AN EFFECTIVE IMAGE STEGANALYSIS METHOD BASED ON NEIGHBORHOOD INFORMATION OF PIXELS

Qingxiao Guan [a b], Jing Dong [a], and Tieniu Tan [a]

[a] National Laboratory of Pattern Recognition, CAS Institute of Automation
[b] Department of Automation, University of Science and Technology of China
E-mail: qingxiao@mail.ustc.edu.cn, { jdong, tnt }@nlpr.ia.ac.cn

ABSTRACT

This paper focuses on image steganalysis. We use higher order image statistics based on neighborhood information of pixels (NIP) to detect the stego images from original ones. We use subtracting gray values of adjacent pixels to capture neighborhood information, and also make use of "rotation invariant" property to reduce the dimensionality for the whole feature sets. We tested two kinds of NIP feature, the experimental results illustrates that our proposed feature sets are with good performance and even outperform the state-of-art in certain aspect.

Index Terms— steganalysis, steganalysis evaluation

1. INTRODUCTION

Digital images can be used for hiding secret information if it was slightly modified while caused no change in visual content. Steganography and steganalysis are two complementary techniques for secret communication scenario. In the language of steganography, the image which is embedded with secret message usually called a stego image. Since it has no abnormal trait in image format or content appearance, it can be abused for illegal messages transmission. Steganalysis is then studied to avoid steganography for bad purposes by detection of the stego images from innocent ones (cover images). Effective steganalysis methods are often developed by extracting distinguishable features from images themselves. Features for steganalysis are usually in form of a high dimension vector and contain statistical information of image. As usually be seemed as a hypothesis problem [1], features of cover and stego image are respectively derived from two different distributions, and the detector is constructed by supervised learning methods with training samples from these two distribution. Previously, steganalysis methods count on linear discriminate analysis as classifier. Recently Support Vector Machine (SVM) becomes a powerful tool which is wildly used as an analyzer. For a stegnalysis system, good feature reflects discriminative distributions of stego and cover images, and such kind of feature should be of most significant. Format of candidate image is also an important factor that always determines what domain the feature can be extracted from. In this paper we focus on effective feature extraction for image steganalysis and we introduce our recently developed steganalysis feature sets based on neighborhood information of pixels (NIP). The feature sets are extracted from spatial domain and are mainly for detecting spatial domain based steganographic algorithms (i.e. LSB matching revisited [2], HUGO [3] etc.) and cross domain based steganographic algorithms (i.e. YASS [4] etc.).

The rest paper is organized as follow: in Section 2, we introduce two important characters of the existing universal image staganalysis feature sets based on our studying. Then we propose our new and effective image steganlaysis method based on neighborhood information of pixels (NIP feature sets) in Section 3. In Section 4, details and results of designed experiments are presented. Finally the conclusion and discussion are drawn in Section 5.

2. FEATURE ANALYSIS

Based on our observation on previous literatures, there are two characters for universal image steganalysis features:

1. Steganogprphy embeds secret message by small modification in image, this operation equal to adding some weak noise to the original image. We call such noise "stego noise". Image content severely affects feature's sensitivity to "stego noise", thus most effective universal steganalysis features are extracted after a preprocessing step of "depressing" image content which is to enhance effectiveness of the stego features. In previous literatures, there are several approaches for this task [5-10]. Subtracting adjacent pixel values (or DCT coefficients) in the candidate image is simple but very effective, it is adopted by SPAM proposed in [10] and a 324-D Markov based feature set proposed by Shi et al in [9], which are most effective and are considered as the state-of- art feature sets for spatial domain and frequency domain steganalysis methods respectively.
2. Steganalysis are always utilizing image statistics because steganography only slightly change images in a microscopic level. Statistics can be either first order ones or higher order ones. First order ones, such as gray-level histograms of image pixels, are easy to be calculated and with low dimensionality, but they may have less ability to capture the difference between cover and stego images, and can be deliberately revised in message embedding via many steganography scheme to avoid being detected. Higher order statistics are of joint distributions or conditional distributions of local structure unit in an image, such as pairs or triples of adjacent differenced pixels. They are more capable for steganalysis detection, although they require higher dimensionality and may suffer from "dimension of curse". Controlling dimension of high order feature is inevitable and is implemented by various means.

These two characters are related to properties of our proposed feature sets. We will emphasize that part for our discussion in the

978-1-4577-1302-6/11/$26.00 ©2011 IEEE

following sections. Empirically, we believe higher order statistics of adjacent image pixels in a neighborhood area contain more local information of an image itself, and are suitable for steganalysis. Inspired by these ideas, we developed our image steganalysis feature sets based on neighborhood information of pixels (NIP). These feature sets are aimed to preserve the structure of neighboring pixels in an image. In another word, it is an image histogram of complex "differenced" neighborhood structure unit. In this paper, we extend pixel substraction to higher order statistics of neighboring image pixels and meanwhile avoiding "curse of dimensional" problem for high-dimension features. We design several experiments to test our NIP feature sets for different cases including detection on several kinds of stego images by different feature sets, and we make a comparison of our proposed feature sets with SPAM, which is the state-of-art steganalysis technique. Experimental results show that our feature is more effective.

2. DETAILS OF THE PROPOSED FEATURE SETS

Traditionally, most spatial domain based feature, especially Markov based features [9], only consider variation of pixels in single directions. However, adjacent pixels in neighbor also present stego noise, we want to further exploit such virtues for steganalysis. This is the main motivation of our proposed feature. As discussed before, we may cope with very high dimensionality if we comprise more pixels in multiple neighborhoods. Heuristically we believe statistics of image symmetric in orthogonal directions counts for staganlaysis features, hence we combine a certain "rotation invariant" states in the feature sets to reduce dimensionality to alleviate over-learning problem. We will describe the details of our proposed feature sets in the following section.

First, we define the neighbor of a pixel in our method. Let $a_{x,y}$ be the gray value of image pixel in coordinates (x,y). (x,y) along any directions go over the entire image. Specifically, we define neighbor set $N(x,y)$ of pixel (x,y) as a sequence $N(x,y)=\{a_{x-1,y},\ a_{x,y+1},\ a_{x+1,y},\ a_{x,y-1}\}$ with elements of gray values $a_{x-1,y},\ a_{x,y+1},\ a_{x+1,y},\ a_{x,y-1}$ around pixel (x,y) in clockwise sequence. Note it is a sequence so that relative position of element is preserved. And elements in defined neighbor are symmetry in position. This is important for our combining "rotation invariant" states.

Then we subtract gray value of pixels of neighbor with that of center and then threshold them with T. More details are presented as follows:

We define differenced and thresholded sets $DS_1(x,y)$ for a neighbor set $N(x,y)$.

$$DS(x,y)=\{DS^1(x,y),\ DS^2(x,y),\ DS^3(x,y),\ DS^4(x,y)\}$$

$$DS^1=Tsh(a_{x-1,y}-a_{x,y}),\ DS^2=Tsh(a_{x,y+1}-a_{x,y})$$

$$DS^3=Tsh(a_{x+1,y}-a_{x,y}),\ DS^4=Tsh(a_{x,y-1}-a_{x,y})$$

While $Tsh(\bullet)$ denote thresholding if input number is larger (or smaller) than *T*(-*T*), as following definition:

$$Tsh(x)=\begin{cases} x, & -T<x<T \\ -T, & x\le -T \\ T, & x\ge T \end{cases}$$

After thresholding, elements of DS take values from $-T$ to T, thus DS have $(2T+1)^4$ possible states for any single pixel. This operation is reasonable because most differenced gray values of adjacent pixels distribute in a limited range. Beside, differenced gray values out of this range are mainly caused by sharp edge of image. Stego noise is a kind of weak noise, therefore information of sharp edge is not only useless for steganalysis, but also excessive for introducing disturbance. Although we reduced number of possible states of DS by taking threshold, even we set value T to a very small number, the states of $(2T+1)^4$ are still too large to get a histogram of DS. Hence we combine "rotation invariant" states then. Here we explain "rotation invariant" first. For two states of differenced neighbor units, we say they are "rotation invariant" state, only if they can be turn to an identical state by rotating one of them by 0, 90, 180, or 270 degree. For example, as fig.1 shows, they are "rotation invariant" states in terms of each other. Their sequences of differenced values are identical to each other when rotating them around their center by 0, 90, 180, or 270 degree respectively.

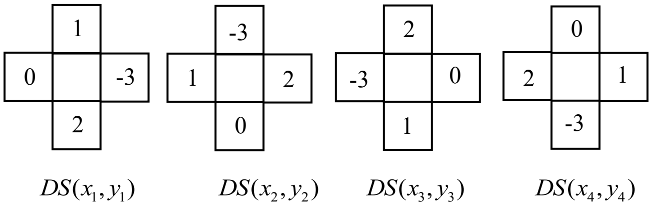

Fig.1: An example of "rotation invariant" state

We then describe how to implement "rotation invariant" coding. In principal we need to map states to same value if they are "rotation invariant". We map any $DS(x,y)$ to a code $C(x,y)\in[1,\ (2T+1)^4]$ that ensure "rotate invariant" ones be identically and uniquely coded. This implies that if two differenced neighbor unit $DS(x_1,y_1)$ and $DS(x_2,y_2)$ are "rotation invariant", they will have identical code satisfying $C(x_1,y_1)=C(x_2,y_2)$.

After coded, we can calculate histogram H for coded $DS(x,y)$, as our feature sets:

$$H=(h_1^1,h_1^2,\ldots h_1^{(2T+1)^4})$$

$$h_1^i=\sum_{x,y}\delta(C(x,y),i)\quad i=1,2\ldots(2T+1)^4$$

Where $\delta(x,y)=\begin{cases}1 & if\ x=y\\ 0 & if\ x\ne y\end{cases}$

Although in this step, dimensionality of H equals to $(2T+1)^4$, but it is obvious that some bins of H constantly equal to zero due to this special encoding method. These definite zero bins can be easily distinguished by a simple analysis. We remove those redundant zero bins yielding a feature set denoted as F. Dimensionality of F is less than H. For $T=3$, the dimensionality of F is 616. Finally F are normalized to adapt any size of images. The flowchart of extracting our feature is presented in Fig.2.

Considering the trade-off between preserving neighbor structure and low dimensionality of feature, it is acceptable when we set $T=3$.Except previous defined neighbor, we can also define the neighbor of pixel (x,y) as a set of adjacent pixels in diagonal and mirror diagonal: $\{a_{x-1,y-1},\ a_{x+1,y-1},\ a_{x+1,y+1},\ a_{x-1,y+1}\}$, and extract NIP feature with the same procedure as described. The dimensionality of this type of NIP feature is also 616.

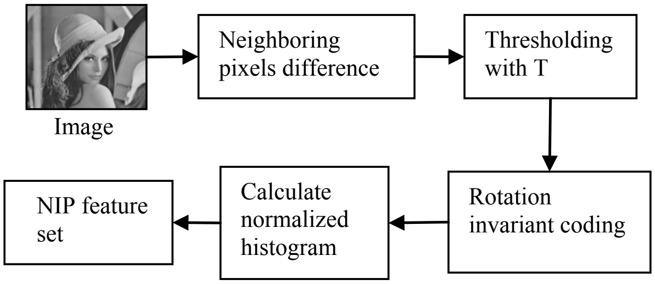
Image
Neighboring pixels difference
Thresholding with T
Rotation invariant coding
Calculate normalized histogram
NIP feature set

Fig.2: The flow chart of extracting our feature

3. EXPERIMENTAL RESULTS

The experiments are implemented on BOWS2 [11] image database. There are 10000 gray images in BOWS2 with fixed size 512 x 512. They were restored in PGM format. We randomly select 80% images as cover for training set and rest as cover for testing set. These cover images are embedded using N kinds of steganographic methods to generate stego images for training set and testing set. This result in a training set of 8000 images and a testing set of 2000 images for each kind of steganographic methods. There are enough training samples and testing samples to avoid over-learning and meanwhile obtain credible testing result. Performance of feature sets is assessed by their detection rate of test samples. We use true positive (TP), true negative (TN), and average rate (AR) to compare the detection performance. True positive rate stands for proportion of stego samples be correctly classified, and vice versa the true negative. Average rate is the average value of TP and TN. In this paper we compared our feature with SPAM feature. Our experiments consist of two parts: Detecting spatial domain based steganography: LSB matching revisited, Hugo, and cross domain based steganography: YASS. In both parts, we use SVM with RBF kernel as classifier.

The first part is detecting LSB matching revisited steganography [2] and Hugo steganography [3]. LSB matching revisited is an improved version of LSB matching. It firstly divides pixels of cover into non-overlap pairs. By using a binary function, it at most changes one of pixel in a pair to embed 2 bits of secret message. It is a very secure steganographic algorithm.

Hugo [3] is another spatial domain based steganography, it is more complex than LSB matching or LSB matching revisited, Hugo is aimed to preserve higher order feature. It evaluates each component of SPAM, and selects adding or subtracting 1 to gray values by evaluating its influence to SPAM feature. Hugo is the latest spatial domain based steganography in the literature.

Digital image can be embedded with secret message of different length, thus contain different embedding payload. Higher embedding payload introduce more modification in embedding, and consequently bring higher risk of been detected. In another experiment, we use bit per pixel (bpp) to evaluate embedding payload for comparison. We tested LSB matching with 0.15bpp and 0.25bpp payload, and Hugo with 0.4 bpp. Although 0.4bpp is much higher, Hugo with 0.4bpp payload is much more undetectable than LSB matching revisited with 0.25bpp payload (see Table 1-3).

The other part is a testing on YASS steganography only. YASS [10] is a recent developed steganography algorithm. It modifies s image data neither in JPEG coefficient nor spatial domain (gray value of pixels). It firstly decompresses image to spatial domain and use RA coding to code bit stream of secret message, then use QIM method to embed them in first 19 DCT AC coefficients of 8×8 blocks, each of which is randomly selected from non-overlaping big blocks of image. Note these big blocks are larger than 8×8, and the locations of embedding blocks randomly distribute in whole image so that it do not always coincident with 8×8 blocks compressed in JPEG. Even modified coefficients in "cross domain", stego images of YASS finally were advertised in JPEG format. RA coding and QIM embedding made it more robust to resist distortion introduced by JPEG compression in final step. However, payload embedding rate of YASS is not proper to be assigned in the same way of other spatial domain based embedding method because embedding rate of YASS do not increase linearly refer to length of secret message. Payload of YASS is dominated by size of big blocks. Larger size of big blocks correspond to lower number of modified blocks, and consequently lower embedding payload. In this part, we tested YASS of two settings:

Setting 1: big block size equal to 13, Qh is quantitize matrix and fixed in standard quality 50 in QIM embedding, and image was compressed to JPEG standard quality 75 at last.

Setting 2: big block size equal to 11, Qh is automatically selected from the set {65, 70, 75} according to block variance of embedding 8×8 blocks [13]. Image was also compressed to JPEG standard quality 75 at last. Both these two settings produce very low payload rate in embedding.

Table 1 to Table 3 are experimental results of detecting LSB matching revisited and Hugo. And Table 4 to Table 5 are experimental results of detecting YASS. In the first column of Table 1 to Table 5, NIP (horiz & vert) denotes our feature extracted from neighbor pixels in horizontal and vertical directions, NIP (diag & mirror diag) denotes our feature extracted from neighbor pixels in diagonal and mirror diagonal directions, these two feature sets are specified in section 2. SPAM denote SPAM feature. AR, TN, TP in the first rows of Table 1-5 respectively denote average rate, true negative rate, true positive rate.

Table 1: Experimental results of detecting LSB matching revisited (0.15 bpp)

LSB matching revisited (0.15 bpp)	**AR**	**TN**	**TP**
NIP (horiz & vert)	80.475	80.2	80.75
NIP (diag & mirror diag)	67.125	64.5	69.75
SPAM	**82.2**	**80.65**	**83.75**

Table 2: Experimental results of detecting LSB matching revisited (0.25 bpp)

LSB matching revisited (0.25 bpp)	**AR**	**TN**	**TP**
NIP (horiz & vert)	84.75	86.2	83.3
NIP (diag & mirror diag)	73.75	71	75.65
SPAM	**88.6**	**87.45**	**89.75**

Table 3: Experimental results of detecting Hugo (0.4 bpp)

Hugo (0.4 bpp)	**AR**	**TN**	**TP**
NIP (horiz & vert)	**66.9421**	**66.3361**	**67.5482**
NIP (diag & mirror diag)	61.8457	60.4408	63.2507
SPAM	62.8099	60.1653	63.9669

Table 4: Experimental results of detecting YASS (setting 1)

YASS Setting 1 (B=14, Qh=50, Qa=75)	**AR**	**TN**	**TP**
NIP (horiz & vert)	80.125	77.4	82.85
NIP (diag & mirror diag)	**84.225**	82.05	**86.4**
SPAM	83.825	**83.55**	84.1

Table 5: Experimental results of detecting YASS (setting 2)

YASS Setting 2 (B=14,Qh=60,65,70, Qa=75)	**AR**	**TN**	**TP**
NIP (horiz & vert)	81.225	78.35	84.1
NIP (diag & mirror diag)	**85.675**	83.65	**87.7**
SPAM	85.525	**84.8**	86.25

4. CONCLUSION

From experimental result, it is clear that our feature is comparable to SPAM. There is an interesting phenomenon. For both settings of YASS algorithm, our NIP feature setsin terms of diagonal and mirror diagonal directions is more effective than that of vertical and horizontal directions. But it reversed in experiment of detecting spatial domain based steganography LSB matching revisited and Hugo. Considering the distance of pixels in spatial domain, adjacent pixels in diagonal directions have larger distance with central pixels than adjacent pixels in vertical and horizontal. If we assume each pixels are generated in an identical way in obtaining digital image, and view gray value subtract as a “high pass” filter in certain scale, probably we can suppose that cross domain based steganography and spatial domain based steganography respectively leaves more artifacts in different scales. However, there still no more complex high pass filter is more effective than subtracting gray values of adjacent pixels. In detecting YASS steganography, for both settings, accuracy of our feature (diagonal) is very close to SPAM, the effectiveness of our proposed feature sets is proved., In detection of spatial domain based steganography, SPAM has higher accuracy for detecting LSB matching, but do not have absolute superiority. Moverover, our feature outperforms SPAM in detecting HUGO steganography. It is no surprising because HUGO is designed with a mechanism aimed to resist detection by SPAM..

Extracting effective feature for steganalysis is really a challenging topic. Our feature proposed in this paper is proved effective. For future analysis, we compared our proposed features with SPAM more. From literature [10] we know that SPAM have remarkable promotion when use second order Markov feature with $T = 3$ rather than first order with $T = 4$. Our feature considered more complex neighbor structure and is comparable to SPAM, but so far it have not come up to the performance we expected. There are some reasons. Our feature is a joint probability, while SPAM is a conditional probability. Although our feature is competent to present stego noise, image content severely disturbed it performance. Because we finally calculate histogram for the whole image, different image content definitely have different statistical properties, such as images of snow ground has less edges than images of woods. SPAM only considered 3 differenced pixels in one direction. It is a condition probability, which means it is divided by a factor of marginal probability. This operation somehow diminishes the influences of image content. To solve this problem, we argue that there are several approaches. Applying more complex learning strategy, such as hierarchical learning, feature fusion, or adding more complement features would be feasible and promising. Also, the dimensionality of our feature is less than SPAM. Part of SPAM is redundant since we know there is a underlying constrain for SPAM that: $\sum_{x_i} p(x_i \mid x_{i-1}, x_{i-2}) = 1$. From this point, the SPAM may be improved by lower dimensionality while containing equivalent information for steganalysis.

5. REFERENCES

[1] Cachin, C., “An information-theoretic model for stega -nography,” information and computation 192(1), 41-56 2004

[2] Mielikainen, J., "LSB matching revisited," *Signal Process -ing Letters, IEEE* , vol.13, no.5, pp. 285- 287, May 2006 doi: 10.1109/LSP.2006.870357

[3] Tomas Pevny and Patrick Bas, “Using High-Dimensional Image Models to Perform Highly Undetectable Steganography,” *12th Information Hiding*, Calgary, Alberta, Canada, June 28-30, 2010

[4] K. Solanki, A. Sarkar, and B. S. Manjunath. YASS: Yet another steganographic scheme that resists blind steganalysis.*9th International Workshop on Information Hiding,* Saint Malo, France, Jun. 2007.

[5] Zou, D, Shi, Y.Q, Wei Su; Guorong Xuan; , "Steganalysis based on Markov Model of Thresholded Prediction-Error Image," *Multimedia and Expo, 2006 IEEE International Conference on* , vol., no., pp.1365-1368, 9-12 July 2006 doi: 10.1109/ICME.2006.262792

[6] H. Farid, “detecting hiden messages using higher order statistics and support vector machines,” in *5th International Workshop on Information Hidding, 2002*

[7] Tomáˇs Pevný, Jessica Fridrich, “Multiclass Detector of C -urrent Steganographic Methods for JPEG Format,” IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 3, NO. 4, DECEMBER 2008.

[8] J. Kodovský, J. Fridrich. “Calibration revisited,” In J. D -ittmann, S. Craver, and J. Fridrich, editors, Proceedings of the 11th ACM Multimedia & Security Workshop, pages 63–74,Princeton, NJ, September 7–8, 2009.

[9] Yun Q. Shi, Chunhua Chen, and Wen Chen, “A Markov Process Based Approach to Effective Attacking JPEG Steganography,” 8th International Workshop on Information Hiding**.** JUL 10-12, 2006.

[10] Pevny T, Bas P, Fridrich J, "Steganalysis by Subtractive Pixel Adjacency Matrix," *Information Forensics and Security, IEEE Transactions on* , vol.5, no.2, pp.215-224, June 201

[11] http://bows2.gipsa-lab.inpg.fr/

[12] http://www.agents.cz/boss/BOSSFinal/

[13] A. Sarkar, K. Solanki, and B. S. Manjunath, “Further study on YASS: Steganography based on randomized embedding to resist blind steganalysis,” Proc of the Society of Photo-Optical Instrumentation Engineers (SPIE), vol. 6819, 2008, p. 81917

[14] A. Westfeld, I. S. Moskowitz, Ed, “High capacity despite b-etter steganalysis(F5—A steganographic algorithm),” in *Proc. 4^{th} Int. Workshop Information Hiding*, Pittsburgh, PA, Apr. 25–27, 2001, vol. 2137,Lecture Notes in Computer Science, pp. 289–302.