Detecting Hidden Messages Using Higher-Order Statistics and Support Vector Machines

Siwei Lyu and Hany Farid

Dartmouth College, Hanover NH 03755, USA,
{lsw,farid}@cs.dartmouth.edu,
www.cs.dartmouth.edu/~{lsw,farid}

Abstract. Techniques for information hiding have become increasingly more sophisticated and widespread. With high-resolution digital images as carriers, detecting hidden messages has become considerably more difficult. This paper describes an approach to detecting hidden messages in images that uses a wavelet-like decomposition to build higher-order statistical models of natural images. Support vector machines are then used to discriminate between untouched and adulterated images.

1 Introduction

Information hiding techniques (e.g., steganography and watermarking) have recently received quite a bit of attention (see [12, 1, 10, 15] for general reviews). With digital images as carriers, detecting the presence of hidden messages poses significant challenges. Although the presence of embedded messages is often imperceptible to the human eye, it may nevertheless disturb the statistics of an image. Previous approaches to detecting such deviations [11, 28, 17] typically examine first-order statistical distributions of intensity or transform coefficients (e.g., discrete cosine transform, DCT). The drawback of this analysis is that simple counter-measures that match first-order statistics are likely to foil detection. In contrast, the approach taken here relies on building higher-order statistical models for natural images [13, 19, 29, 14, 21] and looking for deviations from these models. We show that, across a large number of natural images, there exist strong higher-order statistical regularities within a wavelet-like decomposition, see also [9]. The embedding of a message significantly alters these statistics and thus becomes detectable. Support vector machines (linear and non-linear) are employed to detect these statistical deviations.

2 Image Statistics

The decomposition of images using basis functions that are localized in spatial position, orientation, and scale (e.g., wavelets) has proven extremely useful in a range of applications (e.g., image compression, image coding, noise removal, and texture synthesis). One reason is that such decompositions exhibit statistical

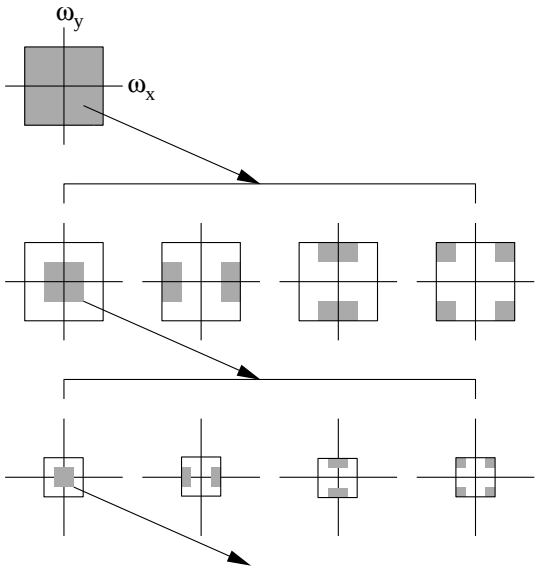

Fig. 1. An idealized multi-scale and orientation decomposition of frequency space. Shown, from top to bottom, are levels 0, 1, and 2, and from left to right, are the lowpass, vertical, horizontal, and diagonal subbands.

regularities that can be exploited (e.g., [20, 18, 2]). Described below is one such decomposition, and a set of statistics collected from this decomposition.

The decomposition employed here is based on separable quadrature mirror filters (QMFs) [23, 26, 22]. As illustrated in Fig. 1, this decomposition splits the frequency space into multiple scales and orientations. This is accomplished by applying separable lowpass and highpass filters along the image axes generating a vertical, horizontal, diagonal and lowpass subband. Subsequent scales are created by recursively filtering the lowpass subband. The vertical, horizontal, and diagonal subbands at scale $i = 1, ..., n$ are denoted as $V_i(x, y)$, $H_i(x, y)$, and $D_i(x, y)$, respectively. Shown in Fig. 2 is a three-level decomposition of a "disc" image.

Given this image decomposition, the statistical model is composed of the mean, variance, skewness and kurtosis of the subband coefficients at each orientation and at scales $i = 1, ..., n - 1$. These statistics characterize the basic coefficient distributions. The second set of statistics is based on the errors in an optimal linear predictor of coefficient magnitude. As described in [2], the subband coefficients are correlated to their spatial, orientation and scale neighbors. For purposes of illustration, consider first a vertical band, $V_i(x, y)$, at scale i. A linear predictor for the magnitude of these coefficients in a subset of all possible

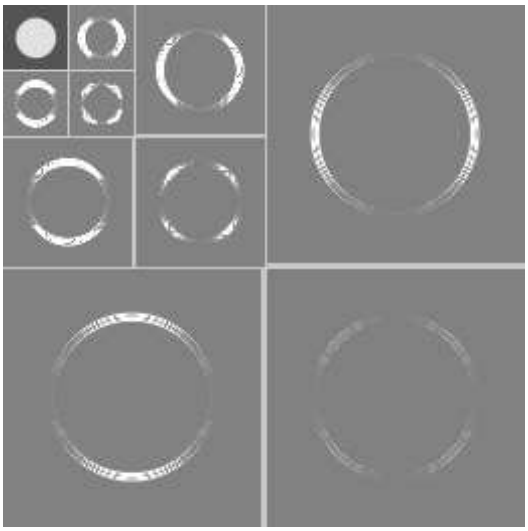

Fig. 2. Shown are the absolute values of the subband coefficients at three scales and three orientations for a "disc" image. The residual lowpass subband is shown in the upper-left corner.

neighbors [1] is given by:

$$\begin{aligned} V_i(x,y) = {} & w_1 V_i(x-1,y) + w_2 V_i(x+1,y) \\ & + w_3 V_i(x,y-1) + w_4 V_i(x,y+1) \\ & + w_5 V_{i+1}(x/2,y/2) + w_6 D_i(x,y) \\ & + w_7 D_{i+1}(x/2,y/2), \end{aligned} \tag{1}$$

where w_k denotes scalar weighting values. This linear relationship is expressed more compactly in matrix form as:

$$\boldsymbol{V} = Q\boldsymbol{w}, \tag{2}$$

where the column vector $\boldsymbol{w} = (w_1 \quad \ldots \quad w_7)^T$, the vector $\boldsymbol{V}$ contains the coefficient magnitudes of $V_i(x,y)$ strung out into a column vector, and the columns of the matrix Q contain the neighboring coefficient magnitudes as specified in Equation (1) also strung out into column vectors. The coefficients are determined by minimizing the quadratic error function:

$$E(\boldsymbol{w}) = [\boldsymbol{V} - Q\boldsymbol{w}]^2. \tag{3}$$

This error function is minimized by differentiating with respect to $\boldsymbol{w}$:

$$dE(\boldsymbol{w})/d\boldsymbol{w} = 2Q^T[\boldsymbol{V} - Q\boldsymbol{w}], \tag{4}$$

[1] The particular choice of spatial, orientation and scale neighbors was motivated by the observations of [2] and modified to include non-casual neighbors.

setting the result equal to zero, and solving for $\boldsymbol{w}$ to yield:

$$\boldsymbol{w} = (Q^T Q)^{-1} Q^T \boldsymbol{V}. \tag{5}$$

The log error in the linear predictor is then given by:

$$\boldsymbol{E} = \log_2(\boldsymbol{V}) - \log_2(|Q\boldsymbol{w}|). \tag{6}$$

It is from this error that additional statistics are collected, namely the mean, variance, skewness, and kurtosis. This process is repeated for each vertical subband at scales $i = 1, ..., n-1$, where at each scale a new linear predictor is estimated. A similar process is repeated for the horizontal and diagonal subbands. The linear predictor for the horizontal subbands is of the form:

$$\begin{aligned} H_i(x,y) = {} & w_1 H_i(x-1,y) + w_2 H_i(x+1,y) \\ & + w_3 H_i(x,y-1) + w_4 H_i(x,y+1) \\ & + w_5 H_{i+1}(x/2,y/2) + w_6 D_i(x,y) \\ & + w_7 D_{i+1}(x/2,y/2), \end{aligned} \tag{7}$$

and for the diagonal subbands:

$$\begin{aligned} D_i(x,y) = {} & w_1 D_i(x-1,y) + w_2 D_i(x+1,y) \\ & + w_3 D_i(x,y-1) + w_4 D_i(x,y+1) \\ & + w_5 D_{i+1}(x/2,y/2) + w_6 H_i(x,y) \\ & + w_7 V_i(x,y). \end{aligned} \tag{8}$$

The same error metric, Equation (6), and error statistics computed for the vertical subbands, are computed for the horizontal and diagonal bands, for a total of $12(n-1)$ error statistics. Combining these statistics with the $12(n-1)$ coefficient statistics yields a total of $24(n-1)$ statistics that form a feature vector which is used to discriminate between images that contain hidden messages and those that do not.

3 Classification

From the measured statistics of a training set of images with and without hidden messages, the goal is to determine whether a test image contains a message. In earlier work [6], we performed this classification using a Fisher linear discriminant (FLD) analysis [7, 5]. Here a more flexible support vector machine (SVM) classifier is employed [24, 25, 3]. We briefly describe, in increasing complexity, three classes of SVMs. The first, linear separable case is mathematically the most straight-forward. The second, linear non-separable case, contends with situations in which a solution cannot be found in the former case, and is most similar to a FLD. The third, non-linear case, affords the most flexible classification scheme and, in and our application, the best classification accuracy. For simplicity a two-class SVM is described throughout.

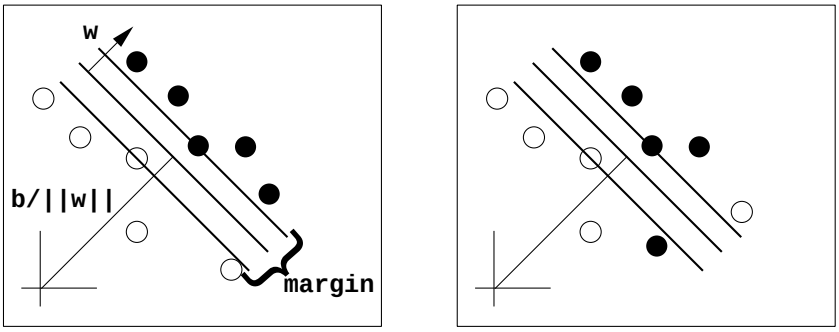

Fig. 3. Linear (a) separable and (b) non-separable support vector machines. Shown is a toy example of a two-class discriminator (white and black dots) for data in $\mathcal{R}^2$.

3.1 Linear Separable SVM

Denote the tuple $(\boldsymbol{x}_i, y_i)$, $i = 1, ..., N$ as exemplars from a training set of images with and without hidden messages. The column vector $\boldsymbol{x}_i$ contains the measured image statistics as outlined in the previous section, and $y_i = -1$ for images with a hidden message, and $y_i = 1$ for images without a hidden message. The linear separable SVM classifier amounts to a hyperplane that separates the positive and negative exemplars, Fig. 3(a). Points which lie on the hyperplane satisfy the constraint:

$$\boldsymbol{w}^t \boldsymbol{x}_i + b = 0, \tag{9}$$

where $\boldsymbol{w}$ is normal to the hyperplane, $|b|/||\boldsymbol{w}||$ is the perpendicular distance from the origin to the hyperplane, and $||\cdot||$ denotes the Euclidean norm. Define now the margin for any given hyperplane to be the sum of the distances from the hyperplane to the nearest positive and negative exemplar, Fig. 3(a). The separating hyperplane is chosen so as to maximize the margin. If a hyperplane exists that separates all the data then, within a scale factor:

$$\boldsymbol{w}^t \boldsymbol{x}_i + b \geq 1, \qquad \text{if } y_i = 1 \tag{10}$$

$$\boldsymbol{w}^t \boldsymbol{x}_i + b \leq -1, \qquad \text{if } y_i = -1. \tag{11}$$

These pair of constraints can be combined into a single set of inequalities:

$$(\boldsymbol{w}^t \boldsymbol{x}_i + b)\ y_i - 1 \geq 0, \qquad i = 1, ..., N. \tag{12}$$

For any given hyperplane that satisfies this constraint, the margin is $2/||\boldsymbol{w}||$. We seek, therefore, to minimize $||\boldsymbol{w}||^2$ subject to the constraints in Equation (12).

For largely computational reasons, this optimization problem is reformulated using Lagrange multipliers, yielding the following Lagrangian:

$$L(\boldsymbol{w}, b, \alpha_1, ..., \alpha_N) = \frac{1}{2}||\boldsymbol{w}||^2 - \sum_{i=1}^{N} \alpha_i\ (\boldsymbol{w}^t \boldsymbol{x}_i + b)\ y_i + \sum_{i=1}^{N} \alpha_i, \tag{13}$$

where α_i are the positive Lagrange multipliers. This error function should be minimized with respect to $\boldsymbol{w}$ and b, while requiring that the derivatives of $L(\cdot)$ with respect to each α_i is zero and constraining $\alpha_i \geq 0$, for all i. Because this is a convex quadratic programming problem, a solution to the dual problem yields the same solution for $\boldsymbol{w}$, b, and $\alpha_1, ..., \alpha_N$. In the dual problem, the same error function $L(\cdot)$ is maximized with respect to α_i, while requiring that the derivatives of $L(\cdot)$ with respect to $\boldsymbol{w}$ and b are zero and the constraint that $\alpha_i \geq 0$. Differentiating with respect to $\boldsymbol{w}$ and b, and setting the results equal to zero yields:

$$\boldsymbol{w} = \sum_{i=1}^{N} \alpha_i \boldsymbol{x}_i y_i \tag{14}$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0. \tag{15}$$

Substituting these equalities back into Equation (13) yields:

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j \boldsymbol{x}_i^t \boldsymbol{x}_j y_i y_j. \tag{16}$$

Maximization of this error function may be realized using any of a number of general purpose optimization packages that solve linearly constrained convex quadratic problems (see e.g., [8]).

A solution to the linear separable classifier, if it exists, yields values of α_i, from which the normal to the hyperplane can be calculated as in Equation (14), and from the Karush-Kuhn-Tucker [8] (KKT) condition:

$$b = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \boldsymbol{w}^t \boldsymbol{x}_i \right), \tag{17}$$

for all i, such that $\alpha_i \neq 0$. From the separating hyperplane, $\boldsymbol{w}$ and b, a novel exemplar, $\boldsymbol{z}$, can be classified by simply determining on which side of the hyperplane it lies. If the quantity $\boldsymbol{w}^t \boldsymbol{z} + b$ is greater than or equal to zero, then the exemplar is classified as not having a hidden message, otherwise the exemplar is classified as containing a hidden message.

3.2 Linear Non-Separable SVM

It is possible, and even likely, that the linear separable SVM will not yield a solution when, for example, the training data do not uniformly lie on either side of a separating hyperplane, as illustrated in Fig. 3(b). Such a situation can be handled by softening the initial constraints of Equation (10) and (11). Specifically, these constraints are modified with "slack" variables, ξ_i, as follows:

$$\boldsymbol{w}^t \boldsymbol{x}_i + b \geq 1 - \xi_i, \qquad \text{if } y_i = 1 \tag{18}$$

$$\boldsymbol{w}^t \boldsymbol{x}_i + b \leq -1 + \xi_i, \qquad \text{if } y_i = -1, \tag{19}$$

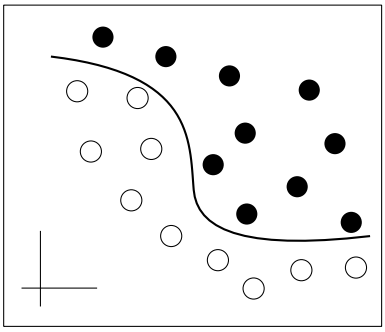

Fig. 4. Non-linear support vector machine, as compared with the linear support vector machine of Fig. 3.

with $\xi_i \geq 0,\ i = 1, ..., N$. A training exemplar which lies on the "wrong" side of the separating hyperplane will have a value of ξ_i greater than unity. We seek a hyperplane that minimizes the total training error, $\sum_i\ \xi_i$, while still maximizing the margin. A simple error function to be minimized is $||\boldsymbol{w}||^2/2 + C\sum_i\ \xi_i$, where C is a user selected scalar value, whose chosen value controls the relative penalty for training errors. Minimization of this error is still a quadratic programming problem. Following the same procedure as the previous section, the dual problem is expressed as maximizing the error function:

$$L_D = \sum_{i=1}^{N}\ \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j\boldsymbol{x}_i^t\boldsymbol{x}_j y_i y_j, \tag{20}$$

with the constraint that $0 \leq \alpha_i \leq C$. Note that this is the same error function as before, Equation (16) with the slightly different constraint that α_i is bounded above by C. Maximization of this error function and computation of the hyperplane parameters are accomplished as described in the previous section.

3.3 Non-Linear SVM

Fundamental to the SVMs outlined in the previous two sections is the limitation that the classifier is constrained to a linear hyperplane. Shown in Fig. 4 is an example of where a non-linear separating surface would greatly improve the classification accuracy. Non-linear SVMs afford such a classifier by first mapping the training exemplars into a higher (possibly infinite) dimensional Euclidean space in which a linear SVM is then employed. Denote this mapping as:

$$\Phi : \mathcal{L} \rightarrow \mathcal{H}, \tag{21}$$

which maps the original training data from $\mathcal{L}$ into $\mathcal{H}$. Replacing $\boldsymbol{x_i}$ with $\Phi(\boldsymbol{x_i})$ everywhere in the training portion of the linear separable or non-separable SVMs of the previous sections yields an SVM in the higher-dimensional space $\mathcal{H}$.

It can, unfortunately, be quite inconvenient to work in the space $\mathcal{H}$ as this space can be considerably larger than the original $\mathcal{L}$, or even infinite. Note, however, that the error function of Equation (20) to be maximized depends only on the inner products of the training exemplars, $\boldsymbol{x}_i^t \boldsymbol{x}_j$. Given a "kernel" function such that:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i)^t \Phi(\boldsymbol{x}_j), \tag{22}$$

an explicit computation of Φ can be completely avoided. There are several choices for the form of the kernel function, for example, radial basis functions or polynomials. Replacing the inner products $\Phi(\boldsymbol{x}_i)^t \Phi(\boldsymbol{x}_j)$ with the kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ yields an SVM in the space $\mathcal{H}$ with minimal computational impact over working in the original space $\mathcal{L}$.

With the training stage complete, recall that a novel exemplar, $\boldsymbol{z}$, is classified by determining on which side of the separating hyperplane (specified by $\boldsymbol{w}$ and b) it lies. Specifically, if the quantity $\boldsymbol{w}^t \Phi(\boldsymbol{z}) + b$ is greater than or equal to zero, then the exemplar is classified as not having a hidden message, otherwise the exemplar is classified as containing a hidden message. The normal to the hyperplane, $\boldsymbol{w}$, of course now lives in the space $\mathcal{H}$, making this testing impractical. As in the training stage, the classification can again be performed via inner products. From Equation (14):

$$\begin{aligned} \boldsymbol{w}^t \Phi(\boldsymbol{z}) + b &= \sum_{i=1}^{N} \alpha_i \Phi(\boldsymbol{x}_i)^t \Phi(\boldsymbol{z}) y_i + b \\ &= \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{z}) y_i + b. \end{aligned} \tag{23}$$

Thus both the training and classification can be performed in the higher-dimensional space, affording a more flexible separating hyperplane and hence better classification accuracy. We next show the performance of a linear non-separable and non-linear SVM in the detection of hidden messages. The SVMs classify images based on the 72-dimensional feature vector as described in Section 2.

4 Results

Shown in Fig. 5 are several examples taken from a database of natural images [2]. These images span decades of digital and traditional photography and consist of a broad range of intdoor and outdoor scenes. Each 8-bit per channel RGB image is cropped to a central 640×480 pixel area. Statistics from 1,800 such images are collected as follows. Each image is first converted from RGB to gray-scale (gray = 0.299R + 0.587G + 0.114B). A four-level, three-orientation QMF pyramid is constructed for each image, from which a 72-length feature vector of coefficient

[2] Images were downloaded from: `philip.greenspun.com` and reproduced here with permission from Philip Greeenspun.

Fig. 5. Sample images.

and error statistics is collected, Section 2. To reduce sensitivity to noise in the linear predictor, only coefficient magnitudes greater than 1.0 are considered. The training set of "no-steg" statistics comes from either 1,800 JPEG images (quality ≈ 75), 1,800 GIF images (LZW compression), or 1,800 TIFF images (no compression). The GIF and TIFF images are converted from their original JPEG format.

Messages are embedded into JPEG images using either Jsteg[3] or OutGuess[4] (run with $(+)$ and without $(-)$ statistical correction). Jsteg and OutGuess are transform-based systems that embed messages by modulating the DCT coefficients. Unique to OutGuess is a technique for embedding into only one-half of the redundant bits and then using the remaining redundant bits to preserve the first-order distribution of DCT coefficients [16]. Messages are embedded into GIF images using EzStego[5] which modulates the least significant bits of the sorted color palette index. Messages are embedded into the TIFF images using a generic LSB embedding that modulates the least-significant bit of a random subset of the pixel intensities. In each case, a message consists of a $n \times n$ pixel ($n \in [32, 256]$) central portion of a random image chosen from the same image database. After the message is embedded into the cover image, the same transformation, decomposition, and collection of statistics as described above is performed. In all cases the embedded message consists of only the raw pixel intensities (i.e., no image headers).

Shown in Fig. 6 is an example cover and message image, and the result of embedding the message into the cover image. In this example, the mean of the absolute value of the difference between the cover and steg image is 3.1 intensity values with a standard deviation of 3.2. For display purposes the difference image is renormalized into the range $[0, 255]$.

In the first set of results linear non-separable SVMs, implemented using the freely available package LIBSVM [4], were separately trained to classify the

[3] Jsteg V4, by Derek Upham, is available at `ftp.funet.fi`
[4] OutGuess, by Niels Provos, is available at `www.outguess.org`
[5] EZStego, by Romana Machado, is available at `www.stego.com`

cover message steg=cover+message |cover - steg|

Fig. 6. Shown is a cover image and a steg image containing an embedded message. Also shown is the the 256×256 message (at scale), and the absolute value of the difference between the cover and steg image (renormalized into the range [0,255] for display purposes).

JPEG, GIF and TIFF embeddings. In each case, the training set consists of the 1,800 "no-steg" images, and a random subset of 1,800 "steg" images embedded either with OutGuess$^+$, EzStego or LSB, and with varying message sizes. [6] The SVM parameters were chosen to yield a 1.0% false-positive rate. The trained SVM is then used to classify all of the remaining previously unseen steg images of the same format, Table 1. In this table, the columns correspond to separate classification results for JPEG, GIF and TIFF format images. Note that the JPEG classifier generalizes to the different embedding programs not previously seen by the classifier. Also shown in this table are results from classification employing a Fisher linear discriminant analysis used in our earlier work [6]. Both the FLD and SVM classifiers employ a linear separating hyperplane for classification so, as expected, performance is similar across these different classifiers.

Shown in Table 2 are classification results for a non-linear SVM (using a radial basis kernel function), also implemented using LIBSVM [4]. In this table results are shown for a 1.0% and 0.0% false positive rate. Note the significant improvement over the linear classifiers of Table 1. Shown in Fig. 7 is a graphical comparison of all of these results.

Lastly, we also tested detection accuracy under the F5 embedding algorithm [27]. Detection, as described above, was just slightly above chance. Significantly better detection rates were achieved by collecting statistics from within and across all three color channels (as opposed to analyzing only a grayscale converted image). In this case basic coefficient statistics are collected from all three RGB color channels, and the linear predictor incorporates color as well

[6] OutGuess is run with unlimited iterations to find the best embedding. OutGuess imposes limits on the message size, so not all images were able to be used for cover. This is significant only for message sizes of 256×256, where less than 300 steg images were generated.

Embedding	Messsage	FLD			SVM		
		JPEG	GIF	TIFF	JPEG	GIF	TIFF
Jsteg	256×256	94.0	-	-	92.8	-	-
Jsteg	128×128	95.7	-	-	94.3	-	-
Jsteg	64×64	95.3	-	-	94.3	-	-
Jsteg	32×32	51.7	-	-	52.2	-	-
OutGuess$^-$	256×256	92.8	-	-	92.8	-	-
OutGuess$^-$	128×128	63.4	-	-	63.2	-	-
OutGuess$^-$	64×64	27.7	-	-	28.6	-	-
OutGuess$^-$	32×32	5.9	-	-	6.9	-	-
OutGuess$^+$	256×256	74.4	-	-	78.4	-	-
OutGuess$^+$	128×128	41.4	-	-	40.7	-	-
OutGuess$^+$	64×64	14.0	-	-	14.8	-	-
OutGuess$^+$	32×32	4.1	-	-	3.7	-	-
EzStego	194×194	-	45.2	-	-	44.5	-
EzStego	128×128	-	13.8	-	-	14.3	-
EzStego	64×64	-	2.9	-	-	2.5	-
EzStego	32×32	-	1.6	-	-	1.5	-
LSB	194×194	-	-	42.3	-	-	42.7
LSB	128×128	-	-	16.8	-	-	17.2
LSB	64×64	-	-	2.8	-	-	2.2
LSB	32×32	-	-	1.3	-	-	1.3

Table 1. Classification accuracy (percent) with 1.0% false positives for varying message sizes (the maximum message size for EzStego and LSB is 194×194). Classification is from an earlier Fisher Linear discriminant (FLD) analysis [6] and a linear non-separable SVM, Section 3.2. As expected, the classification accuracy is comparable. See also Fig. 7.

as scale, orientation, and spatial neighbors. The resulting feature vector is of size 216 (3×72). With a 0.0% false positive rate, a non-linear SVM is able to correctly detect messages sizes of 160×160, 128×128, 64×64, and 32×32 with an accuracy of 90.2%, 83.1%, 57.0%, and 42.2%, respectively.

5 Discussion

Messages can be embedded into digital images in ways that are imperceptible to the human eye, and yet, these manipulations can fundamentally alter the underlying statistics of an image. To detect the presence of hidden messages we have employed a model based on statistics taken from a multi-scale decomposition. This model includes basic coefficient statistics as well as error statistics from an optimal linear predictor of coefficient magnitude. These higher-order statistics appear to capture certain properties of "natural" images, and more importantly, these statistics are significantly altered when a message is embedded within an image. This makes it possible to detect, with a reasonable degree of accuracy,

Embedding	Messsage	SVM(1.0%) JPEG	GIF	TIFF	SVM(0.0%) JPEG	GIF	TIFF
Jsteg	256×256	99.0	-	-	98.5	-	-
Jsteg	128×128	99.3	-	-	99.0	-	-
Jsteg	64×64	99.1	-	-	98.7	-	-
Jsteg	32×32	86.0	-	-	74.5	-	-
OutGuess$^-$	256×256	98.9	-	-	97.1	-	-
OutGuess$^-$	128×128	93.8	-	-	85.8	-	-
OutGuess$^-$	64×64	72.6	-	-	53.1	-	-
OutGuess$^-$	32×32	33.2	-	-	14.4	-	-
OutGuess$^+$	256×256	95.6	-	-	89.5	-	-
OutGuess$^+$	128×128	82.2	-	-	63.7	-	-
OutGuess$^+$	64×64	54.7	-	-	32.1	-	-
OutGuess$^+$	32×32	21.4	-	-	7.2	-	-
EzStego	194×194	-	77.2	-	-	76.9	-
EzStego	128×128	-	39.2	-	-	36.6	-
EzStego	64×64	-	6.5	-	-	4.6	-
EzStego	32×32	-	2.7	-	-	1.5	-
LSB	194×194	-	-	78.0	-	-	77.0
LSB	128×128	-	-	44.7	-	-	40.5
LSB	64×64	-	-	6.2	-	-	4.2
LSB	32×32	-	-	1.9	-	-	1.3

Table 2. Classification accuracy (percent) with 1.0% or 0.0% false positives and for varying message sizes (the maximum message size for EzStego and LSB is 194×194). Classification is from a non-linear SVM, Section 3.3. Note the significant improvement in accuracy as compared to a linear classifier, Table 1. See also Fig. 7.

the presence of hidden messages in digital images. This detection is achieved with either linear or non-linear pattern classification techniques, with the latter providing significantly better performance. To avoid detection, of course, one need only embed a small enough message that does not significantly disturb the image statistics.

Although not tested here, it is likely that the presence of digital watermarks could also be detected. Since one of the goals of watermarking is robustness to attack and not necessarily concealment, watermarks typically alter the image in a more substantial way. As such, it is likely that the underlying statistics will be more significantly disrupted. Although only tested on images, there is no inherent reason why the approaches described here would not work for audio signals or video sequences.

The techniques described here would almost certainly benefit from several extensions: (1) the higher-order statistical model should incorporate correlations within and between all three color channels; (2) the classifier should be trained separately on different classes of images (e.g., indoor vs. outdoor); and (3) the classifier should be trained separately on images with varying compression rates.

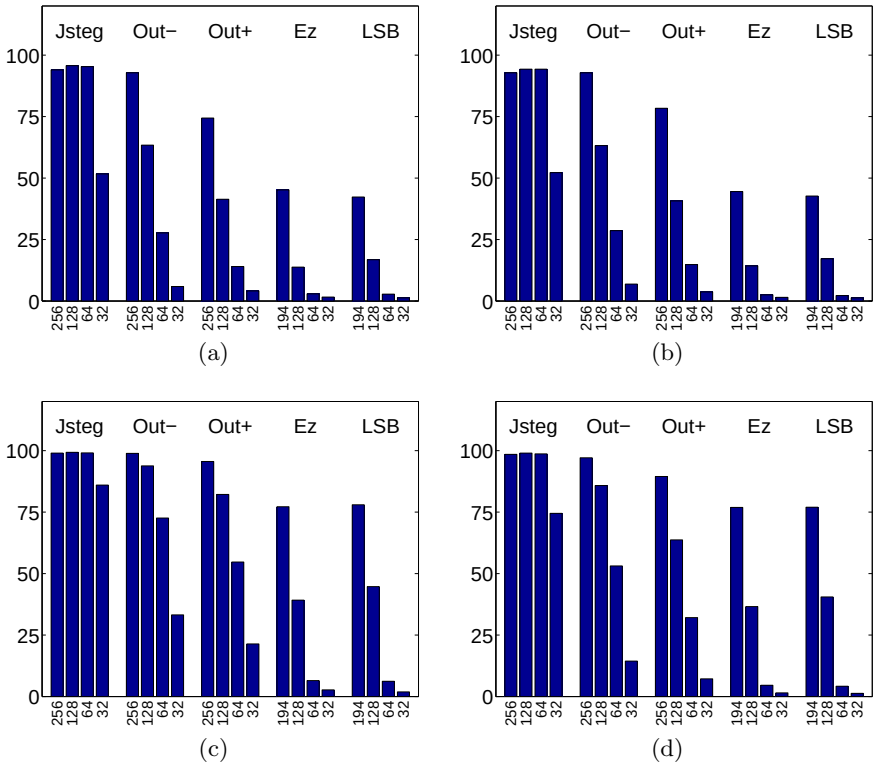

Fig. 7. Classification accuracy for (a) Fisher linear discriminant with 1.0% false positives; (b) linear non-separable SVM with 1.0% false positives; (c) non-linear SVM with 1.0% false positives; and (d) non-linear SVM with 0.0% false positive rates. The values along the horizontal axis denote the size of the embedded message. See also Tables 1 and 2.

One benefit of the higher-order models employed here is that they are not as vulnerable to counter-attacks that match first-order statistical distributions of pixel intensity or transform coefficients. It is possible, however, that countermeasures will be developed that can foil the detection scheme outlined here. The development of such techniques will in turn lead to better detection schemes, and so on.

Acknowledgments

This work is supported by an Alfred P. Sloan Fellowship, a National Science Foundation CAREER Award (IIS-99-83806), a Department of Justice Grant (2000-DT-CS-K001), and a departmental National Science Foundation Infrastructure Grant (EIA-98-02068).

References

1. R.J. Anderson and F.A.P. Petitcolas. On the limits of steganography. *IEEE Journal on Selected Areas in Communications*, 16(4):474–481, 1998.
2. R.W. Buccigrossi and E.P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Transactions on Image Processing*, 8(12):1688–1701, 1999.
3. C.J.C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.
4. Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.
5. R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.
6. H. Farid. Detecting hidden messages using higher-order statistical models. In *International Conference on Image Processing*, page (to appear), Rochester, New York, 2002.
7. R. Fisher. The use of multiple measures in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.
8. R. Fletcher. *Practical Methods of Optimization*. John Wiley and Sons, 2nd edition, 1987.
9. J. Fridrich and M. Goljan. Practical steganalysis: State of the art. In *SPIE Photonics West, Electronic Imaging*, San Jose, CA, 2002.
10. N. Johnson and S. Jajodia. Exploring steganography: seeing the unseen. *IEEE Computer*, pages 26–34, 1998.
11. N. Johnson and S. Jajodia. Steganalysis of images created using current steganography software. *Lecture notes in Computer Science*, pages 273–289, 1998.
12. D. Kahn. The history of steganography. In *Proceedings of Information Hiding, First International Workshop*, Cambridge, UK, 1996.
13. D. Kersten. Predictability and redundancy of natural images. *Journal of the Optical Society of America A*, 4(12):2395–2400, 1987.
14. G. Krieger, C. Zetzsche, and E. Barth. Higher-order statistics of natural images and their exploitation by operators selective to intrinsic dimensionality. In *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics*, pages 147–151, Banff, Alta., Canada, 1997.
15. E.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn. Information hiding - a survey. *Proceedings of the IEEE*, 87(7):1062–1078, 1999.
16. N. Provos. Defending against statistical steganalysis. In *10th USENIX Security Symposium*, Washington, DC, 2001.
17. N. Provos and P. Honeyman. Detecting steganographic content on the internet. Technical Report CITI 01-1a, University of Michigan, 2001.
18. R. Rinaldo and G. Calvagno. Image coding by block prediction of multiresolution subimages. *IEEE Transactions on Image Processing*, 4(7):909–920, 1995.
19. D.L. Ruderman and W. Bialek. Statistics of natural image: Scaling in the woods. *Phys. Rev. Letters*, 73(6):814–817, 1994.
20. J. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, 1993.
21. E.P. Simoncelli. Modeling the joint statistics of images in the wavelet domain. In *Proceedings of the 44th Annual Meeting*, volume 3813, Denver, CO, USA, 1999.
22. E.P. Simoncelli and E.H. Adelson. *Subband image coding*, chapter Subband transforms, pages 143–192. Kluwer Academic Publishers, 1990.

23. P.P. Vaidyanathan. Quadrature mirror filter banks, M-band extensions and perfect reconstruction techniques. *IEEE ASSP Magazine*, pages 4–20, 1987.
24. V. Vapnik. *The nature of statistical learning theory.* Spring Verlag, 1995.
25. V. Vapnik. *Statistical learning theory.* John Wiley and Sons, 1998.
26. M. Vetterli. A theory of multirate filter banks. *IEEE Transactions on ASSP*, 35(3):356–372, 1987.
27. A. Westfeld. High capacity depsite better steganalysis: F5- a steganographic algorithm. In *Fourth Information Hiding Workshop*, pages 301–315, Pittsburgh, PA, USA, 2001.
28. A. Westfeld and A. Pfitzmann. Attacks on steganographic systems. In *Proceedings of Information Hiding, Third International Workshop*, Dresden, Germany, 1999.
29. S.C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame) - towards the unified theory for texture modeling. In *IEEE Conference Computer Vision and Pattern Recognition*, pages 686–693, 1996.