Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

An Application of Linear Mixed Effects Model to Steganography Detection

Mei-Ching Chen*, Anuradha Roy†, and Benjamin M. Rodriguez‡
*Department of Electrical and Computer Engineering
†Department of Management Science and Statistics
The University of Texas at San Antonio, San Antonio, TX, U.S.A.
‡Space Department, Johns Hopkins University Applied Physics Laboratory
Laurel, MD, U.S.A.
*axf710@my.utsa.edu, †Anuradha.Roy@utsa.edu, ‡Benjamin.Rodriguez@jhuapl.edu

***Abstract*—Current technology allows steganography applications to conceal any digital file inside of another digital file. Due to the large number of steganography tools available over the Internet, a particular threat exists when criminals use steganography to conceal their activities within digital images in cyber space. In this paper, a set of statistical features are generated using linear mixed effects models in conjunction with wavelet decomposition for image steganography detection. It is important to generate features capable of distinguishing between a set of clean and steganography images for steganalysts in commercial industry, Department of Defense, government as well as law enforcement. In the experimental results, seven sets of images are used to measure the performance of the proposed method, a clean set and two JPEG steganography methods with three different embedding file sizes to create steganography images. The number of correct predictions that an instance is clean or steganographic are improved by as much as 38% when using the proposed linear mixed effects models compared to the linear fixed effects models.**

***Index Terms*—Linear mixed effects model, steganography, steganalysis, feature generation, maximum likelihood estimates**

I. Introduction

Steganography has been used throughout history from ancient Greece to todays cyber space for secret communications [3], [9]. In recent years, digital steganography has emerged as a significant research area [8]. The use of freely available steganography tools allows secret information to be imperceptibly hidden within digital signals. Signals containing concealed information are stored or transmitted through public channels to protect pertinent data. However, criminals have sought ways to use steganography to conceal their activity in real or physical space as well as in virtual or cyber space. In a special report by the National Institute of Justice, it is stated that potential digital evidences in child abuse, child exploitation and terrorism investigations include information regarding steganography [12].

The use of steganography as a means of computer and cyber crime leads to the problem of steganalysis [3], [16]. There are three areas a steganalyst is challenged with; detecting the existence of steganographic content, identification of the steganography method used, and the extraction of the hidden information [14]. This paper focuses on generating statistical features using image wavelet decomposition along with linear mixed effects models for the detection of image steganography. The proposed concept predicts wavelet coefficients that are either clean or steganography using linear mixed effects models [4], [11], in which images are considered random from both clean or steganography image sets. For an input image, there are three scales of the wavelet decompositions along with three orientations, vertical, horizontal, and diagonal, for each scale. Four statistics, mean, variance, skewness, and kurtosis, are used to measure the differences between the predicted coefficients and the wavelet coefficients for each scale and orientation. This results in a total of 36 statistical features capable of distinguishing between clean and steganography images.

The detection of steganography images based on wavelet features was first introduced by Farid [6]. A total of 72 features were generated, 36 derived from original wavelet coefficients and 36 from the log differences between the original and the predicted coefficients. A linear regression model of fixed effects was used by Farid to estimate the wavelet coefficients. However, the problem arises from correlations between the predicted and the original wavelet transform coefficients when using the same input image. The presented mixed effects models allow this correlation to be broken by considering the randomness among images. This allows a greater separation when determining which class an image is associated with. Features are generated from a set of clean and steganography images using the previous method [6] as well as the proposed method and classified using a neural network with Gaussian kernels. Two JPEG steganography methods, JPEG-JSteg v1 [13] and JPEG-JSteg v4 [15], are used in the experiment with three different embedding text file sizes, 0.21KB, 0.32KB, and 1.04KB. The results show that mixed effects models keep the consistency of classification accuracy no matter how much information is hidden within images.

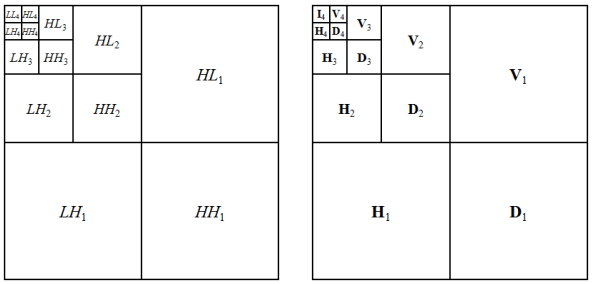
The paper is organized as follows. Section II gives the background knowledge of wavelet decomposition which will be used before generating features. The benchmark feature generation method [6] is then briefly explained in Section III followed by the proposed method in Section IV. A classifier utilized here is illustrated in Section V [2]. With 10-fold cross

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

validation [10], Section VI shows and compares classification accuracies between methods, ensuing the conclusion and discussion of possible future work in Section VII.

II. Wavelet Decomposition

This paper employs wavelet transforms as a preprocess for deriving image features. The image decomposition employed here is based on separable quadrature mirror filters (QMF) originally used in [6]. Applying the wavelet transform on a given image maps the image pixel from the spatial domain to the wavelet transform domain, i.e., to the frequency space. The image decomposition splits the frequency space into multiple orientations and levels. For each level, four subbands, LL, LH, HL, and HH, are created inheriting the characteristics of the image, identical ($\mathbf{I}$), vertical ($\mathbf{V}$), horizontal ($\mathbf{H}$) and diagonal ($\mathbf{D}$) information, correspondingly. The symbol L is denoted as a lowpass filter being used, while H would be the case of a highpass filter. The number of iterations applied, also named



(a) Subband information (b) Edge information

Fig. 1. A four-scale wavelet decomposition

as a scale or level, is denoted by a subscript i. Recursively applying QMF to LL_i, Fig. 1 presents an i^{th}-scale wavelet decomposition, where $i = 1, 2, 3, 4$. Suppose, there is a set of n images. For each image, a four-scale three-orientation QMF is utilized. A coefficient located within a wavelet transformed image is denoted by its edge characteristics and its position, i.e., $v_{j,i}(x, y)$, $h_{j,i}(x, y)$, $d_{j,i}(x, y)$, where $i = 1, 2, 3, 4$, $j = 1, 2, \ldots, n$, and (x, y) is the row and column index within a characteristic block. The coordinate (x, y) denotes the coefficient location within a block.

III. Wavelet Feature Generation using Linear Fixed Effects Models

Given a wavelet transformed image, linear fixed effects models are used for predicting wavelet coefficients [6]. Note that only the wavelet coefficients having the absolute values greater than one and located at the first three scales are considered. For a particular coefficient, seven neighboring coefficients are selected based on each characteristic, formulated as linear fixed effects models in the following.

$$\begin{aligned} v_{j,i}(x,y) &= \beta^v_{i,1} v_{j,i}(x-1,y) + \beta^v_{i,2} v_{j,i}(x+1,y) \\ &\quad + \beta^v_{i,3} v_{j,i}(x,y-1) + \beta^v_{i,4} v_{j,i}(x,y+1) \\ &\quad + \beta^v_{i,5} v_{j,i+1}(x/2,y/2) + \beta^v_{i,6} d_{j,i}(x,y) \\ &\quad + \beta^v_{i,7} d_{j,i+1}(x/2,y/2) + \epsilon^v_{j,i}, \end{aligned}$$

$$\begin{aligned} h_{j,i}(x,y) &= \beta^h_{i,1} h_{j,i}(x-1,y) + \beta^h_{i,2} h_{j,i}(x+1,y) \\ &\quad + \beta^h_{i,3} h_{j,i}(x,y-1) + \beta^h_{i,4} h_{j,i}(x,y+1) \\ &\quad + \beta^h_{i,5} h_{j,i+1}(x/2,y/2) + \beta^h_{i,6} d_{j,i}(x,y) \\ &\quad + \beta^h_{i,7} d_{j,i+1}(x/2,y/2) + \epsilon^h_{j,i}, \end{aligned}$$

$$\begin{aligned} d_{j,i}(x,y) &= \beta^d_{i,1} d_{j,i}(x-1,y) + \beta^d_{i,2} d_{j,i}(x+1,y) \\ &\quad + \beta^d_{i,3} d_{j,i}(x,y-1) + \beta^d_{i,4} d_{j,i}(x,y+1) \\ &\quad + \beta^d_{i,5} d_{j,i+1}(x/2,y/2) + \beta^d_{i,6} h_{j,i}(x,y) \\ &\quad + \beta^d_{i,7} v_{j,i}(x,y) + \epsilon^d_{j,i}, \end{aligned}$$

for $i = 1, 2, 3$ and $j = 1, 2, \ldots, n$. The model coefficient $\beta^r_{i,k}$ for $r = v, h, d$, $i = 1, 2, 3$ and $k = 1, 2, \ldots, 7$ denotes the weight corresponding to a neighboring wavelet coefficient, and $\epsilon^r_{j,i}$ denotes the corresponding error.

There may be unequal number of measurements in the images due to the number of absolute coefficient values greater than one. Let n_j denote the number of measurements in the jth image. The above equations can be expressed in matrix forms as follows:

$$\boldsymbol{v}_{j,i} = \boldsymbol{Q}^v_{n_j,i} \boldsymbol{\beta}^v_i + \boldsymbol{\epsilon}^v_{j,i}, \quad (1)$$

$$\boldsymbol{h}_{j,i} = \boldsymbol{Q}^h_{n_j,i} \boldsymbol{\beta}^h_i + \boldsymbol{\epsilon}^h_{j,i}, \quad (2)$$

$$\boldsymbol{d}_{j,i} = \boldsymbol{Q}^d_{n_j,i} \boldsymbol{\beta}^d_i + \boldsymbol{\epsilon}^d_{j,i}, \quad (3)$$

for $i = 1, 2, 3$ and $j = 1, 2, \ldots, n$. The $n_j \times 1$ dimensional error vectors $\boldsymbol{\epsilon}^v_{j,i}$, $\boldsymbol{\epsilon}^h_{j,i}$ and $\boldsymbol{\epsilon}^d_{j,i}$ are all normally distributed with mean zero. The coefficients $\boldsymbol{\beta}^v_i$, $\boldsymbol{\beta}^h_i$ and $\boldsymbol{\beta}^d_i$ are given by

$$\begin{aligned} \boldsymbol{\beta}^v_i &= (\beta^v_{i,1}, \beta^v_{i,2}, \beta^v_{i,3}, \beta^v_{i,4}, \beta^v_{i,5}, \beta^v_{i,6}, \beta^v_{i,7})', \\ \boldsymbol{\beta}^h_i &= (\beta^h_{i,1}, \beta^h_{i,2}, \beta^h_{i,3}, \beta^h_{i,4}, \beta^h_{i,5}, \beta^h_{i,6}, \beta^h_{i,7})', \\ \boldsymbol{\beta}^d_i &= (\beta^d_{i,1}, \beta^d_{i,2}, \beta^d_{i,3}, \beta^d_{i,4}, \beta^d_{i,5}, \beta^d_{i,6}, \beta^d_{i,7})', \end{aligned}$$

and $\boldsymbol{Q}^v_{n_j,i}$, $\boldsymbol{Q}^h_{n_j,i}$ and $\boldsymbol{Q}^d_{n_j,i}$ are the $n_j \times 7-$dimensional design matrices of the fixed effects in the three orientations as defined before. The n submodels for n images can be stacked one below the other for each characteristic block (1), (2) and (3), giving a single model for each block as follows:

$$\boldsymbol{v}_i = \boldsymbol{Q}^v_i \boldsymbol{\beta}^v_i + \boldsymbol{\epsilon}^v_i, \quad (4)$$

$$\boldsymbol{h}_i = \boldsymbol{Q}^h_i \boldsymbol{\beta}^h_i + \boldsymbol{\epsilon}^h_i, \quad (5)$$

$$\boldsymbol{d}_i = \boldsymbol{Q}^d_i \boldsymbol{\beta}^d_i + \boldsymbol{\epsilon}^d_i, \quad (6)$$

where $\boldsymbol{v}_i$, $\boldsymbol{h}_i$ and $\boldsymbol{d}_i$ are all $m \times 1$ dimensional vectors with $m = \sum_{j=1}^{n} n_j$. The error vectors

$$\boldsymbol{\epsilon}^r_i \sim N_m(\boldsymbol{0}, \sigma^2_{ri} \boldsymbol{I}_m),$$

for $r = v, h, d$, and $i = 1, 2, 3$.

The system of equations (4), (5) and (6) represent nine models for $i = 1, 2$, and 3. It must be noted that one model is designed for one orientation in every scale. The coefficients $\boldsymbol{Q}_i^v, \boldsymbol{Q}_i^h$ and $\boldsymbol{Q}_i^d$ in the above models are written as

$$\boldsymbol{Q}_i^r = \begin{bmatrix} \boldsymbol{Q}_{n_1,i}^r \\ \boldsymbol{Q}_{n_2,i}^r \\ \vdots \\ \boldsymbol{Q}_{n_n,i}^r \end{bmatrix},$$

where r can take value v, h or d. And, $\boldsymbol{r}_i$, for each $\boldsymbol{r}$ and i is given by $\boldsymbol{r}_i = (\boldsymbol{r}'_{1,i}, \boldsymbol{r}'_{2,i}, \ldots, \boldsymbol{r}'_{n,i})'$ and $\boldsymbol{\epsilon}_i^r = (\boldsymbol{\epsilon}_{1,i}^{r\prime}, \boldsymbol{\epsilon}_{2,i}^{r\prime}, \ldots, \boldsymbol{\epsilon}_{n,i}^{r\prime})'$. Now, the best linear unbiased estimates (BLUEs) of the predicted model coefficients $\boldsymbol{\beta}_i^r$ are given by

$$\begin{aligned} \widehat{\boldsymbol{\beta}}_i^v &= (\boldsymbol{Q}_i^{v\prime}\boldsymbol{Q}_i^v)^{-1}\boldsymbol{Q}_i^{v\prime}\boldsymbol{v}_i, \\ \widehat{\boldsymbol{\beta}}_i^h &= (\boldsymbol{Q}_i^{h\prime}\boldsymbol{Q}_i^h)^{-1}\boldsymbol{Q}_i^{h\prime}\boldsymbol{h}_i, \\ \widehat{\boldsymbol{\beta}}_i^d &= (\boldsymbol{Q}_i^{d\prime}\boldsymbol{Q}_i^d)^{-1}\boldsymbol{Q}_i^{d\prime}\boldsymbol{d}_i. \end{aligned}$$

The predicted wavelet coefficients are then given by

$$\begin{aligned} \widehat{\boldsymbol{v}}_i &= \boldsymbol{Q}_i^v\widehat{\boldsymbol{\beta}}_i^v, \\ \widehat{\boldsymbol{h}}_i &= \boldsymbol{Q}_i^h\widehat{\boldsymbol{\beta}}_i^h, \\ \widehat{\boldsymbol{d}}_i &= \boldsymbol{Q}_i^d\widehat{\boldsymbol{\beta}}_i^d, \end{aligned}$$

and the log error measures for each orientation in each scale are determined with

$$\begin{aligned} \boldsymbol{e}_i^v &= \log_2|\boldsymbol{v}_i| - \log_2|\widehat{\boldsymbol{v}}_i|, \\ \boldsymbol{e}_i^h &= \log_2|\boldsymbol{h}_i| - \log_2|\widehat{\boldsymbol{h}}_i|, \\ \boldsymbol{e}_i^d &= \log_2|\boldsymbol{d}_i| - \log_2|\widehat{\boldsymbol{d}}_i|, \end{aligned}$$

for $i = 1, 2$, and 3. The statistics mean, variance, skewness and kurtosis are calculated on both original coefficient vectors and these error vectors. Table I shows these statistics and their corresponding equations for $r = v, h, d$, and $i = 1, 2, 3$. For a given image, there are 72 features generated for the nine fixed effects models by deriving from the statistics calculations on both original and error wavelet coefficients for each characteristic block, where 36 features are from original coefficients and the remaining 36 features are from error coefficients.

TABLE I
THE STATISTICS AND THEIR FORMULAS

Statistic	Formula
Mean	$s_{i,1}^r = \frac{1}{n}\sum_{j=1}^{n}\epsilon_{j,i}^r$
Variance	$s_{i,2}^r = \frac{1}{n}\sum_{j=1}^{n}\epsilon_{j,i}^r - s_{i,1}^r)^2$
Skewness	$s_{i,3}^r = \frac{\frac{1}{n}\sum_{j=1}^{n}(\epsilon_{j,i}^r - s_{i,1}^r)^3}{(s_{i,2}^r)^{3/2}}$
Kurtosis	$s_{i,4}^r = \frac{\frac{1}{n}\sum_{j=1}^{n}(\epsilon_{j,i}^r - s_{i,1}^r)^4}{(s_{i,2}^r)^2}$

IV. Wavelet Feature Generation using Linear Mixed Effects Models

In this section, n random images from a population of images are considered. Then the nine fixed effects models (4), (5) and (6) can be written as the mixed effects models [11] in a compact form as follows:

$$\boldsymbol{r}_i = \boldsymbol{Q}_i^r\boldsymbol{\beta}_i^r + \boldsymbol{Z}_i^r\boldsymbol{\alpha}_i^r + \boldsymbol{\epsilon}_i^r,$$

for $r = v, h, d$, and $i = 1, 2, 3$, where $\boldsymbol{Q}_i^r$ and $\boldsymbol{Z}_i^r$ are respectively $m \times 7$ and $m \times n$ design matrices, and $\boldsymbol{\epsilon}_i^r$ is the $m \times 1$ column vector of residuals assumed to be distributed independently of the $n \times 1$ random effects $\boldsymbol{\alpha}_i^r$. The 7-dimensional vector $\boldsymbol{\beta}_i^r$ contains the fixed effects parameters in each of the nine models. These are fixed, but unknown constants. Assume

$$\begin{aligned} \boldsymbol{\alpha}_i^r &\sim N_n(\boldsymbol{0}, \sigma_{\alpha,ri}^2\boldsymbol{I}_n), \\ \boldsymbol{\epsilon}_i^r &\sim N_m(\boldsymbol{0}, \sigma_{\epsilon,ri}^2\boldsymbol{I}_m). \end{aligned}$$

The BLUEs of $\boldsymbol{\beta}_i^r$ and the best linear unbiased predictors (BLUPs) of $\boldsymbol{\alpha}_i^r$ can be obtained by solving the system of mixed model equations [7]

$$\begin{bmatrix} \boldsymbol{Q}_i^{r\prime}\boldsymbol{Q}_i^r & \boldsymbol{Q}_i^{r\prime}\boldsymbol{Z}_i^r \\ \boldsymbol{Z}_i^{r\prime}\boldsymbol{Q}^r & \boldsymbol{Z}_i^{r\prime}\boldsymbol{Z}_i^r + k\boldsymbol{I}_n \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}}_i^r \\ \widehat{\boldsymbol{\alpha}}_i^r \end{bmatrix} = \begin{bmatrix} \boldsymbol{Q}_i^{r\prime}\boldsymbol{r}_i \\ \boldsymbol{Z}_i^{r\prime}\boldsymbol{r}_i \end{bmatrix},$$

where $k = \sigma_{\epsilon,ri}^2/\sigma_{\alpha,ri}^2$. Thus, the predicted wavelet coefficients and the log error measures for each orientation in each level are calculated as follows:

$$\begin{aligned} \widehat{\boldsymbol{r}}_i &= \boldsymbol{Q}_i^r\widehat{\boldsymbol{\beta}}_i^r + \boldsymbol{Z}_i^r\widehat{\boldsymbol{\alpha}}_i^r, \\ \boldsymbol{e}_i^r &= \log_2|\boldsymbol{r}_i| - \log_2|\widehat{\boldsymbol{r}}_i|, \end{aligned}$$

for $r = v, h, d$ and $i = 1, 2, 3$. The four statistics are calculated as described in Table I in Section III using these log error measures. This results in a feature vector $\boldsymbol{p}$ of 36 features from the log error measures.

V. Classification

Before using the features directly derived from the previous sections for classification purpose, the classifier inputs are standardized in order to derive more impartial results. In many practical situations, the classification model may receive input features whose values lie within different dynamic ranges. Thus, features with large values may inadvertently influence classification over features with small values. In this research, each feature is separately standardized by subtracting its mean and dividing by the standard deviation. Standardization allows for the centroid of the data to be moved to the origin along with stretching or compressing the data according to individual feature's standard deviation. This keeps the data in the same dynamic range according to the standard deviation.

The classification model used in this paper is a neural network with Gaussian kernels as a supervised classification network model [2], [5]. A schematic view of the network is shown in Fig. 2. The network model typically includes three layers: input layer, hidden layer, and output layer. The first layer is an input layer with an input vector $\boldsymbol{p} = [p_1, p_2, \ldots, p_n]' \in \mathbb{R}^n$. The second layer is a hidden layer consisting of nonlinear

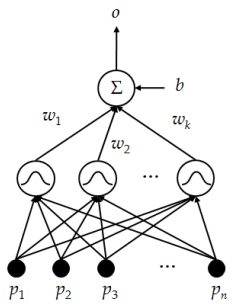


Fig. 2. A neural network model with Gaussian kernels

transformation functions $f_i(\boldsymbol{p}) : \mathbb{R}^n \rightarrow \mathbb{R}$ known as the activation functions having the standard radial basis functions (RBF). The Gaussian function

$$f_i(\boldsymbol{p}) = e^{-\frac{(\boldsymbol{p}-c_i)^2}{s}}$$

is often used as a radial basis function. The variable c_i is the center of the i^{th} receptive field and $s = \frac{(-\log(0.5))^{1/2}}{spread}$ is referred to as the width of the Gaussian function. The spread $spread$ is a constant used to ensure that neurons respond strongly to overlapping regions of the input space [5]. The width of the Gaussian function s is set to be the same as in [5]. The third layer is a linear mapping collecting the weighted outputs from the hidden layer allowing for a class label to be assigned. The output o is described by the equation

$$o = b + \sum_{i=1}^{k} w_i f_i(\boldsymbol{p}), \tag{7}$$

where b is a threshold value and w_i indicating the weight is a scalar. For the application on steganography detection, consider a feature vector as the input, the output indicates a given image belongs to either a clean or a steganography image category.

VI. Experiments

The clean image set used in this paper are from [1]. The steganography data sets are created using carrier images, steganography message to be embedded and two JPEG steganography methods, JPEG-JSteg v1 [13] and JPEG-JSteg v4 [15]. The hidden data consists of three text files containing 212 characters, 320 characters and 1040 characters, respectively. Each character is represented as a byte. Correspondingly, 1040 characters need approximately the same memory size as a 32×32 grayscale image when a pixel value is stored as a byte. To generate the features using the fixed effects models as well as training the classification model, 40 images of size 512×512 are randomly selected from each dataset. To ensure proper training of the classification model, the clean images and the six sets of steganography images do not overlap.

A N-fold cross validation is used in training and testing the proposed feature generation method, where $N = 10$ [10]. Additionally, the network for classification is designed to have a $spread = 10$. The training set of images consists of randomly chosen clean images and the steganographic images for each fold. By applying the steganography detection system on the above mentioned steganography methods, percentage confusion matrices are demonstrated for each method in Tables II and III. The overall classification accuracy of the classification rule can be calculated by averaging the diagonal elements in the confusion matrix. However, in steganography detection systems, it is more important not to misclassify a steganography image as a clean image (*false positive*) as compared to misclassifying a clean image as a steganography image (*false negative*). The consequences of a false positive would be very expensive from the security point of view for steganalysts. Thus, in this paper, the primary focus is on the classification accuracy of steganography images (*true positive*), consequently the confusion matrices for all the experiments are reported rather than reporting only the overall classification accuracy.

TABLE II
Classification Accuracies (%) for JPEG-JSteg v1

	Observed					
	Prev. Method (72 Features)		Prev. Method (36 Features)		Our Method (36 Features)	
	Stego	Clean	Stego	Clean	Stego	Clean
Stego 0.21KB	66.0	34.0	79.0	21.0	98.3	1.7
Clean	36.7	63.3	26.7	73.3	11.8	88.2
Stego 0.32KB	70.2	29.8	76.5	23.5	98.3	1.7
Clean	34.7	65.3	24.3	75.7	11.8	88.2
Stego 1.04KB	87.5	12.5	82.8	17.2	98.3	1.7
Clean	22.2	77.8	19.3	80.7	8.3	91.7

TABLE III
Classification Accuracies (%) for JPEG-JSteg v4

	Observed					
	Prev. Method (72 Features)		Prev. Method (36 Features)		Our Method (36 Features)	
	Stego	Clean	Stego	Clean	Stego	Clean
Stego 0.21KB	61.9	38.1	74.3	25.7	100.0	0.0
Clean	43.3	56.7	30.3	69.7	11.8	88.2
Stego 0.32KB	60.9	39.1	78.7	21.3	98.3	1.7
Clean	43.3	56.7	23.7	76.3	11.8	88.2
Stego 1.04KB	80.2	19.8	74.5	25.5	100.0	0.0
Clean	18.2	81.8	21.8	78.2	10.3	89.7

Tables II and III give the classification accuracies (%) in nine confusion matrices. The results are shown for three sets of features; the previous method described in Section III with all 72 features, 36 error features from log error measures only, and 36 error features from log error measures for the new method presented in Section IV. The clean images are compared with JPEG-JSteg v1 and JPEG-JSteg v4 Stego images for 0.21KB, 0.32KB and 1.04KB steganography file size respectively. The columns represent the observed values and the rows represent the predicted values. It can be seen that with the integration of random effects the percentage of the classification accuracies is

improved for both steganography and clean images by as much as 38% in comparison to the previous fixed effects method. It should also be noticed that using the 36 error features from Section III instead of all 72 features improves the classification accuracies of steganography images, except the case of high amounts of hidden information. Note, the new method in each case classifies the steganography images with more than 98% accuracy for each of the steganography embedding methods.

VII. Conclusion

The need to extract hidden information from a steganography file is necessary for law enforcement to build a criminal case if it is to hold up in court. This problem of extraction leads to an intermediate step of identifying the embedding method used to create the steganography file. One of the goals of detection is to generate a set of features that is capable of detecting small amounts of hidden information. The results in Section VI indicate that the mixed effects models maintain the consistency of the classification accuracy no matter how much information is hidden within images, while Farid's method has a low detection accuracy in detecting small amounts of hidden information. This finding is important for steganalysts in understanding that a combination of several feature generation methods is important when attempting to solve detection of steganography. In future work, more embedding methods will be experimented with the proposed method. Additionally, the feature generation methods will be combined and feature selection will be used in attempts to increase the detection of various steganography methods.

Acknowledgment

The authors would like to thank the Department of Electrical and Computer Engineering along with the College of Business at the University of Texas at San Antonio for supporting this research. A special thanks goes to Hany Farid for providing his code online [6].

References

[1] BOWS2 Image Set, http://bows2.gipsa-lab.inpg.fr/.

[2] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," IEEE Transactions on Neural Networks, vol. 2, no. 2, pp. 302-309, 1991.

[3] M. Conway, "Code wars: steganography, signals intelligence, and terrorism," Knowledge, Technology and Policy, vol. 16, no. 2, pp. 45-62, 2003.

[4] E. P. Cunningham and C. R. Henderson, "An iterative procedure for estimating fixed effects and variance components in mixed model situations," Biometrics, vol. 24, no. 1, pp. 13-25, 1968.

[5] H. Demuth, M. Beale, and M. Hagan, Raidal Basis Networks, Neural Network Toolbox^{TM} 6 User's Guide, The MathWorks, Inc., 2009.

[6] H. Farid, "Detecting hidden messages using higher-order statistics," Proceedings of the International Conference on Image Processing, vol. 2, pp. 905-908, 2002.

[7] C. R. Henderson, Application of Linear Models in Animal Breeding, Guelph: University of Guelph Press, 1984.

[8] N. F. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," IEEE Computer, pp. 26-34, 1998.

[9] D. Kahn, "The history of steganography," Proceedings of the First International Workshop on Information Hiding, Lecture Notes in Computer Science, pp. 1-5, 1996.

[10] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, vol. 2, pp. 1137-1145, 1995.

[11] N. M Laird and J. H. Ware, "Random-effects models for longitudinal data," Biometrics, vol. 38, pp. 963-974, 1982.

[12] National Institute of Justice, "Electronic crime scene investigation: A guide for first responders," Second Edition, 2008, http://www.ncjrs.gov/pdffiles1/nij/219941.pdf.

[13] N. Provos and P. Honneyman, "Hide and seek: an introduction to steganography," IEEE Security and Privacy Magazine, vol. 1, no. 3, pp. 32-44, 2003.

[14] B. M. Rodriguez and G. L. Peterson, "Multi-class classification fusion using boosting for identifying steganography methods," Proceedings of SPIE, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications, vol. 6974, pp. 697407.1-697407.10, 2008.

[15] D. Upham, JPEG-JSteg-v4, http://www.nic.funet.fi/pub/crypt/steganography/.

[16] J. Wingate, "The perfect dead drop," White Paper, Backbone Security, 2005.