4-2007

Steganalysis Feature Improvement Using Expectation Maximization

Benjamin M. Rodriguez
Air Force Institute of Technology

Gilbert L. Peterson
Air Force Institute of Technology

Sos S. Agaian
University of Texas at San Antonio

Recommended Citation

Benjamin M. Rodriguez, Gilbert L. Peterson, and Sos S. Agaian "Steganalysis feature improvement using expectation maximization", Proc. SPIE 6575, Visual Information Processing XVI, 657506 (30 April 2007); https://doi.org/10.1117/12.720794

Steganalysis Feature Improvement using Expectation Maximization

Benjamin M. Rodriguez*[a], Gilbert L. Peterson[a], Sos S. Agaian[b]
[a]Department of Electrical and Computer Engineering
Graduate School of Engineering and Management, Air Force Institute of Technology
[b]Multimedia and Mobile Signal Processing Laboratory
The University of Texas at San Antonio, Department of Electrical and Computer Engineering

ABSTRACT

Images and data files provide an excellent opportunity for concealing illegal or clandestine material. Currently, there are over 250 different tools which embed data into an image without causing noticeable changes to the image. From a forensics perspective, when a system is confiscated or an image of a system is generated the investigator needs a tool that can scan and accurately identify files suspected of containing malicious information. The identification process is termed the steganalysis problem which focuses on both blind identification, in which only normal images are available for training, and multi-class identification, in which both the clean and stego images at several embedding rates are available for training. In this paper an investigation of a clustering and classification technique (Expectation Maximization with mixture models) is used to determine if a digital image contains hidden information. The steganalysis problem is for both anomaly detection and multi-class detection. The various clusters represent clean images and stego images with between 1% and 10% embedding percentage. Based on the results it is concluded that the EM classification technique is highly suitable for both blind detection and the multi-class problem.

Keywords: Steganography, Steganalysis, Feature Extraction, Expectation Maximization, Mixture Models, Bayes Classifier

1. INTRODUCTION

The least significant bit (LSB) insertion method is probably the most well known digital image steganography technique in practice [1]. In this method the least significant bits of each pixel in the cover image are replaced with one bit from the secret message. Because the LSB can only contain zeros and ones, approximately half the bits do not need to be altered in order to embed the data from the secret message.

In this paper the presence of steganography is conducted using two spatial domain feature extraction methods based upon simple first order statistical characteristics of the image. One of the feature extraction methods is RS Steganalysis which was developed by Fridrich *et al*. [2]. This method focuses on detection of least significant bit embedding within digital images and determines the amount of steganographic content (message length) with the use of a polynomial "classifier." For the purpose of this paper the method was modified to generate features which separate the feature space between clean images and stego embedded images. The second spatial domain feature extraction method focuses on the development of weighted multi pixel comparison within digital images which is used to generate features to determine if steganographic content exists or not [5]. By using the features from RS Steganalysis and weighted multi pixel comparison during classification, an improved LSB steganography detector is created.

The classification method used in this paper is Expectation Maximization (EM) which is typically used as a clustering method with a Gaussian probabilistic model [3]. The focus of EM algorithm is that given unknown data values the distribution from samples can be used to determine an estimate for the posterior probability of belonging to a given cluster. This method can be considered as two maximum likelihood approaches used for mixture analysis. With the various orientations, shapes and sizes that data clusters represent in feature space the distribution is improved with the use of mixed models which represent the distribution. The first is the mixture approach which maximizes the likelihood over the mixture parameters. The second is by classification based on maximizing the likelihood over the mixture parameters and identifying the class of the mixture component origin for each exemplar. The classification is improved with the use of a Bayes classification technique which uses the posterior probability, covariance matrix, and the mean of the distribution representing the cluster [4].

In the next section, the LSB features used for classification are described for the two feature extraction methods used. Section 3 presents the classification algorithm details. The classification algorithm is used to determine if steganographic content exists within digital images, and also the percentage of data hidden in the image. The classification method is a Gaussian mixture model used with Bayesian classifier. In Section 4 the findings and analysis are shown for each of the detection methods. This is followed by a discussion on future improvements and other potential analysis.

2. FEATURE EXTRACTION

In this section two methods are introduced which overcome the difficulties encountered when generating features for detecting least significant bit modifications. The first method alters the output of RS Steganalysis which instead of estimating the amount of steganographic content within the LSB of a digital images, provides features for use with a classifier to separate clean images from stego images. The second method focuses on features which emphasize minor changes within the LSB of a digital image using weights when measuring pixel variations among adjacent pixels.

2.1 RS Steganalysis Feature Extraction

RS Steganalysis focuses on the detection of least significant bit embedding within digital images [2]. This method is based on the statistics of sample pairs rather than individual samples which are sensitive to least significant bit embedding. In [2], Fridrich *et al*. states that: 1) the Pairs of Values method provides reliable results when the message placement is known, and 2) only randomly scattered messages can be detected with this method when the message length becomes comparable with the number of pixels in the image.

We use the terminology and definitions presented by Fridrich *et al***.** [2], the input cover image is of size M x N pixels with pixel values from the set P, for an 8-bit color layer, $P = \{0,\ldots, 255\}$.

The image is divided into 2 x 2 blocks and mapped into a vector,

$$\begin{bmatrix} x_{i,j} & x_{i,j+1} \\ x_{i+1,j} & x_{i+1,j+1} \end{bmatrix} \Rightarrow \begin{bmatrix} x_{i,j} & x_{i,j+1} & x_{i+1,j} & x_{i+1,j+1} \end{bmatrix} \Rightarrow \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}.$$

By mapping the 2 x 2 block into a one dimension vector, statistical measurements taken identify changes before and after the least significant bit embedding.

For example the flipping operation can be applied to a set of pixels from image I denoted as $G=[x_1,\ldots,x_n]$ and M is a mask $M=[m_1,\ldots m_n]$, where $m_i \in \{-2, -1, 0, 1, 2\}$ for $i = 1,\ldots, n$. It can be shown that the following operation:

$$F(G) = G \oplus M = [x_1 \oplus m_1, ..., x_n \oplus m_n] \tag{1}$$

where $\oplus$, is a modulus 2 operator on least significant bit of m_i and x_i, i=1,…,n, is the flipping operation.

The flipped group $F(G)$ is the set

$$F_M(G) = \left\{ F_{M(1)}(G) \quad F_{M(2)}(G) \quad \cdots \quad F_{M(n)}(G) \right\} \tag{2}$$

which specifies where and how pixel values are to be modified.

The procedure of applying the masks M to G may be shown as a continuous mapping operation:

$$\begin{bmatrix} x_{i,j} & x_{i,j+1} \\ x_{i+1,j} & x_{i+1,j+1} \end{bmatrix} \Rightarrow \begin{bmatrix} x_{i,j} & x_{i,j+1} & x_{i+1,j} & x_{i+1,j+1} \end{bmatrix} = G \Rightarrow F_{M_k}(G) = \begin{cases} F_{M_1}(G) \\ F_{M_2}(G) \\ \vdots \\ F_{M_n}(G) \end{cases} \tag{3}$$

Next, we classify the set $F_M(G)=\{F_{M_1}(G)\ F_{M_2}(G)\ \dots\ F_{M_n}(G)\}$ (the example case $F_{M_4}(G)$) into three groups smooth (singular), regular and unusable with the use of the so called discrimination function f

$$f\left(F_M(G)\right) = f\left(x_1, x_2, ..., x_n\right) = \sum_{i=1}^{n-1} \left|x_{i+1} - x_i\right|, \quad f \in \mathbb{R} \tag{4}$$

As a rule:

Regular group: $G \in R \Leftrightarrow f\left(F_M(G)\right) > f(G)$

Singular group: $G \in S \Leftrightarrow f\left(F_M(G)\right) < f(G)$

Unusable group: $G \in U \Leftrightarrow f\left(F_M(G)\right) = f(G)$.

Note that the function f is a real function of each pixel group $G=(x_1,x_2,\dots x_n)$ and it is based on some statistical measure of sample pairs $[x_i, x_{i+1}]$, $i = 1,\dots, n$-1, that are sensitive to least significant bit embedding operations.

For example, the larger the pixel variation of the pixels $f(F_M(G))$ the larger the values of the discrimination function becomes. The total number of regular groups is larger than the total number of singular groups [2]. This procedure repeats for each 2 x 2 block within the input image. The features generated can be defined as follows:

$$\begin{aligned} &F_n = \{f_1, f_2, ..., f_n\} \\ &f_n = \frac{1}{NM}\sum_i f\left(F_M(G)\right) > (G), \qquad n \text{ odd} \\ &f_n = \frac{1}{NM}\sum_i f\left(F_M(G)\right) < (G), \qquad n \text{ even} \end{aligned} \tag{5}$$

2.2 Multi Weighted Masks

This section presents the multi pixel comparison used for generating features which are sensitive to changes made to spatial domain images [5]. The presented method alleviates the need to consider multiple bit-plane embedding by considering the overall amplitude of the image pixels.

Consider an input image of size M x N, the rows and columns denoting the number of adjacent pixels surrounding a center pixel at location i, j. The masks used can be of various sizes for the number of compared pixels. As an example a simple mask consists of the following set of analyzed pixels which are represented as:

$$\begin{bmatrix} x_{i,j} & x_{i,j+1} \\ x_{i+1,j} & x_{i+1,j+1} \\ x_{i+2,j} & x_{i+2,j+1} \end{bmatrix} \Rightarrow \begin{bmatrix} x_{i,j} & x_{i+1,j} & x_{i+2,j} & x_{i,j+1} & x_{i+1,j+1} & x_{i+2,j+1} \end{bmatrix} \Rightarrow \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{bmatrix} = X\ .$$

By mapping the 3 x 2 block into a one dimensional 1 x 6 vector provides a means to easily measure statistical changes before and after embedding. This allows the structure of a set of masks to be applied to the pixels $[x_1\ x_2\ x_3\ x_4\ x_5\ x_6]$. Two classes of masks may be used which are $M_1 = [m_1\ m_2\ m_3\ m_4\ m_5\ m_6]$ and $M_2 = [m_1\ m_4\ m_5\ m_2\ m_3\ m_6]$ where $m_i \in (-A,0,A)$, $A = 2^n$, n = 0,1,2,3. Note, that by rearranging the values of M_1 the second mask M_2 is generated.

The two classes of masks generate six sets of sub masks, three for each class, which are added to the pixel region being analyzed. The three sub masks are generated with the use of the A values. The series of weights are assigned according to the distance from the current central pixel at the current location i, j for both of the arranged pixel vectors, as follows;

$$W = \left[w_1 \quad w_2 \quad w_3 \quad w_4 \quad w_5 \right], \qquad w_i \in \mathbb{R}$$

We apply the weights to our modulus operation between the pixels and corresponding mask values. The weight vector only consists of five values since five calculations are made between adjacent pixels. The weighted statistical average of adjacent pixel pairs within a comparative mask and block is defined as follows:

$$\begin{aligned} \hat{f}^{\,n}_{i,j}\left(M_k, W\right) &= \frac{1}{\hat{n}-1} \sum_{\hat{i}=2}^{\hat{n}} \left(w_{i-1} \times \left| \left(m_1 \oplus x_1\right) - \left(m_{\hat{i}} \oplus x_{\hat{i}}\right) \right| \right) \\ F_n &= \frac{1}{NM} \sum_{i,j} \hat{f}^{\,n}_{i,j}\left(M_k, W\right) \\ F_n &= \left\{ f_1, f_2, \ldots, f_n \right\} \end{aligned} \tag{6}$$

where, $\oplus$ is the modulus operation of adding to the pixel value of one bit plane only, k represents the mask used, $\hat{n}$ is the number of pixels in the mask, n represents the number of features (n = 7 in our case) and i, j subscripts represent the pixel location throughout the image.

The first six features extracted represent statistics over any modifications made to the selected six bit planes and the seventh feature is calculated with no modifications made to the bit planes. These features are used in discriminating between clean images and those containing hidden messages.

3. CLASSIFICATION

In this section, the expectation maximization mixture model clustering method is extended to handle the two-class classification problem with the addition of a Bayesian classifier. Initial testing results are shown using a small database of 200 images. The classification method identifies images that are considered clean and stego images which have been generated with random embedding. The classification goal is proper classification of images containing hidden information and images which are considered clean. The image attributes used for classification are the features in section 2 which represent the various domains. In a more general case n features $x_n \in \mathbb{R}^n$ in the subset D_k which form the feature vectors represents the domain space. This section is represented in three subsections which are: the definition of expectation maximization, mixture models, and Bayes classifier.

3.1 Expectation Maximization (EM)

The idea behind the EM algorithm is that even though the data values of x_n, feature vectors $x_n \in \mathbb{R}^n$, are unknown/incomplete the distribution $f(x_n|p)$ can be used to determine an estimate for the maximum likelihood [6]. In maximum likelihood estimation, the estimate to be modeled is the parameter(s) for which the observed data are the most likely. This is done by iteratively estimating the data parameters, then using the data to update the estimated parameters, until a desired convergence is met. The two major steps of the EM algorithm are the E-Step and the M-Step.

The EM algorithm consists of choosing initial parameters for the means ($\mathbf{m}_k^{(i)}$), standard deviations ($\sigma_\kappa^{(i)}$), and mixing probabilities ($p_k^{(i)}$), then performing the E-Step and M-Step successively until convergence. The convergence criteria is determined by examining when the parameters quit changing, i.e., determine "yes" when $|\mathbf{m}_k^{(i)} - \mathbf{m}_k^{(i+1)}| < \varepsilon$ & $|\sigma_\kappa^{(i)}$ -

$\sigma_{\kappa}^{(i+1)}| < \varepsilon$ & $| p^{(i)}(k|n) - p^{(i+1)}(k|n)| < \varepsilon$ for some epsilon (ε) and distance calculation (Euclidian distance). The maximum likelihood estimation is a method of estimating the parameters of the distributions based upon the observed data.

The Expectation Step (E-Step) calculates the *membership probabilities*, $p(k|n)$, which consists of computing the expected value of the data point, x_n, using the probability that n was generated by the component k [6]. The mixing probabilities p_k are viewed as the sample mean of the membership probabilities $p(k|n)$ assuming a uniform distribution over all the data points. The Gaussian function, $g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\Sigma_{\kappa}^{(i)}\right)$, is used to compute mixture of Gaussian functions as shown in the denominator of $p(k|n)$.

$$p^{(i)}(k|n) = \frac{p_k^{(i)} g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\sigma_{\kappa}^{(i)}\right)}{\sum_{k=1}^{K} p_k^{(i)} g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\sigma_{\kappa}^{(i)}\right)} \tag{7}$$

$$g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\sigma_{\kappa}^{(i)}\right) = \frac{1}{\left(\sqrt{2\pi}\sigma_k\right)^D} e^{-\frac{1}{2}\left(\frac{\left\|\mathbf{x}_n - \mathbf{m}_k\right\|}{\sigma_k}\right)^2} \tag{8}$$

The Maximization Step (M-Step) uses the data from the expectation step as if it were measured data to determine the maximum likelihood estimate of the parameter [6]. This estimated data is often referred to as the “imputed” data. This step is dependent upon the membership probabilities $p(k|n)$ which are computed in the E-Step. The EM algorithm consists of iterating the mean, standard deviation, and mixing probabilities until convergence. The mean is a simple intuitive computation of the sample data weighted by the conditional probability that data point n was generated with k. The standard deviation is also a simple computation which is generated by the sample data weighted by the conditional probability that data point n was generated with k. The mixing probabilities is the sample mean of the conditional probabilities $p(k|n)$ assuming a uniform distribution over all the data points.

$$\mathbf{m}_k^{(i+1)} = \frac{\sum_{n=1}^{N} p^i(k|n)\, x_n}{\sum_{n=1}^{N} p^i(k|n)} \tag{9}$$

$$\sigma_k^{(i+1)} = \sqrt{\frac{1}{D}\frac{\sum_{n=1}^{N} p^i(k|n)\left\|\mathbf{x}_n - \mathbf{m}_k^{(i+1)}\right\|^2}{\sum_{n=1}^{N} p^i(k|n)}} \tag{10}$$

$$p_k^{(i+1)} = \frac{1}{N}\sum_{n=1}^{N} p^i(k|n) \tag{11}$$

3.2 Mixture Models

In mixture models also known as model-based Gaussian clustering the multivariate Gaussian normal is used as a density function similarly described in the EM section [4]. The general multivariate normal density for d dimensions is defined as follows:

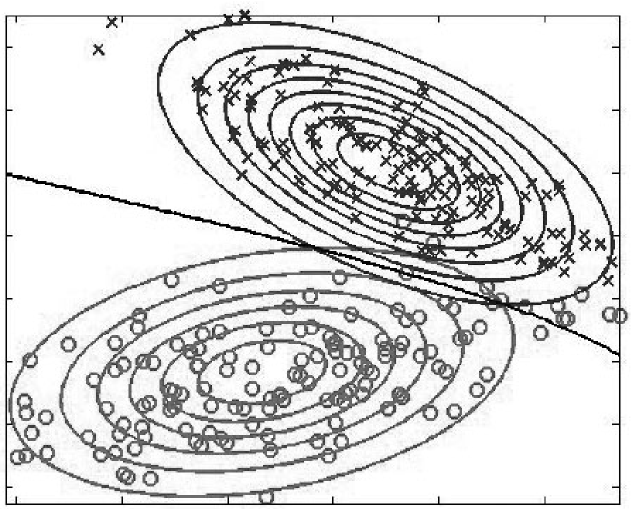
$$g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\Sigma_\kappa^{(i)}\right)=\frac{\exp\left(-\frac{1}{2}\left(x_n-m_k\right)^T\Sigma_k\left(x_n-m_k\right)\right)}{\left(\sqrt{2\pi}\right)^D\left|\Sigma_k\right|^{1/2}} \tag{12}$$

The geometric characteristics (size, shape and orientation) of the clusters are determined by the covariance matrix Σ_k which is generated in terms of eigenvalue decomposition described in [7]. The decomposition of the covariance matrix Σ_k is used as a suitable model for the geometric characteristics of the cluster. The structure of the covariance matrix is as follows:

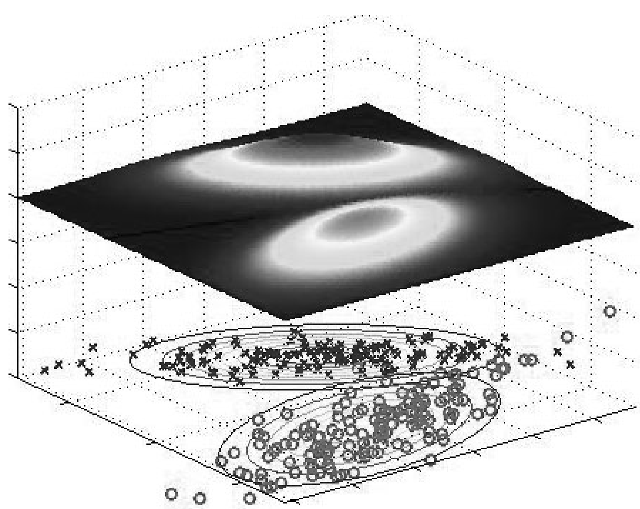
$$\Sigma_k=\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k^T \tag{13}$$

where λ_k is a scalar, D_k is the orthogonal matrix of eigenvectors and A_k is a diagonal matrix whose elements are proportional to the eigenvalues of Σ_k. Note that in EM the values p_k, m_k, and σ_k are updated after each iteration and in the mixture models σ_k is replaced by Σ_k to represent the geometric characteristics of the clusters.

The eigenvalue decomposition can be modeled as various clustering arrangements, i.e., spheres, ellipsoids and rotations of ellipsoids. Allowing the orientation, volume, shape and size of the clusters defines the various models used. Figure 1 shows the mixture model using rotated ellipsoids to generate the decision boundary around each class.



a) 2D example with decision boundary



b) 3D example with Gaussian curve

Fig. 1. Expectation Maximization using mixture models.

3.3 Bayes Classifier

Classification uses input samples described by feature vectors $x_n \in \mathbb{R}^n$ to assign the samples to a given class ω_i. The Bayes classifier extends a general multivariate normal case where the covariance matrix Σ_i for each class is different. For the multi-class classifier each class must have individual conditional probability densities where the densities are modeled as normal distributions. The classes ω_i are defined as normal distributions centered about the mean vector μ_i. The vector x_n is a d-component vector of the observed data, and $|\Sigma_i|$ and Σ_i^{-1} are the determinants and inverse covariance matrix of the given class. Using the density function $g\left(\mathbf{x}_n;\mathbf{m}_k^{(i)},\Sigma_\kappa^{(i)}\right)$ [4], the Bayes classifier can be expressed in terms of the prior probabilities and conditional probability densities as follows:

$$\omega_{ni} = \frac{1}{\sqrt{(2\pi)^d \left|\Sigma_i\right|}} \exp\left[-\frac{1}{2}\left(x_n - \mu_i\right)^T \Sigma_i^{-1}\left(x_n - \mu_i\right)\right] P\left(\omega_i\right) \tag{14}$$

where the a priori probabilities $P(\omega_i)$ are the estimates of belonging to a class.

The posterior probability of class membership can be calculated by Bayes rule if ω_i is defined as the event of belonging to class i;

$$\begin{aligned} P\left(\omega_i\right) &= \frac{\frac{1}{\sqrt{(2\pi)^d \left|\Sigma_i\right|}} \exp\left[-\frac{1}{2}\left(x_n - \mu_i\right)^T \Sigma_i^{-1}\left(x_n - \mu_i\right)\right]}{\sum_{x \in D_i} \frac{1}{\sqrt{(2\pi)^d \left|\Sigma_i\right|}} \exp\left[-\frac{1}{2}\left(x_n - \mu_i\right)^T \Sigma_i^{-1}\left(x_n - \mu_i\right)\right]} \\ &= \frac{\exp\left[-\frac{1}{2}\left(x_n - \mu_i\right)^T \Sigma_i^{-1}\left(x_n - \mu_i\right)\right]}{\sum_{x \in D_i} \exp\left[-\frac{1}{2}\left(x_n - \mu_i\right)^T \Sigma_i^{-1}\left(x_n - \mu_i\right)\right]} \end{aligned} \tag{15}$$

under the assumption that $\Sigma_i = \Sigma$ for $i = \{1,2\}$.

4. RESULTS

A number of experiments have been performed by applying the proposed detection approaches. The spatial domain images are of size 512 x 512 RGB images with 100 clean images and 100 containing random hidden information with a capacity percentages of 1%, 2%, 3%, 4%, 5%, 6%, 7%, 8%, 9% and 10% altered pixel in the LSB. Three sets of tests were run. The first was a blind detection analysis, in which the classification model is trained with only clean image sets. The anomalous data consists of the stego image sets and image sets which contain 5% Gaussian noise. The second test is that of binary classification in which EM is executed separately on the set of images which considered as stego (random embedding) and on the clean images, and then the models are combined for testing. The final test uses EM for multi-class classification, in which the classification must identify the percentage of hidden information. The classification testing is done with a 5-fold cross validation analysis.

Tables 1 and 2 depict the analysis results for blind detection where images that are not in the space of the clean image feature are considered anomalies, and the system is trained only on clean images [8, 9]. This classification method takes advantage of the covariance matrix present in the Expectation Maximization Gaussian mixture models in order to closely fit the clean data. The mixture models create elongated ellipsoids rotated from the feature axis. The classification with this method is based on the probability of an exemplar belonging to a clean class with a probability greater than 20%.

Table 1. Results for RS Steganalysis features using 5-fold cross validation for blind detection.

	Actual Clean Images	**Actual Anomalous Images**
Predicted Clean Images	97.0 ± 2.7 %	3.0 ± 2.7 %
Predicted Anomalous Images	13.8 ± 4.6 %	86.2 ± 4.6 %

Table 2. Results for Multi Weighted Masks features using 5-fold cross validation for blind detection.

	Actual Clean Images	**Actual Anomalous Images**
Predicted Clean Images	94.8 ± 3.4 %	5.2 ± 3.4 %
Predicted Anomalous Images	10.8 ± 4.8 %	89.2 ± 4.8 %

For the two-class case the proposed approach is used to separate the clean image set from the stego image set. The spatial domain images used were 512 x 512 RGB images where 100 clean images and 100 stego images containing stego ranging from 1% to 10% randomly altered LSB pixels is used as the training and testing set. Testing is conducted using 5-fold cross validation, with the results shown in Tables 3 and 4. In this case, the separation between clean and dirty is based upon which cluster the sample has the highest probability of resembling.

Table 3. Results for RS Steganalysis features using 5-fold cross validation for the two-class data.

	Actual Clean Images	**Actual Dirty Images**
Predicted Clean Images	97.94 ± 2.8 %	2.05 ± 2.8 %
Predicted Dirty Images	6.57 ± 4.2 %	93.43 ± 4.2 %

Table 4. Results for Multi Weighted Masks features using 5-fold cross validation for two-class data.

	Actual Clean Images	**Actual Dirty Images**
Predicted Clean Images	96.95 ± 2.3 %	2.78 ± 2.3 %
Predicted Dirty Images	3.3 ± 5.1 %	96.7 ± 5.1 %

In Table 5 Expectation Maximization Gaussian mixture models applied to classifying a multi-class case is used to determine the amount of steganographic content within an image given a low embedding percentage. Here, the clustering algorithm separated the clean image set and the stego image sets containing 1%, 2%, 3%, 4%, 5%, 6%, 7%, 8%, 9% and 10% altered pixel in the LSB from each other.

Table 5. Results for RS Steganalysis and Multi Weighted Masks Multi-Class Classification using 5-fold cross validation.

Classification Accuracy = (TP+TN)/2		
Percent Stego	RS Stego	Weighted Masks
1%	63.9 ± 5.5 %	74.1 ± 6.7 %
2%	80.0 ± 5.8 %	85.6 ± 4.8 %
3%	89.6 ± 5.8 %	93.6 ± 3.9 %
4%	92.3 ± 4.5 %	94.3 ± 3.5 %
5%	95.4 ± 5.5 %	95.7 ± 3.1 %
6%	96.5 ± 3.0 %	96.4 ± 2.7 %
7%	96.3 ± 2.5 %	97.1 ± 2.9 %
8%	97.3 ± 2.0 %	97.6 ± 2.6 %
9%	97.7 ± 2.3 %	98.5 ± 1.7 %
10%	99.5 ± 1.0 %	99.1 ± 1.0 %

The multi-class classification model begins to reach a point of reliability when the stego images contain 4% hidden information for RS Steganalysis and 3% hidden information for Multi Weighted Masks.

5. CONCLUSION

In this paper analysis was conducted using Expectation Maximization with mixture models. Three separate tests were conducted to show that the classification method could be used to improve the two feature extraction methods. The first test showed that when using the classification method as a blind detector, i.e., training with only 20% clean image sets, could determine if an input image is in fact clean. The blind detector was only able to reach 89% classification accuracy when an anomalous image was presented to the system. This was improved with the use of the two-class classification system where both the clean and stego image sets were used in the training process. With the two-class system beginning to reach a point of reliability, the classification method was tested against 10 classes. In the final test the images were classified based on the amount of hidden information within the images. In this testing RS Steganalysis was able to provide reliable results starting at 4% embedding with an accuracy of 92.3% classification accuracy. RS Steganalysis was able to reach 99.5% classification accuracy when 10% of the images were manipulated. Multi Weighted Masks was able to provide reliable results starting at 3% embedding with an accuracy of 93.6% classification accuracy. Multi Weighted Masks was also able to reach 99.1% classification accuracy when 10% of the images were manipulated.

ACKNOWLEDGEMENTS

The work on this paper was partially supported by the Digital Data Embedding Technologies group of the Air Force Research Laboratory, Information Directorate. The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Air Force Research Laboratory, or the U.S. Government. We would additionally like to express our appreciation to June Rodriguez for the contribution of a multitude of digital images for analytical support.

REFERENCES

1. StegoArchive.com, http://www.stegoarchive.com
2. J. Fridrich, M. Goljan and R. Du, “Detecting LSB Steganography in Color and Gray-Scale Images,” Magazine of IEEE Multimedia Special Issue on Security, October-November 2001, pp. 22-28
3. A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm”, *Journal of the Royal Statistical Society B*, 39(1):1–22, 1977
4. R. Duda, P. Hart and D. Stork, *Pattern Classification*, Second Edition, Wiley 2001
5. B. M. Rodriguez, G. L. Peterson, K. W. Bauer, and S. S. Agaian, “Steganalysis Embedding Percentage Determination with Learning Vector Quantization”, 2006 IEEE International Conference on Systems, Man, and Cybernetics, Oct. 8 - Oct. 11, 2006
6. C. Tomasi, “Estimating Gaussian Mixture Densities with EM – A Tutorial”, Duke University Course Notes, http://www.cs.duke.edu/courses/spring04/cps196.1/handouts/EM/tomasiEM.pdf, retrieved Sept 2006
7. W. L. Martinez and A. R. Martinez, *Computational Statistics Handbook with MATLAB*, Chapman & Hall/CRC, Boca Raton, FL 2002
8. B. McBride and G. L. Peterson, “Blind Data Classification using Hyper-Dimensional Convex Polytopes,” Proceedings of the 17th International FLAIRS Conference, pp 520-525, 2004
9. B. McBride and G. L. Peterson, “A new blind method for detecting novel steganography”, Digital Investigation, Vol 2, 50-70, 2005

*benjamin.rodriguez@afit.edu; phone 1 (937) 255-3636 ext. 7543; fax 1 (937) 656-4055