

IMAGE STEGANALYSIS WITH BINARY SIMILARITY MEASURES

Ismail Avcibas^a, Nasir Memon^b, Bülent Sankur^c

^aDept. of Electronics Eng., Uludag University, Bursa, Turkey.

^bDept. of Comp. and Inf. Science, Polytechnic University, Brooklyn, NY, USA.

^cDept. of Electrical and Electronics Eng., Bogaziçi University, Istanbul, Turkey.

ABSTRACT

We present a novel technique for steganalysis of images that have been subjected to Least Significant Bit (LSB) type steganographic algorithms. The seventh and eight bit planes in an image are used for the computation of several binary similarity measures. The basic idea is that, the correlation between the bit planes as well the binary texture characteristics within the bit planes will differ between a stego-image and a cover-image. These telltale marks can be used to construct a steganalyzer, that is, a multivariate regression scheme to detect the presence of a steganographic message in an image.

1. INTRODUCTION

Steganography refers to the science of “invisible” communication. Unlike cryptography, where the goal is to secure communications from an eavesdropper, steganographic techniques strive to hide the very presence of the message itself from an observer [1]. Given the proliferation of digital images, and given the high degree of redundancy present in a digital representation of an image (despite compression), there has been an increased interest in using digital images for the purpose of steganography. The simplest image steganography techniques essentially embed the message in a subset of the LSB (least significant bit) plane of the image, possibly after encryption [2]. Popular steganographic tools based on LSB-embedding vary in their approach for hiding information. Methods like Steganos and Stools use LSB embedding in the spatial domain, while others like Jsteg embed in the frequency domain. Non-LSB steganography techniques include the use of quantization and dithering [2].

Since the main goal of steganography is to communicate securely in a completely undetectable manner, an adversary should not be able to distinguish in any sense between *cover-objects* (objects not containing any secret message) and *stego-objects* (objects containing a secret message). In this context, *steganalysis* refers to the body of techniques that are

conceived to distinguish between cover-objects and stego-objects.

Recent years have seen many different steganalysis techniques proposed in the literature. Some of the earliest work in this regard was reported by Johnson and Jajodia [3],[4]. They mainly look at palette tables in GIF images and anomalies caused therein by common stego-tools. A more principled approach to LSB steganalysis was presented in [5] by Westfeld and Pfitzmann. They identify Pairs of Values (PoV's), which consist of pixel values that get mapped to one another on LSB flipping. Fridrich, Du and Long [6] define pixels that are close in color intensity to be a difference of not more than one count in any of the three color planes. They then show that the ratio of close colors to the total number of unique colors increases significantly when a new message of a selected length is embedded in a cover image as opposed to when the same message is embedded in a stego-image. A more sophisticated technique that provides remarkable detection accuracy for LSB embedding, even for short messages, was presented by Fridrich et al. in [7]. Avcibas, Memon and Sankur [8] present a general- technique for steganalysis of images that is applicable to a wide variety of embedding techniques including but not limited to LSB embedding. They demonstrate that steganographic schemes leave statistical evidence that can be exploited for detection with the aid of image quality features and multivariate regression analysis. Chandramouli and Memon [9] do a theoretical analysis of LSB steganography and derive a closed form expression of the probability of false detection in terms of the number of bits that are hidden. This leads to the notion of steganographic capacity, that is, the number of bits one can hide in an image using LSB techniques without causing statistically significant modifications.

In this paper, we present a new steganalysis technique for detecting stego-images. The technique uses binary similarity measures between successive bit planes of an image to determine the presence of a hidden message. In comparison to previous work, the technique we present differs as follows:

- [3] and [4] present visual techniques and work for palette images. Our technique is based on statistical analysis and works with any image format.
- [5], [6] and [7] work only with LSB encoding. Our technique aims to detect messages embedded in other bit planes as well.
- [5], [6] and [7] detect messages embedded in the spatial domain. The proposed technique works with both spatial and transform-domain embedding.
- Our technique is more sensitive than [5], [6] and [8]. However, in its current form it is not as accurate as [7] and cannot estimate the length of the embedded message like [7].

Notice that our scheme, like [5,6,7] does not need a reference image for steganalysis. The rest of this paper is organized as follows: In Section 2 we review binary similarity measures. In Section 3 we describe our steganalysis technique. In Section 4 we give simulation results and conclude with a brief discussion in Section 5.

2. BINARY SIMILARITY MEASURES

There are various ways to determine similarity between two binary images. Classical measures are based on the bit-by-bit matching between the corresponding pixels of the two images. Typically, such measures are obtained from the scores based on a contingency table (or matrix of agreement) summed over all the pixels in an image. In this study, where we examine lower order bit-planes of images, for the presence of hidden messages, we have found that it is more relevant to make a comparison based on *binary texture statistics*. Let $x_i = \{x_{i-k}\}$, $k = 1, \dots, K$ and $y_i = \{y_{i-k}\}$, $k = 1, \dots, K$ be the sequences of bits representing the 4-neighborhood pixels, where the index i runs over all the image pixels. Let

$$\chi'_i = \begin{cases} 1 & \text{if } x_i = 0 \text{ and } x_i = 0 \\ 2 & \text{if } x_i = 0 \text{ and } x_i = 1 \\ 3 & \text{if } x_i = 1 \text{ and } x_i = 0 \\ 4 & \text{if } x_i = 1 \text{ and } x_i = 1 \end{cases} \quad (1)$$

Then we can define the agreement variable for the pixel x_i

as: $\alpha'_i = \sum_{j=1}^4 \delta(\chi'_i, j)$, $j = 1, \dots, 4$, $K = 4$, where

$$\delta(m, n) = \begin{cases} 1 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases} \quad (2)$$

The accumulated agreements can be defined as:

$$\begin{aligned} a &= \frac{1}{MN} \sum_i \alpha'_i, & b &= \frac{1}{MN} \sum_i \alpha'_i, \\ c &= \frac{1}{MN} \sum_i \alpha'_i, & d &= \frac{1}{MN} \sum_i \alpha'_i. \end{aligned} \quad (3)$$

These four variables {a,b,c,d} can be interpreted as the one-step co-occurrence values of the binary images. Normalizing the histograms of the agreement scores for the 7th bit-plane can be defined as follows:

$$p_i^j = \sum_i \alpha'_i / \sum_i \sum_j \alpha'_i. \quad (4)$$

Similarly, one can define p_i^j for the 8th bit plane. In addition to these we calculate the Ojala texture measure as follows. For each binary image we obtain a 16-bin histogram based on the weighted neighborhood as shown in Fig. 1, where the score is given by: $S = \sum_{i=0}^3 x_i 2^i$ by weighting the four directional neighbors as in Fig. 1.

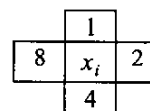


Fig. 1 The weighting of the neighbors in the computation of Ojala score. $S = 4+8=12$ given W , S bits 1 and E , N bits 0.

The resulting Ojala measure is the mutual entropy between the two distributions, that is

$$m_7 = -\sum_{n=1}^N S_n^7 \log S_n^8, \quad (5)$$

where N is the total number of bins in the histogram, S_n^7 is the count of the n 'th histogram bin in the 7th bit plane and S_n^8 is the corresponding one in the 8th plane.

Table 1: Binary Similarity Measures

Similarity Measure	Description
Sokal & Sneath Similarity Measure 1	$m_1 = \frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{b+d} + \frac{d}{c+d}$
Sokal & Sneath Similarity Measure 2	$m_2 = \frac{ad}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
Sokal & Sneath Similarity Measure 3	$m_3 = \frac{2(a+d)}{2(a+d)+b+c}$
Variance Dissimilarity Measure	$m_4 = \frac{b+c}{4(a+b+c+d)}$
Dispersion Similarity Measure	$m_5 = \frac{ad-bc}{(a+b+c+d)^2}$
Co-occurrence Entropy	$dm_6 = \sum_{j=1}^4 p_j^7 \log p_j^8$
Ojala Mutual Entropy	$dm_7 = -\sum_{n=0}^{15} S_n^7 \log S_n^8$

Using the above definitions various binary image similarity measures are defined as shown in Table 1. The measures m_1 to m_5 are obtained for seventh and eighth bits separately by adapting the parameters $\{a,b,c,d\}$ (3) to the classical binary string similarity measures, such as Sokal & Sneath. Then their differences $dm_i = m_i^{7th} - m_i^{8th}$ $i=1,\dots,5$ are used as the final measures. The measure dm_6 is defined as the co-occurrence entropies using the 4-bin histograms of the 7th and 8th bit planes. Finally the measure dm_7 is somewhat different in that we use the neighborhood-weighting mask proposed by Ojala [16]. Thus we obtain a 16-bin histogram for each of the planes and then calculate their mutual entropy.

3. STEGANALYSIS TECHNIQUE BASED ON BINARY MEASURES

Our approach is based on the fact that embedding a message in an image has a telltale effect on the nature of correlation between contiguous bit-planes. Hence we hypothesize that binary similarity measures between bit planes will cluster differently for clean and stego-images. This is the basis of our steganalyzer that aims to classify images as marked and unmarked.

We conjecture that hiding information in any bit plane decreases the correlation between that plane and its contiguous neighbors. For example, for LSB steganography, one expects a decreased similarity between the seventh and the eighth bit planes of the image as compared to its unmarked version. Hence, similarity measures between these two LSB's should yield higher scores in a clean image as compared to a stego-image, as the embedding process destroys the preponderance of bit pair matches.

Since the complex bit pair similarity between bit planes cannot be represented by one measure only, we decided to use several similarity measures to capture different aspects of bit plane correlation. The steganalyzer is based on the regression of the seven similarity measures listed in Table 1:

$$y = \beta_1 m_1 + \beta_2 m_2 + \dots + \beta_q m_q \quad (6)$$

where $\{m_1, m_2, \dots, m_q\}$ are the q similarity scores and $\{\beta_1, \beta_2, \dots, \beta_q\}$ are their regression coefficients. In other words we try to predict the state y , whether the image contains a stego-message ($y = 1$) or not ($y = -1$), based on the bit plane similarity measures. Since we have n observations, we have the set of equations

$$\begin{aligned} y_1 &= \beta_1 m_{11} + \beta_2 m_{12} + \dots + \beta_q m_{1q} + \epsilon_1 \\ y_n &= \beta_1 m_{n1} + \beta_2 m_{n2} + \dots + \beta_q m_{nq} + \epsilon_n \end{aligned} \quad (7)$$

where m_{kr} is the r 'th similarity measure observed in the k 'th test image. The corresponding optimal MMSE linear

predictor $\hat{\mathbf{b}}$ can be obtained by using the matrix M of similarity measures:

$$\hat{\mathbf{b}} = (M^T M)^{-1} (M^T \mathbf{y}). \quad (8)$$

Once prediction coefficients are obtained in the training phase, these coefficients can then be used in the testing phase. Given an image in the test phase, binary measures are computed and using the prediction coefficients, these scores are regressed to the output value. If the output exceeds the threshold 0 then the decision is that the image is embedded, otherwise the decision is that the image is not embedded. That is, using the prediction

$$\hat{y} = \hat{\beta}_1 m_1 + \hat{\beta}_2 m_2 + \dots + \hat{\beta}_q m_q \quad (9)$$

the condition $\hat{y} \geq 0$ implies that the image contains a stego-message, and the condition $\hat{y} < 0$ signifies that it does not.

The above shows how one can design a steganalyzer for the specific case of LSB embedding. The same procedure generalizes quite easily to detect messages in any other bit plane. Furthermore, our initial results indicate that we can even build steganalyzer for non-LSB embedding techniques like the recently designed algorithm F5 [11]. This is because a technique like F5 (and many other robust watermarking techniques which can be used for steganography in an active warden framework [8]) results in the modification of the correlation between bit planes. We note that LSB techniques randomize the last bit plane. On the other hand Jsteg or F5 introduce more correlation between 7th and 8th bit planes, due to compression that filters out the natural noise in a clean image. In other words whereas spatial domain techniques decrease correlation, frequency domain techniques increase it.

4. SIMULATION RESULTS

We have designed a steganalyzer based on a training set and using various image steganographic tools. The steganographic tools were Steganos [12], STools [13] and Jsteg [14], since these were among the most popular and cited tools in the literature. The image database for the simulations was selected from [15] containing a variety of images such as computer generated images, images with bright colors, images with reduced and dark colors, images with textures and fine details like lines and edges, and well-known images like Lena, peppers etc.

In the experiments 12 images were used for training and 10 images for testing. The embedded message size were 1/10 of the cover image size for Steganos and STools, while the message size were 1/100 of the cover image size for Jsteg. The 12 training and 10 test images were embedded with separate algorithms (Steganos, S-

Tools and Jsteg). They were compared against their non-embedded versions in the test and training phases.

The performance of the steganalyzers is given in Table II. In this table we compare two steganalyzers: the one marked *Binary* is the scheme discussed in this paper. The one marked as *IQM* is based on the technique developed in [8]. This technique likewise uses regression analysis, but it is based on several image quality measures (IQM) such as block spectral phase distance, normalized mean square error, angle mean etc. The quality attributes are calculated between the test image and its low-pass filtered version. The steganalyzer scheme denoted as *IQM* [8] is more laborious in the computation of the quality measures and preprocessing.

Table 2: Performance of the Steganalyzer

	False Alarm Rate		Miss Rate		Detection Rate	
	IQM	BSM	IQM	BSM	IQM	BSM
Steganos	2/5	1/5	1/5	1/5	7/10	8/10
Stools	4/10	1/10	1/10	2/10	15/20	17/20
Jsteg	3/10	2/10	3/10	1/10	14/20	17/20
F5		2/10		2/10		16/20

Simulation results indicate that the binary measures form a multidimensional feature space whose points cluster well enough to do a classification of marked and non-marked images and in a manner comparable to the previous technique presented in [8].

5. CONCLUSIONS

In this paper, we have addressed the problem of steganalysis of marked images. We have developed a technique for discriminating between cover-images and stego-images that have been subjected to the LSB type steganographic marking. Our approach is based on the hypothesis that steganographic schemes leave telltale evidence between 7th and 8th bit planes that can be exploited for detection. The steganalyzer has been instrumented with binary image similarity measures and multivariate regression. Simulation results with commercially available steganographic techniques indicate that the new steganalyzer is effective in classifying marked and non-marked images.

As described above, the proposed technique is not suitable for active warden steganography (unlike [8]) where a message is hidden in higher bit depths. But initial results have shown that it can easily generalize for the active warden case by taking deeper bit plane correlations into account. For example, we are able to detect Digimarc when the measures are computed for 3rd and 4th bit planes.

6. REFERENCES

- [1] G. J. Simmons, Prisoners' Problem and the Subliminal Channel (The), CRYPTO83 - Advances in Cryptology, August 22-24, 1984. pp. 51-67.
- [2] N. F. Johnson, S. Katzenbeisser, "A Survey of steganographic techniques", in S. Katzenbeisser and F. Petitcolas (Eds.): *Information Hiding*, pp. 43-78. Artech House, Norwood, MA, 2000.
- [3] N. F. Johnson, S. Jajodia, "Steganalysis: The investigation of Hidden Information", *IEEE Information Technology Conference*, Syracuse, NY, USA, 1998.
- [4] N. F. Johnson, S. Jajodia, "Steganalysis of Images created using current steganography software", in David Aucsmith (Ed.): *Information Hiding*, LNCS 1525, pp. 32-47. Springer-Verlag Berlin Heidelberg 1998.
- [5] A. Westfield, A. Pfizmann, "Attacks on Steganographic Systems", in *Information Hiding*, LNCS 1768, pp. 61-76, Springer-Verlag Heidelberg, 1999.
- [6] J. Fridrich, R. Du, M. Long, "Steganalysis of LSB Encoding in Color Images", Proceedings of ICME 2000, New York City, July 31-August 2, New York, USA
- [7] J. Fridrich, M. Goljan and R. Du, "Reliable Detection of LSB Steganography in Color and Grayscale Images". *Proc. of the ACM Workshop on Multimedia and Security*, Ottawa, CA, October 5, 2001, pp. 27-30.
- [8] I. Avciabas, N. Memon and B. Sankur, "Steganalysis Using Image Quality Metrics", *Security and Watermarking of Multimedia Contents*, SPIE, San Jose, 2001.
- [9] R. Chandramouli and N. Memon, "Analysis of LSB Based Image Steganography Techniques", *Proceedings of the International Conference on Image Processing*, Thessalonica, Greece, October 2001.
- [10] C. Rencher, *Methods of Multivariate Analysis*, New York, John Wiley (1995).
- [11] F5—A Steganographic Algorithm: High Capacity Despite Better Steganalysis. *Information Hiding*. Proceedings, LNCS 2137, Springer-Verlag Berlin 2001
- [12] Steganos II Security Suite, <http://www.steganos.com/english/steganos/download.htm>
- [13] A. Brown, S-Tools Version 4.0, Copyright © 1996, <http://members.tripod.com/steganography/stego/s-tools4.html>.
- [14] J. Korejwa, Jsteg Shell 2.0, <http://www.tiac.net/users/korejwa/steg.htm>.
- [15] http://www.cl.cam.ac.uk/~fapp2/watermarking/benchmark/image_database.html
- [16] T. Ojala, M. Pietikainen, D. Harwood, A Comparative Study of Texture Measures with Classification Based on Feature distributions, *Pattern Recognition*, vol. 29, pp. 51-59.