Chapter 1

STEGANOGRAPHY DETECTION USING MULTI-CLASS CLASSIFICATION

Benjamin M. Rodriguez and Gilbert L. Peterson

Abstract Several steganography tools are freely available over the Internet, ranging from straight forward least significant embedding to complex transform domain encrypted algorithms within digital images. Given the number of tools available, the digital forensics investigator must not only determine that an image contains embedded hidden information but the method used during the embedding process, also know as detecting a stego fingerprint. The determination of the embedding method is the first step in extracting the hidden information. This paper focuses on identifying the stego fingerprint within jpeg images. The classes of methods targeted are F5, JSteg, Model Based, OutGuess, and StegHide. Each of these embedding methods presents different challenges when attempting to extract the hidden information because each of them embed data in dramatically different ways. The embedding methods are separated by using features developed from sets of stego images developed with the mentioned embedding methods as training data for a multi-class support vector machine classifier. For new images, the image features are calculated and evaluated based on their associated label to the most similar class, i.e. clean or embedding method feature space. Results from the SVM demonstrate that, in the worst case scenario, it is possible to separate the embedding methods by 87.0%.

Keywords: Steganalysis, Multi-class Classification, Support Vector Machine

1. Introduction

Steganography is a data hiding and transmission approach which strives to conceal and prevent detection of the true content of a message. The steganography process makes use of a cover object, which in this paper is an image, to conceal the stego data, or hidden message. The steganographic process uses an embedding procedure which given a cover image and the stego data produces a stego image, which is an

image containing a hidden message. To counter unethical steganography practices steganalysis is used. Steganalysis examines a set of cover objects to determine if the presence of steganography exists, determine a potential fingerprint of the embedding procedure used, and potentially extract the embedded content.

In the area of detecting hidden information in images, several researchers have introduced various detection methods. However, for several of these detection methods to be truly effective, the embedding method used must be known. This type of steganography fingerprinting is an issue of concern, given that there are over 250 steganography programs available. In order to combat this situation, methods of detection that use a combination of features and identify the class or type of embedding method used are needed.

In this paper a multi-class classification method is presented which focuses on the classification of unseen instances to their specific embedding method (class). The proposed detection method classifies jpg stego images based on feature classification in which instances are associated with exactly one element of the label set. The multi-level energy band features presented in this paper are used with the multi-class Support Vector Machine (SVM) classification technique. The features are generated from higher order statistics of the discrete cosine transforms' (DCT) multi-level energy bands.

The testing results presented are based on an image database of 1000 high quality jpg images taken with a Nikon Coolpix 5. The stego images are created using five embedding methods which include F5, JSteg, Model Based Embedding, Outguess and StegHide. Each of these steganography tools embeds data using different embedding techniques with the exception of OutGuess and StegHide which embed similarly but use a different randomization technique.

In Section 1.2, various embedding methods and multi-class classifiers are covered. In Section 1.3, the multi-level energy feature generation technique is described. This is followed by a description of the multi-class SVM classification method in Section 1.4, and the results of the SVM classifier using the multi-level energy features are presented in Section 5. Results reveal that it is possible for the presented algorithm to identify the various embedding methods. This paper ends with the discussion of future work and concluding remarks.

2. Related Work

Each class of embedding method leaves a fingerprint on the stego image representative of the technique used to create the stego image. As

a step toward exact tool identification and information extraction, multi-class classifiers are used to detect specific embedding methods using this stego fingerprint. This section discusses two related topics, five jpg image data embedding methods which are commonly available over the Internet, and different forms of developing multi-class classification methods.

2.1 Embedding Methods

Image data embedding methods which are commonly available over the Internet are described [9]. One of the primary reasons images are used for the stego data is due to the number of redundant portions within a digital image that can be altered without affecting the image quality as observed by the human eye. In this work, five tools for embedding hidden information into jpg images are of interest F5, JSteg, Model Based, OutGuess, and StegHide.

The jpeg image format is defined by and named after the Joint Photographic Experts Group, is currently the most prevalent image storage format in use today [9]. The vast number of jpg images on the Internet makes them ideal cover images for hiding secrete data and transmitting them as stego images. The discrete cosine transform (DCT) coefficients are given by $F(u, v)$ of an 8 by 8 block from image pixels $f(x, y)$. The DCT block is followed by the division of the quantization matrix which quantizes the coefficients for compression. After this process most jpeg embedding methods use the least-significant bits (LSB) of the quantized DCT coefficients. The embedding uses redundant bits in which to embed the hidden message having no effect on the binary encoder. While the embedding has no effect on the compression process the modifications of a single DCT coefficient affects all 64 pixels in the image 8 by 8 block.

F5 [10] was developed as a challenge to the steganalysis community. This method takes advantage of the jpg compression algorithm by decrementing the DCT coefficients absolute value in a process known as matrix encoding. An estimated embedding capacity is computed which is based on the total number of DCT coefficients. A recursive algorithm is used in this method which is repeated until the hash function matches the bits of the message, or until one of the coefficients is reduced to zero. The identifiable characteristics are that the embedding procedure leaves unnatural coefficient histograms after embedding.

Model based [11] embedding fits the coefficient histogram into an exponential model with the use of maximum likelihood. This method provides simple solutions to previous embedding methods faults, such as how large a message can be hidden without risking detection by cer-

tain statistical methods, and how to achieve this maximum capacity. Model Based embedding is accomplished by identifying the ideal embedding structure based on the statistical model of the discrete cosine coefficients of the original cover image, and then during embedding make sure that this statistical model is unchanged. The embedding technique is similar to F5, yet the algorithm is attempting to not leave unnatural histogram frequency of adjacent DCT coefficients. The identification of this embedding method requires the combination of several higher order statistics.

JSteg embedding encodes messages into jpg images by means of manipulating the least significant bit of the quantified DCT coefficients. The message is formatted such that the first five bits of the frequency band coefficient indicate the length of the band, indicating the size of the embedded message also known as the capacity of the block. The next set of bits indicates the bit length of the actual message. The message length indication scheme avoids generating large numbers of zeros that occur when short messages are embedded using a fixed bit length to indicate the size of the message [12]. This type of embedding procedure does not spread out the encoded bits among the 14 coefficients; it is identifiable by first order statistics such as mean.

OutGuess [13] modifies the LSB of the DCT coefficients by statistically checking the original image DCT heuristics against the embedded image and manipulates nearby DCT blocks to maintain the original DCT histogram. OutGuess is designed to avoid detection of statistical steganalysis such as the Chi-Square statistical attack. This embedding procedure avoids simple statistical detection by selecting coefficients (AC coefficient $F(u, v) \notin [0, 1]$) with a pseudo-random number generator. This method of statistical correction embeds the hidden data within the least significant bit of the coefficients and offsetting nearby LSB coefficients with minor bit changes to try and prevent Chi-Square statistical image changes.

StegHide [14] is a steganography program that hides data in multiple types of image and audio files. In regards to jpg images, the color representation sample frequencies are not changed which make this method robust to first-order statistical tests. This robustness is a direct correlation of embedding stego data within the LSB of coefficients which contain large variations between adjacent coefficients of the discrete cosine transform. However, the fingerprint left by this embedding method is detectable by higher order statistics such as energy.

Identification of embedding methods is essential in attempts to extract the hidden information. These methods embed data into the rounded DCT coefficients of the jpg file format. The DCT encoding introduces

statistical irregularities that are used to detect the presence of hidden information. The variations of each of the embedding methods leave a signature of how the jpg image was altered. The problem is how to classify the numerous signatures left by the steganalysis tool. In the next section multi-class classification solutions are presented.

2.1.1 Multi-Class Classification. There are approximately 250 tools available for performing steganography on digital images. In order to classify an embedding algorithm, multi-class classification is used to achieve the signature identification of the targeted method. Several multi-class classification solutions have been presented by researchers. The problem is in determining the best classification method to use based on the problem at hand. In this section two solutions are presented.

In several multi-class classification methods two-class classifiers are combined by using posterior probabilities of the binary classifiers. Learning architectures then combine the results of the two-class classifiers to create a multi-class classifier. The learning architectures use hyperplanes which results in boundaries depending on the margin achieved at the nodes separating the space and not the dimension of the space. These types of algorithms operate in a kernel-induced feature space and use two-class maximal margin hyperplanes at each decision-node. This can be achieved by combining simple two-class classification methods using voting and combinations of approximate posterior probabilities. One of the key concepts of this method is that it uses each binary or two-class classifier output in the creation of the multi-class classifier. This method is enhanced with a simple procedure that estimates the posterior probabilities in order to avoid ties and inconsequent in class labeling [2]. Another approach is to use combinations of binary classification methods with Bayes classifier by generalizing the multiple classes. This generalization is done in order to preserve the properties of Bayes classifiers [1].

Several multi-class SVM classifiers are built on a winner-take-all approach which associates available classes from sets of exemplars [3]. This type of approach extends the multi-class SVM to multiple prototypes per class based on a multi-prototype SVM. The method allows the combination of several vectors to obtain a large decision boundary with the use of defined functions. Using a compact constrained quadratic formulation, a greedy optimization algorithm is developed which is able to find locally optimal solutions for non convex objective function. This is a winner take all strategy over a set of linear functions that every

exemplar provides an ordering of the classes where the "winner" is the first class in this ordering.

In this research a majority vote strategy is used. In order to perform classification in any domain, a suitable set of features are required. The features that are used in performing the steganalysis multi-class classification problem are described in the following section.

3. Features

This section introduces the Discrete Cosine Transform (DCT) Multi Level Energy Bands method for calculating the transform domain features from a jpeg image. The features are derived from calculating the DCT energy bands for each block of 8 by 8 coefficients.

The transform domain features presented focus on the energy bands of the DCT coefficients. Figure 1b shows the representation of the energy bands after the DCT. The DCT used in jpg compression does not generate the multilevel energy bands that wavelet decomposition creates. And the multilevel energy band representation, Figure 1b, does not allow for the energy levels to be extracted based on the edges of the original image as shown in Figure 1c. In order to properly extract the various energy bands the DCT transform is rearranged in a wavelet decomposition structure. This structure is created by using 8 by 8 pixel blocks, which are the same blocks used during jpg compression. For each 8x8 block the DCT energy band decomposition of vertical, diagonal and horizontal edges are formed with the use of zig-zag, Figure 1d, and peano scans, Figure 1e. Rearranging the coefficients of the DCT splits the frequency spectrum into uniform spaced bands containing vertical, horizontal and diagonal energy. The ideal representation of the energy bands is shown in Figure 1f. An example of the inverse DCT after the multi level energy bands construction is shown in Figure 1f.

The presented structure captures the energy better than the normal DCT, and as well as some commonly used wavelet decompositions used in image processing. The transformed coefficients are matched to higher level linear predicted neighboring coefficients, which result in an unstructured (non-Gaussian) distribution. To measure the coefficients, higher-order statistics are applicable when dealing with non-Gaussian processes. The features are gathered from higher-order statistics and predicted log errors. Feature parameters are calculated from the symmetric restructured 8 by 8 block as shown in Figure 1e. The result is significantly better energy concentration and similar properties of jpg compression.

For example, the features are calculated from the matrix of neighboring coefficients selected using the predictor pattern for the direction

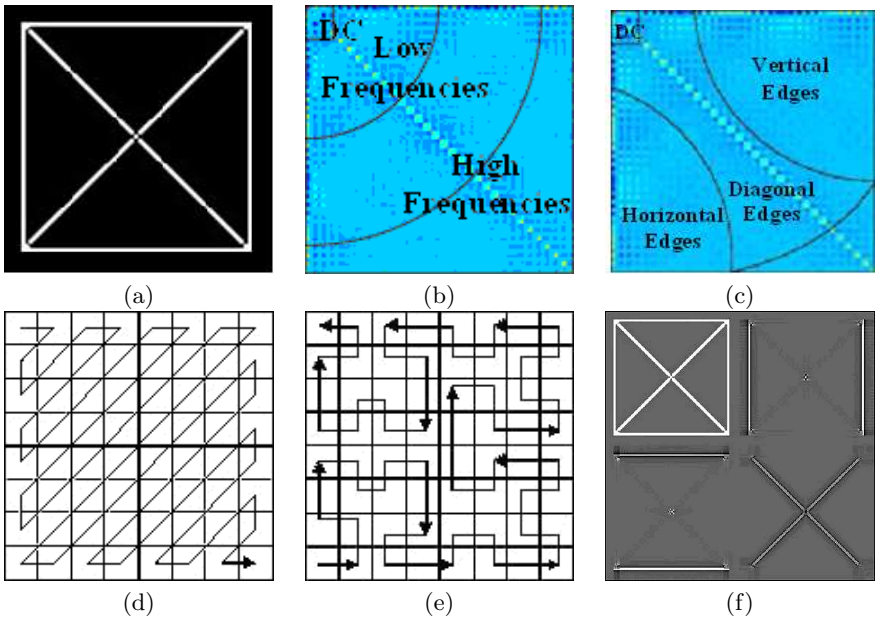


Figure 1. DCT Multi Level Energy Bands a) Input Image, b) Energy Band Representation, c) Extracted Edges, d) Vector w/Zig-Zag, e) Peano Scan Matrix, f) Level1 Representation.)

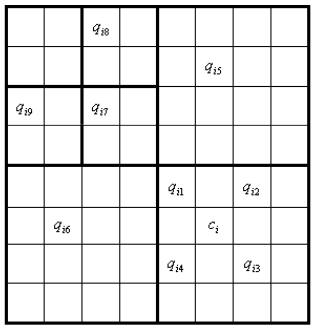


Figure 2. Target Coefficients.

(edges, i.e. vertical, horizontal, and diagonal) under consideration. The coefficient c is used as the target position to predict neighboring coefficients represented by q. In Figure 2, the predictors are represented as neighboring coefficients. Similar methods have been used for pattern recognition [15] and detection [16, 17] with the wavelet structure.

The features that are calculated on a jpg image using the DCT Multi Level Energy Bands are then used in order to differentiate the class of

embedding. The difference in [16] are the coefficients and in [17] wavelets are used. In [18] Fridrich uses features which are specifically developed for separating the embedding methods, e.g. features generated after recompression of F5 and Outguess, giving her good results. The features developed in this paper are for general detection of all jpeg images which make them applicable to the more general anomalous detection case as well. The multi-class classification technique of the support vector machine is discussed in the following section.

4. Multi-Class Classification

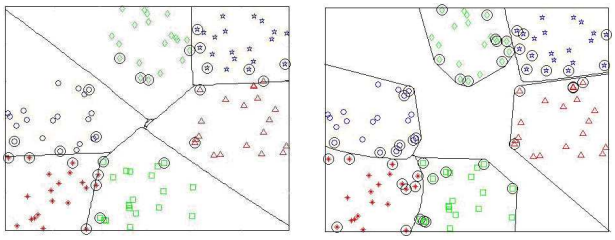
Classification can be used in various forms based on the number of classes being used for training, i.e. single-class, two-class or multi-class classification. In this section, multi-class classification methods are of interest as the digital forensics investigator needs a tool which given an image will provide a single answer as to what type of steganalysis exists in the image. The multi-class classification task requires training and testing over a set of instances (data set) which contains three or more training classes. Each instance includes the values of the features and the class to which it belongs. The multi-class classification objective is for the algorithm to learn a decision model which separates the instances into their classes so that the degree of association is strong between the instances with similar features and the same class and weak between members of different classes. Using the multi-level energy band jpg features each image is classified as belonging to a class in which the degree of association is strongest.

Multi-class classifiers are built on a winner-take-all set of class labels each representing one of the available classes. The classification method used in this paper is a multi-class Support Vector Machine (SVM) classifier.

4.1 SVM Multi-Class

The main idea behind the SVM is to separate classes with a hypersurface that maximizes the margin between each of the classes. Traditionally, SVMs have been shown using two class problems or a binary classification [4].

However, many real-world problems have more than two classes, and there are several approaches to solving existing problems with multi-class classification. Among theses solutions are a) the one versus all approach which uses binary classifiers to encode and train the output labels; b) the one versus one approach which uses a multi-class rule based on the majority wins approach; and c) the training of two-class



(a) (b)

Figure 3. Multi-Class SVM a) SVM w/No Hyperplane b) SVM w/Hyperplane.

classifiers and using voting and combinations of approximate posterior probabilities. Another approach to multi-class SVM classification is to train multi-class kernel-based predictors which use a compact quadratic optimization solution [8]. In this paper a majority vote strategy is used.

The following figures show a multi-class SVM with the support vectors encapsulated in circles and the decision boundaries between the classes. Classification in a SVM is performed by placing the classifying hyperplane, which separates the classes, in the center of the margin of separation. The margin of separation is calculated by locating the training points, x_i, which are closest to the opposing class, and result in the largest margin of separation. Under this condition, the decision surface is referred to as the optimal hyperplane [6]. The maximally separating hyperplane is shown in Figure 3a. By using the maximization of the margin of separation, as the classification mechanism, result shown in Figure 3b, results in fewer false positives for each of the classes but increases the anomalies when the data is unknown.

For the presented classification problem there are several aspects of detecting hidden information, i.e., amount of stego, the embedding method, and the compression scaling factor. This leads to a complex multi-class classifier. The basic concept to generate a multi-class classifier, a set of binary classifiers $f^1, ...f^M$ are constructed, each trained to separate one class from the rest, and combined by using a majority vote wins [6]. For the SVM the multi-class generalization involves a set of discriminant functions designed according to the maximal output. This is a majority voting strategy used to implement the multi-class classifier [5]:

$$f(x) = \arg \max_{j=1,...,M} g^j(x) \tag{1}$$

where $g^j(x) = \sum y_i \alpha_i^j K(x, x_i) + b^j$. The classification y_i provides the sign of the coefficients of x_i. The weight values α_i^j are proportional the number of times the misclassification of x_i causes the weights to update.

The kernel used is a radial basis kernel (RBF) given by $K(x, x_i)$. The bias vector is b^j. The values $g^j(x)$ can be used for the reject decisions.

The kernel function is used to classify input data sets, which were not linearly separable, in the feature space into the kernel feature space and perform the linear classification there. In general, the kernel function is directly defined, and implicitly defines the feature space. By defining the kernel function, the complexity of non-linear class separation is avoided not only in the computation of inner product, but also in the design of the learning machine. In this paper the training vectors x_i are mapped into a higher dimensional space by the mapping function ϕ. The kernel function used in this paper is a Gaussian radial basis function kernel $< \phi(x)\dot{\phi}(x_i) >= K(x, x_i) = e^{|x-x_i|^2/\sigma^2}$[7].

In order to see this, consider the difference between the two largest $g^j(x)$ as a measure of confidence in the classification of x. If that measure falls short of a threshold, the classifier rejects the pattern and does not assign it to a class (anomalies). This has the consequence that on the remaining patterns, a lower error rate can be achieved.

5. Results

In this section the results are based on data sets from the five embedding methods used (F5, JSteg, Model Based, OutGuess, and StegHide), and the clean data set. These images consist of a mixture of 1000 images (512 by 512 RGB *.jpeg) which have clean images sets and images embedded with the six methods mentioned in the related work section. The amount of hidden information embedded within each of the files was 4000 character which is equivalent to one page of text. The percentage of altered coefficients varies with the embedding method. The numbers of features which are used to represent the images are reduced from 120 to 40 features by eliminating features with similar correlation. These results show that the embedding type can be identified by training with a small data set (80% of the images) and tested with the remaining images.

Table 1 shows the confusion matrix from the classification of the clean images sets and five embedding method sets. As can be seen in the matrix, the clean set is clearly separable from the remaining feature sets (clean column and clean row). In this multi-class classification OutGuess and StegHide had the largest number of exemplars that are misclassified as each other. While these two methods avoid statistical methods such as the Chi-Square statistical test they are vulnerable to higher ordered statistics and transforms. These statistics such as inertial, also known as contrast, "compresses" the energy bands of the DCT when no mod-

Table 1. Image class classified with six-class SVM classification.

	Actual					
Predicted	Clean	F5	MB	OutGuess	JSteg	StegHide
Clean	90.2 ± 4.5	3.4 ± 2.0	4.9 ± 2.2	1.4 ± 1.6	0.1 ± 0.0	0.0 ± 0.0
F5	4.2 ± 1.5	83.0 ± 5.4	6.7 ± 3.2	4.8 ± 1.1	0.2 ± 0.0	1.1 ± 0.9
MB	3.6 ± 2.3	16.8 ± 5.2	75.1 ± 9.1	2.4 ± 1.2	0.1 ± 0.0	2.0 ± 1.3
OutGuess	0.4 ± 0.01	1.4 ± 1.6	0.4 ± 0.2	52.3 ± 12.2	6.6 ± 2.9	38.9 ± 7.6
JSteg	1.0 ± 0.5	3.4 ± 1.6	2.2 ± 2.0	6.8 ± 3.8	82.2 ± 5.8	4.4 ± 3.0
StegHide	0.6 ± 0.0	1.2 ± 0.7	1.7 ± 1.8	40.0 ± 7.0	7.1 ± 2.8	49.4 ± 10.9

Table 2. Image class classified with four-class SVM classification.

	Actual			
Predicted	Clean	F5 & MB	OutGuess & StegHide	JSteg
Clean	94.8 ± 3.3	2.4 ± 1.7	1.5 ± 0.4	1.3 ± 0.8
F5 & MB	4.5 ± 2.9	87.0 ± 7.6	6.5 ± 2.6	2.0 ± 1.8
OutGuess & StegHide	3.2 ± 0.9	3.6 ± 2.0	90.7 ± 3.8	2.5 ± 2.2
JSteg	0.0 ± 0.0	4.0 ± 1.7	6.4 ± 2.4	89.6 ± 6.7

ifications to the coefficients is present and "expands" the energy bands when modifications have been made to the coefficients. The next set of embedding methods that had mixed classification results were F5 and Model Based embedding. The methods OutGuess and StegHide were combined along with F5 and Model Based to create a four-class classification. Unlike OutGuess and StegHide, F5 and Model Based embedding avoid detection with sophisticated statistical measure when embedding within the DCT coefficients. Features for F5 and OutGuess are not affected by the re-compression of the embedding. The statistical measures of inertial, energy and entropy each show prominent features in the diagonal, vertical and horizontal energy bands respectively. These findings show the importance of separating the energy bands into the edge components and measuring each energy band with various statistics.

The similarities in embedding method resulted in performances of the multi-class classifier that did not identify the embedding method as shown in Table 1. By identifying similarities in embedding procedures the methods could be identified with higher classification accuracy. As shown in Table 2 the combination of OutGuess and StegHide resulted in a classification accuracy of 90.7% from roughly 50%. This allows the identification of the two embedding method. While the results from Table 1 to Table 2 for F5 and Model Based embedding are not as substantial as OutGuess and StegHide, an increase in classification accuracy is achieved and does allow the methods to be separated from the other embedding methods.

6. Conclusion

In this paper a novel approach was presented for building a steganographic detection method with the use of multi-class SVM classifier and features built from the energy bands of the DCT. What has been shown is that the fingerprinting of the embedding methods is possible. The presented method also showed that methods which have similarities in embedding are classified as the same. Combining the methods that have similarities in the embedding process improved the detection accuracy.

For future research, after the two embedding methods have been identified as either OutGuess or StegHide, a two-class SVM classifier will be used to properly identify the exact embedding fingerprints. Similar research is being done with F5 and Model Based embedding. While multi-class SVM has provided acceptable results, other classification methods are being investigated which are common in multi-class classification. Some of these classification methods include Expectation Maximization (EM) with Bayes Classifier, k-Nearest Neighbor (k-NN) and Parzen-window. These other investigated methods can also be combined to train on specific methods such as OutGuess and StegHide to improve the fingerprinting of these methods as well as F5 and Model Based embedding.

7. Acknowledgments

This research was partially funded by the US Air Force Research Laboratory, Information Directorate/Multi-Sensor Exploitation Branch, Rome New York. The views expressed in this article are those of the authors and do not reflect the official policy or position of the Air Force, Department of Defense or the U.S. Government.

References

[1] A. Tewari and P. L. Bartlett, On the consistency of multiclass classification methods, *Proceedings of the 18th Annual Conference on Learning Theory*, volume 3559, pages 143-157. Springer, 2005.

[2] J. C. Platt, N. Cristianini and J. Shawe-Taylor, Large Margin DAGs for Multiclass Classification, *Advances in Neural Information Processing Systems 12*, pp. 547-553, MIT Press, 2000.

[3] S. Har-Peled, D. Roth and D. Zimak, *Constraint Classification for Multiclass Classification and Ranking*, S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2003.

[4] B. Scholkopf, A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*, The MIT Press, 2002.

[5] N. Cristianini, J. Shawe-Taylor, *An Introduction to Support Vector Machines and other kernel-based learning methods*, Cambridge University Press, 2000.

[6] C. Burgers, A tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery

[7] C.-W. Hsu, C.-C. Chang, C.-J. Lin. A practical guide to support vector classification, http://www.csie.ntu.edu.tw/ cjlin/papers/guide/guide.pdf

[8] J. Fridrich, G. Miroslav, and H. Dorin, New methodology for breaking steganographic techniques for JPEGs, *Proc. EI SPIE*, Santa Clara, CA, pp. 143-155, Jan 2003

[9] StegoArchive.com, http://www.stegoarchive.com/

[10] A. Westfeld, High Capacity Despite Better Steganalysis (F5-A Steganographic Algorithm), *Information Hiding, 4th International Workshop. Lecture Notes in Computer Science*, eds. Moskowitz, I.S. Vol.2137. Springer-Verlag, Berlin Heidelberg New York (2001) 289-302

[11] P. Sallee, Model-based steganography, *International Workshop on Digital Watermarking*, Seoul, Korea, 2003.

[12] T. Lane, P. Gladstone, L. Ortiz, L. Crocker, G. Weijers, and other members of the Independent JPEG Group, JSteg, available, http://www.stegoarchive.com/

[13] N. Provos, OutGuess. http://www.outguess.org/

[14] S. Hetzl, StegHide. http://steghide.sourceforge.net/

[15] R .W. Buccigrossi and E.P.Simoncelli. Image compression via joint statistical characterization in the wavelet domain, *IEEE Tran on: Image Processing*, 8 (12):1688-1701.1999.

[16] S.Agaian, Color Wavelet Based Universal Blind Steganalysis, *The 2004 International Workshop on Spectral Methods and Multirate Signal Processing, SMMSP*, 2004.

[17] S. Lyu and H. Farid, Steganalysis using color wavelet statistics and one-class support vector machines, *SPIE Symposium on Electronic Imaging*, San Jose, CA, 2004.

[18] J. Fridrich and T. Pevny, Determining the Stego Algorithm for JPEG Images, *Proc. SPIE Electronic Imaging, Photonics West*, January 2006.