Digital Image Steganography Using Universal Distortion

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

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

ABSTRACT

Currently, the most secure practical steganographic schemes for empirical cover sources embed their payload while minimizing a distortion function designed to capture statistical detectability. Since there exists a general framework for this embedding paradigm with established payload–distortion bounds as well as near-optimal practical coding schemes, building an embedding scheme has been essentially reduced to the distortion design. This is not an easy task as relating distortion to statistical detectability is a hard and open problem. In this article, we propose an innovative idea to measure the embedding distortion in one fixed domain independently of the domain where the embedding changes (and coding) are carried out. The proposed universal distortion is additive and evaluates the cost of changing an image element (e.g., pixel or DCT coefficient) from directional residuals obtained using a Daubechies wavelet filter bank. The intuition is to limit the embedding changes only to those parts of the cover that are difficult to model in multiple directions while avoiding smooth regions and clean edges. The utility of the universal distortion is demonstrated by constructing steganographic schemes in the spatial, JPEG, and side-informed JPEG domains, and comparing their security to current state-of-the-art methods using classifiers trained with rich media models.

Categories and Subject Descriptors

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

General Terms

Security, Algorithms, Theory

Keywords

Steganography, distortion function, JPEG, side-informed embedding

IH&MMSec'13, June 17–19, 2013, Montpellier, France.

1. MOTIVATION

The lack of accurate models for complex sources, such as digital media, significantly complicates the construction of secure steganographic schemes. One may even argue that, fundamentally, perfect steganographic security in such empirical sources is not possible [1]. Postulating this as an assumption gave birth to the study of imperfect steganography as a subdiscipline that better corresponds to real-world conditions and lead to fundamental new results, such as scaling laws of secure payload w.r.t. various cover source attributes, such as the length (the square root law [16, 21, 6]), quantization [8], and resolution [24]. Steganographic capacity of imperfect steganography is zero as the secure payload scales sublinearly with cover length.[1] Thus, one can say that in practice, steganographers merely try to increase the steganographic Fisher information, which defines the root rate [18, 19, 2] – the proper measure of secure payload for imperfect steganography.

The mainstream (and by far the most successful) approach is framing the embedding as source coding with a fidelity constraint [30] and build the embedding around a distortion function that is minimized to embed a desired payload [5, 3, 22, 31, 28, 13, 15]. Upon closer inspection of these references, one discovers that the distortion functions are always designed either in the embedding domain or in a selected model (feature) space. The first alternative can be rightfully challenged as, for example, changing a DCT coefficient has an effect on an entire block of pixels, and the detectability of this embedding change needs to consider this fact. Designing distortion in a model space [28, 4] is more appealing but can only succeed with a sufficiently comprehensive source model to avoid creating security holes for the Warden who chooses to work outside of the model [25].

In this paper, we propose a distortion function that allows careful analysis of the impact of making an embedding change on the local content and thus introduce less detectable artifacts. We work with a wavelet representation of the cover image (if the image is represented in some other domain, such as JPEG, it is first decompressed to the spatial domain prior to the wavelet transform), which can be viewed as a representation obtained using a bank of directional filters. Interpreting the highest frequency undecimated subbands as directional residuals, one can assess the impact of an embedding change in multiple directions, which allows us to constrain the embedding changes to textures and noisy regions of the image while avoiding smooth content as well

[1] This result was derived for a fully-informed Warden and it may change, depending on Warden's ignorance [20].

as clean edges. This is a model-free approach as we do not work with a feature representation of the cover image.

We implement three versions of one embedding algorithm depending on the cover representation – spatial, JPEG, and side-informed JPEG domains. To prove the merit of our construction, a comparison to the current state-of-the-art steganographic algorithms is included for each domain. Since the proposed distortion is in the form of a sum of *relative* changes between the stego and cover images represented in the wavelet domain, we named it UNIversal WAvelet Relative Distortion (UNIWARD). We would like to point out that this paper is a shortened version of our recent journal submission to IEEE TIFS, which differs from this article in several aspects. The journal version contains a thorough analysis of embedding using the Gibbs construction with the non-additive version of UNIWARD in the spatial domain. Furthermore, to stay within the page limits of this workshop, we steered this article more towards the JPEG domain and limited the scale of experiments in the spatial domain. The wording in this article has also been altered to avoid copyright conflicts and the flow restructured to give it a form that is more suitable for a workshop article and more likely to elicit audience discussions.

The purpose of Section 2 is to introduce notation and basic concepts. In Section (3), we describe the UNIWARD costs for images represented in an arbitrary domain as well as for side-informed JPEG steganography when the sender has the raw, uncompressed cover available and wishes to embed in its JPEG compressed form. The common core of all experiments is summarized in Section 4, where we provide details about the cover source, machine learning, and the measure used for empirical evaluation of security. Section 5 contains the results of all experiments in the spatial, JPEG, and side-informed JPEG domains including the comparison with previous art. The paper is concluded in Section 6.

2. NOTATION AND BASIC CONCEPTS

To improve the readability of this article, we adopt the following conventions. Capital and lower-case boldface symbols will be used solely for matrices and vectors, respectively. The symbols $\mathbf{X} = (X_{ij}), \mathbf{Y} = (Y_{ij}) \in \{0, \ldots, 255\}^{n_1 \times n_2}$ will always stand for matrices representing a cover and the corresponding stego image with $n_1 \times n_2$ pixels/DCT coefficients. For simplicity we only work with 8-bit grayscale images, which means that $X_{ij}, Y_{ij} \in \{0, \ldots, 255\}$. For JPEG images, $X_{ij}, Y_{ij} \in \{-1024, \ldots, 1023\}$ stand for quantized JPEG DCT coefficients arranged into an $n_1 \times n_2$ matrix by replacing each 8×8 pixel block with the corresponding block of quantized DCT coefficients. For simplicity and without any loss on generality, we will assume that n_1 and n_2 are integer multiples of 8.

For matrix $\mathbf{A}$, its transpose is $\mathbf{A}^{\mathrm{T}}$, while $|\mathbf{A}| = (|a_{ij}|)$ is the matrix of absolute values. Furthermore, we reserve the indices i, j to index pixels or DCT coefficients, while u, v will always index wavelet decomposition coefficients. The abbreviation 'w.r.t.' stands for "with respect to." Finally, $[S]$ is the Iverson bracket equal to 1 when the statement S is true and 0 when S is false.

2.1 JPEG compression

The raw image before JPEG compression will be denoted as $\mathbf{P} = (P_{ij}) \in \{0, \ldots, 255\}^{n_1 \times n_2}$. When applying JPEG compression to $\mathbf{P}$, first a blockwise DCT transform is applied to each 8×8 block of pixels from a fixed non-overlapping grid. Then, the DCT coefficients are divided by quantization steps and rounded to integers. Formally, let $\mathbf{P}^{(b)}$ be the bth 8×8 block when ordering the blocks, e.g., in a row-by-row fashion ($b = 1, \ldots, n_1 \times n_2/64$). With an 8×8 luminance quantization matrix $\mathbf{Q} = \{Q_{kl}\}$, $1 \leq k, l \leq 8$, we denote $\mathbf{D}^{(b)} = \mathrm{DCT}(\mathbf{P}^{(b)})./\mathbf{Q}$ the raw (non-rounded) values of DCT coefficients. Here, the operation $'./'$ is elementwise division of matrices and DCT(.) is the DCT transform used in the JPEG compressor. Finally, we denote by $\mathbf{X}^{(b)} = \mathrm{round}(\mathbf{D}^{(b)})$ the quantized DCT coefficients rounded to integers. We use the symbols $\mathbf{D}$ and $\mathbf{X}$ to denote the arrays of all raw and quantized DCT coefficients when arranging all blocks $\mathbf{D}^{(b)}$ and $\mathbf{X}^{(b)}$ in the same manner as the 8×8 pixel blocks in the uncompressed image.

2.2 DCT transform

The JPEG format allows several different implementations of the DCT transform, DCT(.), which may especially impact the security of side-informed steganographic methods that assign costs based on the DCT coefficients' rounding errors. In this work, we use the DCT(.) implemented as 'dct2' in Matlab with the input matrix of pixel values represented as 'double'. In particular, a block of 8×8 DCT coefficients is computed from a block $\mathbf{P}^{(b)}$ as

$$\mathrm{DCT}(\mathbf{P}^{(b)})_{kl} = \sum_{i,j=0}^{7} \frac{w_k w_l}{4} \cos \frac{\pi k(2i+1)}{16} \times \cos \frac{\pi l(2j+1)}{16} P_{ij}^{(b)}, \qquad (1)$$

where $k, l \in \{0, \ldots, 7\}$ index the DCT mode (spatial frequency) and $w_0 = 1/\sqrt{2}$, $w_k = 1$ for $k > 0$.

To make sure that both the cover and stego images were created using the same JPEG compressor and to guarantee that our steganalyzers will not be detecting compressor artifacts but only the impact of embedding, we adopt the following procedure for all steganographic algorithms that output JPEG stego images. To obtain an actual JPEG image from a two-dimensional array of quantized DCT coefficients $\mathbf{X}$ (cover) or $\mathbf{Y}$ (stego), we first create an (arbitrary) JPEG image of the same dimensions $n_1 \times n_2$ using Matlab's 'imwrite' with the same quality factor, read its JPEG data structure using 'jpeg_read' from Sallee's JPEG Toolbox (`http://www.philsallee.com/jpegtbx/index.html`) and then merely replace the array of quantized coefficients in this structure with $\mathbf{X}$ and $\mathbf{Y}$ to obtain the corresponding cover and stego images.

3. UNIVERSAL DISTORTION FUNCTION

In this section, we describe the universal distortion function UNIWARD that will be used to construct steganographic schemes in all embedding domains. To this end, in Section 3.1 we first introduce the wavelet directional filter bank using which UNIWARD is built and then, in Sections 3.2–3.3 we define the distortion between cover and stego images as a sum of relative changes between wavelet coefficients. We do so separately for steganography in the spatial and JPEG domains and for side-informed JPEG domain when the sender has an uncompressed image available. Since the distortion defined in this manner is non-additive, in Section 3.4 we explain a general procedure (originally introduced in [3]) that gives UNIWARD an additive form to be able to embed in practice near the payload–distortion bound using Syndrome–Trellis Codes (STCs) [5].

3.1 Wavelet directional filter bank

For a given image $\mathbf{X}$ represented in the spatial domain, we evaluate its smoothness in multiple directions using the Daubechies 8-tap Wavelet Directional Filter Bank (D-WDFB) $\mathcal{B} = \{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\}$ consisting of the LH, HL, and HH directional high-pass filters (kernels $\mathbf{K}$). These three filters are built from one-dimensional low-pass ($\mathbf{h}$) and high-pass ($\mathbf{g}$) decomposition filters shown in Table 1:

$$\mathbf{K}^{(1)} = \mathbf{h} \cdot \mathbf{g}^{\mathrm{T}},\ \mathbf{K}^{(2)} = \mathbf{g} \cdot \mathbf{h}^{\mathrm{T}},\ \mathbf{K}^{(3)} = \mathbf{g} \cdot \mathbf{g}^{\mathrm{T}}. \qquad (2)$$

Observe from Table 1 that the support of each one-dimensional filter is 16, which gives the kernels the size of 16×16. We define the kth *directional residual* as $\mathbf{R}^{(k)} = \mathbf{K}^{(k)} \star \mathbf{X}$, $k = 1, 2, 3$, where '$\star$' is a mirror-padded convolution that gives each $\mathbf{R}^{(k)}$ the dimension of $n_1 \times n_2$. The purpose of the mirror-padding is to prevent introducing embedding artifacts at the image boundary. Also notice that the directional residuals are essentially the first-level[2] *undecimated* wavelet LH, HL, and HH directional decomposition of $\mathbf{X}$. The reason for selecting this filter bank for constructing UNIWARD is found in [14], where, among the Daubechies wavelets, the authors studied several different filter banks, including the Sobel edge detector, non-directional kernels, and Haar wavelets. Since the Daubechies wavelets gave consistently the best results, we use this filter bank in this article as well.

3.2 UNIWARD for spatial and JPEG domains

Given a pair of cover and stego images, $\mathbf{X}$, and $\mathbf{Y}$, we will denote with $W_{uv}^{(k)}(\mathbf{X})$ and $W_{uv}^{(k)}(\mathbf{Y})$ the uvth wavelet coefficient in the kth decomposition obtained using kernels (2). If $\mathbf{X}$, and $\mathbf{Y}$ are JPEG images, they are first decompressed to the spatial domain and then the wavelet transform is applied. The distortion between both images is the sum of relative changes of the wavelet coefficients w.r.t. the cover image:

$$D(\mathbf{X}, \mathbf{Y}) \triangleq \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{X})|}, \qquad (3)$$

where the sum over uv is taken over all $n_1 \times n_2$ subband coefficients and $\varepsilon > 0$ is a stabilizing constant to avoid dividing by zero. In our implementation, we set $\varepsilon = 10 \times \text{eps}$ (in Matlab), which means that $\varepsilon \approx 10^{-15}$. From experiments, we found out that the security of embedding using UNIWARD is rather insensitive to the exact value of this parameter.

To understand the logic behind this definition, realize that the ratio in (3) is smaller when a large cover wavelet coefficient is changed, which will happen in textures/noisy regions and near edges. On the other hand, if at least one coefficient, which is small, is changed by a relatively large amount, the distortion value will also be large. Thus, (3) discourages making changes in regions where the content is smooth (and thus modelable) in at least one direction.

3.3 UNIWARD for side-informed JPEG embedding

In general, by side-informed embedding we understand any embedding method where the sender has a higher-quality version of the cover available (the so-called 'precover'). Historically, the first method that used precover was the Embedding by Dithering algorithm [9] that utilized a true-color image on its input to embed while converting the image to a 256-color palette GIF. The term precover is due to Ker [17].

Specifically in the JPEG domain, the precover attains the form of the unquantized DCT coefficients D_{ij} obtained from the raw precover image $\mathbf{P}$. In this case, the embedder may choose to round D_{ij} "up" or "down" to modulate its parity (e.g., the least significant bit of the rounded value). When compressing the precover $\mathbf{P}$ to the cover image $\mathbf{X}$, the rounding error for the ijth DCT coefficient is

$$e_{ij} = |D_{ij} - X_{ij}|, \quad e_{ij} \in [0, 0.5]. \qquad (4)$$

By rounding "to the other side," the sender introduces the following embedding change

$$Y_{ij} = X_{ij} + \operatorname{sign}(D_{ij} - X_{ij}), \qquad (5)$$

which corresponds to a "rounding error" of $1 - e_{ij}$. Therefore, every embedding change increases the distortion *w.r.t. the precover* by the *difference* between both rounding errors:

$$|D_{ij} - Y_{ij}| - |D_{ij} - X_{ij}| = 1 - 2e_{ij}. \qquad (6)$$

It is thus natural to define the distortion for side-informed embedding in JPEG domain as the difference:

$$\begin{aligned} D^{(\mathrm{SI})}(\mathbf{X}, \mathbf{Y}) &\triangleq D(\mathbf{P}, \mathbf{Y}) - D(\mathbf{P}, \mathbf{X}) \\ &= \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{Y})| - |W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{X})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{P})|}. \end{aligned} \qquad (7)$$

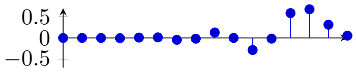
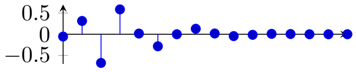
We would like to point out that the linearity of DCT and the wavelet transforms guarantee that $D^{(\mathrm{SI})}(\mathbf{X}, \mathbf{Y}) \geq 0$. This is because rounding a DCT coefficient to obtain a cover $\mathbf{X}$ corresponds to adding a certain two-dimensional 8×8 pattern in the spatial domain, which depends on the modified DCT mode, and thus a 23×23 pattern in the wavelet domain because the support of the 8-tap Daubechies wavelets is 16×16. On the other hand, rounding "to the other side" to obtain the stego image $\mathbf{Y}$ corresponds to *subtracting* the same pattern but with a *larger* amplitude, which is why $|W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{Y})| - |W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{X})| \geq 0$.

3.3.1 Relationship of UNIWARD to prior art

Equation (7) bears some similarity to the distortion utilized in the recently proposed Normalized Perturbed Quantization (NPQ) [15]. There, the authors also proposed to compute the embedding distortion as a *relative* change of cover DCT coefficients. What distinguishes UNIWARD from the distortion function of NPQ is the fact that we compute the distortion using a directional filter bank in the wavelet domain, which brings in a very important ingredient – directional sensitivity – and thus potentially better content adaptability. Furthermore, in our approach we treat all DCT coefficients equivalently and do not exclude those that are zeros in the cover. UNIWARD also naturally incorporates the influence of the quantization step because the

[2] Experiments with multiple decomposition levels did not improve security in any noticeable manner.

Table 1: One-dimensional filters used to construct the kernels of the D-WDFB using (2).

$\mathbf{h}$ = Daubechies 8-tap wavelet decomposition low-pass filter	
$\mathbf{g}$ = Daubechies 8-tap wavelet decomposition high-pass filter	

wavelet coefficients are computed from the decompressed JPEG image.

Also, distortion (3) is built similarly as the embedding distortion used in WOW [14] in that it is also capable of assessing local cover content using directional residuals computed in the wavelet domain. The pixel costs of WOW are, however, obtained in a different manner. First, for every subband and every pixel it computes the so-called embedding suitabilities, which are sums of weighted changes of wavelet coefficients. Then, the suitabilities are aggregated using a reciprocal Hölder norm to obtain costs with the property that if at least one suitability is zero (or very small), the embedding cost is infinite (very large). We refer the reader to the original publication for more details.

3.4 Additive form of UNIWARD

Note that both (3) and (7) are non-additive because changing pixel X_{ij} will affect a 16×16 neighborhood of wavelet coefficients (the support size of the Daubechies 8-tap wavelet). As already mentioned above, for images represented in the JPEG domain, changing a JPEG coefficient X_{ij} will affect a block of 8×8 pixels and thus 23×23 wavelet coefficients. Therefore, when changing neighboring pixels (or DCT coefficients), the embedding patterns overlap and the changes "interact," causing the non-additivity of D. Even though there exist methods for embedding using non-additive distortion functions (e.g., the Gibbs construction [3]), realizing the embedding using additive distortion is significantly easier. Moreover, in the case of UNIWARD, it appears that the interactions among nearby embedding changes are strong enough to make the Gibbs construction ineffective in practice. The Gibbs construction is only capable of embedding the so-called erasure entropy but with a distortion corresponding to the actual entropy of the Markov field. The stronger the interactions among embedding changes are, the larger is the difference between both entropies, and the less effective the Gibbs construction becomes. Detailed technical explanation of this issue supported with experiments on real images appears in the journal version of this paper.

As shown in [3], any distortion function $D(\mathbf{X},\mathbf{Y})$ can be used for embedding in its so-called *additive approximation* by using D to compute the cost of changing each pixel/DCT coefficient. In particular, the cost, ρ_{ij}, of changing X_{ij} to Y_{ij} when leaving all other cover elements unchanged is:

$$\rho_{ij}(\mathbf{X},Y_{ij}) \triangleq D(\mathbf{X},\mathbf{X}_{\sim ij}Y_{ij}), \tag{8}$$

where $\mathbf{X}_{\sim ij}Y_{ij}$ is the cover image $\mathbf{X}$ with only its ijth element changed: $X_{ij}\to Y_{ij}$.[3] Note that $\rho_{ij}=0$ when $\mathbf{X}=\mathbf{Y}$. We will denote the additive approximations to (3) and (7) with a subscript "A." For example, the additive approximation to $D(\mathbf{X},\mathbf{Y})$ is:

$$D_{\mathrm{A}}(\mathbf{X},\mathbf{Y}) = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\rho_{ij}(\mathbf{X},Y_{ij})[X_{ij}\neq Y_{ij}]. \tag{9}$$

Note that the presence of absolute values in $D(\mathbf{X},\mathbf{Y})$ (3) implies

$$\rho_{ij}(\mathbf{X},X_{ij}+1) = \rho_{ij}(\mathbf{X},X_{ij}-1), \quad \forall i,j, \text{ and } X_{ij}, \tag{10}$$

which permits us to use a *ternary* embedding operation for the spatial and JPEG domains. Practical embedding algorithms can be constructed using the ternary multi-layered version of STCs (Section IV in [5]). One might seemingly rightfully argue that the embedding cost should depend on the polarity of the change, however the equal cost for both possible changes is given by the distortion function. Moreover, since UNIWARD restricts the embedding changes to textures, the potential disadvantage of having equal costs for both polarities is reduced and it allows us to reduce the embedding distortion for a fixed payload by utilizing stronger ternary codes. This is expected to become especially advantageous for larger payloads. Finally, note that for the side-informed JPEG steganography, $D_{\mathrm{A}}^{(\mathrm{SI})}(\mathbf{X},\mathbf{Y})$ is inherently limited to a *binary* embedding operation because the sender has only two options – either rounding X_{ij} up or down.

The embedding methods that use the additive approximations of UNIWARD for the spatial, JPEG, and side-informed JPEG domain will be called S-UNIWARD, J-UNIWARD, and SI-UNIWARD, respectively.

4. SETUP OF ALL EXPERIMENTS

Before reporting the experimental results of embedding with UNIWARD in all three domains in the next section, we summarize the common core of all experiments.

4.1 Cover source

All experiments are conducted on the BOSSbase database ver. 1.01 [7] containing 10,000 512×512 8-bit grayscale images coming from eight different cameras. This database is very convenient for our purposes because it contains uncompressed images that serve as precovers for side-informed JPEG embedding that can be compressed to any desirable quality factor for the JPEG domain. The fact that the images are downsampled rather than raw has an effect on the statistical detectability, especially for algorithms operating in the spatial domain. According to the study carried out in [24], downsampling without antialiasing (that is with a fixed-size interpolation kernel) as is done for the BOSSbase makes detection of steganography *more difficult*

[3] This notation was used in [3] and is also standard in the literature on Markov random fields [32].

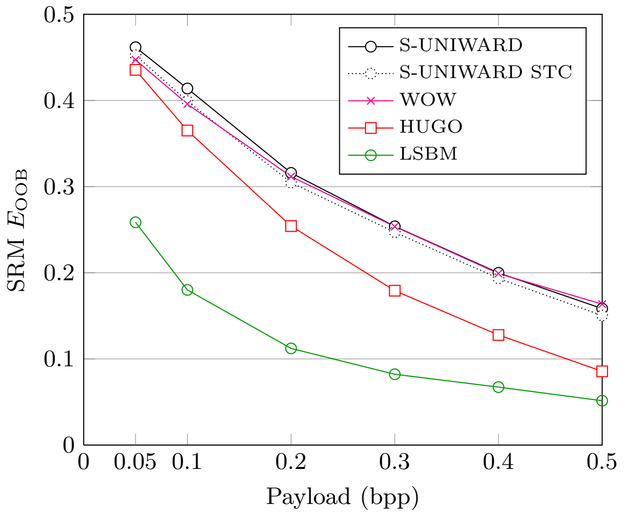


Figure 1: Detection error E_{OOB} as a function of relative payload for S-UNIWARD, HUGO, and LSBM. The dotted curve shows the performance of UNIWARD when implemented with STCs with constraint height $h = 12$.

rather than easier[4] despite the presence of resizing artifacts that might aid detection. This is because resizing in general decreases statistical dependencies between adjacent pixels, which seems to have a much stronger effect than the weak resizing artifacts. Perhaps a more careful statement would be to say that current empirical steganalyzers built using rich models are unable to utilize the resizing artifacts to the point that would outweigh the lowered dependencies among pixels.

The steganographic security is evaluated empirically using binary classifiers trained on a given cover source and its stego version embedded with a fixed payload. Even though this setup is artificial and does not correspond to real-life applications, it allows assessment of security w.r.t. the payload size, which is the goal of academic investigations of this type.

4.2 Features and machine learning

Spatial-domain steganography methods will be analyzed using the Spatial Rich Model (SRM) [11] consisting of 39 symmetrized sub-models quantized with three different quantization factors with a total dimension of 34, 671. JPEG-domain methods (including the side-informed algorithms) will be steganalyzed using the union of a downscaled version of the SRM with a single quantization step $q = 1$ (SRMQ1) with dimension 12, 753 and the JPEG Rich Model (JRM) [23] with dimension 22, 510, giving the total feature dimension of 35, 263.

All classifiers were implemented using the ensemble [26] with Fisher linear discriminant as the base learner. Security is quantified using the ensemble's "out-of-bag" (OOB) error E_{OOB}, which is an unbiased estimate of the minimal total *testing* error under equal priors [26] (equal a priori probabilities of encountering a cover or stego image):

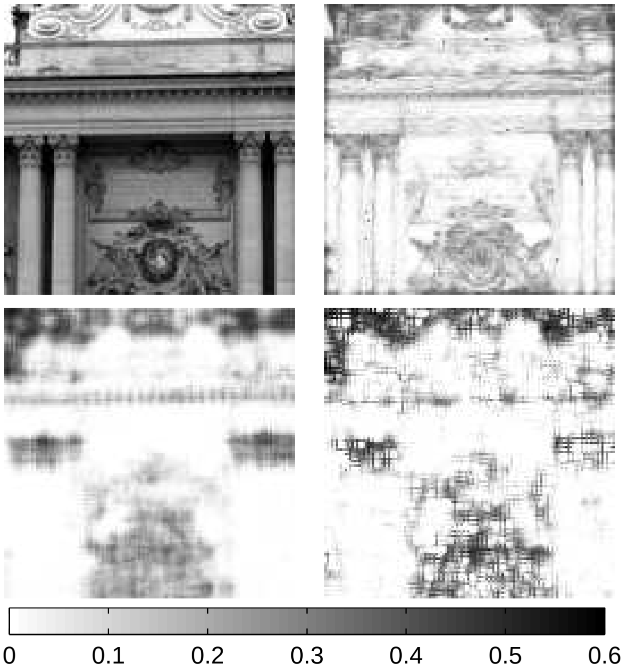


Figure 2: Embedding probability for payload 0.4 bpp using HUGO (top right), WOW (bottom left), and S-UNIWARD (bottom right) for a 128×128 grayscale cover image shown in top left.

$$P_{\mathrm{E}} = \min_{P_{\mathrm{FA}}} \frac{1}{2}(P_{\mathrm{FA}} + P_{\mathrm{MD}}). \qquad (11)$$

To show how the statistical detectability increases with payload, we produce graphs showing E_{OOB} as a function of the relative payload. With the feature dimensionality and the database size, the statistical scatter of E_{OOB} over multiple ensemble runs with different seeds was typically so small that drawing error bars around the data points in the graphs would not show two visually discernible horizontal lines, which is why we omit this information in our graphs. As will be seen later, the differences in detectability between the proposed methods and prior art are so large that there should be no doubt about the statistical significance of the improvement. The code for extractors of all rich models as well as the ensemble is available at `http://dde.binghamton.edu/download`.

5. EXPERIMENTS AND COMPARISON TO PRIOR ART

This section contains the results of all experiments carried out with the costs obtained from the additive approximation of UNIWARD for all three embedding domains – spatial, JPEG, and side-informed JPEG. Spatial-domain methods are tested for relative payloads 0.05, 0.1, 0.2, . . ., 0.5 bits per pixel (bpp), while JPEG-domain and side-informed JPEG methods will be tested on the same payloads expressed in bits per non-zero cover AC DCT coefficient (bpnzAC). Even though J-UNIWARD and SI-UNIWARD embed into DC modes and zero coefficients, we express the payload in terms of bpnzAC in order to be compatible with previous art.

[4] For a fixed root rate [2], embedding in resized images is more difficult to detect than in cropped images.

5.1 Spatial domain

In the spatial domain, we compare S-UNIWARD with HUGO [28], Wavelet Obtained Weights (WOW) [14], and LSB Matching (LSBM). HUGO [28] embeds by minimizing an embedding distortion defined as a weighted norm between the features of the cover and stego image in the SPAM feature space [27]. It assigns large weights to well-populated feature bins and low weights to sparsely populated bins that correspond to more complex content. We used the HUGO embedding simulator [7] with default settings $\gamma = 1$, $\sigma = 1$, and the switch --T with $T = 255$ to remove the weakness reported in [25].

Given the similarity of distortion functions employed in WOW and S-UNIWARD (see Section 3.3.1 on comparison to prior art), one can expect a correspondingly similar performance of both algorithms, which is confirmed below. Since both algorithms are highly adaptive, they are expected to better resists steganalysis using rich models [11] than HUGO.

We report the results of all algorithms for their embedding simulators that operate at the theoretical payload–distortion bound. The non-adaptive LSBM was simulated at the ternary bound corresponding to uniform costs, $\rho_{ij} = 1$ for all i, j. The only algorithm that we implemented using STCs (with constraint height $h = 12$) to assess the coding loss is the proposed S-UNIWARD method.

Figure 1 shows the E_{OOB} error for all stego methods as a function of the relative payload expressed in bpp. As expected, the security of the S-UNIWARD and WOW is practically the same due to the similarity of their distortion functions. The improvement over HUGO is, however, quite significant especially for large payloads. As expected, the non-adaptive LSBM performs poorly across all payloads.

5.1.1 Content adaptivity

In Figure 2, we contrast the placement of embedding changes for HUGO, WOW, and S-UNIWARD. Observe that the cover image has numerous horizontal and vertical edges and also some textured areas. While HUGO embeds with high probability into the pillar edges as well as the horizontal lines above the pillars, S-UNIWARD directional costs force the changes solely into the textured areas.

While the placement of embedding changes for WOW and S-UNIWARD is quite similar, S-UNIWARD seems to be more discriminative than WOW. This higher sensitivity to content is due to the fact that it only takes one wavelet coefficient (among 3×16^2 coefficients affected by changing a single pixel $X_{ij} \to Y_{ij}$) to be close to zero to have a very large embedding cost ρ_{ij}. In contrast, in WOW, the costs are obtained by adding reciprocal values of three "embedding suitabilities," which are themselves sums over many wavelet coefficients. This makes encountering a high embedding cost less likely than in S-UNIWARD.

Upon closer inspection of the embedding probabilities for S-UNIWARD in Figure 2, one observes alternating short streaks with large differences in embedding probabilities. This is caused by the properties of the Daubechies 8-tap filter bank, which has proved to be ideal for compression of natural images due to its ability to produce many small coefficients even in textured regions. In combination with the oscillation of its high-pass component between positive and negative values, it creates the streaks as well as some small low-probability areas in textured regions. While the streaks may increase the statistical detectability, steganalysis with rich media models showed no evidence for this.

5.2 JPEG domain

To the best knowledge of the authors, currently the most secure embedding method for JPEG images that does not use any side information is the heuristic Uniform Embedding Distortion UED method [13]. It offers a substantially better empirical security than the nsF5 algorithm [12]. The authors of UED implemented their algorithm with binary codes. However, since the UED costs do not depend on the polarity of the embedding change direction, we included for comparison the UED implemented using *ternary* codes rather than binary as this is likely to produce an even more secure method.[5]

All methods were again simulated at their corresponding payload–distortion bounds. The costs for nsF5 were uniform over all non-zero DCTs with zeros assigned infinite costs (the so-called wet elements [10]). Figure 3 shows the results for JPEG quality factors 75, 85, and 95. J-UNIWARD clearly outperforms nsF5 as well as both versions of UED by a sizeable margin across all three quality factors. Furthermore, the coding loss of J-UNIWARD implemented using STCs with constraint height $h = 12$ appears rather negligible.

5.3 Side-informed JPEG domain

In the JPEG domain, by far the most successful paradigm is to minimize the distortion w.r.t. the raw, uncompressed image, if available [22, 29, 31, 15]. In this section, we compare SI-UNIWARD with three other side-informed JPEG steganographic schemes that constitute the current state of the art. The first is the Entropy Block Steganography (EBS) [31] with the cost of DCT coefficient ij corresponding to the DCT mode kl:

$$\rho_{ij}^{(kl)} = \left(\frac{q_{kl}(0.5 - |e_{ij}|)}{H(\mathbf{X}^{(b)})} \right)^2, \qquad (12)$$

where $H(\mathbf{X}^{(b)})$ is the block entropy defined as $H(\mathbf{X}^{(b)}) = -\sum_m h_m^{(b)} \log h_m^{(b)}$, where $h_m^{(b)}$ is the normalized histogram of all non-zero DCT coefficients in block $\mathbf{X}^{(b)}$. EBS embeds into all DCT coefficients, including the DC term and coefficients that would otherwise round to zero ($X_{ij} = 0$).

The second method is the already mentioned Normalized Perturbed Quantization (NPQ) [15] with embedding costs

$$\rho_{ij}^{(kl)} = \frac{q_{kl}^{\lambda_1}(1 - 2|e_{ij}|)}{(\mu + |X_{ij}|)^{\lambda_2}}, \qquad (13)$$

where, per the experiments reported in [15], we set $\mu = 0$ as NPQ embeds only in non-zero AC DCT coefficients. We also set $\lambda_1 = \lambda_2 = 1/2$ as this setting seemed to produce the most secure scheme across a wide range of payloads when tested with various feature sets.

The third algorithm is the BCHopt [29] introduced in 2009. We refer the reader to this publication for more details about its cost assignment and the actual coding.

5.3.1 Problem with zero embedding costs

We want to point out that the cost ρ_{ij} for all three prior-art methods as well as for SI-UNIWARD is equal to zero

[5] The authors of UED were apparently unaware of this possibility to further boost the security of their algorithm using ternary codes.

when the rounding error $e_{ij} = 1/2$. This, however, inevitably leads to a technical problem that, to the best knowledge of the authors, has not been disclosed elsewhere. It is connected to the fact that when $e_{ij} = 1/2$ the cost of rounding D_{ij} "down" instead of "up" should not be zero because, after all, this does constitute an embedding change. This does not affect security much when the number of such DCT coefficients is small. With an increasing number of coefficients with $e_{ij} = 1/2$ (we will call them 1/2-coefficients), however, the distortion is no longer a good measure of statistical detectability and one starts observing a rather pathological behavior – with payload approaching zero, the detection error does not saturate at 50% (random guessing) but rather at a lower value and only reaches 50% for payloads nearly equal to zero.[6] The strength with which this phenomenon manifests depends on how many 1/2-coefficients are in the image, which in turn depends on the implementation of the DCT used to compute the costs and the JPEG quality factor.

The slow DCT (implemented using 'dct2' in Matlab) typically produces a negligible number of 1/2-coefficients to cause any pathological behavior with the exception of high quality factors (see below). However, in the fast-integer implementation of DCT (e.g., Matlab's 'imwrite'), all D_{ij} are multiples of 1/8, which increases the number of 1/2-coefficients especially for high JPEG quality factors. To avoid dealing with this issue in this paper, we computed the embedding costs using the slow DCT implemented using Matlab's 'dct2' as explained in Section 2.2.

Even with the slow DCT implementation, however, the effect of 1/2-coefficients does not disappear. As can be easily verified from the formula for the DCT (1), when $k, l \in \{0, 4\}$, the value of D_{kl} is always a rational number because the cosines are either 1 or $\sqrt{2}/2$, which, together with the multiplicative weights $\mathbf{w}$, gives again a rational number. In particular, the DC coefficient (mode 00) is always a multiple of 1/4, the coefficients of modes 04 and 40 are multiples of 1/8, and the coefficients corresponding to mode 44 are multiples of 1/16. For all other combinations of $k, l \in \{0, \ldots, 7\}$, D_{ij} is an irrational number. In practice, *any* embedding whose costs are zero for 1/2-coefficients will thus strongly prefer these four DCT modes, which will cause a highly uneven distribution of embedding changes among the DCT coefficients. Because rich JPEG models [23] utilize statistics collected for each mode separately, they are capable of detecting this statistical peculiarity even at low payloads.

To demonstrate the pathological behavior of all four embedding schemes due to concentrating their embedding changes in DCT modes 00, 04, 40, and 44, we subjected all embedding methods to steganalysis using the JRM+SRMQ1 rich media model (see Section 4.2) for the JPEG quality factor 95. The results displayed in Figure 4 clearly show the saturation of the testing error at $\sim 25-30\%$ for small–medium payloads. Note that NPQ and BCHopt do not exhibit the pathological error saturation as strongly because they do not embed into the DC term (mode 00).

To eliminate this problem, we decided to modify all four side-informed JPEG embedding schemes in the following manner. We prohibit embedding changes into all 1/2-coefficients in modes 00, 04, 40, and 44.[7]

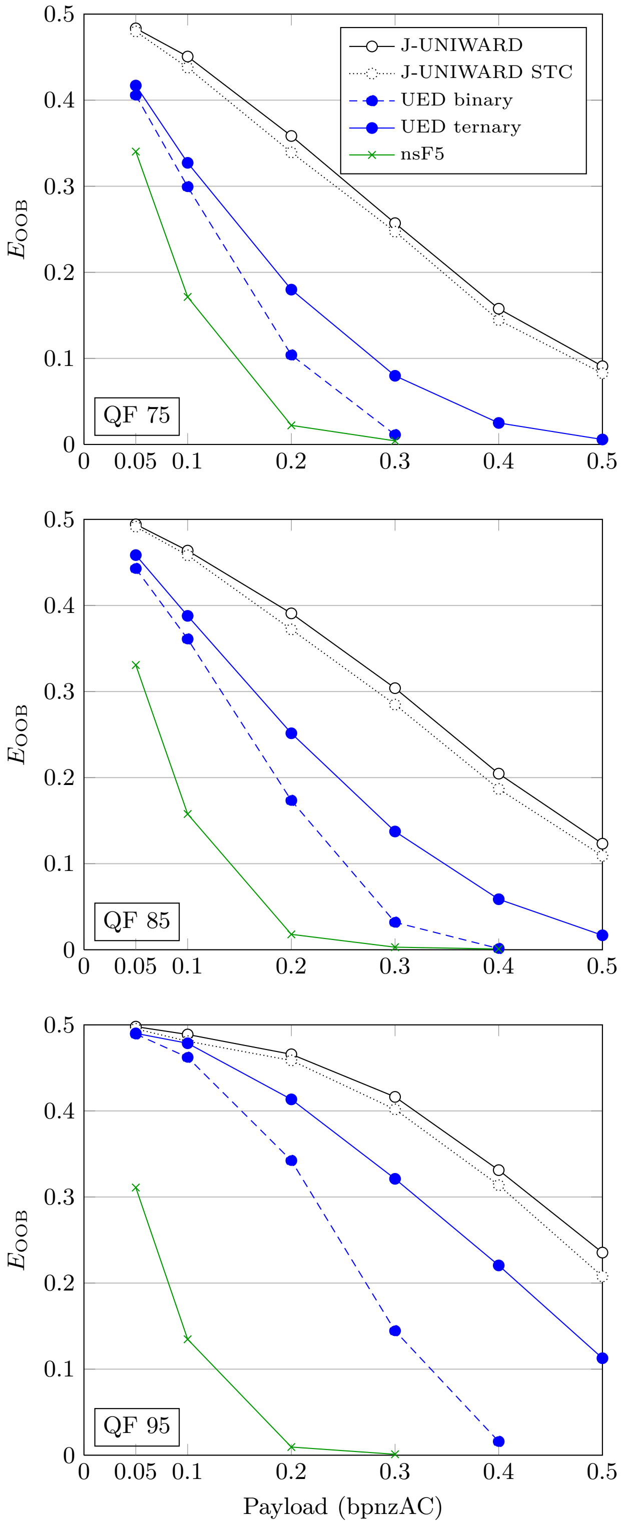


Figure 3: Testing error E_{OOB} for J-UNIWARD, nsF5, and binary (ternary) UED on BOSSbase 1.01 with the union of SRMQ1 and JRM and ensemble classifier for quality factors 75, 85, and 95.

[6] This is because the embedding strongly prefers 1/2-coefficients.

[7] In practice, we assign very large costs to such coefficients.

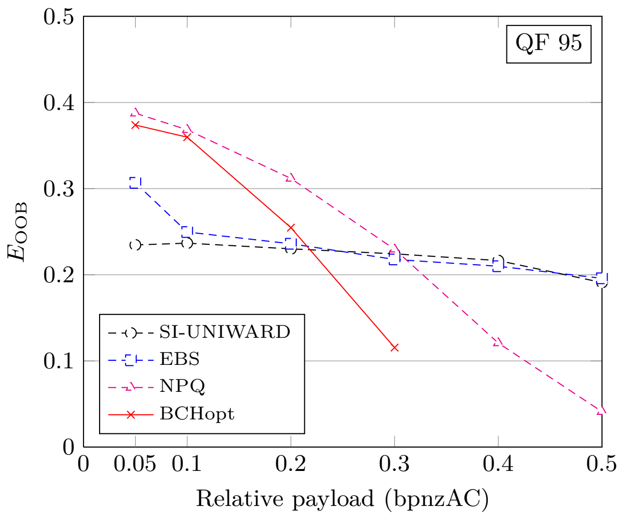


Figure 4: Pathological behavior of all four embedding schemes with zero embedding cost for 1/2-coefficients (JPEG quality factor 95). Notice that the testing error saturates for small–middle payloads due to the fact that the embedding strongly prefers DCT coefficients with zero costs, which are mostly located in DCT modes: 00, 04, 40, and 44. NPQ and BCHopt exhibit this phenomenon to a lesser degree because they avoid embedding in the DC term.

While this measure seems to have largely solved the problem (see Figure 5), we are obviously facing a much more fundamental problem, which is how exactly the side-information in the form of an uncompressed image should be utilized for the design of steganographic distortion functions. The authors postpone a detailed study of this quite intriguing problem to a separate paper.

Figure 5 shows that SI-UNIWARD achieves the best security among the tested methods for all payloads and all JPEG quality factors while its coding loss is quite small.

6. CONCLUSION

The modern paradigm for building steganographic schemes in empirical cover sources is to formulate the data hiding problem as source coding with a fidelity constraint and implement the embedding using existing codes operating near the rate–distortion bound. Technically, in imperfect steganography one minimizes the steganographic Fisher information or, equivalently, embeds as large payload at a given level of statistical detectability as possible. In practice, one first defines the fidelity measure (embedding distortion) and obtains feedback regarding the statistical detectability empirically on a given source (database of images) using a steganalyzer built using machine-learning and the best available cover models.

The main contribution of this paper is a clean, parameter-free, universal design of the distortion function called UNIWARD. What distinguishes our approach from previous art is that UNIWARD evaluates the embedding impact independently of the embedding domain. Whether one embeds

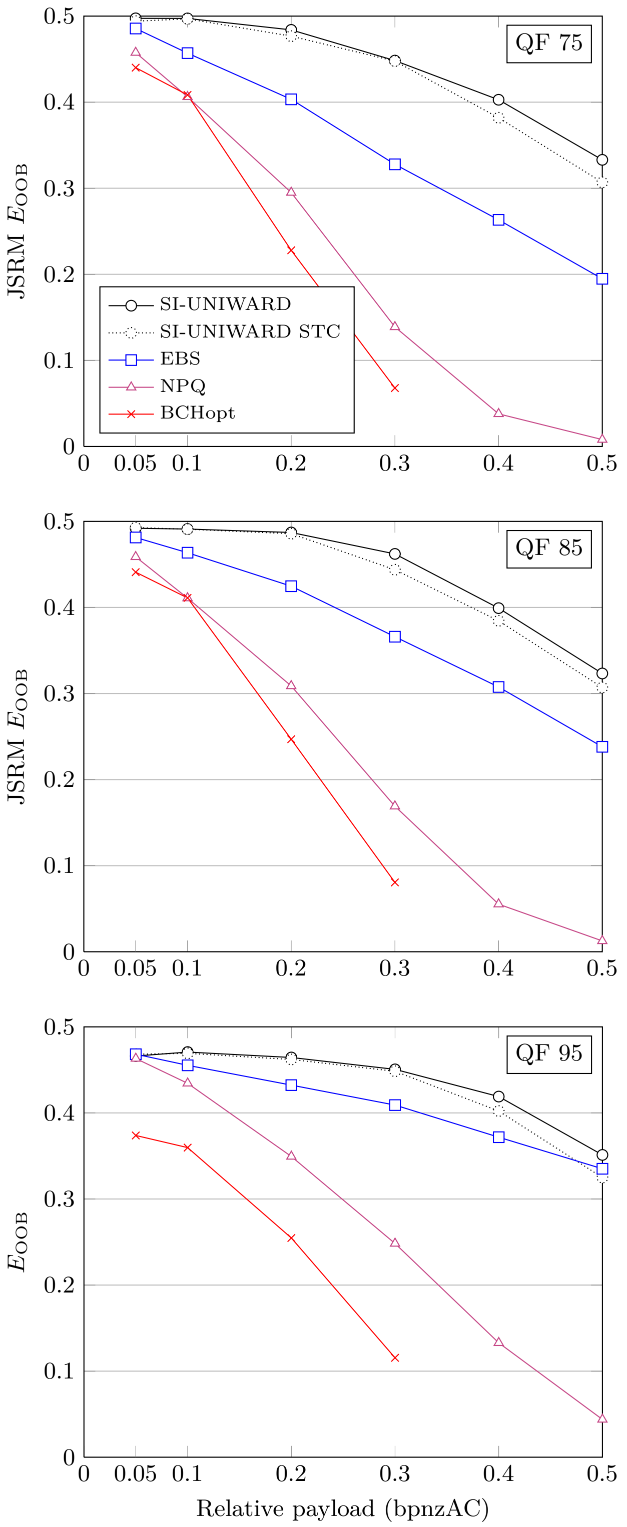


Figure 5: Testing error E_{OOB} for SI-UNIWARD and three other methods with the union of SRMQ1 and JRM and the ensemble classifier for JPEG quality factors 75, 85, and 95.

in the spatial or JPEG domain, the distortion is always computed in the wavelet domain as a sum of relative changes of wavelet coefficients in the highest frequency undecimated subbands. Since the wavelet basis functions are directional, UNIWARD can assess the neighborhood of each pixel (DCT block) for the presence of discontinuities in multiple directions and directs the embedding into the most complex textures and "noisy" regions in the cover image. In particular, UNIWARD discourages embedding in regions that can be modeled along at least one direction, such as "clean edges."

We implemented this model-free heuristic approach in the spatial, JPEG, and side-informed JPEG domains. The merit of the proposed construction is proved in this article by showing (sometimes quite significant) improvement over previous art when detecting steganography using rich media models. This applies especially to the JPEG and side-informed JPEG domains. The innovative concept to assess the costs of changing a JPEG coefficient in an alternative domain is, indeed, quite promising.

Finally, we have discovered that side-informed JPEG steganographic schemes that assign zero embedding distortion when the quantization error of DCT coefficients is 1/2 exhibit a pathological behavior that is especially striking for high quality factors and for fast integer implementation of the DCT. This is because any embedding that minimizes distortion starts introducing embedding artifacts that are quite detectable using the JPEG rich model. This finding raises an important question, which is how to best utilize the side information in the form of an uncompressed image when embedding data into the JPEG compressed form. The authors postpone detailed investigation of this open problem to their future effort.

Matlab, MEX, and C++ code for all three UNIWARD algorithms is available at `http://dde.binghamton.edu/download/stego_algorithms/`.

7. ACKNOWLEDGMENTS

The work on this paper was supported by Air Force Office of Scientific Research under the research grant number FA9950-12-1-0124. 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. The authors would like to thank Tomáš Filler and Jan Kodovský for useful discussions.

8. REFERENCES

[1] R. Böhme. *Advanced Statistical Steganalysis.* Springer-Verlag, Berlin Heidelberg, 2010.

[2] T. Filler and J. Fridrich. Fisher information determines capacity of ϵ-secure steganography. In S. Katzenbeisser and A.-R. Sadeghi, editors, *Information Hiding, 11th International Conference*, volume 5806 of *Lecture Notes in Computer Science*, pages 31–47, Darmstadt, Germany, June 7–10, 2009. Springer-Verlag, New York.

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

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

[5] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security*, 6(3):920–935, September 2011.

[6] T. Filler, A. D. Ker, and J. Fridrich. The Square Root Law of steganographic capacity for Markov covers. In N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Security and Forensics of Multimedia XI*, volume 7254, pages 08 1–11, San Jose, CA, January 18–21, 2009.

[7] T. Filler, T. Pevný, and P. Bas. BOSS (Break Our Steganography System). `http://www.agents.cz/boss`, July 2010.

[8] J. Fridrich. Effect of cover quantization on steganographic fisher information. *IEEE Transactions on Information Forensics and Security*, 2013. To appear.

[9] 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.

[10] J. Fridrich, M. Goljan, D. Soukal, and P. Lisoněk. Writing on wet paper. In T. Kalker and P. Moulin, editors, *IEEE Transactions on Signal Processing, Special Issue on Media Security*, volume 53, pages 3923–3935, October 2005. (journal version).

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] A. D. Ker. A capacity result for batch steganography. *IEEE Signal Processing Letters*, 14(8):525–528, 2007.

[17] 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.

[18] A. D. Ker. The ultimate steganalysis benchmark? In J. Dittmann and J. Fridrich, editors, *Proceedings of the 9th ACM Multimedia & Security Workshop*, pages 141–148, Dallas, TX, September 20–21, 2007.

[19] A. D. Ker. Estimating steganographic fisher information in real images. In S. Katzenbeisser and A.-R. Sadeghi, editors, *Information Hiding, 11th International Conference*, volume 5806 of *Lecture Notes in Computer Science*, pages 73–88, Darmstadt, Germany, June 7–10, 2009. Springer-Verlag, New York.

[20] A. D. Ker. The square root law in stegosystems with imperfect information. In R. Böhme and R. Safavi-Naini, editors, *Information Hiding, 12th International Conference*, volume 6387 of *Lecture Notes in Computer Science*, pages 145–160, Calgary, Canada, June 28–30, 2010. Springer-Verlag, New York.

[21] A. D. Ker, T. Pevný, J. Kodovský, and J. Fridrich. The Square Root Law of steganographic capacity. In A. D. Ker, J. Dittmann, and J. Fridrich, editors, *Proceedings of the 10th ACM Multimedia & Security Workshop*, pages 107–116, Oxford, UK, September 22–23, 2008.

[22] 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.

[23] J. Kodovský and J. Fridrich. Steganalysis of JPEG images using rich models. In A. Alattar, N. D. Memon, and E. J. Delp, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics of Multimedia XIV*, volume 8303, pages 0A 1–13, San Francisco, CA, January 23–26, 2012.

[24] J. Kodovský and J. Fridrich. Steganalysis in resized images. In *Proc. of IEEE ICASSP*, Vancouver, Canada, May 26–31, 2013. Under review.

[25] J. Kodovský, J. Fridrich, and V. Holub. On dangers of overtraining steganography to incomplete cover model. In J. Dittmann, S. Craver, and C. Heitzenrater, editors, *Proceedings of the 13th ACM Multimedia & Security Workshop*, pages 69–76, Niagara Falls, NY, September 29–30, 2011.

[26] 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, 2012.

[27] 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.

[28] 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.

[29] 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.

[30] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec.*, 4:142–163, 1959.

[31] 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.

[32] G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods: A Mathematical Introduction (Stochastic Modelling and Applied Probability)*. Springer-Verlag, Berlin Heidelberg, 2nd edition, 2003.