On Dangers of Overtraining Steganography to Incomplete Cover Model

Jan Kodovský
Binghamton University
Department of ECE
Binghamton, NY 13902-6000
jan.kodovsky@
binghamton.edu

Jessica Fridrich
Binghamton University
Department of ECE
Binghamton, NY 13902-6000
fridrich@
binghamton.edu

Vojtěch Holub
Binghamton University
Department of ECE
Binghamton, NY 13902-6000
vholub1@
binghamton.edu

ABSTRACT

A modern direction in steganography calls for embedding while minimizing a distortion function defined in a sufficiently complex model space. In this paper we show that, quite surprisingly, even a high-dimensional cover model does not automatically guarantee immunity to simple attacks. Moreover, the security can be compromised if the distortion is optimized to an incomplete cover model. We demonstrate these pitfalls with two recently proposed steganographic schemes and support our arguments experimentally. Finally, we discuss how the corresponding models might be modified to eliminate the security flaws.

Categories and Subject Descriptors

I.4.9 [**Computing Methodologies**]: Image Processing and Computer Vision—*Applications*

General Terms

Security, Algorithms, Theory

Keywords

Cover model, overtraining, HUGO, distortion, steganalysis

1. INTRODUCTION

There appears a simple recipe how to construct secure steganographic systems – adopt a model for the cover source and make the embedding preserve it exactly. This can, however, be achieved only for very simple cover models that do not describe empirical sources, such as digital images, well. For more complex (high-dimensional) models, the underlying distributions of cover and stego objects cannot be accurately estimated due to lack of data.[1] A different and rather promising direction is to use complex models but abandon the goal of exactly preserving the cover distribution and the need to sample it. Instead, the embedding minimizes a distortion function in the model space defined to correlate with statistical detectability. This philosophy is somewhat reminiscent of classification using support vector machines where one maximizes a margin between examples from both classes instead of estimating the underlying distributions and constructing a likelihood ratio test.

The first embedding scheme that dared to approximately preserve a complex cover model was the Feature-Correction Method (FCM) [14]. After adopting a model for images (selecting a feature space), a given payload was embedded while minimizing the distortion induced in the feature space and by introducing additional changes to bring the feature vector near its original position.[2] The FCM, however, has one familiar disease – even though the embedding could be made significantly less detectable with steganalyzers using the same feature space, it became in fact more detectable in alternative feature spaces that could even be just slight modifications of the original feature space. The steganography was "overtrained" to a cover model that was not a complete statistical description of the cover source.

Aware of this deficiency, the authors of [18] designed what can be interpreted as a more advanced version of the FCM without the feature correction with a cover model of dimensionality 10^7 to make the model "more complete" and make it hopefully impossible for Eve to detect embedding by working outside the model. The algorithm, whose name is HUGO (Highly Undetectable steGO), is a case of the so-called minimum-embedding-impact steganography that embeds a given payload while minimizing the impact of introduced modifications measured using a suitably defined distortion function. As long as the distortion corresponds to statistical distinguishability, steganography cast within this framework formally becomes source coding with a fidelity criterion for which near-optimal coding schemes were developed [6, 4]. HUGO was used in the steganalysis contest BOSS (Break Our Steganographic System) conceived as Alice's challenge to Eve. The first contribution of this paper reveals an unexpected weakness of HUGO caused by an abrupt end of the model that enables Eve to build a simple low-dimensional detector with unusual properties: the detection accuracy is almost flat w.r.t. payload and the detector is more accurate on highly textured images than

[1] The number of samples increases exponentially with model dimensionality.

[2] A more detailed extensive study of the FCM appeared recently in [3].

MM&Sec'11, September 29–30, 2011, Buffalo, New York, USA.
Copyright 2011 ACM 978-1-4503-0806-9/11/09 ...$10.00.

on images with a smooth content. We also show that this weakness can be easily prevented by adjusting HUGO's embedding parameters.

The steganographic security of stegosystems built from the principle of minimum impact depends on how well the distortion captures the statistical impact of embedding. A general method for designing distortion functions that correlate with statistical detectability was recently introduced in [5]. The authors proposed to learn the parameters of the distortion function (essentially a look-up table) by minimizing the margin of a linear support vector machine (L-SVM) on a sample of cover and stego image features. One interesting result of this work was a new algorithm for the JPEG domain whose security as tested using blind steganalyzers on known feature spaces, was shown to be significantly better when compared to the state of the art. The second contribution of this paper is another surprising revelation that this new algorithm with optimized distortion is, in fact, highly detectable using an appropriately enlarged cover model of a relatively low dimension. The steganography is again overtrained to an incomplete model, allowing Eve to mount an accurate attack.

This paper is organized as follows. In the next section, we introduce the notation and summarize the methodology adopted in all our steganalysis experiments. Selected elements of the HUGO algorithm relevant to this work are described in Section 3, where we also point out a weakness of HUGO's model and study its effect on statistical detectability experimentally for different embedding parameters and payloads. Section 4 starts with a discussion of the methodology for optimizing the distortion function as introduced in [5]. Then, a low-dimensional feature space is introduced and shown to detect embedding with optimized distortion functions with very high accuracy. Each experimental section is closed with a discussion on how the model flaw exploited in the attack can be eliminated and the security of the algorithm thus improved. The paper is summarized in Section 5.

2. NOTATION AND SETUP

Everywhere in this article, lower-case boldface symbols are used for vectors and capital-case boldface symbols for matrices or higher-dimensional arrays. The symbols $\mathbf{X} = (\mathbf{X}_{ij}) \in \mathcal{X} = \mathcal{I}^{n_1 \times n_2}$ and $\mathbf{Y} = (\mathbf{Y}_{ij}) \in \mathcal{X}$ will always represent either pixel values of grayscale cover and stego images with $n = n_1 n_2$ pixels ($\mathcal{I} = \{0, \ldots, 255\}$) or quantized DCT coefficients in the corresponding JPEG files ($\mathcal{I} = \{-1023, \ldots, 1024\}$). For any $x \in \mathbb{R}$ and $T > 0$, we define $\mathrm{trunc}_T(x) = x$ for $x \in [-T, T]$ and $\mathrm{trunc}_T(x) = T\mathrm{sign}(x)$ otherwise. The symbol $\lceil x \rceil$ stands for the smallest integer larger than or equal to x.

All experiments in this paper will be carried out on two databases of grayscale images: CAMERA and BOSSbase. CAMERA contains $6,500$ images originally taken by 22 digital cameras in their RAW format, resized so that the smaller side was 512 pixels, converted to grayscale, and compressed using the Matlab command 'imwrite' with JPEG quality factor 75. BOSSbase consists of $9,074$ images taken with seven digital cameras in their RAW format, converted to grayscale, and resized/cropped to 512×512 using the script provided by the BOSS organizers [7]. All steganalyzers are binary classifiers implemented using libSVM [2] as Gaussian SVMs (G-SVMs) for each case of cover/stego images (for each payload). The hyperparameters C and γ (the penalty and kernel width) of the G-SVMs were determined by a search on the following grid: $\mathcal{G}_C \times \mathcal{G}_\gamma$, $\mathcal{G}_C = \{10^i | i = 0, \ldots, 5\}$, $\mathcal{G}_\gamma = \{2^j/d | j = -4, \ldots, 3\}$, where d is the feature dimensionality. All experiments are realized by randomly splitting the database into two halves, training on one half and testing on the other half. The detection performance is reported as the minimal total error $P_{\mathrm{E}} \triangleq \min_{P_{\mathrm{FA}}} \frac{1}{2}(P_{\mathrm{FA}} + P_{\mathrm{MD}}(P_{\mathrm{FA}}))$ averaged over ten independent database splits; P_{FA} and P_{MD} are the probabilities of false alarm and missed detection and $P_{\mathrm{MD}}(P_{\mathrm{FA}})$ makes the dependency of P_{MD} on P_{FA} explicit as both are functions of a single threshold parameter of the classifier.

3. ABRUPT MODEL END IN HUGO

In this section, we analyze the HUGO algorithm with the same settings as in the BOSS competition. First, we point out a weakness in its model and then demonstrate its effect on detectability through a series of experiments. Finally, we show that the weakness can be easily removed by adjusting HUGO's embedding parameters. This contribution was inspired by one of the features proposed to attack HUGO in [11]. The authors, however, did not analyze why the feature was so effective for detecting HUGO neither did they make any connections with a flaw in HUGO's model.

3.1 HUGO's model

Starting with a cover image $\mathbf{X}$, HUGO represents it with a feature vector computed from four three-dimensional co-occurrence matrices obtained from differences of horizontally, vertically, and diagonally neighboring pairs of pixels. Using $\mathbf{d} = (d_1, d_2, d_3) \in \mathcal{T}_3 = \{-T, \ldots, T\}^3$, the horizontal co-occurrence matrix is defined as:

$$\mathbf{C}^{\rightarrow}_{\mathbf{d}} = \Pr(\mathbf{D}^{\rightarrow}_{i,j} = d_1, \mathbf{D}^{\rightarrow}_{i,j+1} = d_2, \mathbf{D}^{\rightarrow}_{i,j+2} = d_3), \quad (1)$$

where $\mathbf{D}^{\rightarrow}_{i,j} = \mathrm{trunc}_T(\mathbf{X}_{i,j} - \mathbf{X}_{i,j+1})$, $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2 - 2$, and $\Pr(\cdot)$ stands for a sampled probability distribution. The vertical, diagonal, and minor diagonal matrices are defined similarly. Denoting the co-occurrence matrix computed from $\mathbf{X}$ in direction $k \in \{\rightarrow, \leftarrow, \uparrow, \downarrow\}$ as $\mathbf{C}^{\mathbf{X},k}_{\mathbf{d}}$, $\mathbf{d} \in \mathcal{T}_3$, the feature vector is $(\mathbf{F}^{\mathbf{X}}, \mathbf{G}^{\mathbf{X}}) \in \mathbb{R}^{2(2T+1)^3}$ with

$$\mathbf{F}^{\mathbf{X}}_{\mathbf{d}} = \sum_{k \in \{\rightarrow, \leftarrow, \uparrow, \downarrow\}} \mathbf{C}^{\mathbf{X},k}_{\mathbf{d}},$$
$$\mathbf{G}^{\mathbf{X}}_{\mathbf{d}} = \sum_{k \in \{\searrow, \nwarrow, \swarrow, \nearrow\}} \mathbf{C}^{\mathbf{X},k}_{\mathbf{d}}. \quad (2)$$

The distortion function that is minimized in HUGO is a weighted L_1-norm between the cover and stego feature vectors:

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{\mathbf{d} \in \mathcal{T}_3}^{T} \left[w(\mathbf{d}) \left| \mathbf{F}^{\mathbf{X}}_{\mathbf{d}} - \mathbf{F}^{\mathbf{Y}}_{\mathbf{d}} \right| + \right.$$
$$\left. + w(\mathbf{d}) \left| \mathbf{G}^{\mathbf{X}}_{\mathbf{d}} - \mathbf{G}^{\mathbf{Y}}_{\mathbf{d}} \right| \right], \quad (3)$$

where the heuristically defined weights

$$w(\mathbf{d}) = \left(\sqrt{d_1^2 + d_2^2 + d_3^2} + \sigma \right)^{-\gamma} \quad (4)$$

quantify the detectability of an embedding change in the $\mathbf{d}$th element of $\mathbf{F}$ and $\mathbf{G}$. In (4), σ and γ are scalar parameters with values $\sigma = 1$ and $\gamma = 1$.

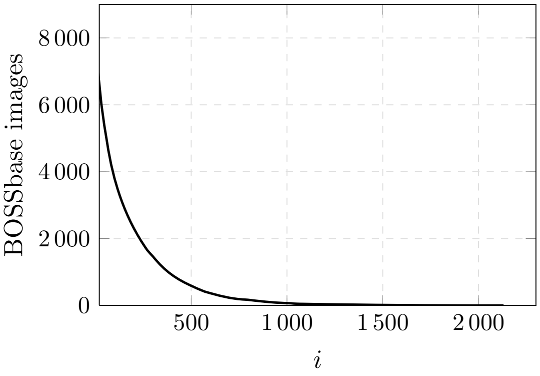

Figure 1: The number of BOSSbase images with histogram bin $\mathbf{h}_{90} > i$.

The secret message is embedded by modifying pixels by ±1 while minimizing the distortion (3) using syndrome-trellis codes [6].

3.2 Model weakness

The default setting for the threshold T used in (3) was $T = 90$, which means that the embedding approximately preserves a $2(2T+1)^3 = 11,859,482$-dimensional feature vector. HUGO designers have likely opted for such a high dimension to make it as hard as possible for Eve to mount an attack. Indeed, the individual co-occurrence bins with $|\mathbf{d}| > 90$ are mostly empty or very sparsely populated and the steganalyst cannot use them to make any reliable inference about the presence of a secret message. However, this does not mean that the *marginals* of the feature vector (2) will necessarily be sparsely populated as well.

For image $\mathbf{X}$, let us define the vector $\mathbf{h}_i^{\mathbf{X}}$, $i = 0, \ldots, 255$, as the total number of pixel pairs adjacent either in the horizontal, vertical, diagonal, or minor-diagonal direction whose difference in absolute value is equal to i. Figure 1 shows the total number of cover images $\mathbf{X}$ from BOSSbase with $\mathbf{h}_{90}^{\mathbf{X}} \geq i$ as a function of i. For example, there are 3,799 images with $\mathbf{h}_{90}^{\mathbf{X}} \geq 100$. Because pixel pairs with differences below 90 are treated differently by the embedding algorithm than pairs with differences above 90, the histogram contains a detectable artifact around the value of 90, where HUGO's model ends. This is confirmed in Figure 2 where we show the histogram bins of cover and stego images (HUGO with payload 0.4 bpp) averaged over all BOSSbase images. Note that the bins $\mathbf{h}_{89}^{\mathbf{X}}$ and $\mathbf{h}_{90}^{\mathbf{X}}$ decrease while $\mathbf{h}_{91}^{\mathbf{X}}$ and $\mathbf{h}_{92}^{\mathbf{X}}$ increase after embedding. This is because the difference 90 is more likely to be changed to 91 than to 89 as this change is less costly, increasing thus the bin 91. Since, 89 receives mostly values from 88 rather than from 90 it decreases. Finally, since the change from 91 to 92 is less costly than to 90 (which is within the model), 91 changes more often to 92 than to 90, further increasing 92 and decreasing 90.

Figure 3 shows, that the two-dimensional feature vector $(\mathbf{h}_{90}^{\mathbf{X}}, \mathbf{h}_{91}^{\mathbf{X}})$ already has a non-trivial distinguishing power. Note that better detection with this feature vector will be obtained for images with higher counts in these two bins, which are images with noise, textures, and edges. On the contrary, images with mostly smooth content contain fewer pixel pairs with difference 90 and will thus be less reliably detected.

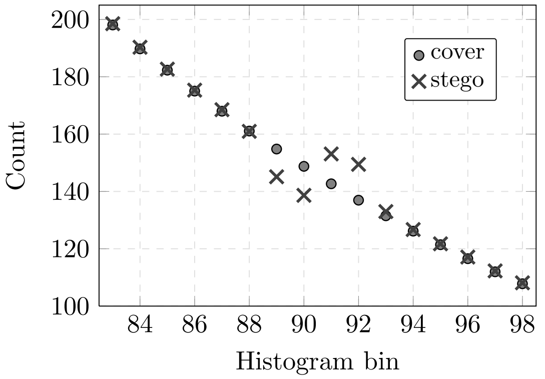

Figure 2: Histogram bins $\mathbf{h}_i^{\mathbf{X}}$, $i = 83, \ldots, 98$ for cover images (dots) and $\mathbf{h}_i^{\mathbf{Y}}$ for stego images embedded with HUGO with payload 0.4 bpp (crosses) averaged over all BOSSbase images.

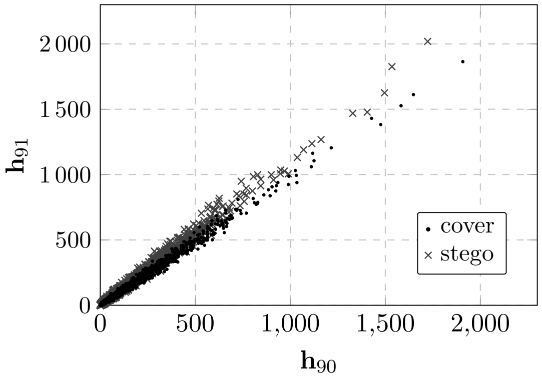

Figure 3: Histogram bin $\mathbf{h}_{91}^{\mathbf{X}}$ versus $\mathbf{h}_{90}^{\mathbf{Y}}$ for cover (dots) and stego images (crosses) with payload 0.4 bpp for all BOSSbase images X.

Since the effect of the model end at 90 "spills over" to the neighboring bins, the detection can be improved by including more than just two bins:

$$\mathbf{h}^{\mathbf{X}} = (\mathbf{h}_{90-r}^{\mathbf{X}}, \ldots, \mathbf{h}_{91+r}^{\mathbf{X}}), \quad r \geq 0. \tag{5}$$

Even though the bins outside of the interval [89, 93] do not practically change, they may still be very useful as references for the classifier. While we did observe a drop in the classification error P_{E} with r increasing up to approximately $r = 10$, the difference when using $r = 1$ (four-dimensional feature) and $r = 10$ (24-dimensional feature) was less than 1% depending on the payload. The first column in Table 1 shows the detection accuracy P_{E} on BOSSbase for six relative payloads $\alpha \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$bpp. Note that the error is nearly constant w.r.t. to the payload, which is highly unusual for a detection statistic. We explain this peculiar behavior by HUGO's adaptive embedding mechanism – smaller payloads introduce a higher percentage of changes in textured areas and around edges where the feature vector (5) is effective.

	$T = 90$			$T = 255$		
α [bpp]	$\mathbf{h}^{\mathbf{X}}$	SQUARE	SQUARE+$\mathbf{h}^{\mathbf{X}}$	$\mathbf{h}^{\mathbf{X}}$	SQUARE	SQUARE+$\mathbf{h}^{\mathbf{X}}$
.05	.3361	.4423	.3233	.4950	.4430	.4432
.10	.3288	.4042	.2911	.4975	.4006	.3993
.20	.3082	.3287	.2254	.4973	.3263	.3262
.30	.2959	.2647	.1648	.4974	.2601	.2630
.40	.2941	.1958	.1284	.4984	.2012	.2008
.50	.2851	.1534	.0926	.4956	.1545	.1468

Table 1: Detection error P_{E} for six payloads and two settings of HUGO when using the histogram feature (dim 4), the SQUARE feature (dim 338), and a combination of both (dim 342).

Since the feature vector (5) detects embedding in images that exhibit a higher level of noise and texture, it is tempting to see how well HUGO can be detected when combining the four histogram features with those that better detect embedding in smooth regions. We use the following SQUARE features that are similar in spirit to those described in [9] and also in [11].

To obtain the SQUARE feature vector for image $\mathbf{X}$, we first compute a noise residual by convolving $\mathbf{X}$ with a symmetrical square kernel $\mathbf{K}$ and then truncate the result with $T = 2$:

$$\mathbf{R} = \mathrm{trunc}_2(\mathbf{K} \star \mathbf{X}). \tag{6}$$

One horizontal, $\mathbf{C}^{\mathbf{R},\rightarrow}$, and one vertical, $\mathbf{C}^{\mathbf{R},\downarrow}$, four-dimensional co-occurrence matrix from $\mathbf{R}$ are formed. For example, the horizontal co-occurrence is defined as expected with $\mathbf{d} \in \mathcal{T}_4$:

$$\begin{aligned} \mathbf{C}^{\mathbf{R},\rightarrow}_{\mathbf{d}} = \{(i,j) | \mathbf{R}_{i,j} = d_1, \mathbf{R}_{i,j+1} = d_2,\\ \mathbf{R}_{i,j+2} = d_3, \mathbf{R}_{i,j+3} = d_4\}, \end{aligned} \tag{7}$$

with the vertical version defined analogically. The range of indices i, j in (7) is adjusted so that both $\mathbf{R}_{i,j}$ and $\mathbf{R}_{i,j+3}$ are defined.

The following two kernels will be used in this paper:

$$\mathbf{K}_3 = \begin{pmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{pmatrix}, \tag{8}$$

$$\mathbf{K}_5 = \begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix}. \tag{9}$$

The kernel $\mathbf{K}_3$ has been used for steganalysis in the past (e.g., [12, 1]) and is a good approximation of a translational-invariant linear filter that best approximates local content in the least-square sense. The larger filter was arrived at experimentally by observing the steganalysis error on HUGO images embedded with payload 0.4 bpp on BOSSbase. It seems to nicely complement (8).

Each co-occurrence matrix, $\mathbf{C}^{\mathbf{R},\rightarrow}$ and $\mathbf{C}^{\mathbf{R},\downarrow}$, has $(2T + 1)^4 = 625$ elements. We further reduce the dimensionality to 169 by using the sign symmetry[3] and the directional symmetry of natural images:

$$\tilde{\mathbf{C}}_{\mathbf{d}} \leftarrow \mathbf{C}_{\mathbf{d}} + \mathbf{C}_{-\mathbf{d}}, \tag{10}$$

$$\hat{\mathbf{C}}_{\mathbf{d}} \leftarrow \tilde{\mathbf{C}}_{\mathbf{d}} + \tilde{\mathbf{C}}_{\overleftarrow{\mathbf{d}}}, \tag{11}$$

for all $\mathbf{d} \in \mathcal{T}_4$, where $\overleftarrow{\mathbf{d}} \triangleq (d_4, d_3, d_2, d_1)$. Thus, the final feature vector, $(\hat{\mathbf{C}}^{\mathbf{R},\rightarrow}_{\mathbf{d}}, \hat{\mathbf{C}}^{\mathbf{R},\downarrow}_{\mathbf{d}})$, consists of two symmetrized co-occurrence matrices and has a total dimensionality $2 \times 169 = 338$. The symmetrization makes the resulting detection statistic more robust and thus improves the detection. Because the threshold in the definition of the SQUARE features is $T = 2$, they are more likely to detect images with smooth content rather than textured/noisy images. Thus, we expect the performance of SQUARE and the four histogram features to be complementary.

Table 1 contains all results of detecting HUGO in images of BOSSbase. As expected, the histogram features significantly improve the performance of SQUARE. For example, for payload 0.4 bpp, which was used in the BOSS competition, the detection error decreases from 19.58% for SQUARE to 12.84% after adding only four additional histogram features. This confirms our analysis above that the histogram features and the SQUARE features have approximately complementary performance. Also note that the gain due to the histogram features w.r.t. SQUARE decreases with decreased payload. It is quite remarkable that a mere 342 relatively simple features can detect HUGO with payload 0.4 bpp more accurately than any other attack on HUGO published so far [9, 8, 16, 11]. Finally, as the right half of the table shows, when the threshold used in HUGO's distortion measure (3) is set to $T = 255$,[4] the model flaw disappears and the histogram features become completely ineffective.

4. OVERTRAINED DISTORTION FUNCTION

Currently, the most successful approach to building stegosystems for empirical cover sources is to embed while minimizing a distortion function [13, 20, 18, 4, 6]. The security depends primarily on how well the distortion actually measures statistical detectability. In [5], the authors showed how to minimize detectability by parametrizing the distortion and determining the best parameters by optimization. The objective function was the margin of a linear SVM determined from 80 pairs of cover and stego images represented in the 548-dimensional CC-PEV model (feature) space [15] chosen as a reasonable representative of a current state-of-the-art steganography model. The optimized distortion was used to construct a new JPEG steganography algorithm, which we refer to as MOD (Model Optimized Distortion). To show that the distortion was not overtrained to a fixed

[3] Sign-symmetry means that taking a negative of an image does not change its statistical properties.

[4] This can be achieved by simply running the HUGO simulator with the switch --T.

model, the authors tested MOD with the CC-PEV set with a slightly different cropping in calibration as well as with the Cross-Domain Feature set (CDF) obtained by merging CC-PEV and the 686-dimensional SPAM vector [17] computed from images represented in the spatial-domain. MOD was reported to be significantly more secure than the nsF5 algorithm [10].

While the parametrization of the distortion function was chosen sufficiently rich (see below), the MOD algorithm was optimized to make changes undetectable within the CC-PEV model. While this model considers both inter- and intra-block dependencies among DCT coefficients by forming co-occurrence matrices, it does so only for a rather limited range. In this section, we show that by enlarging the corresponding parts of the CC-PEV model the MOD algorithm quickly becomes highly detectable and, quite paradoxically, less secure than the heuristically designed nsF5.

4.1 Optimized costs

In the MOD algorithm, the cost ρ_{ij} of changing a DCT coefficient $\mathbf{X}_{i,j}$ by ±1 is determined by its immediate intra- and inter-block neighborhood:[5]

$$\mathcal{N}_{\mathrm{ir}} = \{\mathbf{X}_{i+8,j}, \mathbf{X}_{i,j+8}, \mathbf{X}_{i-8,j}, \mathbf{X}_{i,j-8}\}, \tag{12}$$

$$\mathcal{N}_{\mathrm{ia}} = \{\mathbf{X}_{i+1,j}, \mathbf{X}_{i,j+1}, \mathbf{X}_{i-1,j}, \mathbf{X}_{i,j-1}\}. \tag{13}$$

It is determined as a sum

$$\rho_{ij} = \sum_{z\in\mathcal{N}_{\mathrm{ia}}} (\theta^{(\mathrm{ia})}_{\mathbf{X}_{i,j}-z})^2 + \sum_{z\in\mathcal{N}_{\mathrm{ir}}} (\theta^{(\mathrm{ir})}_{\mathbf{X}_{i,j}-z})^2, \tag{14}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(\mathrm{ir})}, \boldsymbol{\theta}^{(\mathrm{ia})})$ is a vector of $2(2\Delta+1+1)$ inter- and intra-block cost parameters:

$$\boldsymbol{\theta}^{(\mathrm{ir})} = (\theta^{(\mathrm{ir})}_{-\Delta}, \ldots, \theta^{(\mathrm{ir})}_{\Delta}, \theta^{(\mathrm{ir})}_{\bullet}), \tag{15}$$

$$\boldsymbol{\theta}^{(\mathrm{ia})} = (\theta^{(\mathrm{ia})}_{-\Delta}, \ldots, \theta^{(\mathrm{ia})}_{\Delta}, \theta^{(\mathrm{ia})}_{\bullet}). \tag{16}$$

Furthermore, we adopt the convention $\theta^{(\mathrm{ia})}_{z} = \theta^{(\mathrm{ia})}_{\bullet}$, $\theta^{(\mathrm{ir})}_{z} = \theta^{(\mathrm{ir})}_{\bullet}$ whenever $|\mathbf{X}_{i,j} - z| > \Delta$. The subscript of each parameter corresponds to the difference between $\mathbf{X}_{i,j}$ and its immediate neighbor. For example, one would expect that $|\theta^{(\mathrm{ir})}_{k}|$ would be larger for small $|k|$ and vice versa, reflecting the fact that changes in smooth regions should be more costly than in textured areas. The value of Δ controls the size of the parameter space and the complexity of the distortion function.

The authors optimized the parameters (15) and (16) for $\Delta = 6$ for the 548-dimensional CC-PEV cover model and stego images embedded with payload 0.5 bpac (bits per non-zero AC DCT coefficient). Two versions of the MOD algorithm were introduced – one in which both intra- and inter-block costs were optimized and the version in which only the inter-block parameters were optimized while $\boldsymbol{\theta}^{(\mathrm{ia})} \equiv (0, \ldots, 0)$. The latter one exhibited better security when tested with the CDF feature set.

4.2 Security weakness due to incomplete cover model

The CC-PEV cover model considers various dependencies among DCT coefficients, including inter-block co-occurrence matrices constrained to a rather limited range of $\{-2, \ldots, 2\}$ and Markov features [21] in the range $\{-4, \ldots, 4\}$ capturing intra-block dependencies. A distortion function with parameters optimized w.r.t. this rather abruptly terminated model will likely underestimate the importance of dependencies among DCT coefficients outside of the range. Indeed, the MOD algorithm with $\boldsymbol{\theta}^{(\mathrm{ia})} = 0$ makes $\sim 95\%$ of all embedding changes to coefficients with absolute value greater than 2 (see Figure 4). Such changes are unlikely to be detected by the small-range co-occurrences in the CC-PEV model. We confirm that this overtraining manifests in practice by first attacking the MOD algorithm with enlarged inter-block co-occurrences (IBCs).

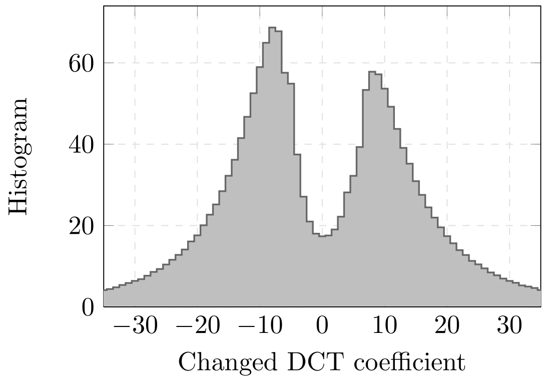

Figure 4: Histogram of changes to DCT coefficient values introduced by the MOD stegosystem with $\boldsymbol{\theta}^{(\mathrm{ia})} = 0$ at payload 0.10 bpac. The chart displays the average counts over 1,000 randomly selected images from the CAMERA database.

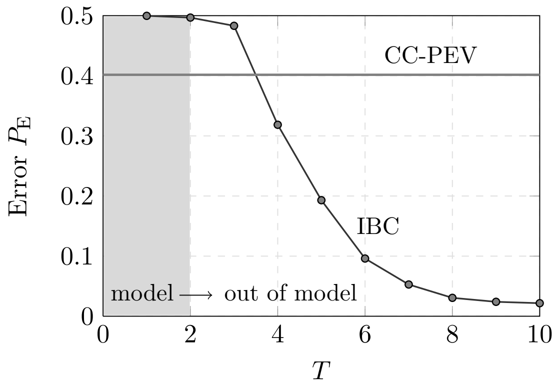

Figure 5: Detection error when steganalyzing the MOD stegosystem with $\boldsymbol{\theta}^{(\mathrm{ia})} = 0$ at payload 0.10 bpac using the IBC features (17) as a function of the threshold T.

Formally, the feature vector is a sum of two two-dimensional co-occurrence matrices:

$$\mathbf{C}^{\mathbf{X}}_{\mathbf{d}} = \mathbf{C}^{\tilde{\mathbf{X}},\rightarrow}_{\mathbf{d}} + \mathbf{C}^{\tilde{\mathbf{X}},\downarrow}_{\mathbf{d}}, \tag{17}$$

where $\mathbf{d} \in \{-T, \ldots, T\}^2$, $\tilde{\mathbf{X}} = \mathrm{trunc}_T(\mathbf{X})$, and

$$\mathbf{C}^{\mathbf{X},\rightarrow}_{\mathbf{d}} = \{(i,j) | \mathbf{X}_{i,j} = d_1 \wedge \mathbf{X}_{i,j+8} = d_2\} \tag{18}$$

$$\mathbf{C}^{\mathbf{X},\downarrow}_{\mathbf{d}} = \{(i,j) | \mathbf{X}_{i,j} = d_1 \wedge \mathbf{X}_{i+8,j} = d_2\}. \tag{19}$$

[5] $\mathbf{X}_{i,j} \in \{-1024, \ldots, 1024\}$ is the DCT mode (i mod 8, j mod 8) in the $\lceil i/8 \rceil, \lceil j/8 \rceil$th block.

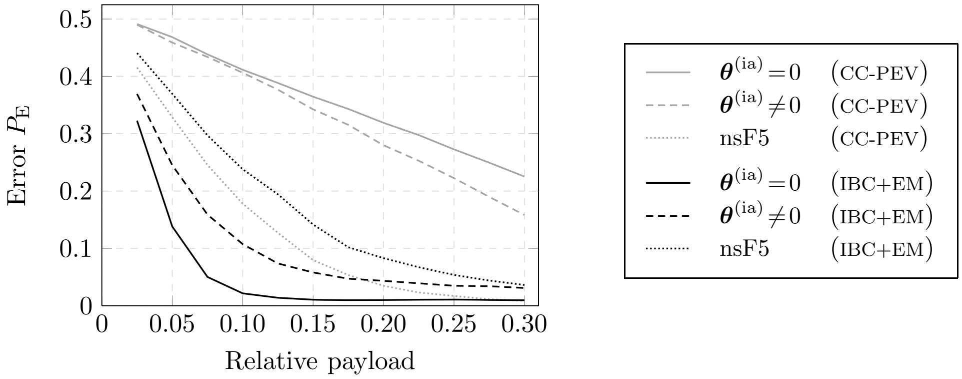

Figure 6: Detection error P_{E} for the MOD algorithm and nsF5 when attacked with CC-PEV features and the union of the IBC and EM models.

Algorithm	Features	.025	.050	.075	.100	.125	.150	.175	.200	.225	.250	.275	.300
$\boldsymbol{\theta}^{(\mathrm{ia})} = 0$	CC-PEV	.4913	.4684	.4384	.4115	.3885	.3646	.3431	.3190	.2976	.2727	.2495	.2253
	IBC	.3214	.1331	.0453	.0196	.0128	.0098	.0095	.0092	.0090	.0092	.0088	.0093
	IBC+EM	.3228	.1382	.0502	.0215	.0138	.0105	.0102	.0095	.0103	.0109	.0100	.0095
$\boldsymbol{\theta}^{(\mathrm{ia})} \neq 0$	CC-PEV	.4900	.4589	.4338	.4063	.3768	.3425	.3154	.2798	.2528	.2225	.1898	.1583
	IBC	.4066	.3117	.2339	.1711	.1330	.1068	.0849	.0706	.0619	.0560	.0506	.0470
	IBC+EM	.3694	.2451	.1589	.1074	.0737	.0580	.0471	.0434	.0391	.0348	.0340	.0309
nsF5	CC-PEV	.4154	.3288	.2454	.1779	.1271	.0795	.0535	.0347	.0230	.0170	.0111	.0087
	IBC	.4697	.4400	.4102	.3776	.3384	.3009	.2625	.2276	.1972	.1726	.1485	.1278
	IBC+EM	.4406	.3695	.2971	.2380	.1943	.1415	.1020	.0829	.0671	.0537	.0438	.0362

Table 2: Detection error P_{E} when attacking nsF5 and the MOD algorithm across different payloads (in bpac) using CC-PEV, IBC, and the union IBC+EM.

With threshold T, the dimensionality of the IBC feature vector (17) is $(2T+1)^2$. Figure 5 shows the results of detecting the MOD stegosystem (the version with $\boldsymbol{\theta}^{(\mathrm{ia})} = 0$) at a fixed payload 0.10 bpac using the IBC features on the CAMERA database. For comparison, the detection error, P_{E}, for CC-PEV features is depicted with a horizontal line. Note that, according to our expectations, as soon as the IBC features get out of the CC-PEV model ($T = 2$), the detection error starts rapidly decreasing and reaches $P_{\mathrm{E}} \approx 2\%$ with $T = 10$.

To complete the picture, we steganalyze both versions of the MOD algorithm across a wider range of payloads using the IBC features with $T = 10$ (model dimensionality 441). The error rates shown in Table 2 should be compared to those obtained using the CC-PEV features. The striking difference clearly supports our argument that the MOD algorithm has been overtrained to an incomplete cover model. Note that the MOD algorithm optimized w.r.t. *both* inter- *and* intra-block dependencies now becomes more secure than the version optimized w.r.t. inter-block dependencies. This is to be expected as the latter will naturally be more overtrained to the incomplete CC-PEV model. Finally, notice that the IBC features, so powerful against the MOD algorithm, are not very effective against nsF5.

The accuracy of the attack can be further improved by also extending the intra-block part of the CC-PEV feature vector formed as a sum of four 9×9 conditional probability matrices modeling the differences between absolute values of neighboring DCT coefficients as a Markov process and thresholded with $T = 4$ (see [21, 19] for details). We enlarge this statistical descriptor by increasing the threshold to $T = 10$, obtaining thus a 441-dimensional feature vector, which we will refer to as the EM (Extended Markov) vector. As Table 2 shows, the enlarged model further decreases the security of the MOD algorithm optimized w.r.t. both inter- and intra-block neighborhood to the extent that it is no longer more secure than the nsF5 algorithm (for payloads less then 0.2 bpac).

Figure 6 summarizes the detection of both versions of the MOD algorithm and nsF5 using the CC-PEV and the 882-dimensional union of the IBC and EM feature sets. Note that the security of nsF5 was not compromised by attacking it with IBC+EM features. In fact, the CC-PEV features are more successful in attacking nsF5 because the IBC+EM model lacks the diversity of the CC-PEV model and also because most changes made by nsF5 are made to DCT coefficients already covered by the small-range co-occurrences in the CC-PEV model.

Also note that the security of the inter-block-only optimized version of MOD ($\boldsymbol{\theta}^{(\mathrm{ia})} = 0$) is now much lower when compared to the case when both inter- and intra-block weights are optimized. This should intuitively be the case as considering both types of dependencies leads to a more accurate (and complete) cover model that should be less prone to overtraining. We conjecture that the security of the MOD algorithm can likely be markedly improved by optimizing the costs w.r.t. an enlarged CC-PEV model.

5. CONCLUSION

The most secure steganographic methods for empirical covers today are designed to embed while minimizing a suitably defined distortion function. However, designing the distortion without introducing fatal flaws appears to be a rather difficult task. We demonstrate this on two examples of modern steganographic schemes – HUGO and a recently proposed embedding algorithm for JPEG images with a distortion function optimized w.r.t. statistical undetectability. In particular, we show that a high-dimensional complex cover model does not automatically guarantee immunity to simple attacks. Moreover, the security of an embedding scheme employing a distortion function optimized to maximize security can be completely compromised and the embedding become highly detectable if its cover model is incomplete.

6. ACKNOWLEDGMENTS

The work on this paper was supported by Air Force Office of Scientific Research under the research grant number FA9550-09-1-0147. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of AFOSR or the U.S. Government.

7. REFERENCES

[1] R. Böhme. *Improved Statistical Steganalysis Using Models of Heterogeneous Cover Signals.* PhD thesis, Faculty of Computer Science, Technische Universität Dresden, Germany, 2008.

[2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[3] V. Chonev and A. D. Ker. Feature restoration and distortion metrics. In N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Security and Forensics of Multimedia XIII*, volume 7880, pages 0G01–0G14, San Francisco, CA, January 23–26, 2011.

[4] T. Filler and J. Fridrich. Gibbs construction in steganography. *IEEE Transactions on Information Forensics and Security*, 5(4):705–720, 2010.

[5] T. Filler and J. Fridrich. Design of adaptive steganographic schemes for digital images. In N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Security and Forensics of Multimedia XIII*, volume 7880, pages OF 1–14, San Francisco, CA, January 23–26, 2011.

[6] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security*, 6(1):1–17, 2011.

[7] T. Filler, T. Pevný, and P. Bas. BOSS (Break Our Steganography System). `http://boss.gipsa-lab.grenoble-inp.fr`, July 2010.

[8] J. Fridrich, J. Kodovský, M. Goljan, and V. Holub. Breaking HUGO – the process discovery. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, Prague, Czech Republic, May 18–20, 2011.

[9] J. Fridrich, J. Kodovský, M. Goljan, and V. Holub. Steganalysis of content-adaptive steganography in spatial domain. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, Prague, Czech Republic, May 18–20, 2011.

[10] J. Fridrich, T. Pevný, and J. Kodovský. Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities. In J. Dittmann and J. Fridrich, editors, *Proceedings of the 9th ACM Multimedia & Security Workshop*, pages 3–14, Dallas, TX, September 20–21, 2007.

[11] G. Gül and F. Kurugollu. A new methodology in steganalysis : Breaking highly undetactable steganograpy (HUGO). In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, Prague, Czech Republic, May 18–20, 2011.

[12] A. D. Ker and R. Böhme. Revisiting weighted stego-image steganalysis. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume 6819, pages 5 1–5 17, San Jose, CA, January 27–31, 2008.

[13] Y. Kim, Z. Duric, and D. Richards. Modified matrix encoding technique for minimal distortion steganography. In J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee, editors, *Information Hiding, 8th International Workshop*, volume 4437 of *Lecture Notes in Computer Science*, pages 314–327, Alexandria, VA, July 10–12, 2006. Springer-Verlag, New York.

[14] J. Kodovský and J. Fridrich. On completeness of feature spaces in blind steganalysis. In A. D. Ker, J. Dittmann, and J. Fridrich, editors, *Proceedings of the 10th ACM Multimedia & Security Workshop*, pages 123–132, Oxford, UK, September 22–23, 2008.

[15] J. Kodovský and J. Fridrich. Calibration revisited. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 63–74, Princeton, NJ, September 7–8, 2009.

[16] J. Kodovský and J. Fridrich. Steganalysis in high dimensions: Fusing classifiers built on random subspaces. In N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, editors, *Proceedings SPIE,*

Electronic Imaging, Security and Forensics of Multimedia XIII, volume 7880, pages OL 1–13, San Francisco, CA, January 23–26, 2011.

[17] T. Pevný, P. Bas, and J. Fridrich. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, 5(2):215–224, June 2010.

[18] T. Pevný, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In R. Böhme and R. Safavi-Naini, editors, *Information Hiding, 12th International Workshop*, volume 6387 of *Lecture Notes in Computer Science*, pages 161–177, Calgary, Canada, June 28–30, 2010. Springer-Verlag, New York.

[19] T. Pevný and J. Fridrich. Merging Markov and DCT features for multi-class JPEG steganalysis. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 3 1–3 14, San Jose, CA, January 29–February 1, 2007.

[20] V. Sachnev, H. J. Kim, and R. Zhang. Less detectable JPEG steganography method based on heuristic optimization and BCH syndrome coding. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 131–140, Princeton, NJ, September 7–8, 2009.

[21] Y. Q. Shi, C. Chen, and W. Chen. A Markov process based approach to effective attacking JPEG steganography. In J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee, editors, *Information Hiding, 8th International Workshop*, volume 4437 of *Lecture Notes in Computer Science*, pages 249–264, Alexandria, VA, July 10–12, 2006. Springer-Verlag, New York.