, $H_{inter}(J_c)$, respectively. Let

$$H_c=(|H_{intra}(J)-H_{intra}(J_c)|,\quad |H_{inter}(J)-H_{inter}(J_c)|) \tag{22}$$

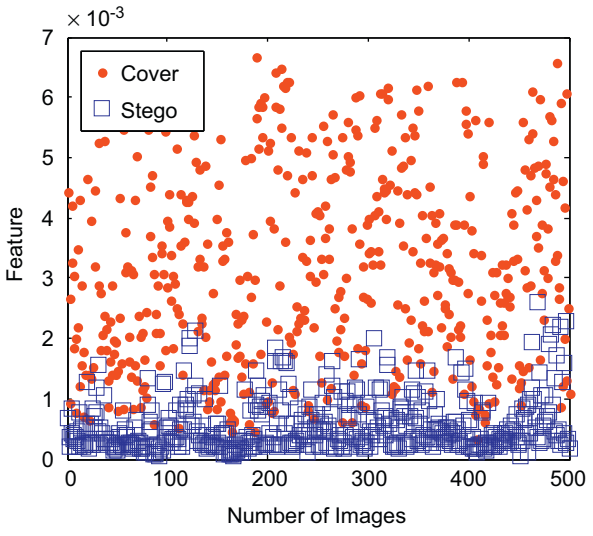

Fig. 7. Scatter plots of the $H_{inter}(0,8)$ for the cover and F5 stego images.

Additionally, we also extract the elements in the first 3 rows of H_c to form the image cropped calibration feature $\overline{H}_c$.

In Fig. 8, we show an example of how the proposed feature $H_c(0,8)$ changes for the cover and stego images. Note that it is different from Fig. 7 such that the squares, which represent stego feature, occupy the higher part, and the points are clustered on the lower part. The reason is that image cropped calibration would recover the statistic of the cover. When the test image is cover image, the difference between $H_{inter}(J)$ and $H_{inter}(J_c)$ is minor; but when the test image is stego image, the difference is large. So absolute difference value between $H_{inter}(J)$ and $H_{inter}(J_c)$ for stego image and calibrated version is larger than the cover image and its calibrated version.

Similarly, we could use the prediction error technique to calibrate features. After the predicted image $\hat{I}$ is attained followed by Eq. (12), recompress it with the same quantization table as the test image. Let $\hat{J}$ denote the predicted version of J, and we could gain the JPEG prediction error coefficients $J_{pe}=J-\hat{J}$. Calculate the intra-block feature $H_{intra}(J_{pe})$ and inter-block feature $H_{inter}(J_{pe})$ by utilizing the proposed method, and combine them in $H_{pe}=(H_{intra}(J_{pe}),\ H_{inter}(J_{pe}))$ which has the dimension 30×16. To decrease the computational complexity, only the elements in the first 3 rows of H_{pe} will be extracted to form a 48-dimensional feature vector $\overline{H}_{pe}$.

In contrast to Fig. 7., the borderline between the cover and stego feature is more distinguishable in Fig. 9 And the points representing cover images are more clustered at the higher part. It is because that the prediction error calibration extends the statistical difference between the cover and stego, which also implies its effective performance for future classification.

Thus, we can extract the features according to the following steps:

(1) Calibrate image I by two different calibration methods, respectively, and then get the JPEG coefficient matrices $\hat{J}$ and J_c.

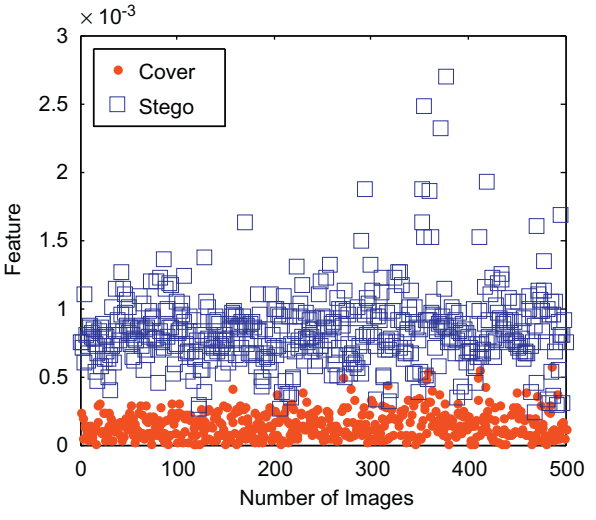

Fig. 8. Scatter plots of the $H_c(0,8)$ for the cover and F5 stego images.

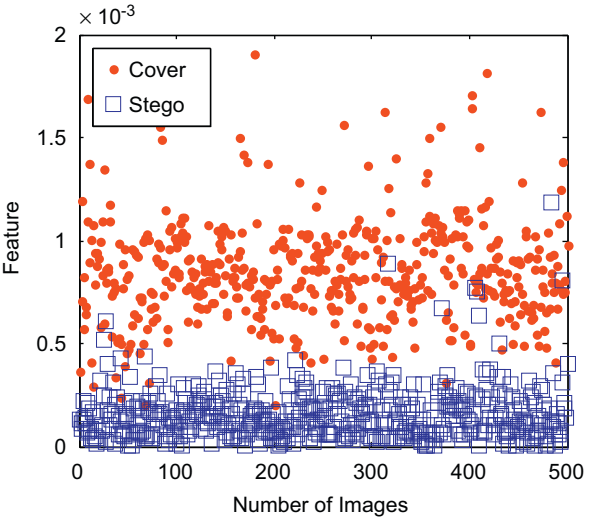

Fig. 9. Scatter plots of the $H_{pe}(0,8)$ for the cover and F5 stego images.

(2) Let $J_{pe}=J-\hat{J}$, and calculate the proposed matrices of J, J_c and J_{pe} on the intra-blocks and inter-blocks, respectively, which are denoted by

$$H_{intra}(J),\quad H_{intra}(J_c),\quad H_{intra}(J_{pe}) \tag{23}$$

and

$$H_{inter}(J),\quad H_{inter}(J_c),\quad H_{inter}(J_{pe}) \tag{24}$$

(3) Merge the features to form

$$H_{pe}=(H_{intra}(J_{pe}),H_{inter}(J_{pe})) \tag{25}$$

and

$$H_c=(H_{intra}(J)-H_{intra}(J_c),\quad H_{inter}(J)-H_{inter}(J_c)), \tag{26}$$

and extract the elements in the first 3 rows of the matrix to compose the 96 dimensional features.

3.3. The detection steps

After the features are computed, whether a test image is cover or stego becomes a two-classification problem. In this section, the SVM [27] will be trained to classify the cover from stego image. The detection method consists of five steps:

(1) Collect cover images and the corresponding stego images of certain steganography algorithm to build the training image set.

(2) Calculate the proposed features of the training set and scale them linearly to the interval $[-1, 1]$. Assume $max(f_i)$ and $min(f_i)$ as the maximal and minimal values of the ith feature, respectively. Denote x and x' as the ith feature before and after scaling, respectively. Scaling to $[-1, 1]$ means

$$x'=\frac{2(x-min(f_i))}{max(f_i)-min(f_i)}-1. \tag{27}$$

(3) Search the training optimal parameters on a multiplicative grid with the cross-validation.

(4) Map the prepared features into a higher (maybe infinite) dimensional space, then use SVM to find a linear separating hyperplane with the optimal parameters, which would classify the cover features from the stego with the maximal margin in the higher dimensional space.
(5) Calculate the test image's features, and send them to the classifier for determining whether it is the cover or the stego.

4. Performance evaluation

In this section, the proposed features and detection are evaluated, respectively. Firstly, the feature extraction methods are compared with the Markov approach. Furthermore, a series of two-class steganalyzers are built to identify the stego from cover images. We compare the proposed approach with other three blind JPEG steganalysis schemes, that is, Shi's [24], Fu's [30] and Fridrich's [26] schemes, in which Fridrich's scheme is regarded as the best effective blind JPEG steganalysis.

4.1. Feature evaluation

The main aim of this study is to develop a classifier for cover images and stego images, which owe to the feature extraction mostly. A good feature for steganalysis should detect the presence of secret messages accurately, and is effective to a large set of steganography methods and various images. In order to measure up to the above capability of features we resort to analysis of variance (ANOVA) [35], which mainly investigates the contribution of different factor variance for final variance.

Suppose the sample space is $A=\{A_1,A_2,\ldots,A_k\}$, and feature of A_i is $F_i=\{f_{i,1},f_{i,2},f_{i,3}\ldots,f_{i,n_i}\}$, where n_i is the number of the A_i. Let

$$S_{inter}=\sum_{i=1}^{k}\frac{n_i(\overline{f}_i-\overline{f})^2}{k-1} \quad S_{intra}=\sum_{i=1}^{k}\sum_{m=1}^{n_i}\frac{(f_{i,m}-\overline{f}_i)^2}{N-k} \tag{28}$$

where

$$\overline{f}_i=\frac{1}{n_i}\sum_{m=1}^{n_i}f_{i,m}, \quad \overline{f}=\frac{1}{N}\sum_{i=1}^{k}\sum_{m=1}^{n_i}f_{im} \text{ and } N=\sum_{i=1}^{k}n_i$$

It is observed that $S_{in\ ter}$ is the square sum of the factors representing the variation of the feature on inter-team and S_{intra} is the square sum of the error representing the variation of the feature on intra-team.

Then, the F-scores could be given by

$$F=S_{inter}/S_{intra} \tag{29}$$

which describes the degree of the feature spreading on inter-team and converging on intra-team. The larger F-scores is, the better the feature could distinguish $A_1,A_2,\ldots,A_k$.

In our case, $A_1,A_2,\ldots,A_k$ represent k different kinds of image set, consisting of cover image set and stego image sets from $k-1$ kinds of steganalytic methods. S_{inter} describes the feature variation between the k image sets, and corresponding S_{intra} represents the feature variation in a certain image set. The F-scores would be regarded as the capability index of the feature classifying the k image sets.

Figs. 10 and 11 show the performances of the proposed features and Markov features tested on five image sets, which are the cover image set and stego image sets from F5, OutGuess, JpHide and StegHide. It is observed that the F-scores of the proposed features are larger than the Markov features in either of the figures. In other words, the proposed features perform better than the Markov features for classifying the five image sets, which are consistent with our theoretical analysis in Section 3.2.1.

Moreover, we also construct 8 feature sets to evaluate their capability for classifying the cover and the stego:

#S1 Shi's 81-D feature set without calibrations. (Average Shi's 324-D feature set to 81 features.)
#S2 Shi's 81-D feature set with predicted error calibration.
#S3 Shi's 81-D feature set with image cropped calibration.
#S4 Shi's 162-D feature set with predicted error calibration and image cropped calibration.
#P1 proposed 48-D feature set without calibrations.
#P2 proposed 48-D feature set with predicted error calibration.
#P3 proposed 48-D feature set with image cropped calibration and
#P4 proposed 96-D feature set with predicted error calibration and image cropped calibration.

The image base contains 1000 TIFF images from the NRCS [34]. Images are cropped into the size of 768×512 or 512×768 from the center and converted into grayscales images. After they have been compressed with the JPEG quality factor 75, random bit-streams of different relative lengths were embedded into the images by using JpHide(ver. 0.5). The lengths of the embedded messages were 10%, 25%, 50%, 75% and 100% of the maximal image embedding capacity, respectively. It should be pointed out

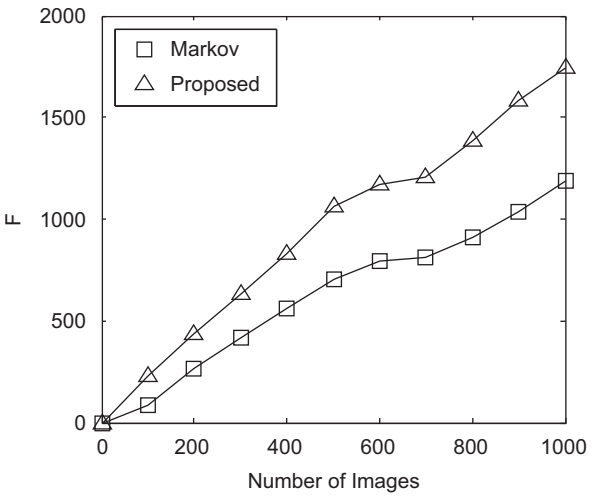

Fig. 10. The comparion of F-scores for the proposed features and Markov features of the same embedding capacity.

that, as JpHide(ver. 0.5) [11] compresses the message before embedding, we just take the image size as the maximal JpHide embedding capacity. In our experiments, an SVM with the radial basis function kernel is chosen to build the two-class steganalyzer, and the optimal parameters are achieved with the tool of LibSVM [27]. The training sets consist of 500 cover images and 500 stego images for each feature set and each embedding ratio. The rest stego images are used for testing different feature sets.

Table 1 shows the detection accuracy represented by the probability such that the steganalyzers distinguish the stego images from cover images successfully. From Table 1, we would see that feature set #P1 (the proposed features without calibrations) performs better than Shi's averaged feature set #S1. Moreover, either of the two calibrations (image cropped calibration and prediction error calibration) does improve the performance of the detector. Especially, the image cropped calibration acts more effectively than the prediction error calibration. It should also be noticed that the combination of the two calibrations performs best when detecting JpHide(ver. 0.5).

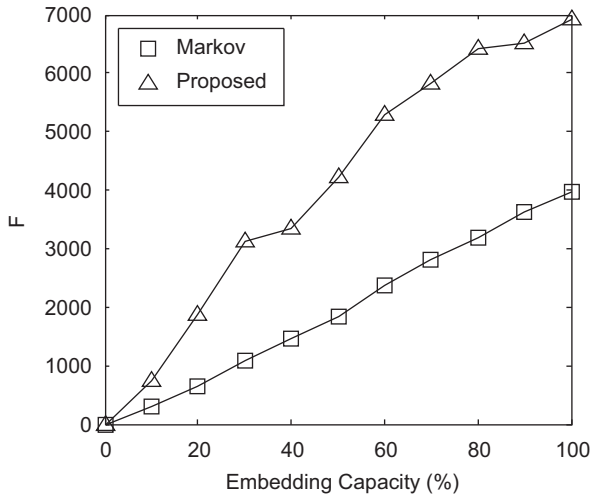

Fig. 11. The comparion of *F*-scores for the proposed features and Markov features of different embedding capacity.

4.2. Classifier evaluation

The same cover set is set up as Section 4.1, and secret messages are embedded by using the following four JPEG steganographic algorithms: F5 [9], OutGuess(ver. 0.2) [10], JpHide(ver. 0.5) [11] and StegHide [12]. It should be noticed that F5 and OutGuess would decompress the JPEG cover into the spatial domain and recompress it before embedding. That means that the corresponding stego image endures double-compress [36], which would make a major impact on the future classification. For simplicity, we adopt F5 and OutGuess to output the stego with the quality factor 75, which ensure that the stego images are single-compressed. The lengths of the embedded messages are 10%, 25%, 50%, 75% and 100% of the maximal image embedding capacity. The training set consists of 500 cover images and 500 stego images for each algorithm and each embedding ratio. Therefore, $4 \times 5 = 20$ classifiers (there were 5 message lengths for 4 algorithms) should be trained in our experiments. The total number of images for training is $20 \times 1000 = 20{,}000$. The rest of the images are used to test the proposed method on a computer equipped with Inter Pentium Dual core 2.5 GHz/2 G. Figs. 12–15 show the detection accuracy of the 20 classifiers.

As shown in Figs. 12–15, the proposed scheme performs best on OutGuess(ver. 0.2). That is because OutGuess(ver. 0.2) only keeps the global histogram of the DCT coefficients while the proposed features describe correlations on intra-block and inter-block. However, as JpHide(ver. 0.5) compresses the message before embedding, the proposed scheme performs commonly on JpHide(ver.0.5). In the meanwhile, the new method significantly outperforms Shi's and Fu's schemes when detecting F5, JpHide and OutGuess. As well as, it outperforms Fridrich's scheme in detecting JpHide, but performs similarly in detecting F5, OutGuess and StegHide. Nevertheless, the proposed scheme could classify 60 images per minute, which are about 2.6 times as many as Fridrich's.

Moreover, a multi-class steganalyzer is also set up to recognize the steganographic algorithms.

Totally 2000 images (500 images for each algorithm) were chosen as training set, the rest were chosen as the testing set, and then the classifier was trained to build multi-class steganalyzer. The probability of successful identification targeting on steganographic algorithms is obtained through a series of experiments.

Table 1
Comparisons of detection accuracy of eight different feature sets against JpHide.

Feature set	The detection accuracy against JpHide steganography in different embedding ratio				
	10%	25%	50%	75%	100%
#S1 (%)	82.3	84	83.5	85.1	89.6
#P1 (%)	85.3	89.6	91.2	94.6	96.1
#S2 (%)	90.6	91.1	92.8	93.3	94.0
#P2 (%)	88.2	93.6	95.6	95.4	96.8
#S3 (%)	93.1	94.8	94.2	94.8	95.5
#P3 (%)	91.6	94.7	95.3	97.5	98
#S4 (%)	94.4	96.3	97	96.2	97.5
#P4 (%)	95.8	96.8	97.8	98.4	99

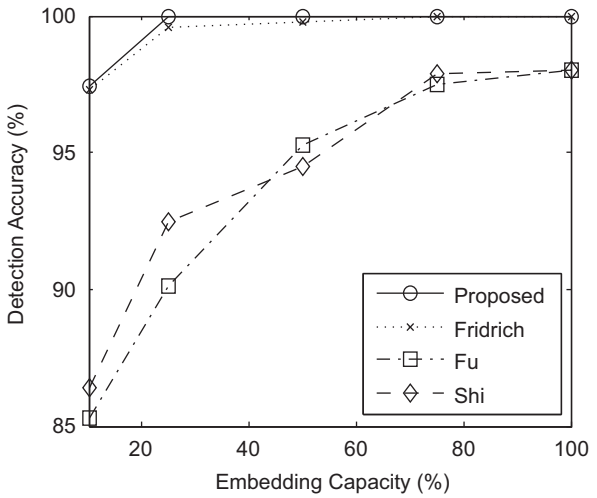

Fig. 12. Comparison of detection accuracy of four different detection schemes against F5.

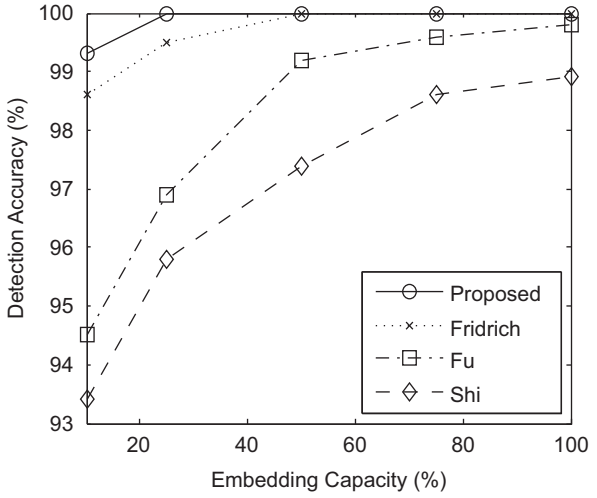

Fig. 13. Comparison of detection accuracy of four different detection schemes against OutGuess.

As Table 2 shows, the proposed scheme performs well on F5, JpHide and StegHide, but could not identify OutGuess from others in respect that the OutGuess stego images were often classified as F5.

5. Conclusions

In this paper, a new multi-directional transition probability matrix is constructed inspired by the Markov approach. Through theoretically analysis, we demonstrate that the Markov transition probability would be considered as a kind of partial probability of the proposed transition probability. The experiments of ANOVA also show the superiority of the proposed feature over the Markov feature. By merging the microscopic and macroscopic calibrations,

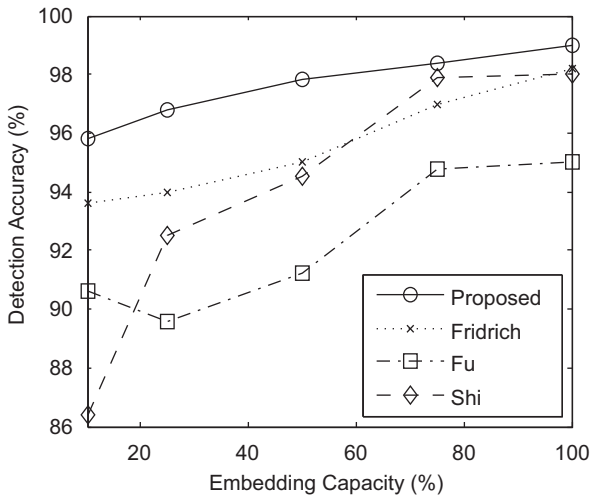

Fig. 14. Comparison of detection accuracy of four different detection schemes against JpHide.

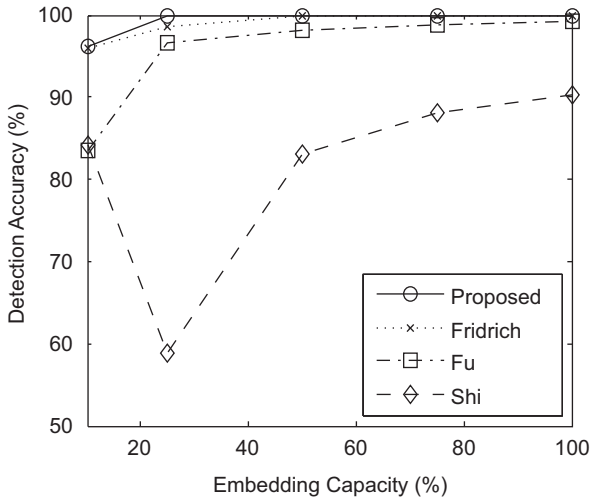

Fig. 15. Comparison of detection accuracy of four different detection schemes against StegHide.

Table 2
Identification of four steganography schemes.

Steganograpy schemes	Embedding ratio				
	10%	25%	50%	75%	100%
F5 (%)	79.8	91	94.4	94.6	92.6
OutGuess (%)	59.6	77.8	85	86.4	91.8
JpHide (%)	87.8	93.6	95.6	96.4	93.8
StegHide (%)	69.6	92.6	96.6	98.2	100

96-dimensional features were extracted. Based on these features, a two-class steganalyzer scheme is set up to reliably detect the stego images embedded by four popular JPEG steganographic algorithms, and a multi-class steganalyzer scheme is also built to distinguish the four steganographic

algorithms. Experimental results demonstrate that our classifier is more reliable and efficient than the best effective blind steganalyzer targeted on JPEG images. The benefits depend on the following properties: (1) based on the multi-directional correlations of the quantized DCT coefficients, more efficient features could be extracted, (2) different kinds of calibrations could be combined to improve the blind steganalysis's performances and (3) merging the intra-block and inter-block correlations of the quantized DCT coefficients is more effective. Although our approach is able to classify the stego image from the cover image reliably, its performance for identifying different kinds of steganography is not satisfactory. To make our system more practical, some means would be taken in future work: (1) combining different kinds of sensitive feature set, such as DCT feature set [29], Wavelet feature set [25], etc., and (2) improving the performance by using appropriate supervised learning technique such as feature selection or classifier fusion, etc.

Acknowledgments

This work was supported by the Natural Science Foundation of China under the Grant no. 60803155, and partially supported by France Telecom's research project AMCS/CRYPTO.

References

[1] S. Lian, D. Kanellopoulos, G. Ruffo, Recent advances in multimedia information system security, Informatica 33 (1) (2009) 3–24.
[2] S. Lian, Y. Zhang (Eds.), Handbook of Research on Secure Multimedia Distribution, IGI Global (formerly Idea Group, Inc), 2009.
[3] S. Lian (Ed.), Multimedia Communication Security: Recent Advances, Nova Publishers, 2009.
[4] J. Hernandez-Castro, L. Blasco-Lopez., Steganography in games: a general methodology and its application to the game of Go, Computers and Security 25 (1) (2006) 64–71.
[5] A. Yılmaz, A. Alatan, Error detection and concealment for video transmission using information hiding, Signal Processing: Image Communication 23 (4) (2008) 298–312.
[6] M. Fu, O. Au, Halftone image data hiding with intensity selection and connection selection, Signal Processing: Image Communication 16 (10) (2001) 909–930.
[7] F.A.P. Petitcolas, R.J. Anderson, M.G. Kuhn, Information hiding—a survey, in: Proceedings of the IEEE, special issue on protection of multimedia content, vol. 87, no. 7, 1999, pp. 1062–1078.
[8] T. Sharp, An implementation of key-based digital signal steganography, in: Proceedings of the 4th International Workshop on Information Hiding, volume 2137 of Springer LNCS, 2001, pp. 13–26.
[9] A. Westfeld, F5 a Steganographic Algorithm High Capacity Despite Better Steganalysis, Information Hiding, in: Proceedings of the 4th International Workshop, Springer, Berlin, 2001, pp. 289–302.
[10] N. Provos, Defending against statistical steganalysis, in: Proceedings of 10th USENIX Security Symposium, Washington DC, 2001, pp. 24–36.
[11] J. Provos, M. Goljan, R. Du, Detecting LSB Steganography in color and gray-scale images, Multimedia IEEE 8 (4) (2001) 22–28.
[12] S. Hetzl, Steghide, 2003, 〈http://steghide.sourceforge.net/〉, accessed on December 28, 2009.
[13] K. Solanki, A. Sarkar, B.S. Manjunath, YASS: Yet another steganographic scheme that resists blind steganalysis, in: Proceedings of 9th International Workshop on Information Hiding, Springer-Verlag, LNCS 4567, 2008, pp. 16–31.
[14] W. Zhang, X. Zhang, S. Wang, Maximizing steganographic embedding efficiency by combining hamming codes and wet paper codes, in: Proceedings of the 10th Information Hiding, LNCS 5284, Springer-Verlag, 2008, pp. 60–71.
[15] J. Fridrich, Asymptotic behavior of the ZZW embedding construction, IEEE Transactions on Information Forensics and Security 4 (1) (2009) 151–154.
[16] W. Zhang, X. Zhang, S. Wang, Near-optimal codes for information embedding in gray-scale signals, IEEE Transactions on Information Theory 55 (3) (2010)preprinted 55 (3) .
[17] W. Zhang, X. Zhu, Improving the embedding efficiency of wet paper codes by paper folding, IEEE Signal Processing Letters 16 (2) (2009) 794–7972009 16 (2009) 794–797.
[18] J. Fridrich, M. Goljan, Practical steganalysis of digital images—state of the art, in: Proceedings of the SPIE, Security and Watermarking of Multimedia Contents IV, vol. 4675, 2002, pp. 1–13.
[19] J. Fridrich, M. Goljan, R. Du, Reliable detection of LSB steganography in color and grayscale images, in: Proceedings of the ACM Workshop on Multimedia Security, 2001, pp. 27–30.
[20] M. Goljan, J. Fridrich, T. Holotyak, New blind steganalysis and its implications, in Proceedings of the SPIE 6072, 2006, pp. 1–13.
[21] A.D. Ker, I. Lubenko, Feature reduction and payload location with WAM steganalysis, in: Proceedings of the SPIE 7254, 2009, pp. 0A01–0A13.
[22] B. Li, J. Huang, Y. Shi, Steganalysis of YASS, IEEE Transactions on Information Forensics and Security 4 (3) (2009) 369–382.
[23] J. Zhang, D. Zhang, Detection of LSB matching steganography in decompressed images, IEEE Signal Processing Letters 17 (2) (2010) 141–144.
[24] Y. Shi, C. Chen, W. Chen, A Markov process based approach to effective attacking JPEG steganography, Information Hiding, in: Proceedings of the 8th International Workshop, Springer, Berlin, 2006, pp. 249–264.
[25] Y. Wang, P. Moulin, Optimized feature extraction for learning-based image steganalysis, IEEE Transactions on Information Forensics and Security 2 (1) (2007) 31–45.
[26] T. Pevny, J. Fridrich, Merging Markov and DCT features for multi-class JPEG steganalysis, in: Proceedings of SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, San Jose, CA, 2007, pp. 650503-1–650503-13.
[27] C. Chang, C. Lin, LIBSVM: a library for support vector machines, 〈http://www.csie.ntu.edu.tw/~cjlin/libsvm〉, accessed on December 28, 2009.
[28] W.N. Lie, G.S. Lin, A feature-based classification technique for blind image steganalysis, IEEE Transactions on Multimedia 7 (6) (2005) 1007–1020.
[29] J. Fridrich, Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes, Information Hiding, in: Proceedings of the 6th International Workshop, Springer, Berlin, 2004, pp. 67–81.
[30] D.D. Fu, Y.Q. Shi, D.K. Zou, G.R. Xuan, JPEG steganalysis using empirical transition matrix in block DCT domain, Multimedia Signal Processing, in: Proceedings of the 8th International Workshop, Victoria, BC, Canada, 2006, pp. 310–313.
[31] J. Fridrich, M. Goljan, D. Hogea, Steganalysis of JPEG Image: breaking the F5 Algorithm. In: Proceedings of the 5th International Workshop on Information Hiding, Springer-Verlag, LNCS 2578, 2002, pp. 310–323.
[32] F. Huang, J. Huang, in: Calibration Based Universal JPEG Steganalysis, Science in China Series F, Information Sciences Science in China Press, 2009, pp. 260–268.
[33] M. Weinberger, G. Seroussi, G. Sapiro, A low complexity context-based lossless image compression algorithm, in: Proceedings of the IEEE Data Compression Conference, 1996, pp. 140–149.
[34] United States Department of Agriculture, NRCS Photo Gallery, 〈http://photogallery.nrcs.usda.gov/〉, accessed on December 28, 2009.
[35] A.C. Rencher, in: Methods of Multivariate Analysis, John Wiley, New York, 2002.
[36] T. Pevny, J. Fridrich, Detection of double-compression in JPEG images for applications in steganography, IEEE Transactions on Information Forensics and Security 3 (2) (2008) 247–258.