Conditional Probabilty Based Steganalysis for JPEG Steganography

* **Ainuddin Wahid Abdul Wahab, *Johann A Briffa, *Hans Georg Schaathun and *Anthony TS Ho

*Department of Computing, University of Surrey
**Department of Computer System and Technology, University of Malaya

Abstract. Inspired by works on the Markov process based steganalysis proposed in [12], in this paper, we proposed a new steganalysis technique based on the conditional probability statistics. Specifically we focus on its performance with F5 steganograpy technique. In our experiment, by using stego images with different size of messages embedded for training and test sets of SVM classifier, we proofed that the proposed technique works on F5. With different number of messages embedded, it also can be seen that the performance of steganalyis depends on the message size embedded. This paper includes the introduction to conditional probability features, how the experiment works, and the discussion of the results.

1 Introduction

Steganography allows a user to hide a secret message in such a way that an adversary will not be able to detect the existence of the secret. Steganography can be dated back to 440 BC, where the tale of Demaratus sending a warning by using a wax tablet and Histiaeus using a tattoo on his slave's shaved head were mentioned by Herodotus in The Histories of Herodotus [11].

A steganography system can be considered defeated if an attacker is able to prove the existence of a secret message [9]. If a steganography system fails to disguise the embedded information, there is no point in using it since the aim for having a secretive communication has now been exposed.

Over the last decade a wide range of steganography techniques have appeared in the literature. Similarly, a wide range of steganalysis techniques were also made available, intended to let an adversary determine whether an intercepted image contains a secret message. In particular, a number of steganalysis techniques based on machine learning have also emerged [10][12]. Such techniques tend to be blind, in the sense that it do not assume any particular steganography algorithm and can usually break a variety of algorithms. Other methods that are specific to certain technique of steganography, such as proposed in [7], are categorized as non-blind techniques.

In this paper, the focus is on the steganalysis of JPEG steganography [2], particularly because JPEG is the most popular image format on the Internet. Steganographic images can be shared by employing a sending and receiving process, or just by placing it on a web page for the receiver to browse and extract the message without being noticed. By focusing on steganalysis for JPEG steganography, this research should be of benefit in the general steganalysis research domain and also be of help to the real world implementation of steganalysis, such as in digital forensic investigation.

This paper consists of four sections, and starts with a review on estimated conditional probability features. Following that, Section 2 presents the description of our works. Section 3 contains the findings and finally Section 4 discusses the conclusion of the research.

2 Conditional Probability Features

The revised probability of B when it is known that A has occurred is called the conditional probability of B given A [3] and is defined by the formula

$$P(B \mid A) = \frac{P(AB)}{P(A)}.$$

Figure 1 illustrates $P(A), P(B)$ and $P(AB)$. Based on the concept of conditional probability, the features for our experiment are collected in horizontal, vertical and diagonal directions from JPEG coefficient values (Figure 2). For each direction, p, q, r, x, y and z will traverse throughout the JPEG coefficient (8×8 block) in horizontal, vertical and diagonal directions, accordingly. This new approach is different than the Markov process approach [12], where the statistics are calculated by considering each entity in JPEG coefficient.

Definition 1 *The JPEG coefficient values consists of all the JPEG coefficients which have been quantized with the JPEG quantization table but have not been zigzag scanned, run-length coded and Huffman coded from JPEG encoding process.*

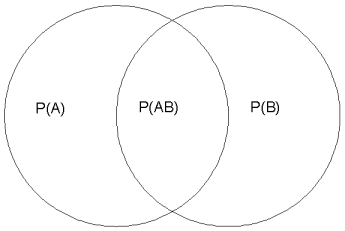


Fig. 1: Venn diagram illustrates $P(A), P(B)$, and $P(AB)$

In the Markov process approach [12], the features are collected by comparing all the values in the JPEG 2-D coefficient array. In this new technique, the statistics are collected in block basis and only certain values from the block were used to generate the statistics. We also exclude the DC coefficient value for each block as what can seen in Figure 2.
For our experiment (Figure 2), we consider six preconditions (event A)

$$A_1 : p < q,\ A_4 : x < y,$$
$$A_2 : p > q,\ A_5 : x > y,$$
$$A_3 : p = q,\ A_6 : x = y.$$

Next, we consider six probabilities (event B)

$$B_1 : r < q,\ B_4 : z < y$$
$$B_2 : r > q,\ B_5 : z > y$$
$$B_3 : r = q,\ B_6 : z = y$$

For three different directions, we calculate 54 statistics (18 statistics for each horizontal, vertical and diagonal directions) values in total

$$X_{i,j} = \widehat{p}(B_i \mid A_j),$$

$$\text{j} = 1,2,3,4,5,6 \text{ and}$$

$$\text{i} = \begin{cases} 1,2,3 & \text{j}\leq 3, \\ 4,5,6 & \text{j}\geq 4. \end{cases}$$

3 Task

To evaluate the performance of the proposed steganalysis algorithm on F5 [2], we used a combination of online database and our own captured images to have a sufficient number of training and testing images. All images were decompress, cropped to the center of 640x480 pixels, and then compressed with a quality factor of 75 in JPEG image format. The cropping technique can help to ensure that the image dimensions is not correlated with spatial characteristics, such as noise or local energy [4]
After the image preparation process, all the images went through the F5 encoding process to produce sets of stego images based on the size of the embedded message. The message is an image with variable file sizes, ranging from 4096 bytes, 1848 bytes, and the smallest is 618 bytes. These contribute to three different sets of stego images as shown in Table 1. With the cover and stego images ready, the proposed technique is conducted to produce the features for the subsequent classification process. The freely available LibSVM [5] was then used as the classifier.
For SVM, the soft margin and γ parameters are determined using parameter selection tool, 'grid.py' that was available from the LibSVM package. Figure 3 shows the result of 'grid.py' (contour of cross-validation accuracy) for our proposed method.

4 Results and Discussions

As reported in [6] and [10], classification accuracy was used to measure the performance of the proposed technique. From Table 1, it can be seen that our proposed technique works on F5 steganography method. While having a classification accuracy of 97.2% for a message size of 4096 bytes, the accuracy increased with the rate of 99.6% for a message size of 1848 bytes and 99.8% for a message size of 618 bytes. Using confidence interval estimation technique [3], we also have computed with 95.0% confidence intervals for the accuracy of the steganalysis technique.

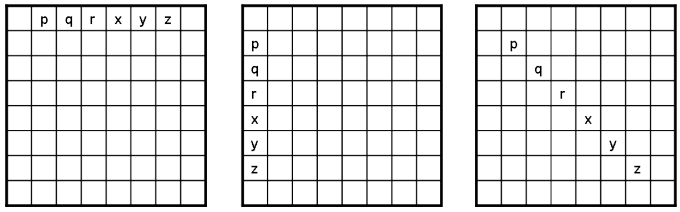
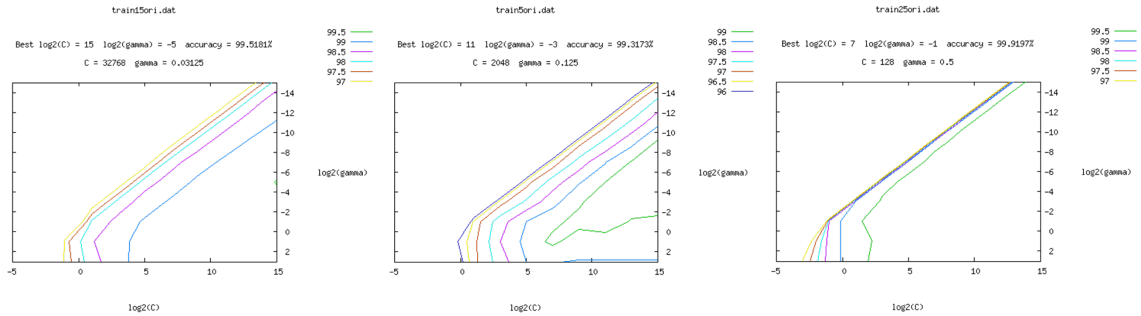


Fig. 2: Conditional probability directions : horizontal, vertical and diagonal



(a) (b) (c)

Fig. 3: Conditional probability based steganalysis classification parameters selection process with different embedded message size; (a)4096 bytes, (b)1848 bytes and (c)618 bytes.

For comparison purposes, we implemented two previously proposed steganalysis methods. The first one is based on the Markov process proposed in [12], and the second technique is based on the higher-order statistic of wavelet component (wavelet decomposition) proposed by Lyu and Farid in [10]. For the second technique, we used the Matlab script available from `http://www.cs.darmouth.edu/~farid`.

Using a total of 5235 images with size of 640×480 pixels and message sizes of 4096, 1848 and 618 bytes, the performance of the three steganalysis approaches were measured (Table 1).

For reference purposes, we have made our own simulation of Lyu-Farid's algorithm, and we note that our results are consistent with those of [1]. Our results also consistent with what published in [12] for Markov process based steganalysis.

In Table 1, for message size of 618 bytes, we note that the 95% confident interval do not overlap, which clearly means that our algorithm is significantly more accurate in this case. However, there is an overlap on confidence interval for message size of 1848 bytes. Although, there is no statistically significant conclusion in this case, this result still shows that our proposed method is more accurate for the predicted accuracy.

Furthermore, it can be seen from Table 2 that our proposed method needs only 1140 CPU-milliseconds compared to 130260 CPU-milliseconds by the Markov process and 10910 CPU-milliseconds by wavelet decomposition techniques for features extraction process of 10 sample images on a standard Dell Optiplex 755 machine with 2.33GHz Intel Core2 Duo processor.

Definition 2 *The CPU-milliseconds is the amount of time the central processing unit (CPU) executes a particular instruction within a computer program.*

There is still another advantage, where our proposed method only used 54 feature vectors compared to 324 features in Markov process [12] and 72 features in wavelet decomposition[10] based steganalysis techniques. This could help to reduce the time needed for training and testing, especially for real world implementation. Using SVMLight [8] as the classification tool, our proposed method was shown to be faster in the SVM training (2480 images) and classification (500 images)

Table 1: Classification accuracy with confidence intervals (optimal soft margin and γ parameters)

	Message Size (bytes)		
	4096	1848	618
Conditional Probability	97.2% (95.3%,99.2%)	99.6% (98.8%,100.0%)	99.8% (99.3%,100.0%)
Markov Model	97.2% (95.3%,99.2%)	97.2% (95.3%,99.2%)	97.2% (95.3%,99.2%)
Wavelet Decomposition	60.0% (54.0%,66.1%)	60.5% (54.4%,66.6%)	61.5% (55.4%,67.6%)

Table 2: Features Extraction, Training and Classification Time (CPU-milliseconds)

	Steganalysis Techniques		
	Conditional Probability	Markov Process	Wavelet Decomposition
Features Extraction (10 images)	1140	130260	10910
Training (2480 images)	770	150	2110
Classification (500 images)	100	100	180

process compared to Farid's technique (Table 2). In addition, the proposed technique works directly on JPEG coefficient values which also means less time spent on the features extraction process.

5 Conclusion

In this study, we have developed a new steganalysis method based on conditional probability statistics. With a hypothesis that F5 steganography technique leaves statistical artifacts on JPEG coefficient values, our steganalysis technique has shown to be able to exploit those artifacts for the detection process. The feature set was obtained by using conditional probability technique that was able to record different patterns in the neighboring pixels in JPEG coefficient values of an image. A total of 54 features for each image are collected and then used for the classification process. From the results, the proposed method works on F5, although the performance is affected with different message sizes. The proposed technique also simplifies the steganalysis process by using a smaller number of features and easier feature extraction process.

References

1. I. Avcibas, M. Kharrazi, N. Memon, and B. Sankur. Image steganalysis with binary similarity measures. *EURASIP Journal on Applied Signal Processing 2005:17*, 2005.
2. A.Westfeld. F5. Software available at `http://www.inf.tu-dresden.de/~westfeld/f5`.
3. G. K. Bhattacharyya and R. A. Johnson. *Statistical Concepts and Methods*. Wiley, 1977.
4. R. Böhme. Assessment of steganalytic methods using multiple regression models. In *Information Hiding*, pages 278–295, 2005.
5. C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.
6. H. Farid. Detecting hidden messages using higher-order statistical models. In *International Conference on Image Processing*, Rochester, NY, 2002.
7. J. Fridrich, M. Goljan, and D.Hogea. Steganalysis of jpeg images: breaking the f5 algorithm. *in Proc. 5th International Workshop on Information Hiding(IH '02)*, pages 310–323, October 2002.
8. T. Joachims. *SVMLight Support Vector Machine, version 6.01*, 2004. Software available at `http://svmlight.joachims.org/`.
9. S. Katzenbeisser and A. P. Fabien. *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House, 2000.
10. S. Lyu and H. Farid. Detecting hidden message using higher-order statistics and support vector machines, 2002.
11. F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn. Information hiding: A survey. *Proceedings of the IEEE, special issue on protection of multimedia content, 87(7)*, pages 1062–1078, July, 1999.
12. Y. Q. Shi, C. Chen, and W. Chen. A markov process based approach to effective attacking jpeg steganography. In *Information Hiding*, pages 249–264, 2006.