Information Sciences 180 (2010) 1643–1655

Contents lists available at ScienceDirect

Information Sciences

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

An improved approach to steganalysis of JPEG images

Qingzhong Liu [a,b], Andrew H. Sung [a,b,*], Mengyu Qiao [a], Zhongxue Chen [c], Bernardete Ribeiro [d]

[a] *Department of Computer Science, New Mexico Tech, Socorro, NM 87801, USA*
[b] *Institute for Complex Additive Systems Analysis, New Mexico Tech, Socorro, NM 87801, USA*
[c] *Department of Epidemiology and Biostatistics, Robert Stempel College of Public Health and Social Work, Florida International University, Miami, FL 33199, USA*
[d] *Department of Informatics Engineering, University of Coimbra, 3030-290 Coimbra, Portugal*

ARTICLE INFO

Article history:
Received 1 December 2008
Received in revised form 9 December 2009
Accepted 4 January 2010

Keywords:
Steganography
Steganalysis
Markov
Polynomial fitting
JPEG
SVM-RFE
MSVM-RFE

ABSTRACT

Steganography secretly embeds additional information in digital products, the potential for covert dissemination of malicious software, mobile code, or information is great. To combat the threat posed by steganography, steganalysis aims at the exposure of the stealthy communication. In this paper, a new scheme is proposed for steganalysis of JPEG images, which, being the most common image format, is believed to be widely used for steganography purposes as there are many free or commercial tools for producing steganography using JPEG covers.

First, a recently proposed Markov approach [27] is expanded to the inter-block of the discrete cosine transform (DCT) and to the discrete wavelet transform (DWT). The features on the joint distributions of the transform coefficients and the features on the polynomial fitting errors of the histogram of the DCT coefficients are also extracted. All features are called original ExPanded Features (EPF). Next, the EPF features are extracted from the calibrated version; these are called reference EPF features. The difference between the original and the reference EPF features is calculated, and then the original EPF features and the difference are merged to form the feature vector for classification.

To handle the large number of developed features, the feature selection method of support vector machine recursive feature elimination (SVM-RFE) and a method of multi-class support vector machine recursive feature elimination (MSVM-RFE) are used to select features for binary classification and multi-class classification, respectively. Finally, support vector machines are applied to the selected features for detecting stego-images.

Experimental results show that, in comparison to the Markov approach [27], this new scheme remarkably improves the detection performance on several JPEG-based steganographic systems, including JPHS, CryptoBola, F5, Steghide, and Model based steganography.

1. Introduction

Steganography is the art and science of hiding data in digital images, audios, videos, and other digital files. The innocent digital media or files are called carriers or covers. The digital media or files, after being embedded with the secret or hidden files, are called steganograms. There is no perceptible difference between the cover and the steganogram. In image steganography, the common techniques implement information hiding by modifying pixel values such as LSB

* Corresponding author. Address: Department of Computer Science, New Mexico Tech, Socorro, NM 87801, USA. Tel.: +1 575 835 5949.
E-mail addresses: liu@cs.nmt.edu (Q. Liu), sung@cs.nmt.edu (A.H. Sung), myuqiao@cs.nmt.edu (M. Qiao), Zhongxue.Chen@fiu.edu (Z. Chen), bribeiro@dei.uc.pt (B. Ribeiro).

0020-0255/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2010.01.001

matching [26] or modifying transform coefficients such as reversible hiding [2,3]. Other information hiding techniques include spread spectrum steganography [22], statistical steganography, distortion, and cover generation steganography [12]. The relevant issue of evaluating how much information can be hidden within an image has been studied by Zhang et al. [30].

Due to the great variety of steganographic tools, the unknown extent that they may be used, and the lack of effective (highly accurate and efficient) detection techniques, the threat posed by steganography is a serious information security concern. Aiming to discover the presence of hidden data in digital media by accurately discriminating steganograms from covers, steganalysis, normally involves feature mining and pattern analysis. To date, a few steganographic methods such as LSB embedding/matching and spread spectrum steganography [5,6,8,9,13,15,16,18–21] have been successfully steganalyzed.

JPEG images are widely used in steganography for carrying hidden data as there are at least several JPEG steganographic methods/tools freely available on the Internet. To detect JPEG steganography, Fridrich et al. presented a method to estimate the cover histogram from the stego-image [6]. Fridrich also extracted features based on an L1 norm of the difference between a specific macroscopic functional calculated from the stego-image and the same functional obtained from a decompressed, cropped, and recompressed stego-image [5]. Harmsen and Pearlman implemented a detection scheme using only the indices of the quantized DCT coefficients in JPEG images [9]. Recently, Shi et al. proposed a Markov approach to effectively detect JPEG stego-images [27]. By applying calibration to Markov features, Pevny and Fridrich merged the DCT features and calibrated Markov features to improve the steganalysis performance in JPEG images [23]. Li et al. successfully detected YASS by using statistical features extracted from the blocks that are probably modified [14]. Based on the Markov approach presented in Ref. [27], Liu et al. expanded the approach for steganalysis of WAV audio streams using the features of the neighboring joint density and transition probability on the second order derivative [17].

Among these works, the Markov approach in Ref. [27] is remarkable due to the promising steganalysis performance on several JPEG-based steganographic systems. In this paper, the Markov approach is expanded to inter-blocks of the DCT domain and to the wavelet domain, additional features on the joint distributions in the DCT domain and the wavelet domain are also designed as well as the features of a polynomial fitting on the histogram of the DCT coefficients. All these features including expanded Markov transition features are called ExPanded Features (EPF). After that, the EPF features are extracted from the calibrated version of the image, called reference EPF features, the difference between the original EPF features and the reference EPF features is calculated, and the original EPF features and the difference features are merged together. Then the feature selection methods of support vector machine recursive feature elimination (SVM-RFE) [7] and multi-class support vector machine recursive feature elimination (MSVM-RFE) [31] are utilized to select features for binary classification and multi-class classification, respectively. Finally, support vector machines [4,28] are applied to the selected features for detecting steganograms. Experimental results show that this new approach successfully improves the steganalysis performance on several JPEG-based steganographic systems, including JPWIN [32], F5 [29], CryptoBola [33], Steghide [10] and Model based steganography [24].

In the following, Section 2 expands the Markov features; Section 3 describes the features of the neighboring joint density in DCT and DWT domains; Section 4 presents the features of the errors of polynomial fitting in the DCT domain; Section 5 explains the feature extraction from the calibrated version of the images, the combination of all features, and the feature selection using SVM-RFE and MSVM-RFE; Section 6 presents experimental results; Section 7 gives conclusions.

2. Expanding markov approach

2.1. Modified markov approach

To detect the existence of hidden data in JPEG images, the approach presented in Ref. [27], which models the differences between absolute values of neighboring DCT coefficients as a Markov process, is successful for steganalysis of several JPEG-based steganographic systems [27]. Let F denote the matrix of absolute values of DCT coefficients of the image. The DCT coefficients in F are arranged in the same way as pixels in the image by replacing each 8×8 block of pixels with the corresponding block of DCT coefficients. Four difference arrays are calculated along four directions: horizontal, vertical, diagonal, and minor diagonal, denoted $F_h(u,v)$, $F_v(u,v)$, $F_d(u,v)$, and $F_m(u,v)$, respectively.

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

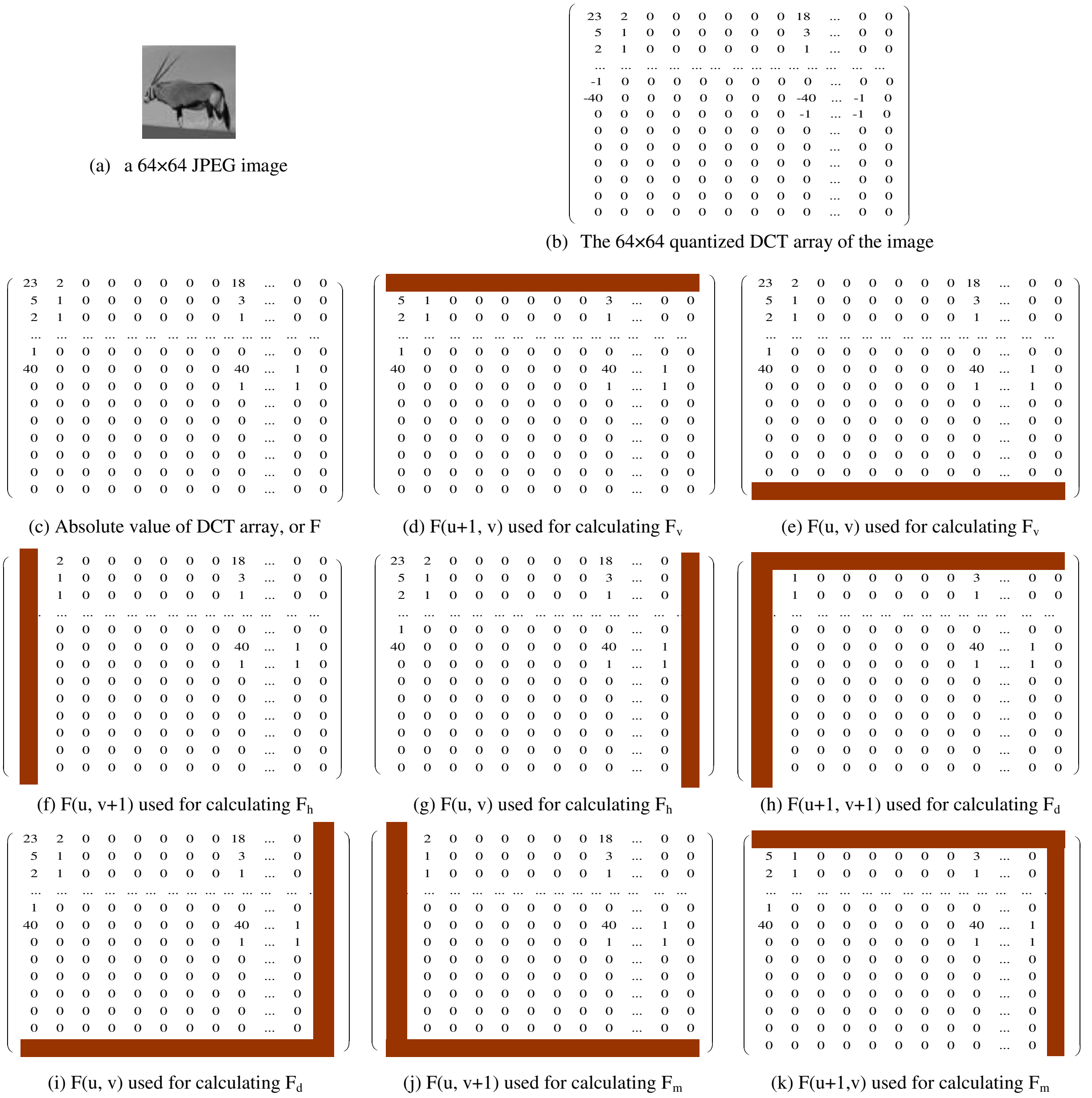
$$F_v(u,v) = F(u,v) - F(u+1,v) \tag{2}$$

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

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

The transition probability matrices are constructed according to the above four difference arrays. Fig. 1 shows an example of the calculations of F_h, F_v, F_d and F_m.

In this paper, only the transition probability matrices along the horizontal and vertical directions are constructed. The four transition probability matrices $M1_{hh}$, $M1_{hv}$, $M1_{vh}$, and $M1_{vv}$, based on $F_h(u,v)$ and $F_v(u,v)$, are calculated as follows:



(a) a 64×64 JPEG image

$$\begin{pmatrix}
23 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 & 0 \\
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
-40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -40 & \cdots & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & \cdots & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(b) The 64×64 quantized DCT array of the image

$$\begin{pmatrix}
23 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 & 0 \\
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(c) Absolute value of DCT array, or F

$$\begin{pmatrix}
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(d) F(u+1, v) used for calculating F$_v$

$$\begin{pmatrix}
23 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 & 0 \\
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(e) F(u, v) used for calculating F$_v$

$$\begin{pmatrix}
2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(f) F(u, v+1) used for calculating F$_h$

$$\begin{pmatrix}
23 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 \\
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0
\end{pmatrix}$$

(g) F(u, v) used for calculating F$_h$

$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(h) F(u+1, v+1) used for calculating F$_d$

$$\begin{pmatrix}
23 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 \\
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0
\end{pmatrix}$$

(i) F(u, v) used for calculating F$_d$

$$\begin{pmatrix}
2 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & \cdots & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}$$

(j) F(u, v+1) used for calculating F$_m$

$$\begin{pmatrix}
5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & \cdots & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
40 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 40 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0
\end{pmatrix}$$

(k) F(u+1,v) used for calculating F$_m$

Fig. 1. An example to show the calculation of Eqs. (1)–(4). The shaded rows or columns of the absolute DCT arrays in (d) to (k) are removed to calculate F_v, F_h, F_d and F_m, respectively.

$$M1_{hh}(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(F_h(u,v) = i, F_h(u+1,v) = j)}{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(F_h(u,v) = i)} \tag{5}$$

$$M1_{hv}(i,j) = \frac{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_h(u,v) = i, F_h(u,v+1) = j)}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_h(u,v) = i)} \tag{6}$$

$$M1_{vh}(i,j) = \frac{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_v(u,v) = i, F_v(u+1,v) = j)}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_v(u,v) = i)} \tag{7}$$

$$M1_{vv}(i,j) = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(F_v(u,v) = i, F_v(u,v+1) = j)}{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(F_v(u,v) = i)} \tag{8}$$

Where S_u and S_v denote the dimensions of the image and $\delta = 1$ if and only if its arguments are satisfied. Due to the differences between absolute values of neighboring DCT coefficients which could be quite large, following the original approach [27], the ranges of i and j are limited to $[-4,+4]$. Thus, all Markov features consist of 4 × 81 = 324 features.

2.2. Expanding Markov transition features

The Markov approach in Ref. [27] utilizes the correlation of neighboring DCT coefficients in the intra-DCT-block. The neighboring DCT coefficients on the inter-block have similar correlation that can be derived from the previous work [5,23]; and thus the Markov approach is expanded to the neighboring DCT coefficients on the inter-blocks. The horizontal and vertical difference arrays on the inter-block are obtained as follows:

$$D_h(u,v) = F(u,v) - F(u+8,v) \tag{9}$$

$$D_v(u,v) = F(u,v) - F(u,v+8) \tag{10}$$

The four transition probability matrices $M2_{hh}$, $M2_{hv}$, $M2_{vh}$, and $M2_{vv}$ are calculated as follows where the ranges of i and j are $[-4,+4]$.

$$M2_{hh}(i,j) = \frac{\sum_{u=1}^{S_u-16}\sum_{v=1}^{S_v}\delta(D_h(u,v)=i, D_h(u+8,v)=j)}{\sum_{u=1}^{S_u-16}\sum_{v=1}^{S_v}\delta(D_h(u,v)=i)} \tag{11}$$

$$M2_{hv}(i,j) = \frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(D_h(u,v)=i, D_h(u,v+8)=j)}{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(D_h(u,v)=i)} \tag{12}$$

$$M2_{vh}(i,j) = \frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(D_v(u,v)=i, D_v(u+8,v)=j)}{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-16}\delta(D_v(u,v)=i)} \tag{13}$$

$$M2_{vv}(i,j) = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-16}\delta(D_v(u,v)=i, D_v(u,v+8)=j)}{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-16}\delta(D_v(u,v)=i)} \tag{14}$$

2.3. Markov features on the DWT approximation subband

Embedding data in JPEG images may happen to the low frequency DCT coefficients, in such case, exploring the modification of the low frequency DCT coefficients is necessary. It is generally difficult for us to directly mine the statistical characteristics of the low frequency DCT coefficients. However, the modification of these DCT coefficients will affect the DWT coefficients in low frequency, or DWT approximation subband. Let WA denote the Haar DWT approximation subband that is multiplied by 2. To obtain it, Haar wavelet transform is applied to uncompressed JPEG image, and the LL subband or approximation subband is multiplied by 2 to enable the minimal interval value to be 1. The horizontal and vertical difference arrays are then calculated as follows:

$$WA_h(u,v) = WA(u,v) - WA(u,v+1) \tag{15}$$

$$WA_v(u,v) = WA(u,v) - WA(u+1,v) \tag{16}$$

Fig. 2 shows an example of the calculations of WA_h and WA_v. The four transition probability matrices $M3_{hh}$, $M3_{hv}$, $M3_{vh}$, and $M3_{vv}$ are constructed as follows, where SW_u and SW_v denote the size of the WA.

$$M3_{hh}(i,j) = \frac{\sum_{u=1}^{SW_u-2}\sum_{v=1}^{SW_v}\delta(WA_h(u,v)=i, WA_h(u+1,v)=j)}{\sum_{u=1}^{SW_u-2}\sum_{v=1}^{SW_v}\delta(WA_h(u,v)=i)} \tag{17}$$

$$M3_{hv}(i,j) = \frac{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_h(u,v)=i, WA_h(u,v+1)=j)}{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_h(u,v)=i)} \tag{18}$$

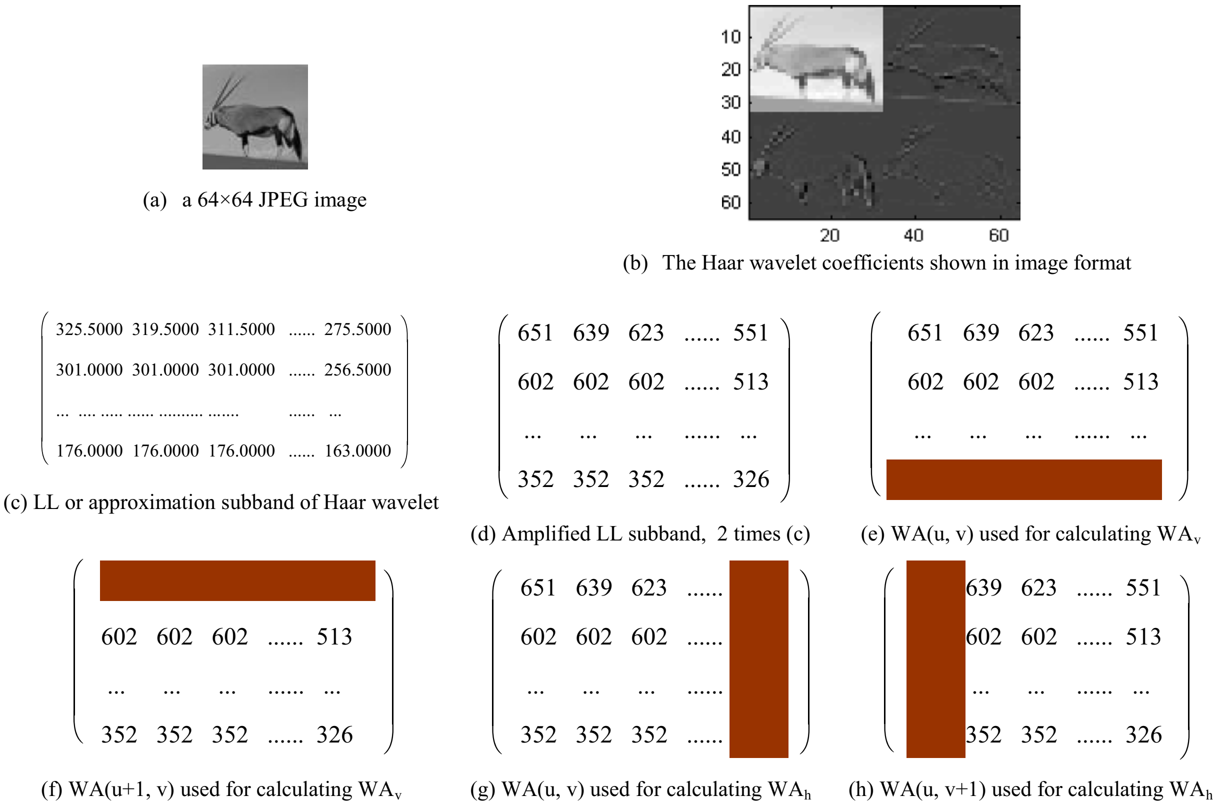
$$M3_{vh}(i,j) = \frac{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_v(u,v)=i, WA_v(u+1,v)=j)}{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_v(u,v)=i)} \tag{19}$$

$$M3_{vv}(i,j) = \frac{\sum_{u=1}^{SW_u}\sum_{v=1}^{SW_v-2}\delta(WA_v(u,v)=i, WA_v(u,v+1)=j)}{\sum_{u=1}^{SW_u}\sum_{v=1}^{SW_v-2}\delta(WA_v(u,v)=i)} \tag{20}$$

Similar to the original Markov approach, the ranges of i and j are $[-4,+4]$.

3. Joint density features

Our previous work in steganalysis [15,16,18–20] has demonstrated that there is high correlation between adjacent pixels, which is consistent with a well-known statistical model, generalized Gaussian distribution (GGD) [25]. GGD is given by



(a) a 64×64 JPEG image

(b) The Haar wavelet coefficients shown in image format

(c) LL or approximation subband of Haar wavelet

(d) Amplified LL subband, 2 times (c)

(e) WA(u, v) used for calculating WA$_v$

(f) WA(u+1, v) used for calculating WA$_v$

(g) WA(u, v) used for calculating WA$_h$

(h) WA(u, v+1) used for calculating WA$_h$

Fig. 2. An example to show the calculation of Eqs. (15) and (16). The shaded rows or columns of the amplified Haar approximation subband in (e) to (h) are removed to calculate WA_v and WA_h.

$$p(x;\alpha,\beta)=\frac{\beta}{2\alpha\Gamma(1/\beta)}e^{-(|x|/\alpha)^{\beta}} \tag{21}$$

where $\Gamma(\cdot)$ is the gamma function; x is the value of transform coefficients; the parameter α, referred to as the scale parameter, models the width of the probability distribution function (PDF) peak; and β is called the shape parameter, which is inversely proportional to the decreasing rate of the peak. The GGD model contains the Gaussian and Laplacian PDFs as special cases, using $\beta=2$ and $\beta=1$, respectively.

According to the GGD of the transform domain, in addition to the high correlation of adjacent pixels, there is also high correlation of the adjacent coefficients in the transform domains. In our point of view, it is the high correlation in DCT domain that makes the Markov features applicable in detecting the hidden information in JPEG images. With this in mind, the neighboring joint distribution matrices $J1$, $J2$ and $J3$ in the DCT and DWT domains, corresponding to the previous Markov transition matrices, are extracted by modifying the Eqs. (5)–(8), (11)–(14), (17)–(20), listed as follows.

$$J1_{hh}(i,j)=\frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(F_h(u,v)=i,F_h(u+1,v)=j)}{(S_u-2)S_v} \tag{22}$$

$$J1_{hv}(i,j)=\frac{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_h(u,v)=i,F_h(u,v+1)=j)}{(S_u-1)(S_v-1)} \tag{23}$$

$$J1_{vh}(i,j)=\frac{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v-1}\delta(F_v(u,v)=i,F_v(u+1,v)=j)}{(S_u-1)(S_v-1)} \tag{24}$$

$$J1_{vv}(i,j)=\frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(F_v(u,v)=i,F_v(u,v+1)=j)}{S_u(S_v-2)} \tag{25}$$

$$J2_{hh}(i,j)=\frac{\sum_{u=1}^{S_u-16}\sum_{v=1}^{S_v}\delta(D_h(u,v)=i,D_h(u+8,v)=j)}{(S_u-16)S_v} \tag{26}$$

$$J2_{hv}(i,j)=\frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(D_h(u,v)=i,D_h(u,v+8)=j)}{(S_u-8)(S_v-8)} \tag{27}$$

$$J2_{vh}(i,j) = \frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(D_v(u,v)=i, D_v(u+8,v)=j)}{(S_u-8)(S_v-8)} \tag{28}$$

$$J2_{vv}(i,j) = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-16}\delta(D_v(u,v)=i, D_v(u,v+8)=j)}{S_u(S_v-16)} \tag{29}$$

$$J3_{hh}(i,j) = \frac{\sum_{u=1}^{SW_u-2}\sum_{v=1}^{SW_v}\delta(WA_h(u,v)=i, WA_h(u+1,v)=j)}{(SW_u-2)SW_v} \tag{30}$$

$$J3_{hv}(i,j) = \frac{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_h(u,v)=i, WA_h(u,v+1)=j)}{(SW_u-1)(SW_v-1)} \tag{31}$$

$$J3_{vh}(i,j) = \frac{\sum_{u=1}^{SW_u-1}\sum_{v=1}^{SW_v-1}\delta(WA_v(u,v)=i, WA_v(u+1,v)=j)}{(SW_u-1)(SW_v-1)} \tag{32}$$

$$J3_{vv}(i,j) = \frac{\sum_{u=1}^{SW_u}\sum_{v=1}^{SW_v-2}\delta(WA_v(u,v)=i, WA_v(u,v+1)=j)}{SW_u(SW_v-2)} \tag{33}$$

4. Errors of polynomial fitting

Some JPEG-based steganographic methods modify the quantized DCT coefficients and affect the marginal density of the DCT coefficients; the distribution may deviate from the GGD in the DCT domain. For the quantized DCT coefficients in JPEG, the values of x in (21) are the discrete values, 0, 1, −1, 2, −2, 3, −3, etc. The marginal density of the quantized DCT coefficients, $h(x)$, can be approximated by (34).

$$h(x) = \frac{\beta}{2\alpha\Gamma(1/\beta)}\exp\{-(|x|/\alpha)^{\beta}\}, \quad x = 0, \ \pm1, \ \pm2, \ldots \tag{34}$$

Applying logarithmic transform to the above formula,

$$f(x) = \log\{h(x)\} = \log\left\{\frac{\beta}{2\alpha\Gamma(1/\beta)}\right\} - (|x|/\alpha)^{\beta} = A - B\cdot|x|^{\beta} \tag{35}$$

Where$A = \log\left\{\frac{\beta}{2\alpha\Gamma(1/\beta)}\right\}$ and $B = \alpha^{-\beta}$. By expanding a Taylor series to (35), $f(x)$ can be expressed as follows:

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \cdots \tag{36}$$

When zero is set to a, $f(x)$ can be approximately represented by the nth polynomial series. $p_n(x)$ is denoted the n^{th} polynomial that fits function $f(x)$ best in terms of least-square and it is calculated as follows:

$$p_n(x) = p(1)\cdot x^n + p(2)\cdot x^{n-1} + \ldots + p(n)\cdot x + p(n+1) \tag{37}$$

The error between (35) and (37) is called errors of polynomial fitting, shown as follows:

$$R_n(x) = f(x) - p_n(x) \tag{38}$$

In our experiments, the 6th order polynomial fitting is adopted and the value of x is set to $0, 1, \ldots, 30$.

The expanded Markov transition features and joint density features, given by (5)–(8), (11)–(14), (17)–(20), (22)–(33), and the errors of polynomial fitting features, given by (38), are collectively called EPF features.

5. Reference features and feature selection

The EPF features that are extracted from the testing image are termed EPF_O. Reference EPF features that are extracted from the calibrated version are termed EPFc. The calibrated version is produced as follows:

1. Uncompress the JPEG image.
2. Crop the pixels, and the distance of these pixels to the boundary is in the range of 0–3.
3. Compress the cropped image in JPEG with the use of the same compression quantization table.

The difference, EPF_D, between the EPF_O and EPF_C is calculated by

$$\text{EPF}_D = \text{EPF}_O - \text{EPF}_C \tag{39}$$

The final features consist of EPF_O features and EPF_D features, and the total number of the features is 3950 for all the images used in the experiments.

Feature selection is an important issue in classification. In steganalysis, Avcibas et al. utilized the analysis of variance (ANOVA) to select statistically significant features [1,11]. However, commonly used statistics including ANOVA only consider the statistical significance of individual features and neglect the interaction among features. It has been demonstrated that the support vector weights based feature selection method SVM-RFE performs better than statistical significance based feature selection method, such as ANOVA, or a typical forward feature selection method [16].

To handle the large number of developed features and to improve the detection accuracy, the feature selection method of support vector machine recursive feature elimination (SVM-RFE) proposed by Guyon et al., an application of recursive feature elimination using the support vector weight magnitude as ranking criterion [7] is utilized for binary classification (for deciding whether the image under test is a steganogram or a cover), and an extension of SVM-RFE termed MSVM-RFE by Zhou and Tuck, which was originally designed to solve the multi-class gene selection problem [31], is utilized for multi-class classification (for deciding whether the image under test is a steganogram, and if so, the specific steganographic tool used for creating it).

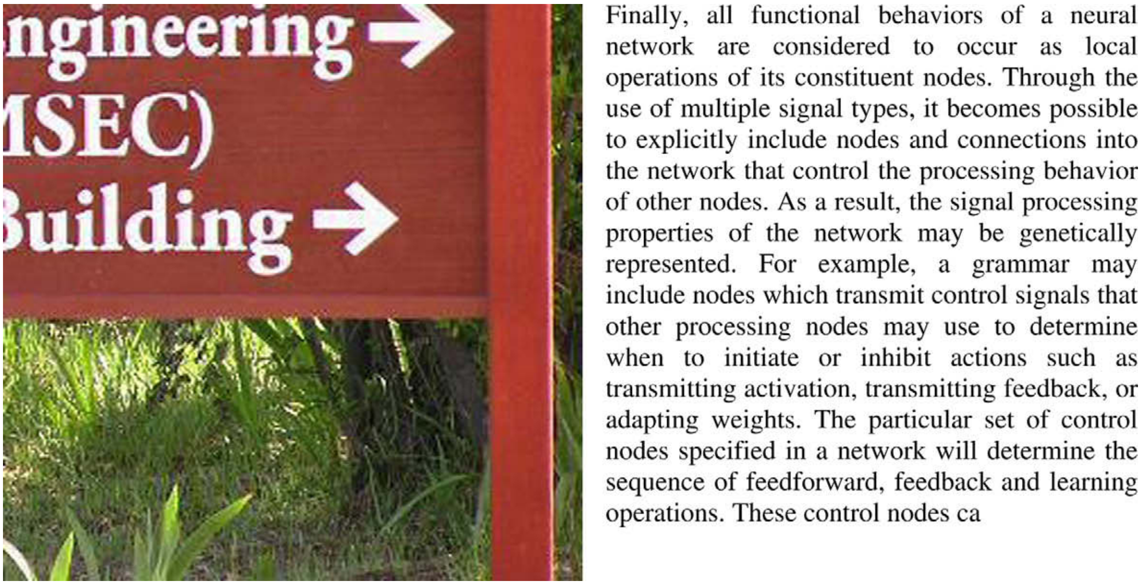
6. Experiments

6.1. Training and testing images

The original images are raw format digital pictures; they are 24-bit, 640 × 480 pixels, lossless true color and never compressed. These images are first cropped into 256 × 256 pixels according to the method of [15,16,19,21], in order to remove the low complexity parts (all original uncompressed images are available at http://www.cs.nmt.edu/~IA/steganalysis.html), and then converted into JPEG format with default quality. Different data, including texts, images, computer codes, and random binary bits, are embedded in these JPEG images to produce steganograms. Fig. 3 shows a JPEG steganogram and the hidden message. In this paper, modification strength in steganograms is used to evaluate the change of DCT coefficient, which is calculated as the ratio of the number of modified quantized DCT coefficients to the number of non-zero quantized DCT coefficients.

In our experiments, the following five types of steganograms and cover images are incorporated:

1. *CryptoBola* (*CB*): The commercial information hiding software CryptoBola determines which parts (bits) of the JPEG-encoded data play the least significant role in the reproduction of the image, and it replaces those bits with the bits of the cipher text. CryptoBola JPEG is available at http://www.cryptobola.com/. 3950 CryptoBola (CB) JPEG stego-images are generated. The average modification strength is about 0.29.
2. *F5*: Westfeld proposed the algorithm F5 that withstands visual and statistical attacks, yet it still offers a large steganographic capacity [29]. F5 implements matrix encoding to improve the efficiency of embedding and reduces the number of necessary changes. F5 employs per mutative straddling to uniformly spread out the changes over the whole steganogram. 5000 JPEG steganograms are produced by using F5 algorithm with the average modification strength of 0.35.



Finally, all functional behaviors of a neural network are considered to occur as local operations of its constituent nodes. Through the use of multiple signal types, it becomes possible to explicitly include nodes and connections into the network that control the processing behavior of other nodes. As a result, the signal processing properties of the network may be genetically represented. For example, a grammar may include nodes which transmit control signals that other processing nodes may use to determine when to initiate or inhibit actions such as transmitting activation, transmitting feedback, or adapting weights. The particular set of control nodes specified in a network will determine the sequence of feedforward, feedback and learning operations. These control nodes ca

Fig. 3. A JPEG steganogram (left) and the covert message (right) in the steganogram.

3. *JPHS* (*JPHIDE* and JPSEEK) *for Windows* (*JPWIN*): The design objective of JPHS was not simply to hide a file, but also to do it in such a way that it is impossible to prove that the host file contains a hidden file. Given a typical visual image, a low insertion rate (under 5%) and the absence of the original file, the author claims that it is not possible to conclude with any worthwhile certainty that the host file contains inserted data. JPHS for Windows (JPWIN) is available at: http://digitalforensics.champlain.edu/download/jphs_05.zip/. 3596 JPHS stego-images are generated with average modification strength of 0.11.
4. *Steghide*: Hetzl and Mutzel designed a graph-theoretic approach for information hiding based on the idea of exchanging rather than overwriting pixels [10]. Their approach preserves first-order statistics and the visual changes can be minimized, and hence the detection on the first-order does not work. In our experiments, 4504 JPEG stego-images are produced by Steghide with average modification strength of 0.06.
5. *Model Based steganography without deblocking* (*MB1*): Sallee presented an information-theoretic method for producing steganography [24]. 5000 JPEG stego-images are produced by Model Based steganography without deblocking (MB1). The average modification strength is about 0.36.
6. *Cover*: 5000 JPEG images without any hidden data are included in our experiments. These JPEG cover images and the steganograms are available upon request.

It should be noted that some cover images have smooth scenes which result in small capacity for information hiding. If the capacity is smaller than the size of embedded data, the embedding procedure failed when using CryptoBola, JPHS, and Steghide. Therefore, only 3950, 3596, and 4505 stego-images were produced by using CryptoBola, JPHS, and Steghide, respectively.

Some of the steganogram samples and cover samples are shown in Fig. 4.

6.2. Comparison of steganalysis performance in multi-class classification

EPF_O and EPF_D features are extracted from the six types of JPEG images and then mixed together. The objective is to recognize each type of JPEG images based on the features. To handle the large number of features and to improve the detection accuracy, MSVM-RFE is applied for feature selection. Support vector machines are applied to each selected feature set, and the classification accuracy is compared against the original Markov approach. To improve the steganalysis performance of the original Markov approach, MSVM-RFE is also applied to Markov features for feature selection. Ten experiments are performed on each feature set that is selected using MSVM-RFE. In each experiment, the ratio of training samples to testing samples is 3:7. Table 1 shows the average classification accuracy in different feature sets. In the comparison of the correct classification for each type of JPEG images by using EPF feature sets and Markov feature sets, if the difference of classification accuracy is over 1%, the higher one is marked in bold.

To evaluate the results listed in the confusion matrix of Table 1(1), the values in, e.g., the first column, indicate the classification results on JPWIN steganograms. With the use of 20 EPF features, 70.92% of the original JPWIN steganograms are correctly classified as such, while 0.58%, 0.16%, 2.54%, and 2.34% of them are misclassified, respectively, as F5, CryptoBola, Steghide, and MB1 stego-images, and the remaining 23.46% of them are misclassified as cover images. With the use of 20 Markov features, 23.60% of the JPWIN steganograms are correctly classified, while 0.57%, 2.83%, 6.00%, and 4.92% of them are misclassified as F5, CyptoBola, Steghide, and MB1 stego-images, respectively, and the remaining 62.07% of the JPWIN steganograms are misclassified as cover images.

The results listed in the confusion matrices of Table 1 show that the classification performances of EPF and Markov feature sets are very similar in case of detecting F5, CB, and MB1 steganograms. However, the rates of correct classification of JPWIN, Steghide, and cover images using EPF feature sets are significantly higher than those of using Markov feature sets. Especially noticeable is the correct detection of JPWIN steganograms where the classification accuracy of EPF feature sets outperforms that of the Markov feature sets by about 20–50%.

Generally, in an N-class classification, the testing sample sizes are $m_1, m_2, \ldots, m_N$, for class 1, class 2, …, class N, respectively. The numbers of the correct testing samples are t_1 for class 1, t_2 for class 2, …, and t_N for class N. The general classification accuracy (GA) is calculated as follows:

$$GA = \frac{\sum_{i=1}^{N} t_i}{\sum_{i=1}^{N} m_i} \tag{40}$$

Due to the difference of the testing numbers among different classes, a more reliable measurement called weighted classification accuracy (WA) is defined by:

$$\begin{aligned} &WA = \sum_{i=1}^{N} \left(w_i \times \frac{t_i}{m_i} \right) \\ &\text{s.t.} \quad \sum_{i=1}^{N} w_i = 1 \quad \text{and} \quad w_i > 0 \end{aligned} \tag{41}$$

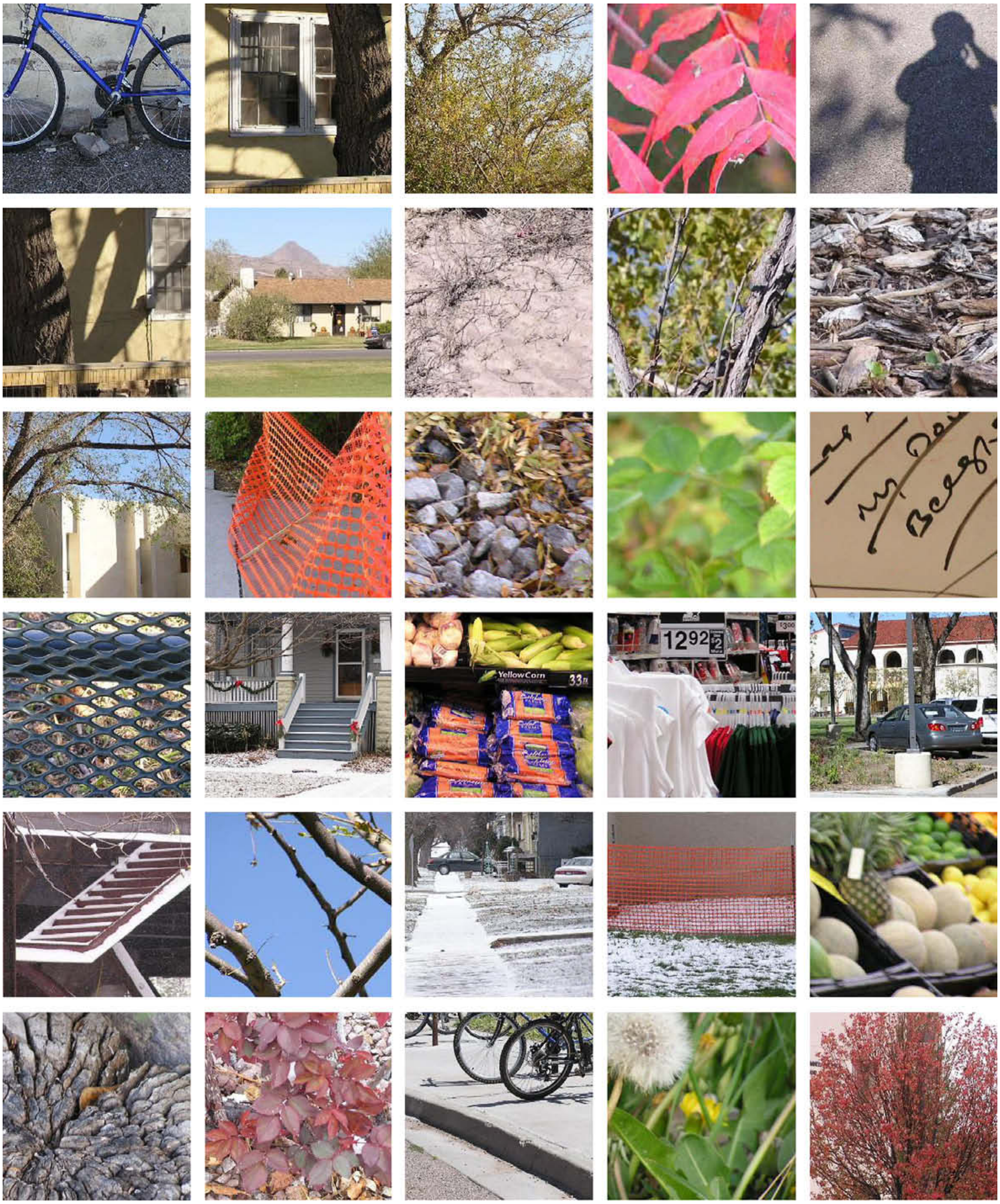


Fig. 4. Some samples of the six-class JPEG images. Row one to row five show some steganograms generated by using CryptoBola, F5, JPWIN, Steghide, and MB1 hiding tools/algorithms. Row six lists some cover images.

Since the testing images consist of six types in our experiments, without lost of generality, 1/6 is set to each weight factor, i.e., $w_i = 1/6, i = 1, 2, \ldots, 6$. Fig. 5 compares the classification accuracy values of *GA* and *WA* with the use of EPF and Markov feature sets, selected by MSVM-RFE.

As seen in Fig. 5, as the number of features selected by MSVM-RFE increases, the classification accuracy with the use of Markov feature sets increases at first; however, as the number of Markov features exceeds 200, the classification accuracy apparently deteriorates with the addition of more features. In our explanation, the original 324 Markov transition features include redundant information or statistically insignificant features, the inclusion of these features results in the decreased

Table 1
EPF vs. Markov — comparison of classification accuracy (%) in multi-class classification.

Method	Results	Testing type					
		JPWIN	F5	CB	Steghide	MB1	Cover
(1) *The results with the use of 20 features selected using MSVM-RFE*							
EPF	JPWIN	**70.92%**	0.06	0	6.46	0.34	9.85
	F5	0.58	99.71	0	0.03	0.39	0.32
	CB	0.16	0	100	0	0.00	0.03
	Steghide	2.54	0.00	0	**36.52**	1.45	5.56
	MB1	2.34	0.20	0	17.93	96.02	3.65
	Cover	23.46	0.03	0	39.06	1.79	**80.59**
Markov	JPWIN	23.60	0.06	0.00	7.76	0.40	8.06
	F5	0.57	99.61	0.00	0.04	1.00	0.81
	CB	2.83	0.06	99.95	1.08	0.29	0.85
	Steghide	6.00	0	0.01	17.86	0.87	4.91
	MB1	4.92	0.10	0.02	22.49	96.07	6.75
	Cover	62.07	0.19	0.01	50.76	1.37	78.62
(2) *The results with the use of 100 features selected using MSVM-RFE*							
EPF	JPWIN	**73.84**%	0	0	1.32	0	6.16
	F5	1.08	99.73	0.01	0.06	0.89	0.70
	CB	0.01	0	99.99	0.00	0	0.00
	Steghide	2.11	0	0	**80.81**	1.73	3.32
	MB1	0.39	0.08	0	3.84	96.99	0.48
	Cover	22.57	0.20	0.00	13.98	0.39	**89.34**
Markov	JPWIN	54.20	0	0	0.91	0.01	8.20
	F5	0.28	99.27	0	0.01	0.63	0.53
	CB	0.21	0	99.99	0.07	0	0.06
	Steghide	5.92	0	0	78.13	0.44	7.79
	MB1	0.60	0.13	0.00	5.32	97.92	1.13
	Cover	38.78	0.59	0.01	15.56	1.01	82.29
(3) *The results with the use of 200 features selected using MSVM-RFE*							
EPF	JPWIN	**72.69**%	0	0	0.90	0	5.07
	F5	2.68	**100**	0.11	0.19	2.16	2.49
	CB	0	0	99.89	0	0	0
	Steghide	1.72	0	0	**85.14**	0.51	2.37
	MB1	0.08	0	0	2.36	97.31	0.19
	Cover	22.82	0	0	11.40	0.03	**89.88**
Markov	JPWIN	51.94	0.00	0.01	1.48	0.00	8.30
	F5	0.27	98.83	0	0	0.60	0.49
	CB	0.04	0	99.99	0.03	0	0.01
	Steghide	5.167	0	0	78.35	0.36	6.59
	MB1	0.71	0.22	0	5.26	97.68	1.25
	Cover	41.89	0.94	0.01	14.87	1.36	83.37
(4) *The results with the use of 300 features selected using MSVM-RFE*							
EPF	JPWIN	**73.22%**	0	0	0.98	0	6.20
	F5	4.49	100	0.33	0.28	3.39	3.62
	CB	0	0	99.67	0	0	0
	Steghide	3.36	0	0	**89.59**	0.56	5.04
	MB1	0.08	0	0	2.1	96.02	0.11
	Cover	18.85	0	0	7.02	0.02	**85.03**
Markov	JPWIN	50.40	0	0	1.79	0.01	11.50
	F5	2.74	99.89	0.05	0.21	2.21	2.73
	CB	0.22	0	99.91	0.08	0	0.06
	Steghide	4.89	0	0	72.62	0.20	6.83
	MB1	0.62	0.11	0.02	6.65	**97.47**	1.36
	Cover	41.14	0	0.02	18.65	0.10	77.53

classification accuracy. Fig. 5 also demonstrates that the classification accuracy with the use of selected EPF features increases as the number of features increases, and there is no obvious deterioration. This is because the selected EPF features include more discriminate information to distinguish different types of JPEG images. However, the detection accuracy may eventually deteriorate when the number of features is increased to, say, a few hundred, due to the inclusion of redundant features and statistically insignificant features. Just as it is in pattern recognition, feature selection is an important and interesting issue in steganalysis.

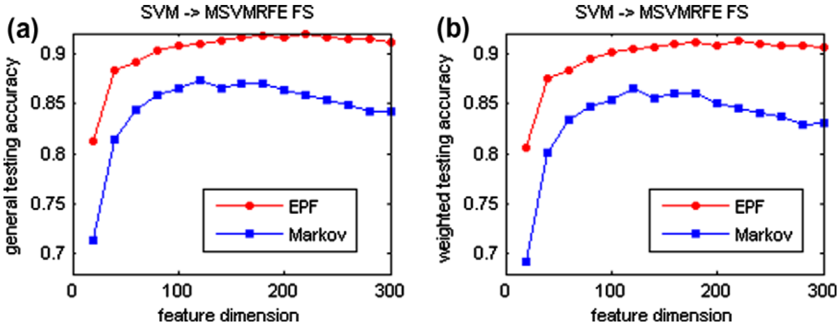


Fig. 5. General classification accuracy and the weighted classification accuracy by applying SVM to the EPF feature sets, and Markov feature sets, respectively. These feature sets are selected by using MSVM-RFE.

Both Table 1 and Fig. 5 indicate that, in comparison to the Markov approach, our method improves the detection performance of the six mixed types of JPEG images.

6.3. Comparison of steganalysis performance in binary classification

In addition to the detection performance on multi-class classification, the steganalysis performance on binary classification is also evaluated. SVM-RFE is applied to EPF features and Markov features, respectively, for feature selection, and then SVM is used for classification. Fig. 6 shows the detection accuracy, which is measured by weighted classification accuracy, with the use of EPF and Markov feature sets, respectively.

In distinguishing CryptoBola steganograms from covers, both the EPF method and the Markov approach obtain almost 100% classification accuracy. In detecting the information hiding behavior of the other four steganographic systems, EPF is superior to the Markov approach. The improvement of classification accuracy is ~14% for JPHS, ~0.3% for F5, ~0.5% for MB1, and ~8% for Steghide.

In the original Markov approach, each of the 324 Markov features is treated as a detector. Therefore, it is sensible to compare the results acquired by using the proposed EPF features (selected by SVM-RFE) to the results acquired by using selected (by SVM-RFE) Markov features and to the results acquired by using the entire Markov features (without feature selection). The results shown in Table 2 demonstrate that, even with fewer features (the numbers of features for steganalysis of JPWIN, F5, CB, steghide, and MB1 are 195, 65, 25, 165, and 165, respectively), the Markov approach with SVM-RFE obtains better

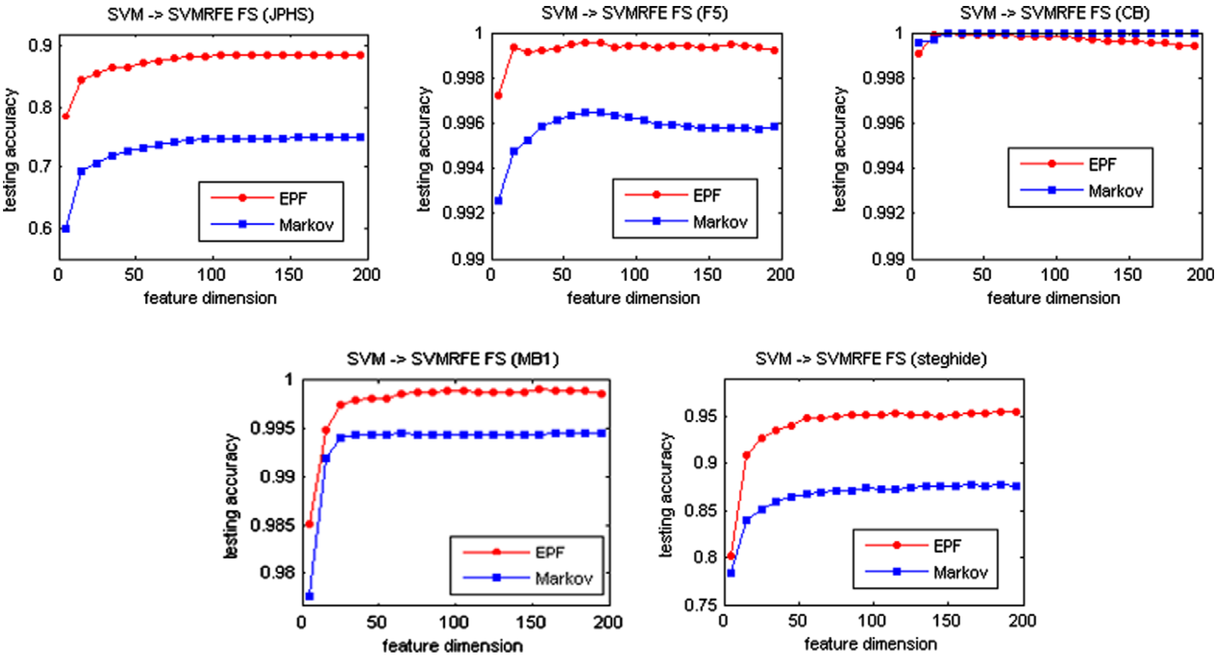


Fig. 6. Comparison of steganalysis performances in binary classification. Feature sets are selected by using SVM-RFE.

Table 2
Classification accuracy of EPF with SVM-RFE, Markov approach with and without SVM-RFE in binary classification.

Feature set	Classification accuracy (number of features)				
	JPWIN	F5	CB	Steghide	MB1
EPF with SVM-RFE	**88.66%** (145)	**99.97%** (50)	99.99% (25)	**95.43%** (195)	**99.91%** (155)
Markov approach with SVM-RFE	75.00% (195)	99.65% (65)	**100%** (25)	87.72% (165)	99.46% (165)
Original Markov approach	73.99% (324)	99.62% (324)	99.97% (324)	84.10% (324)	99.39% (324)

detection performances in comparison to the approach without feature selection. On average, EPF with SVM-RFE delivers the best steganalysis performance.

The superiority of our proposed method is based on the following facts. First, it substantially expands the original Markov approach into DCT inter-blocks and DWT approximation subband, and it extracts the Markov transition features and joint density features as well as the errors of polynomial fitting on the marginal density of DCT coefficients. All these features contain more differentially expressed information for distinguishing the steganograms from the covers. Second, by extracting the same features from the calibrated JPEG image, which was first proposed by Fridrich [5], and calculating the difference, the statistical significance to separate the steganograms from the covers is generally amplified. Third, the employed feature selection methods are effective in selecting feature sets that prove to enhance classification accuracy.

The most time consuming step is the training of SVM to construct the classification model due to the large number of features. However, training and model construction is performed off-line, and thus is not a major performance factor for the conceivable applications of steganalysis.

Feature selection is a very important issue in developing steganalysis methods since, if properly performed, it can reduce the training time (to achieve an overall speedup in spite of the additional time spent on feature selection), reduce the running or testing time of the trained classifiers, as well as improve classification accuracy – with the latter two measures being the most important performance factors of steganalysis.

7. Conclusions

To improve the steganalysis performance on several JPEG-based steganographic systems, a recently proposed Markov approach has been expanded to the inter-blocks of the DCT domain and to the approximation subband of the Haar wavelet domain, and new features of the joint density of the differential neighboring in the DCT and the DWT domains have been presented. These features and the errors of the polynomial fitting on the histogram of the DCT coefficients constitute the EPF features. The difference features between the EPF features, extracted from the testing image, and the reference EPF features, extracted from the calibrated version, are merged with the EPF features as a feature vector. Feature selection methods, SVM-RFE and MSVM-RFE, are applied, respectively, to the merged features for feature selection. Support vector machines are employed for classification. Experimental results show that this new approach obviously improves the steganalysis performance on several JPEG-based steganographic systems for both multi-class and binary classifications.

In this work, the detection performance has not been explored when the embedded message length is some fraction (e.g., 25% or 50%) of the maximum carrier capacity. In our further study, besides designing new detection methods, it will also be necessary and interesting to investigate the relation among detection performance, embedded message length, and image complexity, for steganalysis of JPEG images.

Acknowledgements

The authors gratefully acknowledge the support for this research from the Institute for Complex Additive Systems Analysis, a division of New Mexico Tech.

We are very grateful to Dr. Xin Zhou at Yale University for affording us his MSVM-RFE code and to Drs. Shi and Chunhua Chen at New Jersey Institute of Technology for affording us their Markov code. We appreciate the Editor-in-Chief, Professor Witold Pedrycz, and the anonymous reviewers for their insightful comments and invaluable suggestions. Special thanks go to Keegan and Elisa at the writing center of the New Mexico Tech for their professional editing help.

References

[1] I. Avcibas, N. Memon, B. Sankur, Steganalysis using image quality metrics, IEEE Transactions on Image Processing 12 (2) (2003) 221–229.
[2] C. Chang, C. Lin, Reversible steganographic method using SMVQ approach based on declustering, Information Sciences 177 (8) (2007) 1796–1805.
[3] C. Chang, C. Lin, C. Tseng, W. Tai, Reversible hiding in DCT-based compressed images, Information Sciences 177 (13) (2007) 2768–2786.
[4] R. Duda, P. Hart, D. Stork, Pattern Classification, second ed., Wiley, New York, NY, 2001.
[5] J. Fridrich, Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes, in: J. Fridrich (Ed.), Sixth Information Hiding Workshop, Lecture Notes in Computer Science, vol. 3200, Springer-Verlag, 2004, pp. 67–81.

[6] J. Fridrich, M. Goljan, D. Hogeam, Steganalysis of JPEG images: breaking the F5 algorithm, in: Proceedings of Fifth Information Hiding Workshop, 2002, pp. 310–323.

[7] I. Guyon, J. Weston, S. Barnhill, V.N. Vapnik, Gene selection for cancer classification using support vector machines, Machine Learning 46 (1-3) (2002) 389–422.

[8] J. Harmsen, W. Pearlman, Steganalysis of additive noise modelable information hiding, Proceedings of SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents V 5020 (2003) 131–142.

[9] J. Harmsen, W. Pearlman, Kernel fisher discriminant for steganalysis of JPEG hiding methods, Proceedings of SPIE, Security, Steganography, and Watermarking of Multimedia Contents VI 5306 (2004) 13–22.

[10] S. Hetzl, P. Mutzel, A graph-theoretic approach to steganography, in: Ninth IFIP TC-6 TC-11 International Conference, Lecture Notes in Computer Science, vol. 3677, 2005, pp. 119–128.

[11] T. Hill, P. Lewicki, Statistics: Methods and Applications, StatSoft, Inc., 2005. ISBN: 1884233597.

[12] S. Katzenbeisser, F.A.P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House Books, 2000.

[13] A. Ker, Improved detection of LSB steganography in grayscale images, in: Sixth International Workshop on Information Hiding, Lecture Notes in Computer Science, vol. 3200, Springer-Verlag, 2005, pp. 97–115.

[14] B. Li, Y. Shi, J. Huang, Steganalysis of YASS, in: Proceedings of 10th ACM Workshop on Multimedia and Security, 2008, pp. 139–148.

[15] Q. Liu, A. Sung, Feature mining and nuero-fuzzy inference system for steganalysis of LSB matching steganography in grayscale images, in: Proceedings of 20th International Joint Conference on Artificial Intelligence (IJCAI), 2007, pp. 2808–2813.

[16] Q. Liu, A. Sung, Z. Chen, J. Xu, Feature mining and pattern classification for steganalysis of LSB matching steganography in grayscale images, Pattern Recognition 41 (1) (2008) 56–66, doi:10.1016/j.patcog.2007.06.005.

[17] Q. Liu, A. Sung, M. Qiao, Detecting the information-hiding in WAV audios, in: Proceedings of 19th International Conference on Pattern Recognition, 2008.

[18] Q. Liu, A.H. Sung, B M. Ribeiro, Statistical correlations and machine learning for steganalysis, in: Proceedings of Seventh International Conference on Adaptive and Natural Computing Algorithms, 2005, pp. 437–440.

[19] Q. Liu, A. Sung, B. Ribeiro, M. Wei, Z. Chen, J. Xu, Image complexity and feature mining for steganalysis of least significant bit matching steganography, Information Sciences 178 (1) (2008) 21–36.

[20] Q. Liu, A. Sung, J. Xu, B. Ribeiro, Image complexity and feature extraction for steganalysis of LSB matching steganography, in: Proceedings of 18th International Conference on Pattern Recognition, ICPR, vol. 1, 2006, pp. 1208–1211.

[21] S. Lyu, H. Farid, How realistic is photorealistic, IEEE Transactions on Signal Processing 53 (2) (2005) 845–850.

[22] L. Marvel, C. Boncelet, C. Retter, Spread spectrum image steganography, IEEE Transactions on Image Processing 8 (8) (1999) 1075–1083.

[23] T. Pevny, J. Fridrich, Merging Markov and DCT features for multi-class JPEG steganalysis, in: Proceedings of SPIE Electronic Imaging, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, 2007.

[24] P. Sallee, Model based steganography, in: T. Kalker, I.J. Cox, Yong Man Ro (Eds.), International Workshop on Digital Watermarking, Lecture Notes in Computer Science, vol. 2939, 2004, pp. 154–167.

[25] K. Sharifi, A. Leon-Garcia, Estimation of shape parameter for generalized Gaussian distributions in subband decompositions of video, IEEE Transactions on Circuits and Systems for Video Technology 5 (1995) 52–56.

[26] T. Sharp, An implementation of key-based digital signal steganography, in: I. Moskowitz (Ed.), Information Hiding. Fourth International Workshop, Lecture Notes in Computer Science, vol. 2137, Springer-Verlag, New York, 2001, pp. 13–26.

[27] Y. Shi, C. Chen, W. Chen, A Markov process based approach to effective attacking JPEG steganography, Lecture Notes in Computer Sciences 4437 (2007) 249–264.

[28] V. Vapnik, Statistical Learning Theory, John Wiley, 1998.

[29] A. Westfeld, High capacity despite better steganalysis (F5 – A steganographic algorithm), in: I.S. Moskowitz (Ed.), Fourth Information Hiding Workshop, Lecture Notes in Computer Science, vol. 2137, Springer-Verlag, Berlin Heidelberg New York, 2001, pp. 289–302.

[30] F. Zhang, Z. Pan, K. Cao, F. Zheng, F. Wu, The upper and lower bounds of the information-hiding capacity of digital images, Information Sciences 178 (14) (2008) 2950–2959.

[31] X. Zhou, D.P. Tuck, MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data, Bioinformatics 23 (9) (2007) 1106–1114.

[32] <http://digitalforensics.champlain.edu/download/jphs_05.zip/>.

[33] <http://www.cryptobola.com/>.