BLIND IMAGE STEGANALYSIS BASED ON RUN-LENGTH HISTOGRAM ANALYSIS

Jing Dong and Tieniu Tan

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences,P.O.Box 2728, Beijing, 100190
E-mail: {jdong, tnt}@nlpr.ia.ac.cn

ABSTRACT

In this paper, a new, simple but effective method is proposed for blind image steganalysis, which is based on run-length histogram analysis. Higher-order statistics of characteristic functions of three types of image run-length histograms are selected as features. Support vector machine is used as classifier. Experimental results demonstrate that the proposed scheme significantly outperforms prior arts in detection accuracy and generality.

Index Terms— Blind steganalysis, run length histogram, supervised learning.

1. INTRODUCTION

Steganography has been a hot topic and has drawn much attention in recent years. However, cases have been reported where steganography has been abused for bad purposes. Hence, the research of steganalysis, which is a counter-technology of steganography aimed at detecting the presence of secret message in cover medium, serves the urgent needs of network security to block covert communication with illegal information.

Various steganalysis techniques have been proposed for tacking steganographic algorithms. These techniques can be roughly ascribed to two categories. One is called specific steganalysis which is targeted at a particular known steganographic algorithm. The other is named universal (or blind) steganalysis that can defeat steganography blindly, or in another word, that can detect the hidden data without knowing the embedding methods, which seems to be more desirable in practical applications.

The statistical blind steganalysis schemes using supervised learning on features extracted from both plain cover and stego signals have been proved successful in coping with many existing steganographic methods. In [1], *Farid et al.* proposed a universal supervised learning steganalysis scheme using quadrature mirror filters to decompose a test image into wavelet subbands and the higher-order statistical features are generated from wavelet coefficients to capture the difference between plain cover and stego images. Similar features formulated from the prediction errors of wavelet coefficients of each high-frequency subband are also utilized in their method. *Harmsen and Pealman* in [2] described another method that exploits properties of the center of mass of the Fourier transform of the image histogram. In the work of [3], *Chen et al*. utilized multi-order moments of the projection histogram (PH) of image empirical matrix as well as the characteristic function of PH as their features for steganalysis.

The construction of valid blind steganalysis methods usually starts by extracting a set of features from the original and stego images and then training a classifier on a large number of such images to ensure that even the slightest statistical variation in the features is learned by the machine. The selection of appropriate features plays a crucial role in building the stego classifier. This paper focuses on extracting sensitive features to embedding modification and proposes a new, simple but effective blind image steganalysis approach. Statistical moments of characteristic functions of image run-length histogram and its variants are taken as features. SVM is utilized as classifier.

The rest of this paper is organized as follows. Section 2 discusses the proposed approach based on image run-length histogram analysis. Experimental results and comparisons are presented in Section 3, followed by concluding remarks in Section 4.

2. PROPOSED APPROACH

In this section, we present the details of the proposed method. We first describe three run-length representations and then discuss how to extract effective steganalysis features from run-length histograms.

2.1. Run-length Analysis For Steganalysis

The concept of run-length was proposed in the 1950s and has become the compression standard in fax transmissions and bitmap-file coding [4]. A run is defined as a string of consecutive pixels which have the same gray level intensity along a specific linear orientation (typically in 0^o, 45^o, 90^o, and 135^o). The length of the run is the number of repeating pixels in the run. For a given image, a run-length matrix $p(i, j)$ is defined as the number of runs with pixels of gray level *i* and run length *j*. For a run-length matrix $p_\theta(i, j)$, let *M* be the number of gray levels and *N* be the maximum run length. We can define the image run-length histogram (RLH) as a vector:

$$H_\theta(j) = \sum_{i=1}^{M} p_\theta(i, j). \qquad 1 < j < N \tag{1}$$

This vector represents the sum distribution of the number of runs with run length *j* in the corresponding image. In order to reduce the effect of different image sizes, the RLH may be normalized by the maximal value of the histogram. Short runs (with smaller *j*) refer to those runs with a small number of pixels, while long runs (with larger *j*) imply those runs with a largenumber of pixels.

As most current steganographic schemes hide data based on per-pixel processing, when one bit of message is embedded in the cover image, only one corresponding image pixel would be changed slightly. Such attributes of data embedding process would directly be reflected by the local intensity variations of the image. Our work is inspired by this observation. It is claimed that the run-length statistics could capture the coarseness of a texture in specified directions [4]. The lengths of runs could reflect the details of image

texture element, hence reflect the local intensity variations of the image. This is the major reason for our using the run-length histogram as the basis of the features for steganalysis. After the processing of per-pixel data embedding, the perceived local intensity continuity will be disturbed and the corresponding pixel runs will be alerted. For example, after LSB data embedding, values of some image pixels will be increased or decreased by one as a result. These changes would directly influence the image RLH. A concurrent change occurs: long runs in image would "break into" short runs, leading to a smaller number of long runs and a larger number of short runs. As a result, the image RLH would "shrink". Although there also exist cases that short runs may be combined to a long run, the tendency of these combinations is much less significant than the splitting of long runs because of the spatial correlation of natural images. Fig.1 shows the RLHs of the 'Lena' image before and after data hiding, where the shrinkage is clearly seen.

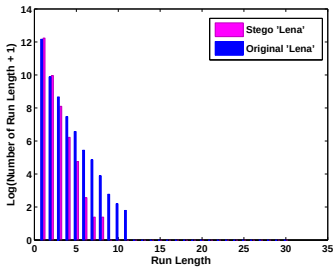

Fig. 1. Example of run length histograms of original and stego Lena.

2.2. Parameterized Run-Length Representations

For natural images, the number of short runs is significantly more than the number of long runs in an image RLH. The maximal length of runs is usually very limited compared to the range of possible length values (see Fig.2 the maximal length of the runs in Lena is less than 15). In order to make the shrinkage of image RLH more obvious so as to make RLH more sensitive to data embedding, we define two new run-length representations, which are variations of the traditional run-length matrix $p(i, j)$, by counting the pixels into a same run with different rules and parameters.

1. **Quantization Run-length Representation:** Instead of calculating the run-length matrix of image pixels directly, we firstly apply intensity quantization on the image plane using a quantization step factor Q. Then we calculate the RLH of the quantized image matrix. For example, for a 256 gray-level image with $Q = 2$, we get a new image matrix whose range of intensity values is from 0 to 127. Hence, the number of long runs in this new image RLH would increase compared to the original image RLH, because each pair of neighboring intensities would fall into the same run. Obviously, the lager the Q is, the more long runs we can expect. The traditional image run-length matrix is just the special case of quantization run-length matrix with $Q = 1$.
2. **Difference Run-length Representation:** A run in this type of representation is defined as a string of pixels with a maximum inter-pixel absolute intensity difference of ϵ along a direction. Thus, a string of consecutive pixels with small intensity difference would form a single run. For example, for a string of 4 image pixels with intensity of 124, 125, 125 and 126, their corresponding traditional run-length matrix is *p(124, 1), p(125, 2) and p(126, 1)*, while their corresponding difference run-length matrix in case of $\epsilon = 1$ is *p(124, 4)*. Similarly, the larger the ϵ is, the more long runs we can obtain. When ϵ is 0, the difference run-length matrix is simply the traditional image run-length matrix.

The two run-length representations defined above make the long runs in image run-length matrix much more than those of traditional run-length matrix. As a result, the tendency of shrinkage of their corresponding RLH caused by data hiding turns to be much more obvious, hence the RLH is more sensitive to data embedding. As shown in Fig.2, the maximal length of runs for the 'Lena' image has been extended to 40 and 60 respectively. At a first glance, it seems that the two new run-length representations are inconsistent with our assumption, that is, data hiding would increase the local intensity variations of images and the run-length statistics could reflect this variation. However, our experiences show that the influence on run-length statistics caused by data embedding could still be captured by global analysis on image RLHs and appropriate setting of parameters Q and ϵ. Fig.3 is the distributions of the two parameterized runs for the 'Lena' image regardless of intensity in raw representation, where the different runs are shown alternatively in white and black. We can still observe that their runs are broken into pieces after data hiding, in another word, there are more short runs and less long runs in stego image.

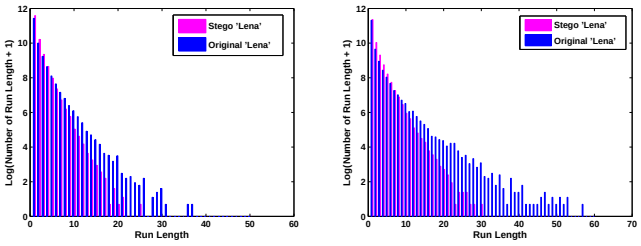

Fig. 2. Examples of parameterized image RLHs of original and stego Lena. (a): quantization RLHs with $Q = 4$; (b): difference RLHs with $\epsilon = 2$.

We now have three types of image run-length histograms, denoted as H_1 for traditional image RLH, H_2 for quantization image RLH, H_3 for difference image RLH. In the following, we introduce the features extracted from these image RLHs for steganalysis.

2.3. Features For Steganalysis

In most blind steganalysis methods, the higher-order statistics of an image and its other representations are introduced as features because these statistics are very sensitive to even slight embedding modifications [5]. The statistical moments of the characteristic function (denoted as CF) of certain histogram are claimed to be very effective features and were applied in a series of previous work [6]. Here, we also utilize multi-order moments of CF of the three types of image RLHs as features, defined as:

$$M_n = \sum_{j=1}^{L/2} f_j^n |F_i(f_j)| / \sum_{j=1}^{L/2} |F_i(f_j)|. \quad i = 1, 2, 3; \qquad (2)$$

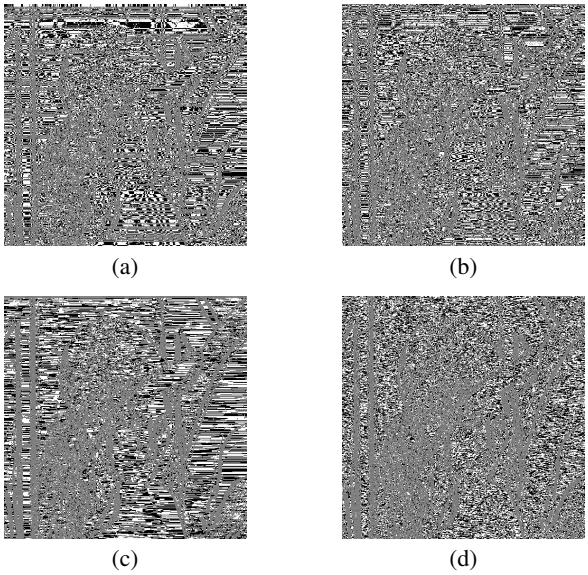

Fig. 3. Distribution of the two parameterized runs regardless of intensity in raw representation of the Lena image, where the different runs are shown alternatively in white and black. (a): original Lena with $Q = 4$; (b): stego Lena with $Q = 4$;(c): original Lena with $\epsilon = 2$; (d):stego Lena with $\epsilon = 2$.

where F_i is the CF of image RLH H_i (i.e. the DFT of H_i), $F_i(f_j)$ is the component of F_i at frequency f_j, and L is the DFT sequence length. In our experiments, we compute the first three order moments of each CF of the RLHs (H_1, H_2, H_3) along four different directions (0^o , 45^o, 90^o, and 135^o) as features for steganalysis. Thus, for each type of image RLH, we get a 12-D feature vector. The fourth and higher order moments of CF are not taken as our features because their discrimination of plain cover and stego images are poorer than the first three order moments in our experiments.

2.4. Classifier

The design of classifier is another important task in a pattern recognition system. Since our work in this paper only focuses on feature extraction rather than classification, Support Vector Machine (SVM) is simply taken as the classifier in our experiments for its optimal and efficient classification performance for large scale learning. More specifically, we utilize the SVM^{light} and a non-linear kernel is chosen. All the comparisons are tested on the same database and classifier.

3. EXPERIMENTS

In this section, several sets of experiments have been conducted to demonstrate the performance of the proposed blind steganalysis method. Comparative experimental studies are also presented to show the superiority of the method over typical existing methods in terms of detection accuracy and generality.

3.1. Database

We choose a commonly used image database, the *CorelDraw* Database,in our experiments. And totally 1142 images from *CorelDraw* version 11 CD *#4* were collected as the original images. Also, six sets of stego images were generated by using the following six common stego-algorithms:

#1: Non-blind spread spectrum (SS)method [7]. (0.15bpp, 36dB)
#2: 8×8 DCT block SS method [8]. (0.1bpp, 48dB)
#3: Generic LSB embedding method. (0.3bpp, 56dB)
#4: Adaptive LSB embedding method[9]. (0.3bpp, 51dB)
#5: LSB matching embedding method. (0.2bpp, 48dB)
#6: Generic quantization index modulation (QIM) method[10]. (0.11bpp, 47dB)

We just embed a small amount of secret data using the above stego-algoritms in order to make our tests more convincing. The approximate average embedding rates and the PSNR are shown in brackets. The same training and testing procedures are used. All the experiments are repeated 5 times under similar conditions, and the average rate is recorded for each run.

Table 1. The detection accuracy of three image RLHs with different parameters Q and ϵ at 5% false positive rate.

Scheme	Detection Accuracy (%)						
	H_1	H_2			H_3		
		Q=2	Q=3	Q=4	ϵ=1	ϵ=2	ϵ=3
#1	94.7	97.2	86.3	78.5	95.6	94.3	87.2
#2	96.4	96.8	84.9	80.9	98.1	75.3	67.1
#3	97.3	98.6	92.6	79.2	97.0	81.3	64.8
#4	95.2	97.1	90.3	85.6	97.3	94.5	87.2
#5	91.2	94.3	87.5	65.1	86.2	59.6	51.8
#6	96.6	98.9	91.5	88.2	98.6	64.2	53.7
Mixture	93.4	96.7	82.3	64.7	95.1	68.4	53.1

3.2. Detection Performance

We tested our method on the above database. There are two modes in the experiments. In a "individual" mode, we randomly select 600 original images and their stego images generated by one of the six stego-algorithms as training samples. The remaining 542 pairs are used for testing. To test the generalization (or the blindness) of the proposed features, in the "mixture" mode, again 600 original images are randomly selected for training. But this time, their corresponding training stego images are not chosen from one particular set of stego images but the mixed six sets of stego images. It means the whole training samples consist of 600 original images and 3600 (6×600) stego images. Then the remaining 542 original images and their corresponding 3252 (6×542) stego images are used to test. The detection accuracy represents the ratio of correct classification of test images.

As for the two new run-length representations, each representation has a parameter for generating the run-length histogram (i.e.Q and ϵ respectively). Different value of these two parameters would result in different image RLH, hence the different features for steganalysis. To evaluate the performance of features from different image RLHs for steganalysis, we firstly design an experiment to test the features from three different RLH representations. Table 1 illustrates the obtained detection accuracy of these features generated by different parameters Q and ϵ. Note that H_1 is the special case of H_2 and H_3 when $Q = 1$ and $\epsilon = 0$.

From Table 1, we can see that the best results are obtained from H_2 when Q= 2. As a whole, H_3 with $\epsilon = 1$ is also better than H_1. In order to select an optimal and rational combination of the three sets

Table 2. Comparison of our proposed approach with existing blind steganalysis methods. The detection accuracy numbers are obtained at 5% false positive rate.

Schemes	**Shi [1]** *78-D*	**Chen [2]** *108-D*	**Proposed** *36-D*
#1	90.56%	86.74%	96.76%
#2	91.83%	93.45%	97.43%
#3	92.51%	92.91%	99.37%
#4	93.28%	92.54%	98.62%
#5	86.07%	88.28%	93.69%
#6	97.24%	98.45%	99.92%
Mixture	90.24%	93.30%	97.92%

of RLH features for blind steganalysis, we use the features generated by H_2 with $Q = 2$, H_3 with $\epsilon = 1$, and the traditional image RLH H_1 together as a final feature vector for further blind image steganalysis. This way, we get a 36-D feature vector for each cover image.

3.3. Comparison With Existing Methods

Furthermore, we compare our final 36-D feature vector with the 78-D feature vector of [6] and the 108-D feature vector of [3]. The former is often taken as the baseline method for using statistical features extracted from CF of histograms and the latter is an effective extension to the former. The comparative results are shown in Table 2. It can be seen that our proposed 36-D feature vector provides clearly better detection accuracy compared with the other two methods.

In addition, another experiment is designed to test the generality of the combined 36-D feature vector to an untrained stego-algorithm. In this test, the training procedures are exactly the same as that in previous "mixture" mode, but during testing, we replace the #4 testing stego samples with another 542 stego images generated by an untrained stego-algorithm designed in [11] (The embedding rates and the PSNR are 0.15bpp and 52dB). The corresponding ROC curves are shown in Fig.4. We can notice that our proposed method has a better generality to an untrained stego-algorithm than others.

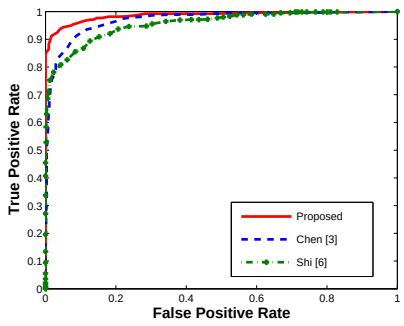

Fig. 4. Comparison of generality test of the proposed method with existing methods in terms of ROC curves.

4. CONCLUSION

In this paper, we have presented a new, simple but effective method for blind image steganalysis based on run-length analysis. As a first attempt, we extracted features through processing image run-length histograms because the distribution of image run-length histogram would be altered after data hiding. Moreover, we designed two alternative representations for image run-length in order to make the changes more obvious. The first three moments of the CF of three image RLHs are selected as features to distinguish the plain cover image from stego images. Experimental results have illustrated the effectiveness of our proposed features for blind steganalysis as well as their superiority over prior arts. Moreover, the proposed method also has a better performance to an untrained stego-algorithm compared to others in our tests.

Acknowledgment

This work is funded by research grants from the National Fundamental Research Program of China (Grant No.2004CB318110), and the National Key Technology R&D Program (Grant No.2006BAH02A13).

5. REFERENCES

[1] H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines," in *5th International Workshop on Information Hiding*, 2002.

[2] J. J. Harmsen and W. A. Pearlman, "Steganalysis of additive noise modelable information hiding," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents VI*, 2003, pp. 131–142.

[3] X. C. Chen, Y. H. Wang, L. Guo, and T. N. Tan, "Blind image steganalysis based on statistical analysis of empirical matrix," in *Proc. ICPR*, 2006, pp. 1107–1110.

[4] M. M. Galloway, "Texture analysis using gray level run lengths," in *Cornput. Graph. Image Proc.*, 1975, vol. 4, pp. 171–179.

[5] Y. Wang and P. Moulin, "Optimized feature extraction for learning-based image steganalysis," in *IEEE Trans.on Information Forensics and Security*, 2007.

[6] Y. Q. Shi and et al, "Image steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image,andneural network," in *ICME 2005*, 2005, pp. 269–272.

[7] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectru, watermarking for multimedia," in *IEEE Trans.Image Process*, 1997 6(12), pp. 1673–1687.

[8] J. Huang and Y. Q. Shi, "Adaptive image watermarking scheme based on visual masking," in *Electron, Letter*, 1998, vol. 34, pp. 748–750.

[9] W. N. Lie and L. C. Chang, "Data hiding in images with adaptive numbers of least significant bits based on human visual system," in *Proc IEEE Int. Conf. Image Processing*, 1999, pp. 286–290.

[10] B. Chen and G. W. Wornell, "Digital watermarking and information embedding using dither modulation," in *Proceedings of IEEE MMSP*, 1998, pp. 273–278.

[11] A. Piva, M. Barni, E. Bartolini, and V. Cappellini, "Dct-based watermark recovering without resorting to the uncorrupted original image," in *Proc ICIP97*, 1998, vol. 1, p. 520.