Synchronizing Embedding Changes in Side-Informed Steganography

Mehdi Boroumand and Jessica Fridrich, Department of ECE, SUNY Binghamton, NY, USA, {mboroum1, fridrich}@binghamton.edu

Abstract

Historically, two different strategies have been proposed for improving steganographic security by allowing each cover element to be modified by $+1$ or -1 with unequal probabilities: side-informed steganography and methods that cluster the polarity of neighboring changes. In the first strategy, the sender typically uses the knowledge of quantization errors when developing / processing the cover before embedding. In the latter, embedding on disjoint sub-lattices employs heuristic rules to increase the probability that the polarities of neighboring changes align. In this paper, we propose a method for combining both strategies and experimentally show an improvement in empirical security for several types of side information on two datasets when steganalyzing with rich models as well as convolutional neural networks.

Motivation

Steganography is a mode of covert communication in which messages are embedded in inconspicuous cover objects to hide the very presence of the communicated secret. Digital images are particularly suitable cover sources because they can hold large amounts of data and are commonly shared on social networks and attached to emails. Moreover, there are thousands of applications available to potential users.[1]

Since statistical detectability increases sharply with the amplitude of embedding changes, steganographic schemes typically modify the individual cover elements, which encode the luminance or DCT coefficients using integer values, by at most ± 1. The vast majority of embedding algorithms do so with *equal* probabilities as this maximizes the entropy (payload) embedded at each pixel [23, 22, 29, 35]. Two exceptions to this rule of thumb include side-informed steganography [23, 20, 6, 19, 34, 17, 15, 14, 7] and steganography that encourages neighboring embedding changes to share the same polarity [30, 5, 24]. Since both strategies have been shown to improve empirical security, in this paper we investigate whether they can be combined to further boost the resistance to steganalysis.

Side-informed (SI) steganography is a general term used for embedding schemes in which the sender makes use of the so-called precover [26] that is subjected to some sort of processing, development, or format conversion before embedding the secret message. Since the last step of the processing pipeline is typically quantization, the sender has access to the rounding errors and uses them to modulate the costs of changes by 1 and -1. SI steganography generally prefers changing those cover elements whose rounding errors are close to $\pm 1/2$ because such elements are the most sensitive to small perturbations. For example, a cover element with a non-rounded value 2.57, which would round to 3, is allowed to be modified during embedding to 2 with a small cost while changing the cover value 3 to 4 incurs a proportionally larger cost.

The first side-informed scheme was the embedding-while-dithering steganography [15], in which the secret message was embedded by perturbing the process of color quantization and dithering when converting a true-color image to a palette format. In perturbed quantization [16], the cover JPEG is recompressed to create side-information. The embedding prefers modifying DCT coefficients that fall close to the middle of the quantization bins during the second compression. The same idea can be applied when the cover image is uncompressed and the sender embeds her message in its JPEG form. The rounding errors of DCT coefficients can again be used to adjust the costs of polarities of embedding changes [27, 34, 39, 25]. This methodology was later further advanced using the paradigm of minimal-distortion steganography with advanced source coding [23, 20, 6].

While not studied in this paper, the authors wish to point out that side-information can have many other forms than rounding errors. In particular, when the sender has access to an acquisition oracle (e. g., a camera or a scanner [11, 13, 12]), she can acquire multiple exposures of the same scene to estimate the preferred polarity of embedding changes for cover elements that are most susceptible to small noise, and thus better mimic the embedding changes as acquisition noise [8]. In the so-called Natural Steganography [1, 2, 38], also recognized as steganography by cover source switching, the sender has access to the RAW image capture and embeds the message in the developed domain by making the stego image look as if it was acquired with a higher ISO setting. When the developing pipeline is modelable, extremely large payloads can be embedded with virtually perfect security.

In general, since side-information is only available to the sender, it can improve empirical security by a rather large margin. In [14], the author has shown that the precover compensates for the lack of the cover model. In particular, for a Gaussian model of acquisition noise, precover-informed rounding is more secure than embed-

[1] N. Johnson, "IoT Forensic Considerations and Steganography Beyond Images." Invited talk presented in the Network and Cloud Forensics Workshop, IEEE Conference on Communications and Network Security, October 9–11, 2017, Las Vegas, Nevada, USA.

ding designed to preserve the cover model estimated from the precover image, assuming the cover is "sufficiently nonstationary." Model-based binary SI embedding has been analyzed in [7]. The authors provided an explanation for why the the Fisher information (costs) in binary SI schemes should be modulated by $1-2|e|$, where e is the rounding error.

The second class of embedding methods with asymmetric embedding probabilities encourages synchronization (clustering) of polarities of neighboring modifications. Interaction of embedding changes can formally be captured with non-additive distortion functions [10]. The Gibbs construction [10] is a general framework for embedding with non-additive distortion, which is applicable whenever the distortion can be written as a sum of locally supported potentials (i.e., no "action at distance"). So far, the Gibbs construction has not produced a practical embedding scheme with improved empirical security. This is because it is not clear how to design non-additive distortion that properly captures the interaction of neighboring embedding changes and their impact on detectability. In [33], the authors introduced a greedy distortion minimization technique that they applied to embedding with the non-additive UNIWARD distortion [23]. Disappointingly, a smaller total embedding distortion did not correlate with empirical detectability.[2]

Nevertheless, the intuition that synchronized neighboring changes should improve security seems correct based on the two following observations. First, since most pixel predictors used to extract noise residuals for steganalysis, e.g., as in the Spatial Rich Model (SRM) [18], use filters with alternating signs, they are less disturbed by embedding changes with positively correlated polarities (see Sec. 5.1 in [5]). Second, since the polarization of the embedding directions depends on the exact embedding changes, the selection channel (the embedding change probabilities) are not available to the steganalyst, which decreases the efficiency of selection-channel-aware steganalysis, such as the maxSRM feature set [9] (Sec. 5.2 in [5]). In the absence of non-additive distortion functions that correlate with empirical detectability, the research community turned their attention to heuristic schemes.

The first method whose embedding mechanism considered the interaction of embedding changes was HUGO [32]. After determining the parities of all pixels to communicate the desired payload, the authors execute the actual embedding changes by $+1$ or -1 sequentially to minimize the impact on the stego image represented using the SPAM features [31]. In Clustering Modification Directions (CMD) steganography [30], the embedding is executed on disjoint interleaved sublattices embedded sequentially. The actual neighboring changes on the previous sublattices are used to decrease the costs in the direction of the majority of neighboring change polarities. A different idea was explored in [5], where the authors started from an additive scheme and purposely designed a non-additive distortion function as a sum of locally supported potentials. Recently, CMD has been improved in [24], where the coupling of neighboring embedding changes was derived by minimizing the variational approximation of the KL divergence between a Gaussian pixel residual model with non-zero mean and an asymmetric Gaussian mixture.

Intuitively, it should be possible to combine both embedding strategies. Note, however, that the evidence provided by side-information and the neighboring changes may be in conflict as the majority of neighboring changes may point in a different direction than the side-information. The solution proposed in this paper is to first modulate all pixel costs based on SI. The embedding proceeds on interleaved sublattices with the coupling enforced by further modulating the costs of different polarities by a multiplicative factor that non-linearly depends on the local mean of embedding changes weighted by their rounding errors.

In the next section, we describe relevant prior art on SI steganography and steganography with coupled embedding changes. In the third section, we investigate several different approaches for combining both strategies and determine the parameters of modulation factors. The results of all experiments appear in Section "Experiments," where we report the empirical security of three different types of side-information in two datasets with both rich models and deep neural networks. Numerical results in the form of tables appear at the end of the paper. The paper is concluded in the last section, where we also elaborate on possible future directions.

Relevant prior art

In this section, we describe relevant details of side-informed steganography and steganography with synchronized embedding changes.

Distortion-minimizing steganography

Currently, all modern steganographic schemes are designed within the paradigm of distortion minimization as this also allows efficient implementation in practice. Let us assume for simplicity that the cover image is grayscale, represented with integer values c_{ij}, $1 \le i \le M$, $1 \le j \le N$, where $M \times N$ are the number of rows and columns in the image. We describe the more general version when each cover element is assigned two potentially different costs, $\rho_{ij}(1)$ and $\rho_{ij}(-1)$, that measure the impact on detectability when the i,jth element is modified by 1 and -1, respectively. The payload is embedded while minimizing the sum of costs of all cover elements changed during embedding,

$$\sum_{i,j} \rho_{ij}(\nu_{ij}), \tag{1}$$

where $\nu_{ij} = s_{ij} - c_{ij}$ is the polarity of the embedding change, and s_{ij} represent the stego image. We note that the cost of no change is $\rho_{ij}(0) \triangleq 0$ for all i,j. The sum is over all elements in the image.

A steganographic scheme that embeds with the minimal expected total cost modifies each cover element with

[2]This observation was already made in [21].

probabilities

$$\beta_{ij}^{\nu} = \frac{\exp(-\lambda\rho_{ij}(\nu))}{1+\exp(-\lambda\rho_{ij}(1))+\exp(-\lambda\rho_{ij}(-1))}. \quad (2)$$

At the boundary of the dynamic range (for an 8-bit image, this means for 0 and 255), one can adopt several different strategies: 1) allow embedding by changing the boundary values to encode the same ternary symbol (e.g., the change by +1 at cover value 255 would be executed as 253), 2) allow only "inward" embedding by making the costs of out-of-range changes infinity, 3) forbidding changes of boundary values altogether. The impact of these three choices on empirical security depends on the image source [36]. For embedding schemes tested in this paper, we simply use their implementations as originally proposed, e.g., MiPOD uses Strategy 1, while S-UNIWARD and HILL use Strategy 2.

Side-informed steganography

Most SI schemes begin with a regular (i.e., not SI) additive embedding algorithm that assigns costs ρ_{ij} to each pixel, which is usually determined by some heuristic rule that assesses the content complexity (noise) in a local neighborhood of pixel i,j. Thus, fundamentally, the costs of both changes are considered to be the same.

Assuming the steganographer has access to a precover (unquantized cover), which we denote x_{ij}, the sender computes the rounding errors $e_{ij} = x_{ij} - [x_{ij}]$, $-1/2 \le e_{ij} \le 1/2$, where $[x]$ denotes the operation of rounding to the nearest integer within the dynamic range of the cover. In the absence of embedding, the steganographer would simply send the cover image $c_{ij} = [x_{ij}]$. In *ternary* SI steganography,[3] the costs of both change polarities are modulated based on the rounding error [6]:

$$\rho_{ij}(\text{sign}(e_{ij})) = (1-2|e_{ij}|)\rho_{ij} \quad (3)$$
$$\rho_{ij}(-\text{sign}(e_{ij})) = \rho_{ij}, \quad (4)$$

where $\rho_{ij}(\nu)$ are the modulated costs. It makes intuitive sense to make the costs proportional to $1-2|e_{ij}|$ because when $|e_{ij}| \approx 1/2$ a small perturbation of x_{ij} could cause x_{ij} to be rounded to "the other side." On the other hand, the costs are unchanged when $e_{ij} \approx 0$, as the SI does not provide any information about preferred polarity of the embedding change. A theoretical justification of this modulation for schemes that minimize detectability rather than cost appears in [7] where the authors showed for binary SI schemes that the steganographic Fisher information should be modulated by $(1-2|e_{ij}|)^2$.

For embedding schemes that do not use costs and, instead, minimize statistical detectability based on a cover model, such as MiPOD [35, 37], their SI version starts by first computing the (symmetric) embedding change probabilities β_{ij} derived to minimize (an approximation to) the KL divergence between the cover and symmetric stego mixture by solving the following equation for each pixel i,j:

$$\beta_{ij} I_{ij} = \lambda \ln \frac{1-2\beta_{ij}}{\beta_{ij}}, \quad (5)$$

where I_{ij} is the steganographic Fisher information, which reflects the impact of embedding on the cover model, and λ is a Lagrange multiplier determined by the payload size. To incorporate side-information, the sender next converts the embedding probabilities into costs by inverting (2)

$$\rho_{ij} = \log\left(1/\beta_{ij} - 2\right), \quad (6)$$

and finally modulates them by rounding errors as in (3).

Steganography with synchronized embedding changes

Another way of improving steganographic security is to encourage neighboring changes to share the same polarity. This has the effect of curbing the range of local pixel differences or noise residuals used for steganalysis. In particular, in [5] the authors linked the improvement in security to the fact that noise residuals from which steganalysis rich features are typically formed use sign-alternating kernels [18] and the observation that the coupling partially masks the embedding change probabilities (the selection channel) from the steganalyst.

Steganographic schemes that cluster the polarity of embedding changes typically work on a collection of disjoint sublattices [30, 5, 24]. In this section, we detail the CMD (Clustering Modification Direction) steganography [30] as the proposed method shares some similarity with this approach. CMD starts with an additive embedding scheme with costs ρ_{ij}, and partitions the cover image into four interleaved sublattices

$$\mathcal{L}_1 = \{(i,j)|\ \text{mod}\ (i,2)=1 \text{ and } \text{mod}\ (j,2)=1\} \quad (7)$$
$$\mathcal{L}_2 = \{(i,j)|\ \text{mod}\ (i,2)=1 \text{ and } \text{mod}\ (j,2)=0\} \quad (8)$$
$$\mathcal{L}_3 = \{(i,j)|\ \text{mod}\ (i,2)=0 \text{ and } \text{mod}\ (j,2)=0\} \quad (9)$$
$$\mathcal{L}_4 = \{(i,j)|\ \text{mod}\ (i,2)=0 \text{ and } \text{mod}\ (j,2)=1\}. \quad (10)$$

It then embeds a quarter of the payload in each sublattice, starting with $\mathcal{L}_1$. In sublattices $\mathcal{L}_2,\dots,\mathcal{L}_4$, the modification at pixel i,j is encouraged to be the same as the majority of actual embedding changes already executed in the local cross-neighborhood

$$\mathcal{C}_{ij} = \{(i-1,j),(i+1,j),(i,j-1),(i,j+1)\} \quad (11)$$

by decreasing the cost at pixel i,j by $\frac{1}{9}$ (this factor was determined experimentally) if there are more changes in that direction in the cross-neighborhood $\mathcal{C}_{ij}$ than in the opposite direction. Formally, let us denote the local average of embedding changes, $\nu_{kl} = s_{kl} - c_{kl}$, as

$$\mu_{ij} = \frac{1}{4}\sum_{(k,l)\in\mathcal{C}_{ij}} \nu_{kl}. \quad (12)$$

[3] Binary SI schemes only allow modification of cover elements by sign(v) [23, 19, 20, 34]. Binary schemes are not considered in this paper.

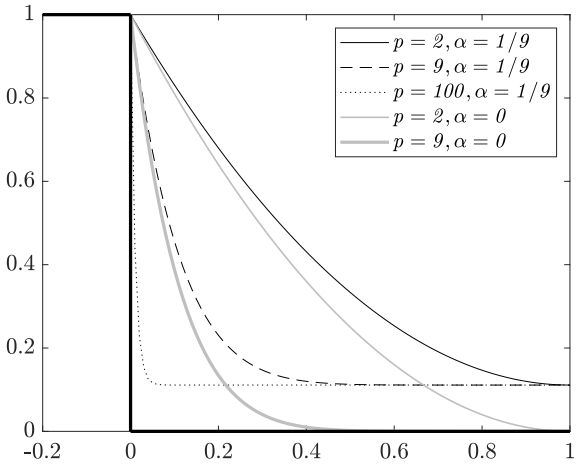


Figure 1: Modulation factor (16) as a function of the local weighted mean $x = \mu^{(\mathrm{w})}$ for $p = 2, 9, 100$, and $\alpha \in \{0, 1/9\}$. The values $p = 9$ and $\alpha = 0$ provided the best empirical security.

If $\mu_{ij} \neq 0$, the costs are modulated as follows:

$$\rho_{ij}(\mathrm{sign}(\mu_{ij})) = \frac{1}{9}\rho_{ij} \tag{13}$$

$$\rho_{ij}(-\mathrm{sign}(\mu_{ij})) = \rho_{ij}, \tag{14}$$

and, when $\mu_{ij} = 0$ the costs of both change polarities are the same as in the original scheme.

Side-informed steganography with synchronized embedding changes

In this section, we explore different ideas for combining side-informed embedding with clustering of polarities of neighboring changes. Note that side-information and the actual changes in a local neighborhood can be "in agreement" or "disagreement." For example, a negative rounding error translates into decreasing the cost of changing the pixel by -1 while a majority of embedding changes in the neighborhood of the same pixel may point in the opposite direction, creating thus a *conflict*. The cover image is split into the same four interleaved sublattices as above and one quarter of the message is embedded in each sublattice. Then, pixel costs of some additive embedding scheme ρ_{ij} are computed and modulated as in (3)–(4) to incorporate the side-information. We denote the costs modulated by SI as $\rho_{ij}^{(\mathrm{SI})}(\pm 1)$.

On the first sublattice $\mathcal{L}_1$, one quarter of the message is embedded with the side-information modulated costs $\rho_{ij}^{(\mathrm{SI})}(\pm 1)$. Moving to the second sublattice, for each pixel $(i, j) \in \mathcal{L}_2$, we first compute a *weighted* average of the actual embedding changes $\nu_{kl} \in \{-1, 0, 1\}$ from its cross-neighborhood $\mathcal{C}_{ij}$

$$\mu_{ij}^{(\mathrm{w})} = \frac{1}{4} \sum_{(k,l) \in \mathcal{C}_{ij}} w_{kl}\nu_{kl}, \tag{15}$$

with weights

$$w_{kl} = \begin{cases} 1 - 2|e_{kl}| & \text{when } \mathrm{sign}(e_{kl}\nu_{kl}) > 0 \\ 1 & \text{otherwise.} \end{cases}$$

The weighting, which is similar to weighting of costs in SI schemes(3), helps take into account how much each pixel in the cross-neighborhood was modified w.r.t. its precover value. In particular, embedding changes of pixels with $|e_{kl}| \approx 1/2$ have $w_{kl} \approx 0$ while $w_{kl} = 1$ when the SI and the actual embedding change "point in opposite directions."

When $\mu_{ij}^{(\mathrm{w})} = 0$, the neighboring changes do not affect the SI-modulated costs $\rho_{ij}^{(\mathrm{SI})}(\pm 1)$. When $\mu_{ij}^{(\mathrm{w})} \neq 0$, the embedding costs $\rho_{ij}^{(\mathrm{SI})}$ are further modulated by a soft step function shown in Figure 1

$$\rho_{ij}^{(\mathrm{nmSI})}(\mathrm{sign}(\mu_{ij}^{(\mathrm{w})})) = \rho_{ij}^{(\mathrm{SI})}(\mathrm{sign}(\mu_{ij}^{(\mathrm{w})})) \tag{16}$$

$$\times \Big((1-\alpha)(1-|\mu_{ij}^{(\mathrm{w})}|)^p + \alpha\Big) \tag{17}$$

$$\rho_{ij}^{(\mathrm{nmSI})}(-\mathrm{sign}(\mu_{ij}^{(\mathrm{w})})) = \rho_{ij}^{(\mathrm{SI})}(-\mathrm{sign}(\mu_{ij}^{(\mathrm{w})})), \tag{18}$$

where p is a positive integer and $0 \leq \alpha \leq 1$ to be determined experimentally.

The embedding in the remaining two sublattices $\mathcal{L}_3$ and $\mathcal{L}_4$ follows the same steps as for sublattice $\mathcal{L}_2$. This method will be called "nmSI" as in "Neighborhood Modulated Side-Informed" steganography.

Note that when removing the weighting in (15) and setting $\alpha = 1/9$ and $p = \infty$ (see Figure 1), we obtain a direct combination of SI embedding and the CMD algorithm [30] because the term $(1-|\mu_{ij}|)^\infty$ is equal to 0 whenever $|\mu_{ij}| > 0$, e.g., when there are more changes in the cross neighborhood in one direction than in the opposite direction. For a tie or no embedding changes in the neighborhood, $\mu_{ij} = 0$, which results in no cost modulation by neighboring changes as in CMD, leaving the rounding error e_{ij} as the only factor affecting the costs of changes of different polarities. We call this method SI-CMD.

Determining the parameters

To determine the parameters α and p for the cost modulation by neighboring changes (16), we adopted the following experimental setup. The additive embedding algorithm is HILL [29] at 0.4 bpp and image source BOSSbase 1.01 with 10,000 true-color images obtained using the same script as the original BOSSbase but removing the RGB to gray conversion. The side-information was obtained by RGB to gray conversion using the formula $0.2989 \times R + 0.5870 \times G + 0.1140 \times B$ in Matlab. The non-rounded pixel values served as the precover. Steganalysis was carried out with SRM and maxSRM feature sets and the ensemble classifier [28]. The total detection error P_{E} with equal priors averaged over ten random 5,000/5,000 splits of the dataset was used for evaluating empirical security.

0.4 bpp	SRM	maxSRMd2
HILL	0.2552 ± 0.0026	0.2264 ± 0.0015
CMD-HILL	0.3015 ± 0.0034	0.2681 ± 0.0016
SI-HILL	0.3338 ± 0.0034	0.3175 ± 0.0037
SI-CMD-HILL	0.3296 ± 0.0034	0.3350 ± 0.0034
nmSI ($p=\infty$, $\alpha=\frac{1}{9}$)	0.3342 ± 0.0021	0.3378 ± 0.0026
nmSI ($p=9$, $\alpha=0$)	$\mathbf{0.3678 \pm 0.0020}$	$\mathbf{0.3499 \pm 0.0022}$

Table 1: Average total detection error under equal priors $\overline{P}_{\mathrm{E}}$ for HILL, CMD-HILL, SI-HILL, SI-CMD-HILL, and two versions of the proposed nmSI-HILL. Steganalysis with SRM and maxSRMd2 [9] and the ensemble classifier [28].

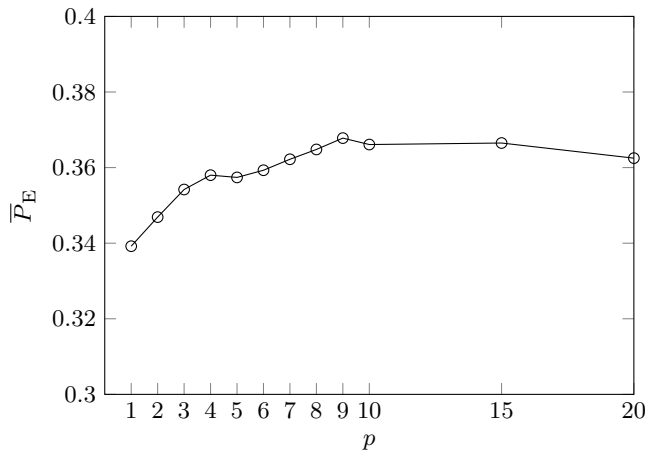


Figure 2: Average total detection error under equal priors $\overline{P}_{\mathrm{E}}$ and its statistical spread over ten splits of BOSSbase into training and testing as a function of the parameter p in (16). Additive embedding scheme HILL at 0.4 bpp, SI from RGB to gray conversion, SRM, ensemble classifier.

According to our experiments, $\alpha = 0$ gave the best results. To determine the value of the exponent p, in Figure 2 we show the detection error for nmSI-HILL at 0.4 bpp as a function of p (with $\alpha = 0$). The value $p \approx 9$ corresponds to a rather flat optimum. Table 1 shows the average total detection error under equal priors $\overline{P}_{\mathrm{E}}$ of HILL [29], CMD-HILL [30], SI-HILL [6], SI-CMD-HILL, and nmSI-HILL with two sets of values for the parameters p and α. The case of nmSI-HILL with $p = \infty$ and $\alpha = 1/9$ was included to show the effect of weighting the embedding changes by $1 - 2|e_{kl}|$ when computing μ_{ij} in (15).

As expected, SI boosts security more than synchronizing the polarities of neighboring modifications. The synchronization improves the security of SI-HILL by 3.5% (SRM) and 3.25% (maxSRMd2). The proposed nmSI-HILL improves upon a naive combination of SI and CMD-HILL by almost 4% (SRM) and 1.5% (maxSRMd2).

Experiments

In this section, we report the results of all experiments to show the benefit of synchronizing the polarity of neighboring embedding changes with SI. We do so for three types of side-information, two different datasets, and with steganalysis implemented with rich models as well as a deep neural network SRNet [3].

Datasets

Two datasets were used for our experiments: BOSSbase 1.01 (as described in the previous section) and a dataset derived from images made available to ALASKA competitors during the recent steganalysis challenge [4].

Experiments on BOSSbase were carried out only for one type of side-information when converting a true color image to gray exactly as explained in the previous section. Since there are only 10,000 images in this dataset, steganalysis was executed only using the SRM and maxSRMd2 feature sets coupled with the ensemble classifier. Detection with deep convolutional neural networks [40, 3, 41, 42] was not included for this dataset because such detectors require much larger datasets for proper training.

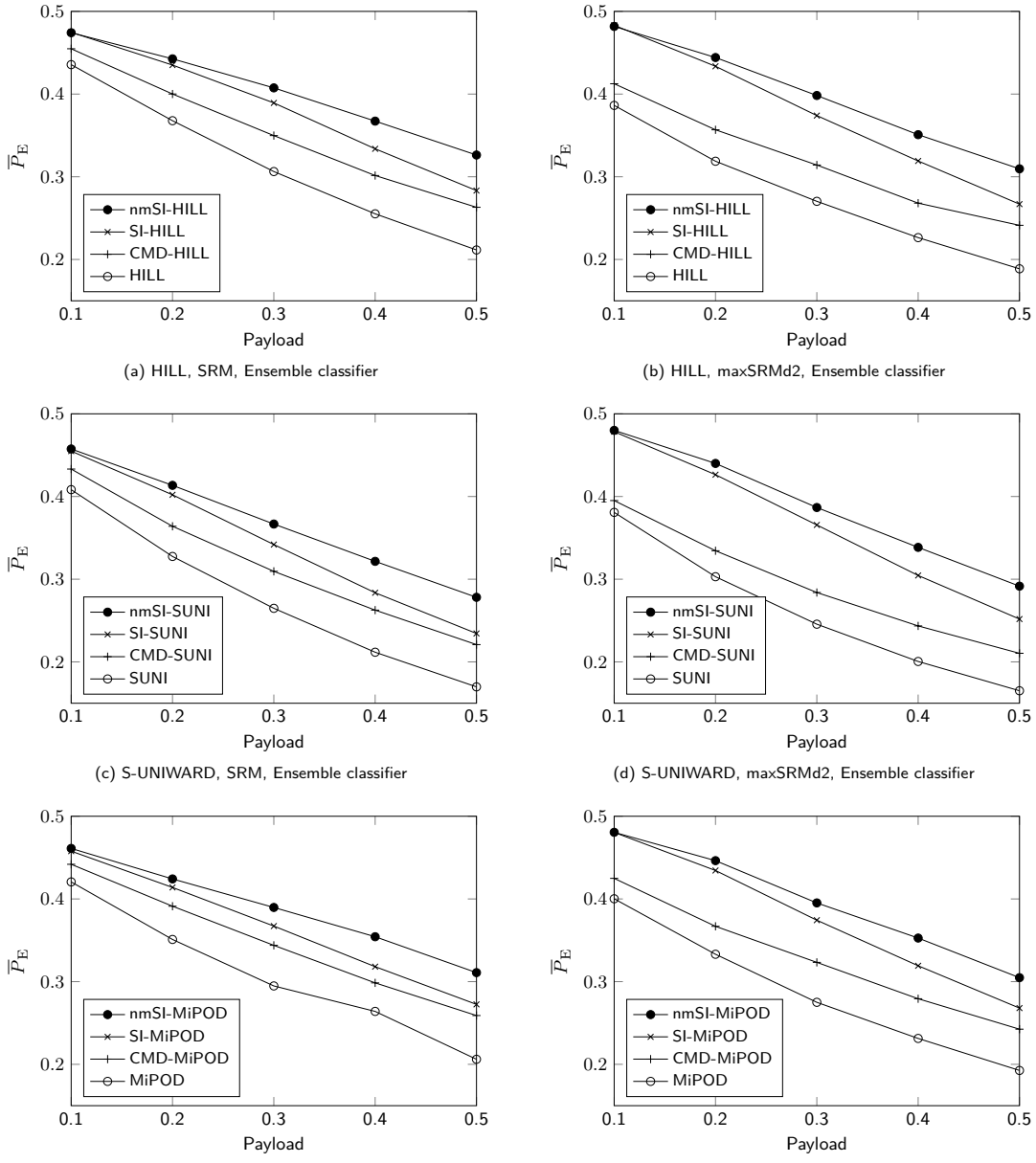
For more realistic conditions, and to be able to investigate other types of side-information with the SRNet, we added experiments on a second dataset derived from 47,260 RAW images provided as part of the steganalysis competition ALASKA.[4] Available from the same web site is the script for developing the RAW images to the TIFF format, which we modified to only use the 'dem_amaze.pp3' RAW converter and output uncompressed images of the same size. The reader is referred to the above-cited ALASKA web site for more information about the script.

Evaluation metric

The detection performance was measured with the total classification error under equal priors $P_{\mathrm{E}} = \frac{1}{2}(P_{\mathrm{FA}} + P_{\mathrm{MD}})$ on the testing set, where P_{FA} and P_{MD} stand for the false-alarm and missed-detection probabilities. For rich models with the ensemble on BOSSbase, we report P_{E} on the testing set averaged over ten random equal size splits into training and testing sets.

On ALASKA, due to its large size and to be able to compare the results with network detectors (SRNet), which are much more computationally demanding to train, in agreement with most prior art, we report the results for one random 40,460 / 3,200 / 3,600 split of the database into training / validation / testing. Rich models were trained on the union of the training and validation sets. Based on the results reported in [3], the statistical spread of the detection error (scaled to [0,1]) for the SRNet in terms of the mean absolute deviation is 0.002–0.003, which is comparable to what has typically been reported for detectors implemented with rich models.

[4] `https://alaska.utt.fr`



(a) HILL, SRM, Ensemble classifier

(b) HILL, maxSRMd2, Ensemble classifier

(c) S-UNIWARD, SRM, Ensemble classifier

(d) S-UNIWARD, maxSRMd2, Ensemble classifier

(e) MIPOD, SRM, Ensemble classifier

(f) MIPOD, maxSRMd2, Ensemble classifier

Figure 3: Average detection error $\overline{P}_{\mathrm{E}}$ as a function of payload size for four versions of HILL, S-UNIWARD, and MiPOD. BOSSbase 1.01, SRM (left), maxSRMd2 (right), ensemble classifier. The graphs contain the original embedding algorithm, CMD and SI versions, and the combination of SI and synchronized embedding polarities (nmSI). SI by RGB to gray conversion.

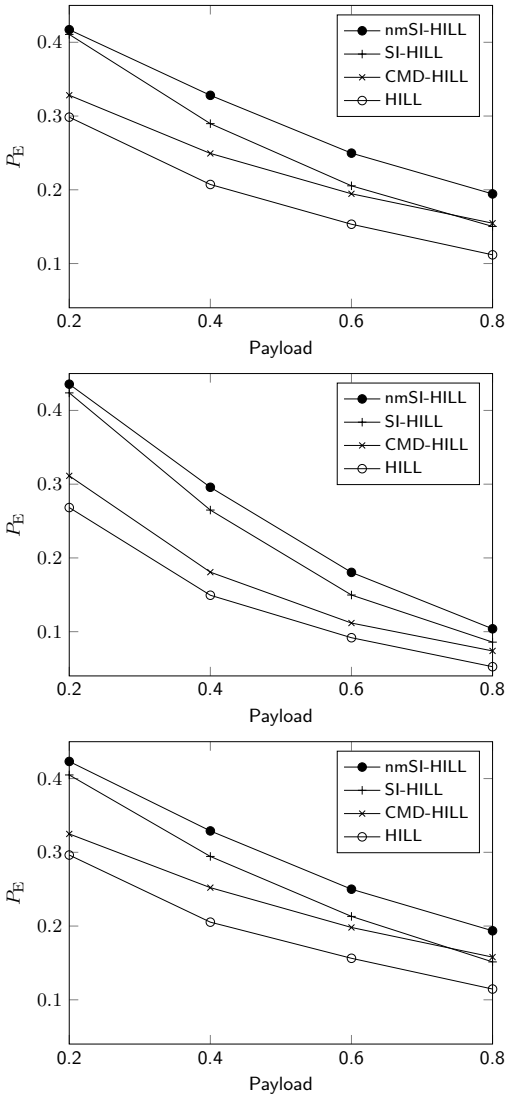


Figure 4: Detection error P_E as a function of payload size for four versions of HILL on ALASKA dataset. Top down, side-information obtained by RGB to gray conversion, quantization from 16 to 8 bits, and resizing. Steganalysis with maxSRMd2 and ensemble.

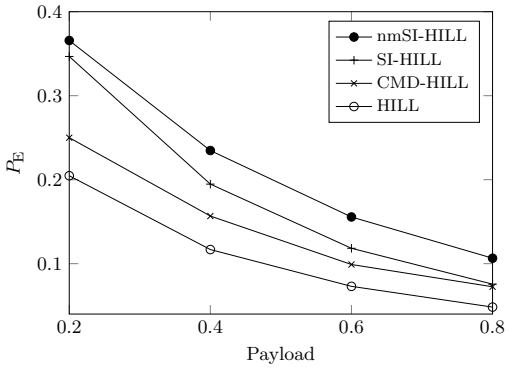


Figure 5: Detection error P_E of SRNet for four versions of HILL. ALASKA, side-information RGB to gray.

BOSSbase

Figure 3 shows the average detection error $\overline{P}_E$ achieved with the SRM and maxSRMd2 features and the ensemble for four different versions of HILL, S-UNIWARD, and MiPOD and five payloads on BOSSbase with SI in the form of the rounding errors when converting a true-color image to grayscale. The graphs do not show error bars as they were too small in the graphs to see and made the markers harder to discern. Tables containing the full numerical results, including the statistical spread, are at the end of the paper.

The results are consistent across the embedding algorithms and both steganalysis feature sets. SI offers a bigger boost in security than CMD. When combined in nmSI, they improve empirical security w.r.t. SI schemes by more than 4% in terms of P_E for the largest tested payload. This improvement naturally diminishes with decreased payload because of the lower density of embedding changes. The gain of synchronization of neighboring change polarities in SI schemes (nmSI vs. SI) is slightly smaller (by about 1%) than the gain provided by CMD vs. regular additive version of the embedding. This is to be expected because the SI and neighboring changes can be in conflict (point in opposite directions).

ALASKA

On this dataset, we experimented with three types of side-information for HILL: resizing, color conversion, and quantization (the same processing was investigated in [6]). For resizing, the RAW image was developed to a full resolution true-color (8 bit per channel) image, then converted to gray, resized (cubic kernel) so that the smaller size was 256, and centrally cropped to 256×256. For color conversion, the full resolution true-color image was resized to a true-color image so that the smaller size was 256, centrally cropped to 256×256, and then converted from RGB to gray as above. For quantization, the image was developed to a 16-bit per channel TIFF, centrally cropped to 256×256, then converted to 16 bit gray, rounded and quantized to 8 bit. The intermediate representation of the data for resizing, color converion, and quantization, respectively, is: RAW $\rightarrow$ RGB-FULL(8B) $\rightarrow$ GRAY-FULL(DBL) $\stackrel{\text{scale}}{\rightarrow}$ RGRAY-256×256(DBL), RAW $\rightarrow$ RGB-FULL(8B) $\rightarrow$ RGB-256×256(8B) $\rightarrow$ GRAY-256×256(DBL), and RAW $\rightarrow$ RGB-FULL(16B) $\stackrel{\text{crop}}{\rightarrow}$ RGB-256×256(16B) $\rightarrow$ GRAY-256×256-DBL(16B) $\rightarrow$ GRAY-256×256(16B) $\rightarrow$ GRAY-256×256-DBL(8B). The abbreviation FULL stands for full resolution, DBL for double Matlab format, and the number in round brackets is the number bits (quantization) representing each pixel / color.

Because the images are smaller and also because this dataset is more noisy than BOSSbase, the tested payloads were increased to 0.2, 0.4, 0.6, and 0.8 bpp. The results obtained with the maxSRMd2 features and the ensemble classifier are shown in Figure 4. The benefit of synchronizing polarities of neighboring embedding changes is more than 4% for the largest payloads and it decreases for smaller payloads. The results for resizing and color conversion are

very similar both in terms of absolute detectability and the improvement of combining synchronization with SI. For quantization, the boost is smaller but consistent with the smaller benefit of CMD applied to HILL in this dataset.

Due to the significantly larger computational complexity of training the SRNet, we report the results only for SI from color conversion. Sample tests with the selection-channel-aware version of the SRNet, the SCA-SRNet, showed that the knowledge of the selection channel either did not help or brought only a small improvement over SRNet. This is probably due to the combined effect of SI and synchronization as both modulate the selection channel in a way that is not available to the steganalyst. Moreover, because training the SCA version is slower, we only report the results with the SRNet. The detection error of the SRNet is significantly lower than what has been achieved with rich models, especially for lower payloads (Figure 5). The results are, however, consistent with rich models in terms of trends and mutual comparison across the four versions of the embedding algorithm.

Conclusions

This paper proposes a novel idea to incorporate synchronization of the polarities of neighboring embedding modifications in steganography with side-information. Both measures have previously been shown to improve empirical security on their own. Also, both lead to asymmetric embedding change probabilities of modifications by ± 1. The proposed method first modulates the costs by the side-information, and then embeds the message on four interleaved sublattices. The first sublattice is embedded using only the side-information, while the side-informed costs of pixels in the remaining three sublattices are further adjusted by a multiplicative factor that non-linearly depends on a weighted average of neighboring changes. The method is tested for three spatial-domain embedding schemes, three types of side-information, on two datasets, and with both rich-feature models and deep neural detectors. The results are rather consistent across all tested algorithms, types of side-information, and detectors. The synchronization of embedding change polarities offers an additional boost in empirical security over purely side-informed schemes that is slightly smaller than the boost of synchronizing the polarities in the original additive schemes. We hypothesize that this due to the fact that side-information may "point to the opposite direction" than the majority of neighboring changes, which leads to a conflict.

Our future effort will be directed towards a model-based scheme in which the coupling of neighboring change polarities is dictated by minimizing the KL divergence between the cover model and its asymmetric stego mixture similar to the approach proposed in [24].

All code used to produce the results in this paper, including the network configuration files are available from `http://dde.binghamton.edu/download/`.

Acknowledgments

The work on this paper was supported by NSF grant No. 1561446. Thanks belong to Jan Butora for preparing the ALASKA datasets for this paper.

References

[1] P. Bas. Steganography via cover-source switching. In *IEEE International Workshop on Information Forensics and Security*, Abu Dhabi, December 4–7 2016.

[2] P. Bas. An embedding mechanism for natural steganography after down-sampling. In *IEEE ICASSP*, New Orleans, March 5–9 2017.

[3] M. Boroumand, M. Chen, and J. Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, May 2019.

[4] R. Cogranne, Q. Giboulot, and P. Bas. The ALASKA steganalysis challenge: A first step towards steganalysis "Into the wild". In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[5] T. Denemark and J. Fridrich. Improving steganographic security by synchronizing the selection channel. In J. Fridrich, P. Comesana, and A. Alattar, editors, *3rd ACM IH&MMSec. Workshop*, Portland, Oregon, June 17–19, 2015.

[6] T. Denemark and J. Fridrich. Side-informed steganography with additive distortion. In *IEEE International Workshop on Information Forensics and Security*, Rome, Italy, November 16–19 2015.

[7] T. Denemark and J. Fridrich. Model based steganography with precover. In A. Alattar and N. D. Memon, editors, *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2017*, San Francisco, CA, January 29–February 1, 2017.

[8] T. Denemark and J. Fridrich. Steganography with two JPEGs of the same scene. In *IEEE ICASSP*, New Orleans, March 5–9 2017.

[9] T. Denemark, V. Sedighi, V. Holub, R. Cogranne, and J. Fridrich. Selection-channel-aware rich model for steganalysis of digital images. In *IEEE International Workshop on Information Forensics and Security*, Atlanta, GA, December 3–5, 2014.

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

[11] E. Franz. Steganography preserving statistical properties. In F. A. P. Petitcolas, editor, *Information Hiding, 5th International Workshop*, volume 2578 of Lecture Notes in Computer Science, pages 278–294, Noordwijkerhout, The Netherlands, October 7–9, 2002. Springer-Verlag, New York.

[12] E. Franz. Embedding considering dependencies between pixels. In E. J. Delp, P. W. Wong, J. Dittmann, and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume

	0.1	0.2	0.3	0.4	0.5
HILL	0.4356 ± 0.0022	0.3678 ± 0.0036	0.3063 ± 0.0031	0.2552 ± 0.0026	0.2115 ± 0.0019
CMD-HILL	0.4548 ± 0.0021	0.4001 ± 0.0032	0.3498 ± 0.0023	0.3015 ± 0.0034	0.2631 ± 0.0027
SI-HILL	0.4747 ± 0.0012	0.4351 ± 0.0025	0.3893 ± 0.0019	0.3338 ± 0.0034	0.2833 ± 0.0018
nmSI-HILL	0.4742 ± 0.0012	0.4426 ± 0.0022	0.4075 ± 0.0024	0.3673 ± 0.0027	0.3263 ± 0.0026

Table 2: $\overline{P}_{\mathrm{E}}$ for four versions of HILL, SRM, ensemble classifier, BOSSbase, side-information RGB to gray.

	0.1	0.2	0.3	0.4	0.5
S-UNIWARD	0.4083 ± 0.0020	0.3275 ± 0.0022	0.2648 ± 0.0030	0.2117 ± 0.0026	0.1698 ± 0.0026
CMD-S-UNIWARD	0.4332 ± 0.0026	0.3640 ± 0.0029	0.3095 ± 0.0027	0.2626 ± 0.0024	0.2209 ± 0.0022
SI-S-UNIWARD	0.4549 ± 0.0018	0.4021 ± 0.0029	0.3419 ± 0.0026	0.2836 ± 0.0029	0.2344 ± 0.0021
nmSI-S-UNIWARD	0.4575 ± 0.0021	0.4135 ± 0.0016	0.3667 ± 0.0027	0.3216 ± 0.0033	0.2781 ± 0.0018

Table 3: $\overline{P}_{\mathrm{E}}$ for four versions of S-UNIWARD, SRM, ensemble classifier, BOSSbase, RGB to gray.

	0.1	0.2	0.3	0.4	0.5
MiPOD	0.4206 ± 0.0021	0.3510 ± 0.0021	0.2948 ± 0.0033	0.2640 ± 0.0033	0.2061 ± 0.0013
CMD-MiPOD	0.4419 ± 0.0026	0.3912 ± 0.0022	0.3439 ± 0.0030	0.2985 ± 0.0025	0.2590 ± 0.0025
SI-MiPOD	0.4578 ± 0.0023	0.4139 ± 0.0031	0.3672 ± 0.0028	0.3181 ± 0.0029	0.2725 ± 0.0033
nmSI-MiPOD	0.4611 ± 0.0017	0.4242 ± 0.0015	0.3897 ± 0.0022	0.3543 ± 0.0033	0.3109 ± 0.0025

Table 4: $\overline{P}_{\mathrm{E}}$ for four versions of MiPOD, SRM, ensemble classifier, BOSSbase, RGB to gray.

	0.1	0.2	0.3	0.4	0.5
HILL	0.3863 ± 0.0014	0.3188 ± 0.0030	0.2703 ± 0.0030	0.2264 ± 0.0015	0.1887 ± 0.0023
CMD-HILL	0.4125 ± 0.0030	0.3568 ± 0.0030	0.3142 ± 0.0030	0.2681 ± 0.0016	0.2413 ± 0.0025
SI-HILL	0.4832 ± 0.0031	0.4336 ± 0.0019	0.3740 ± 0.0020	0.3191 ± 0.0022	0.2669 ± 0.0035
nmSI-HILL	0.4818 ± 0.0020	0.4442 ± 0.0033	0.3983 ± 0.0032	0.3509 ± 0.0029	0.3096 ± 0.0026

Table 5: $\overline{P}_{\mathrm{E}}$ for four versions of HILL, maxSRMd2, ensemble classifier, BOSSbase, RGB to gray.

	0.1	0.2	0.3	0.4	0.5
S-UNIWARD	0.3808 ± 0.0022	0.3030 ± 0.0025	0.2456 ± 0.0029	0.2005 ± 0.0023	0.1651 ± 0.0016
CMD-S-UNIWARD	0.3951 ± 0.0037	0.3344 ± 0.0029	0.2839 ± 0.0026	0.2435 ± 0.0022	0.2104 ± 0.0015
SI-S-UNIWARD	0.4784 ± 0.0013	0.4264 ± 0.0019	0.3656 ± 0.0020	0.3045 ± 0.0031	0.2518 ± 0.0028
nmSI-S-UNIWARD	0.4800 ± 0.0027	0.4400 ± 0.0020	0.3867 ± 0.0035	0.3385 ± 0.0017	0.2916 ± 0.0027

Table 6: $\overline{P}_{\mathrm{E}}$ for four versions of S-UNIWARD, maxSRMd2, ensemble classifier, BOSSbase, RGB to gray.

	0.1	0.2	0.3	0.4	0.5
MiPOD	0.4002 ± 0.0018	0.3331 ± 0.0019	0.2750 ± 0.0026	0.2313 ± 0.0017	0.1926 ± 0.0016
CMD-MiPOD	0.4249 ± 0.0023	0.3669 ± 0.0016	0.3233 ± 0.0030	0.2793 ± 0.0020	0.2425 ± 0.0023
SI-MiPOD	0.4805 ± 0.0021	0.4344 ± 0.0024	0.3743 ± 0.0017	0.3192 ± 0.0023	0.2680 ± 0.0030
nmSI-MiPOD	0.4806 ± 0.0022	0.4463 ± 0.0034	0.3952 ± 0.0023	0.3527 ± 0.0026	0.3048 ± 0.0026

Table 7: $\overline{P}_{\mathrm{E}}$ for four versions of MiPOD, maxSRMd2, ensemble classifier, BOSSbase, RGB to gray.

	0.2	0.4	0.6	0.8
HILL	0.2985	0.2072	0.1533	0.1118
CMD-HILL	0.3282	0.2493	0.1946	0.1547
SI-HILL	0.4107	0.2897	0.2054	0.1503
nmSI-HILL	0.4171	0.3281	0.2497	0.1944

Table 8: P_E for four versions of HILL, maxSRMd2, ensemble classifier, ALASKA, side-information from RGB to gray.

	0.2	0.4	0.6	0.8
HILL	0.2683	0.1494	0.0918	0.0525
CMD-HILL	0.3113	0.1807	0.1117	0.0739
SI-HILL	0.4238	0.2649	0.1497	0.0858
nmSI-HILL	0.4356	0.2958	0.1804	0.1039

Table 9: P_E for four versions of HILL, maxSRMd2, ensemble classifier, ALASKA, side-information from quantization from 16 to 8 bits.

	0.2	0.4	0.6	0.8
HILL	0.2963	0.2053	0.1563	0.1146
CMD-HILL	0.3249	0.2521	0.1981	0.1579
SI-HILL	0.4049	0.2944	0.2131	0.1514
nmSI-HILL	0.4232	0.3290	0.2500	0.1936

Table 10: P_E for four versions of HILL, maxSRMd2, ensemble classifier, ALASKA, side-information from resizing.

	0.2	0.4	0.6	0.8
HILL	0.2047	0.1168	0.073	0.0483
CMD-HILL	0.2500	0.1567	0.099	0.0725
SI-HILL	0.3467	0.1947	0.1183	0.0756
nmSI-HILL	0.3657	0.2347	0.1556	0.1065

Table 11: P_E achieved with SRNet for four different versions of HILL. ALASKA, side-information from RGB to gray.

6819, pages D 1–12, San Jose, CA, January 27–31, 2008.

[13] E. Franz and A. Schneidewind. Pre-processing for adding noise steganography. In M. Barni, J. Herrera, S. Katzenbeisser, and F. Pérez-González, editors, *Information Hiding, 7th International Workshop*, volume 3727 of Lecture Notes in Computer Science, pages 189–203, Barcelona, Spain, June 6–8, 2005. Springer-Verlag, Berlin.

[14] J. Fridrich. On the role of side-information in steganography in empirical covers. In A. Alattar, N. D. Memon, and C. Heitzenrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2013*, volume 8665, pages 0I 1–11, San Francisco, CA, February 5–7, 2013.

[15] J. Fridrich and R. Du. Secure steganographic methods for palette images. In A. Pfitzmann, editor, *Information Hiding, 3rd International Workshop*, volume 1768 of Lecture Notes in Computer Science, pages 47–60, Dresden, Germany, September 29–October 1, 1999. Springer-Verlag, New York.

[16] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography using wet paper codes. In J. Dittmann and J. Fridrich, editors, *Proceedings of the 6th ACM Multimedia & Security Workshop*, pages 4–15, Magdeburg, Germany, September 20–21, 2004.

[17] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography. *ACM Multimedia System Journal*, 11(2):98–107, 2005.

[18] J. Fridrich and J. Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, June 2011.

[19] L. Guo, J. Ni, and Y.-Q. Shi. An efficient JPEG steganographic scheme using uniform embedding. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.

[20] L. Guo, J. Ni, and Y. Q. Shi. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 9(5):814–825, May 2014.

[21] V. Holub. *Content Adaptive Steganography – Design and Detection*. PhD thesis, Binghamton University, May 2014.

[22] V. Holub and J. Fridrich. Designing steganographic distortion using directional filters. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.

[23] V. Holub, J. Fridrich, and T. Denemark. Universal distortion design for steganography in an arbitrary domain. *EURASIP Journal on Information Security, Special Issue on Revised Selected Papers of the 1st ACM IH and MMS Workshop*, 2014:1, 2014.

[24] X. Hu, J. Ni, W. Su, and J. Huang. Model-based image steganography using asymmetric embedding scheme. *Journal of Electronic Imaging*, 27(4):1 – 7, 2018.

[25] F. Huang, J. Huang, and Y.-Q. Shi. New channel selection rule for JPEG steganography. *IEEE*

Transactions on Information Forensics and Security, 7(4):1181–1191, August 2012.

[26] A. D. Ker. A fusion of maximal likelihood and structural steganalysis. In T. Furon, F. Cayre, G. Doërr, and P. Bas, editors, *Information Hiding, 9th International Workshop*, volume 4567 of Lecture Notes in Computer Science, pages 204–219, Saint Malo, France, June 11–13, 2007. Springer-Verlag, Berlin.

[27] 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.

[28] J. Kodovský, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, April 2012.

[29] B. Li, M. Wang, and J. Huang. A new cost function for spatial image steganography. In *Proceedings IEEE, International Conference on Image Processing, ICIP*, Paris, France, October 27–30, 2014.

[30] B. Li, M. Wang, X. Li, S. Tan, and J. Huang. A strategy of clustering modification directions in spatial image steganography. *IEEE Transactions on Information Forensics and Security*, 10(9):1905–1917, September 2015.

[31] 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.

[32] 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 Conference*, volume 6387 of Lecture Notes in Computer Science, pages 161–177, Calgary, Canada, June 28–30, 2010. Springer-Verlag, New York.

[33] T. Pevný and A. D. Ker. Exploring non-additive distortion in steganography. In R. Bohme and C. Pasquini, editors, *The 6th ACM Workshop on Information Hiding and Multimedia Security*, Innsbruck, Austria, June 20–22, 2018. ACM Press.

[34] 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.

[35] V. Sedighi, R. Cogranne, and J. Fridrich. Content-adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security*, 11(2):221–234, 2016.

[36] V. Sedighi and J. Fridrich. Effect of saturated pixels on security of steganographic schemes for digital images. In *IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, September 25–28 2016.

[37] V. Sedighi, J. Fridrich, and R. Cogranne. Content-adaptive pentary steganography using the multivariate generalized Gaussian cover model. In A. Alattar and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2015*, volume 9409, San Francisco, CA, February 8–12, 2015.

[38] T. Taburet, P. Bas, W. Sawaya, and J. Fridrich. A natural steganography embedding scheme dedicated to color sensors in the JPEG domain. In A. Alattar and N. D. Memon, editors, *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2019*, San Francisco, CA, January 14–17, 2019.

[39] C. Wang and J. Ni. An efficient JPEG steganographic scheme based on the block–entropy of DCT coefficents. In *Proc. of IEEE ICASSP*, Kyoto, Japan, March 25–30, 2012.

[40] J. Ye, J. Ni, and Y. Yi. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):2545–2557, November 2017.

[41] M. Yedroudj, M. Chaumont, and F. Comby. How to augment a small learning set for improving the performances of a CNN-based steganalyzer? In A. Alattar and N. D. Memon, editors, *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2018*, San Francisco, CA, January 29–February 1, 2018.

[42] M. Yedroudj, F. Comby, and M. Chaumont. Yedroudj-net: An efficient CNN for spatial steganalysis. In *IEEE ICASSP*, pages 2092–2096, Alberta, Canada, April 15–20, 2018.

Author Biography

Mehdi Boroumand received the B.S. degree in electrical engineering from the K. N. Toosi University of Technology, Iran, in 2004, and the M.S. degree in electrical engineering from the Sahand University of Technology, Iran, in 2007. He graduated with the Ph.D. degree in electrical engineering from Binghamton University in 2019. His areas of research interest include digital image steganalysis and steganography, digital image forensics, image processing and computer vison, and machine learning.

Jessica Fridrich is Distinguished Professor of Electrical and Computer Engineering at Binghamton University. She received her PhD in Systems Science from Binghamton University in 1995 and MS in Applied Mathematics from Czech Technical University in Prague in 1987. Her main interests are in steganography, steganalysis, and digital image forensics. Since 1995, she has received 20 research grants totaling over $12 mil that lead to more than 200 papers and 7 US patents.