STEGANALYSIS BASED ON MARKOV MODEL OF THRESHOLDED PREDICTION-ERROR IMAGE

Dekun Zou[1]*, Yun Q. Shi*[1]*, Wei Su*[1]*, Guorong Xuan*[2]

[1] ECE Dept., New Jersey Institute of Technology, Newark, New Jersey, USA
[2] Computer Science Department, Tongji UniversityShanghai, China

ABSTRACT

A steganalysis system based on 2-D Markov chain of thresholded prediction-error image is proposed in this paper. Image pixels are predicted with their neighboring pixels, and the prediction-error image is generated by subtracting the prediction value from the pixel value and then thresholded with a predefined threshold. The empirical transition matrixes of Markov chain along the horizontal, vertical and diagonal directions serve as features for steganalysis. Support vector machines (SVM) are utilized as classifier. The effectiveness of the proposed system has been demonstrated by extensive experimental investigation. The detection rate for Cox et al.'s non-blind spread spectrum (SS) data hiding method, Piva et al.'s blind SS method, and a generic QIM method (as embedding data rate being 0.1 bpp (bits per pixel)) are all above 90% over an image database consisting of approximately 4000 images. For generic LSB method (with various embedding data rates), our steganalysis system achieves a detection rate above 85% as the embedding data rate is 0.1 bpp and above.

1. INTRODUCTION

In recent years digital data hiding has become an active research field. Various kinds of data hiding methods have been proposed. Some methods aim at digital copy right protection, and authentication, while some aim at covert communication. The latter category of data hiding is called steganography. At the meantime, many public domain available stego software tools can be downloaded freely through the Internet. On the one hand, this can help to protect people's privacy. On the other hand, it provides criminals new high-tech tools for conspiracy. Consequently, various steganalysis methods have been proposed recently.

In [1], Fridrich et al. have discovered that the number of zeros in the block DCT domain of a stego-image will increase if the F5 embedding method is applied to generate the stego-image. This feature can be used to determine whether there exist hidden messages embedded with the F5 method. There are some other findings regarding the steganalysis of particularly targeted data hiding method [2, 3]. In [4], Farid proposed a more general steganalysis method based on image high order statistics, derived from image decomposition with separable quadrature mirror filters. The wavelet high-frequency subbands' high order statistics are extracted as features for steganalysis. It can differentiate stego-images from cover images with a certain success rate. The data hiding methods addressed for the steganalysis in [4] are basically the least significant bit-plane (LSB) modification based steganographic tools.

In [5], a steganalysis method based on Markov model is proposed. The empirical transition matrix of a test image is formed. Because the size of the empirical transition matrix is very large, e.g., the 65536 elements for a grey level image with bit depth of 8, it cannot be used as features directly. The authors of [5] select several largest probabilities along the main diagonal together with their neighbors, and randomly select some other probabilities along the main diagonal as features. It is obvious that some useful information might be ignored due to the random fashion of feature formulation. The data hiding methods addressed in [5] are restricted to spread spectrum (SS) data hiding methods. Although it may not carry as many information bits as the LSB methods in general, the SS methods can still serve for the covert communication purpose. For example, a terrorist command may need only to send a 'GO' command to his cell members for an attack. By the way, some newly developed SS methods can hide a large amount of data. For instance, a data embedding rate from 0.5 bpp (bits per pixel) to 0.75 bpp has been achieved in [6]. In addition, the SS methods are known more robust than the LSB. Therefore, it is necessary to consider the SS methods for steganalysis.

Inspired by [5], we propose in this paper a steganalysis system based on Markov chain model of thresholded prediction-error image. Image pixels are predicted with the neighboring pixels. The prediction error is obtained by subtracting the prediction values from the pixel value. Though the range of the difference values is increased, the majority of the difference values are highly concentrated in a small range near zero owing to the high correlation between neighboring pixels in natural images. Considering the large values in the prediction-error image may mainly be caused by the image content rather than by the data hiding process, a certain threshold is applied to the prediction errors to remove the large values in the prediction error images for steganalysis, thus limiting the dynamic range of the prediction-error image. The prediction-error images are modeled using Markov chain. Empirical transition matrix is calculated and served as features for steganalysis. Owing to the thresholding, the size of the empirical transition matrixes is decreased to a manageable size for classifiers so that all of the probabilities in the matrixes can be included into the feature vectors. For feature classification, the SVM with both linear and non-linear kernels are used as classifier.

The rest of this paper is organized as follows. Section 2 discusses the proposed scheme for feature extraction. In Section 3, a brief introduction of SVM is provided. Experimental results are presented in Section 4. Conclusion is drawn in Section 5.

1-4244-0367-7/06/$20.00 ©2006 IEEE

2. PROPOSED SCHEME FOR FEATURE EXTRACTION

Steganalysis can be considered as a two-class pattern classification problem if the test image needs to be classified as either a cover image, namely, no covert information is hidden in it, or a stego-image which carries hidden messages. Generally, the classification consists of two parts, feature extraction and pattern classification. The best way for classification is to use the image itself as feature since it contains all the information. However, the dimensionality of features thus extracted would then have been too high for most classifiers to handle. Therefore, feature extraction becomes crucial. For computer vision problems, the feature should represent the shape and color of an object. For steganalysis, we should look into different properties of images. The best feature for steganalysis should contain information about the changes incurred by data hiding rather than by the content of the image. In this section, we will discuss about the proposed features for steganalysis.

2.1 Prediction-Error Image

Generally speaking, natural images are continuous, smooth, and tend to have a high correlation between neighboring pixels because any object has a certain size. Often, the hidden data may be independent to the cover media. The watermarking process may change the continuity because it incurs random variation. As a result, it may reduce the correlation among adjacent pixels, bit-planes and image blocks. In steganalysis, the variation caused by data hiding should be amplified. We propose to use neighboring pixels to predict the current pixel. The predictions are made in three directions, namely, horizontal, vertical and diagonal since a digital image is actually a 2-D array. For each prediction we made, the prediction error can be obtained by subtracting the predicted pixel value from the original pixel value as shown in (1),

$$\begin{aligned} e_h(i,j) &= x(i+1,j)-x(i,j)\\ e_v(i,j) &= x(i,j+1)-x(i,j) \qquad (1)\\ e_d(i,j) &= x(i+1,j+1)-x(i,j)\end{aligned}$$

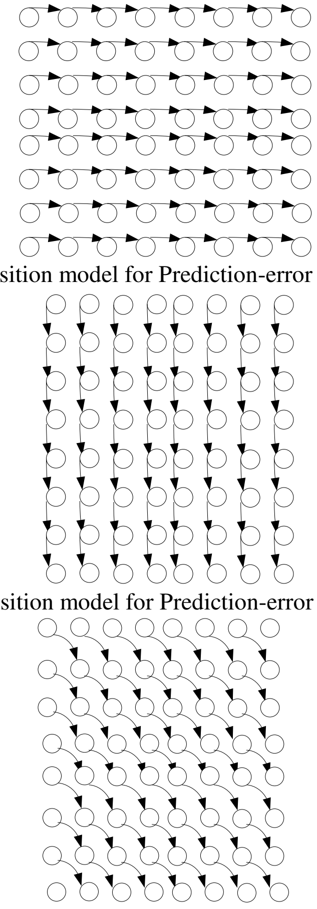
where e_h (i, j) indicates the prediction error for pixel (i, j) along horizontal direction while e_v (i, j) and e_d (i, j) the prediction error for pixel (i, j) on vertical and diagonal directions, respectively. For each pixel of an image, we have three prediction errors. At this point, the prediction errors will form three prediction-error images denoted by E_h, E_v and E_d, respectively.

2.2 Markov Transition Matrix of Thresholded Error Image

It is observed that the distortions introduced by data hiding are usually small comparing to the difference along pixels due to the presence of different objects in an image because, otherwise, the distortion itself will raise alarm when inspected by human eyes, thus breaking the very purpose of covert communication. Upon this observation, we think that large prediction errors reflect more on the image content itself rather than the data hiding process. Therefore, a predefined threshold *T* is adopted and the prediction errors are adjusted according to the following rule:

$$e\ (i,j)=\begin{cases} e(i,j) & |e(i,j)|\le T\\ 0 & |e(i,j)|>T\end{cases} \qquad (2)$$

As a result, the large prediction errors are regarded as 0. In other words, the image pixels are regarded smooth from the data hiding point of view. At this point, the value range of the prediction-error image are limited to [-T, T], with only 2*T+1 possible values.



(a) Transition model for Prediction-error image E_h

(b) Transition model for Prediction-error image E_v

(c) Transition model for Prediction-error image E_d

Figure 1. Transition models of theresholded prediction-error image. (One circle represents one pixel. This diagram showed an image of size 8 by 8. The arrows represent the state change in Markov chain.)

Instead of 1-D [5], a 2-D Markov chain model is applied to the thresholded prediction error images. Figure 1(a) shows the transition model for horizontal prediction-error image E_h, in which the Markov chain is modeled along the horizontal direction. Figure 1(b) and Figure 1(c) are the transition models for E_v and $E_{d,}$ respectively. The elements of the empirical transition matrices for E_h, E_v and E_d are served as features for steganalysis.

3. SUPPORT VECTOR MACHINES

The support vector machines [7] are very powerful for two-class classification. SVM can handle not only linear case but also no-linear case. For the linearly separable case, the SVM classifier simply searches for a hyper-plane that separates the positive pattern from the negative pattern. Denote the training data pair $\{\mathbf{y}_i, \omega_i\}, i = 1, \ldots, l$, $\mathbf{y}_i$ is a feature vector, and $\omega_i = \pm 1$ for positive/negative pattern. The linear support vector algorithms can be formulated as follows: if a separating hyper-plane exists, then all the training data satisfy the following constraints:

$$\mathbf{w}'\mathbf{y}_i + b \geq 1 \quad \text{if } \omega_i = +1 \qquad (3)$$

$$\mathbf{w}'\mathbf{y}_i + b \leq -1 \quad \text{if } \omega_i = -1 \qquad (4)$$

A Lagrangian formulation can be constructed:

$$L = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i \mathrm{y}_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i \qquad (5)$$

where α_i is the positive Lagrange multiplier introduced for each of the inequality constraints (3) & (4). The gradient of *L* with respect to **w** and *b* give the conditions:

$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i \mathbf{y}_i \omega_i \quad \text{and} \quad b = \frac{1}{l}\sum_{i=1}^{l} (\omega_i - \mathbf{w}'\mathbf{y}_i) \qquad (6)$$

Once the SVM classifier has been trained, the novel sample **z** from the testing data can be classified using the **w** and *b*. If $\mathbf{w}'\mathbf{z} + b$ is greater than or equal to zero, the image is classified as having a hidden message, otherwise classified as not containing a hidden message.

For non-linearly separable case, the learning machine maps the input feature vectors to a higher dimensional space where a linear hyper-plane is located. The transformation from the non-linear feature space to linear higher dimensional space is by using kernel function.

There are four basic kernels: linear, polynomial, radial basis function and sigmoid. The linear kernel is for linear SVM and the rest three other kernels are for non-linear SVM. In our experiment, linear and polynomial kernels are used.

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed steganalysis system, we use 2812 images download from the website of Vision Research Lab, University of California, Santa Barbara. [8] and all the 1096 sample images included in the CorelDRAW Version 10.0 software CD#3 [9]. Altogether, we have 3908 images as test image dataset. All color images are converted to grey level images with Irreversible Color Transform [10]:

$$Y = 0.299R + 0.587G + 0.114B \qquad (7)$$

The following typical data hiding methods are used in experiments: Cox et al.'s non-blind SS data hiding method (α=0.1) [11], Piva et. al's blind SS [12], a generic quantization index modulation (QIM) data hiding method [13] (with a quantization step size of 5, an embedding rate of 0.1bits per pixel (bpp)), and generic LSB. For all the data hiding methods, different random signals are embedded into different images. For the generic LSB data hiding, the embedding position is randomly selected for different images. Therefore, this model covers almost all steganographic tools that use LSB as the message embedding method. Various data embedding rates ranging from 0.3 bpp to as low as 0.01 bpp are applied. This range of embedding rates is comparable to that reported in [4] for those LSB based stego tools. The evaluation of the proposed steganalysis system is hence more general.

In our experimental evaluation, the threshold *T* is set to be 4. The effective prediction error values thus range from [-4 to 4], with 9 different values in total. Therefore, the dimension of transition matrix is 9 by 9, which is 81 features for each error image. Since we have three error images in three different directions, the number of total features is 243 for each image.

For each image in the image database, stego-images with each of the above-mentioned data hiding methods are generated. We evaluate the system with each one of the data hiding methods discussed above at a time. Randomly selected half of original images and the corresponding half of stego-images are used for training. The remaining half pairs of the cover images and stego-images are put through the trained SVM to evaluate the performance. The detection rate is defined as the ratio of the number of the correctly classified images with respect to the number of the test images. For each type of test, we do 20 times. All the experimental data reported in this paper are the average of the 20 times of tests.

4.1 Experments with Linear Kernel

At first, we use linear SVM to evaluate our system. The linear SVM has the property of fast training. But it may not perform well for non-linearly separable patterns. The Matlab SVM code from LIBSVM [14] is used. Table 1 is the test results.

Table 1. Experiment results of proposed steganalysis method.

Embedding Method	Detection Rates (243D feature, Linear SVM)		
	TN	TP	Average
Cox's SS	72.51%	88.65%	80.58%
Piva's SS	81.68%	95.46%	88.57%
QIM (0.1bpp)	88.66%	99.97%	94.32%
LSB (0.3bpp)	88.84%	96.98%	92.91%
LSB (0.2bpp)	83.98%	92.56%	88.27%
LSB (0.1bpp)	74.57%	79.97%	77.27%
LSB (0.05bpp)	64.30%	64.34%	64.32%
LSB (0.02bpp)	54.39%	54.62%	54.51%
LSB (0.01bpp)	48.11%	53.78%	50.94%

In Table 1, "TN" stands for "True Negative", i.e. the detection rate of original cover images. "TP" stands for "True Positive", the detection rate of stego-images. "Average" is the arithmetic mean of these two rates. In other words, it is the overall correct classification rate for all test images.

Table 2 Experiment results of steganalysis method of [5].

Embedding Method	Detection Rates (129D feature, Linear SVM)		
	TN	TP	Average
Cox's SS	86.64%	64.98%	75.81%
Piva's SS	71.34%	81.34%	76.34%
QIM (0.1bpp)	91.43%	90.07%	90.75%
LSB (0.3bpp)	56.69%	74.66%	65.68%
LSB (0.2bpp)	51.24%	69.07%	60.15%
LSB (0.1bpp)	45.11%	62.34%	53.73%
LSB (0.05bpp)	42.25%	58.33%	50.29%
LSB (0.02bpp)	39.17%	56.94%	48.05%
LSB (0.01bpp)	41.69%	52.68%	47.19%

We have implemented the Markov chain based method in [5] and applied it to the same set of images and the same data hiding methods. The same training and testing procedures are used. The results are listed in Table 2. It is noted that the QIM and LSB have not been tested and reported in [5]. It can be seen that our proposed system outperforms the method in [5] for all data hiding methods, in particular, for LSB methods.

4.2 Experments with non-Linear Kernel

We use polynomial kernel to train our proposed 243-D features and the 129-D features proposed in [5]. The results are listed in Table 3 and Table 4, respectively. The proposed method has a True Positive rate of over 90% for Cox's SS, Piva's blind SS, QIM and LSB with embedding strength over 0.1 bpp. In [4], evaluation of LSB is also provided. The embedded data are images with the sizes ranging from 32x32 to 194x194. The corresponding embedding data rates are from 0.02 bpp to 0.9 bpp and the detection rates ranges from 1.9% to 78%. Compared with the results reported in [4], the proposed method outperforms [4] with a large margin.

Table 3 Experiment results of proposed steganalysis method.

Embedding Method	Detection Rates (243D feature, Poly SVM)		
	TN	TP	Average
Cox's SS	84.14%	94.16%	89.15%
Piva's SS	89.81%	98.40%	94.10%
QIM (0.1bpp)	94.14%	99.91%	97.03%
LSB (0.3bpp)	96.27%	99.24%	97.75%
LSB (0.2bpp)	91.80%	97.09%	94.45%
LSB (0.1bpp)	83.69%	88.90%	86.30%
LSB (0.05bpp)	72.10%	78.18%	75.14%
LSB (0.02bpp)	57.92%	61.01%	59.46%
LSB (0.01bpp)	52.05%	52.51%	52.28%

Table 4 Experiment results of steganalysis method of [5].

Embedding Method	Detection Rates (129D feature, Poly SVM)		
	TN	TP	Average
Cox's SS	80.54%	74.67%	77.60%
Piva's SS	70.07%	85.10%	77.58%
QIM (0.1bpp)	90.20%	93.73%	91.96%
LSB (0.3bpp)	56.88%	81.09%	68.98%
LSB (0.2bpp)	48.21%	74.05%	61.13%
LSB (0.1bpp)	37.16%	62.47%	49.82%
LSB (0.05bpp)	33.33%	55.44%	44.38%
LSB (0.02bpp)	33.41%	48.21%	40.81%
LSB (0.01bpp)	35.88%	43.21%	39.54%

It can be observed that the detection rate increase by nearly 10% for our proposed features. However, there is no significant gain to use non-linear kernel for the features proposed in [5], that agrees with what the authors of [5] reported in their paper. One possible reason is that since our proposed feature vectors have higher dimensionality, the performance should be better in higher dimensional vector space.

5. CONCLUSION AND FUTURE WORK

This paper proposed a steganalysis method based on 2-D Markov model of thresholded prediction-error image. SVM with both linear and non-linear kernel are used as classifiers. The non-linear SVM performs much better than the linear SVM for our proposed features. The experimental results have proved that the proposed steganalysis features are more effective than that proposed in [5] for spread spectrum data hiding methods and more effective than the wavelet based features proposed [4] for LSB-based data hiding methods.

Although the generic LSB methods can cover most of the commercial steganographic tools, there still exist differences between the two. Further applying the proposed steganalysis scheme to those commercially available steganographic software tools is our future research work.

6. REFERENCES

[1] J. Fridrich, M. Goljan and D. Hogea, "Steganalysis of JPEG Images: Breaking the F5 algorithm", 5th Information Hiding Workshop, 2002, pp. 310-323.

[2] J. Fridrich, M. Goljan and R. Du, "Detecting LSB steganography in color and gray-scale images", Magazine of IEEE Multimedia Special Issue on Security, Oct.-Nov. 2001, pp. 22-28.

[3] R.Chandramouli and N.Memon, "Analysis of LSB based image steganography techniques", Proc. of ICIP 2001, Oct. 7–10, 2001.

[4] S. Lyu and H. Farid, "Detecting Hidden Messages Using Higher-Order Statistics and Support Vector Machines," 5th International Workshop on Information Hiding, Noordwijkerhout, The Netherlands, 2002.

[5] K. Sullivan, U. Madhow, S. Chandrasekaran, and B.S. Manjunath, "Steganalysis of Spread Spectrum Data Hiding Exploiting Cover Memory", SPIE2005, vol. 5681, pp38-46.

[6] G. Xuan, Y. Q. Shi, Z. Ni, "Lossless data hiding using integer wavelet transform and spread spectrum," IEEE International Workshop on Multimedia Signal Processing (MMSP04), Siena, Italy, September 2004.

[7] C. Cortes and V.Vapnik, "Support-vector networks," in Machine Learning, 20, 273-297, Kluwer Academic Publishers, 1995

[8] http://vision.ece.ucsb.edu/~sullivak/Research_imgs/

[9] www.corel.com

[10] M. Rabbani and R. Joshi, "An Overview of the JPEG2000 Still Image Compression Standard", Signal Processing: Image Communication 17 (2002) 3–48.

[11] I.J. Cox, J. Kilian, T. Leighton and T. Shamoon,, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. on Image Processing,* 6, 12, 1673-1687, (1997).

[12] A. Piva, M. Barni, E. Bartolini, V. Cappellini, "DCT-based watermark recovering without resorting to the uncorrupted original image", *Proc. ICIP 97*, vol. 1, pp.520

[13] B. Chen and G.W. Wornell, "Digital watermarking and information embedding using dither modulation," *Proceedings of IEEE MMSP 1998*, pp273 – 278.I.S.

[14] C.C. Chang and C.J. Lin, LIBSVM: a library for support vector machines, 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm