Improving Steganographic Security by Synchronizing the Selection Channel

Tomáš Denemark
Binghamton University
Department of ECE
Binghamton, NY 13902-6000
tdenema1@binghamton.edu

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

ABSTRACT

This paper describes a general method for increasing the security of additive steganographic schemes for digital images represented in the spatial domain. Additive embedding schemes first assign costs to individual pixels and then embed the desired payload by minimizing the sum of costs of all changed pixels. The proposed framework can be applied to any such scheme – it starts with the cost assignment and forms a non-additive distortion function that forces adjacent embedding changes to synchronize. Since the distortion function is purposely designed as a sum of locally supported potentials, one can use the Gibbs construction to realize the embedding in practice. The beneficial impact of synchronizing the embedding changes is linked to the fact that modern steganalysis detectors use higher-order statistics of noise residuals obtained by filters with sign-changing kernels and to the fundamental difficulty of accurately estimating the selection channel of a non-additive embedding scheme implemented with several Gibbs sweeps. Both decrease the accuracy of detectors built using rich media models, including their selection-channel-aware versions.

Categories and Subject Descriptors

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

General Terms

Security, Algorithms, Theory

Keywords

Steganography, Gibbs construction, non-additive distortion, selection channel, synchronization, security

1. MOTIVATION

The prevalent paradigm for designing new steganographic schemes for digital images is based on the concept of minimizing an additive distortion function defined as the sum of costs of all changed pixels. This design has proved extremely successful in both the spatial and JPEG domain [16, 9, 7, 11, 15, 14]. Additive distortion functions, however, cannot capture the fact that executing the embedding changes in a group of adjacent pixels will likely have a smaller statistical impact than changing the same number of isolated pixels. Moreover, spatially synchronized adjacent embedding changes will also be less detectable than independent changes. Both claims can be understood on an intuitive level when one takes into account how steganography is being detected – with various statistical descriptors of noise residuals. Two adjacent pixels will disturb fewer residual values than two spatially separated pixels. Also, the detectability of changing an entire connected patch of pixels by $+1$ should depend only on the length of its boundary. In the extreme case, changing all pixels by $+1$ should not be detectable at least when ignoring the effects of pixels' limited dynamic range.

While the design of additive distortion functions is a well-researched subject, non-additive distortion is much less understood. Surprisingly, the first content-adaptive scheme, HUGO [16], already employed a non-additive element. First, a binary Syndrome-Trellis Code [3] (STC) was used to determine which least significant bits were to be changed. Then, the embedding proceeded in a pixel-by-pixel fashion, each time recomputing the cost of the pixel when changing it by $+1$ or -1 (based on adjacent and potentially already modified pixels) and selecting the option with the smaller cost. This approach, however, gave HUGO only a rather limited ability to consider adjacent embedding changes. A better founded version of this embedding algorithm called HUGO-BD (Bounding Distortion) starts with a distortion between the cover and stego image in the form of a difference between features in a selected feature space. As this distortion is not only non-additive but most importantly non-local, it is upper-bounded by another function that can be written as a sum of locally supported potentials and one that can be implemented using the Gibbs construction [2]. HUGO-BD's empirical security, however, is subpar when compared to current state-of-the-art additive schemes, such as S-UNIWARD [11], HILL [15], and the approach based on minimizing the detectability of an optimal detector within a chosen cover model [17].

The algorithm called S-UNIWARD uses an additive approximation of a distortion function that is natively non-additive. Because it is a sum of locally supported potentials, the Gibbs construction can be used for practical embedding. However, as the recent study reports [8], the Gibbs construction when applied to the distortion of S-UNIWARD does not

IH&MMSec'15, June 17–19, 2015, Portland, Oregon.

provide satisfactory performance in practice. The author attributed this to the suboptimality of the Gibbs construction when the individual sublattices are strongly dependent. The problem of designing non-additive distortion functions for steganography is generally not well understood because it is not clear how the interactions among neighboring embedding changes affect security and how to capture this using a distortion measure.

The approach taken in this paper starts with a cost assignment of an additive scheme and builds from it a simple non-additive distortion function purposely designed to discourage adjacent desynchronized embedding changes and to satisfy certain natural a priori requirements. Because the distortion is a sum of potential functions on two-pixel cliques, only two interleaved sublattices are needed for practical embedding using STCs in a Gibbs-like manner. Since the empirical detectability does not change with an increased number of sweeps, the proposed scheme utilizes merely a single Gibbs sweep. We prove the usefulness of the framework by applying it to the costs of the (ternary) scheme proposed in [17] with the multivariate Gaussian cover model (MVG) and HILL [15]. In both cases, the empirical security is markedly improved when testing with the spatial rich model [5] (SRM) as well as its selection-channel aware version [1] (maxSRMd2).

The entire framework is described in Section 2. In Section 3, we analyze the properties of the embedding and study the influence of the single parameter that controls the severity of penalizing desynchronized adjacent embedding changes. On a test image, we further study the properties of the selection channel with increased number of Gibbs sweeps. All experimental results on the BOSSbase 1.01 source appear in Section 4. In Section 5, we provide further arguments supporting our design and investigate possible avenues for attacks. The paper is closed in Section 6, where we summarize the contribution and list future research directions.

2. STEGANOGRAPHY WITH SYNCHRONIZED EMBEDDING CHANGES

This part of the paper introduces a general procedure how to form a non-additive distortion function from an additive scheme and then describes how to use it for steganography within a Gibbs-like construction.

Let $\mathbf{x}$ be an $M \times N$ grayscale cover image with pixel values $x_{ij} \in \{0, \dots, 255\}$, $1 \le i \le M$, $1 \le j \le N$, and $\mathcal{A}$ be an additive steganographic scheme that modifies each pixel by at most ± 1. We will further assume that $\mathcal{A}$ is such that the cost of changing each cover pixel x_{ij} to $y_{ij} = x_{ij} + 1$ or $y_{ij} = x_{ij} - 1$ is the same and equal to ρ_{ij}.[1] Note that for each pixel, $x_{ij} - y_{ij} \in \{-1, 0, 1\}$. Let us further denote with $\mathcal{C}$ the index set of all two-pixel cliques formed by two vertically or horizontally adjacent pixels. For example, given a pixel index (i, j), there are four cliques that contain this pixel: $((i,j),(i+1,j))$, $((i,j),(i-1,j))$, $((i,j),(i,j+1))$, and $((i,j),(i,j-1))$. The non-additive distortion function has the following form:

$$D(\mathbf{x}, \mathbf{y}) = \sum_{((i,j),(k,l)) \in \mathcal{C}} S_{\mathcal{C}}(x_{ij} - y_{ij}, x_{kl} - y_{kl}), \qquad (1)$$

[1]Virtually all state-of-the-art additive steganographic techniques have this cost symmetry to be able to utilize the more powerful ternary STCs.

where $S_{\mathcal{C}}(a, b)$, $-1 \le a, b \le 1$,

$$S_{\mathcal{C}} = \begin{array}{c|ccc} & -1 & 0 & 1 \\ \hline -1 & 0 & A_{\mathcal{C}} & \nu A_{\mathcal{C}} \\ 0 & A_{\mathcal{C}} & 0 & A_{\mathcal{C}} \\ 1 & \nu A_{\mathcal{C}} & A_{\mathcal{C}} & 0 \end{array} \qquad (2)$$

is a 3×3 array that depends on the average clique cost $A_{\mathcal{C}} = (\rho_{ij} + \rho_{kl})/2$, and $\nu \ge 0$ is a parameter controlling the strength of penalizing *desynchronized* changes. Alternatively and equivalently, one can define the array as

$$S_{\mathcal{C}}(a, b) = \begin{cases} 0 & \text{when } a = b \\ A_{\mathcal{C}} & \text{when } |a| + |b| = 1 \\ \nu A_{\mathcal{C}} & \text{when } a \neq b \text{ and } |a| + |b| = 2. \end{cases} \qquad (3)$$

The distortion function (1) has the form of a sum of locally supported potentials and the embedding can be implemented using a Gibbs-like construction on two interleaved sublattices

$$\mathcal{L}_1 = \{(i,j) | \bmod (i + j, 2) = 0\}, \qquad (4)$$

$$\mathcal{L}_2 = \{(i,j) | \bmod (i + j, 2) = 1\}. \qquad (5)$$

The support of each potential is the two-pixel clique. The zeros on the diagonal of $S_{\mathcal{C}}$ enforce the requirement we formulated in the introduction – changing all pixels by $+1$ (or -1) should not have any effect on detectability. Furthermore, when modifying a connected patch of pixels all by the same amount, only the boundary pixels will intuitively contribute to the distortion.

The parameter ν has a major effect on the properties of the selection channel and needs to be suitably chosen. From our experiments with rich image models (SRM and maxSRMd2), the empirical security is not very sensitive to ν as long as it stays within a certain range (see Figure 3 in Section 3).

The entire embedding algorithm is described using Algorithm 1. The inputs are the cover image $\mathbf{x}$, the cost assignment of the additive scheme $\mathcal{A}$ via an $M \times N$ array ρ_{ij}, and the payload $\mathbf{m}$, while its output is the stego image $\mathbf{y}$. In the pseudo-code, we used the following notation. The additive approximation of the distortion function (1) is defined as

$$D_A(\mathbf{x}, \mathbf{y}) = \sum_{x_{ij} \neq y_{ij}} D(\mathbf{x}, y_{ij}\mathbf{x}_{\sim ij}), \qquad (6)$$

where $D(\mathbf{x}, y_{ij}\mathbf{x}_{\sim ij})$ is the distortion between image $\mathbf{x}$ and $y_{ij}\mathbf{x}_{\sim ij}$, which is a shorthand for $\mathbf{x}$ in which only the (i,j)th

Algorithm 1 Pseudo-code for the embedding algorithm. The initial image can be selected as the cover, $\mathbf{x}$, or the stego image $\mathbf{y}$ embedded with the additive scheme $\mathcal{A}$.

Divide message into two equal size parts $\mathbf{m} = \mathbf{m}_1 \cup \mathbf{m}_2$
Compute the costs ρ_{ij} from the cover image $\mathbf{x}$
Set $\mathbf{y}$ = Initial image
for $k = 1$ to Number of sweeps **do**
  for $l = 1$ to 2 **do**
    Execute for all $(i,j) \in \mathcal{L}_l$
    $\rho^{(+)}_{ij} = D_A(\mathbf{y}, x_{ij} + 1\ \mathbf{y}_{\sim ij})$
    $\rho^{(0)}_{ij} = D_A(\mathbf{y}, x_{ij}\ \mathbf{y}_{\sim ij})$
    $\rho^{(-)}_{ij} = D_A(\mathbf{y}, x_{ij} - 1\ \mathbf{y}_{\sim ij})$
    $\mathbf{y}_{\mathcal{L}_l} = \mathrm{STC}(\mathbf{y}_{\mathcal{L}_l}, \rho^{(+)}, \rho^{(0)}, \rho^{(-)}, \mathbf{m}_l)$
  end for$\{l\}$
end for$\{k\}$

pixel x_{ij} was changed to y_{ij}. The symbol STC(.) stands for the actual embedding using STCs with the specified costs of changes by $+1$, 0, and -1.

In contrast to additive schemes where the cost of *not* making a change is always zero, in a non-additive scheme it may not be so because of the influence of surrounding pixels. A positive cost of no change will increase the payload (entropy) that one can embed at a given pixel but will also increase the number of embedding changes. Whether the increased payload outweighs the increased change rate depends on how well the non-additive distortion captures statistical detectability.

Embedding with different costs of all three possibilities ($+1$, 0, -1) requires the use of the so-called multi-layered STCs [3]. We would also like to stress that the costs A_c are computed only once before the embedding starts and are kept the same throughout the embedding, i.e., they are not recomputed after every sweep. Finally, we note that the recipient reads the secret message using the same STCs applied to each sublattice and concatenating both parts. Also, for security as well as efficiency of the STCs, before applying the code for embedding or reading, the sublattice elements should be rearranged by a permutation that depends on the stego key.

When starting with the costs from an additive embedding scheme $\mathcal{A}$, we will call the embedding algorithm with the synchronized selection channel as Synch-$\mathcal{A}$.

3. SELECTION CHANNEL PROPERTIES

In this section we study the effect of the parameter ν on the selection channel and the overall performance of the proposed scheme, and then discuss some issues related to the Gibbs construction.

Figure 1 illustrates how the value of the parameter ν controls the strength of the separation between neighboring clusters with synchronized changes. The experiment was set up to amplify the effect of separation for easy viewing. The viewer is advised to magnify the figure in the PDF viewer to better see the properties of the selection channel. For larger values of ν the embedding forces areas with changes by $+1$ and by -1 to be separated by a small area with no changes. As we do not use any model of the cover source, the parameter ν has to be set experimentally. Figure 3 shows the detection error on the test set when steganalyzing Synch-HILL (and MVG) with the maxSRMd2 feature set and the FLD ensemble as a function of ν for two different relative payloads. For both embedding schemes and both payloads, the optimal value is near $\nu = 5$. Therefore, we will use this value in the rest of this paper.

The distortion function of Synch-$\mathcal{A}$ is fully defined after selecting the additive scheme $\mathcal{A}$ and the parameter ν. To embed a message in a given cover image while introducing minimal total distortion, one can use the Gibbs construction as introduced in [2]. For the payload limited sender, the task is to find a probability distribution over stego images $\pi(\mathbf{y})$ that carries the required payload expressed by the entropy $H(\pi)$ and has the minimal expected distortion $E_\pi[D(\mathbf{x}, \mathbf{y})]$. The optimal distribution has the Gibbs form $\pi_\lambda(\mathbf{y}) \propto \exp(-\lambda D(\mathbf{x}, \mathbf{y}))$, where λ is a scalar parameter determined from the payload constraint. For any given $\lambda \geq 0$, one can use the Gibbs sampler [20] to obtain a stego image $\mathbf{y}$ drawn with the correct probability $\pi(\mathbf{y})$. In practice, however, the Gibbs sampler cannot be used directly since we need to communicate a specific message and we do not know the value of λ. The Gibbs construction can be thought of as an approximation of the Gibbs sampler that allows embedding the secret message.

Figure 4 (left) shows how the distortion (1) and the change rate evolve with consequent iterations (sweeps) of the Gibbs

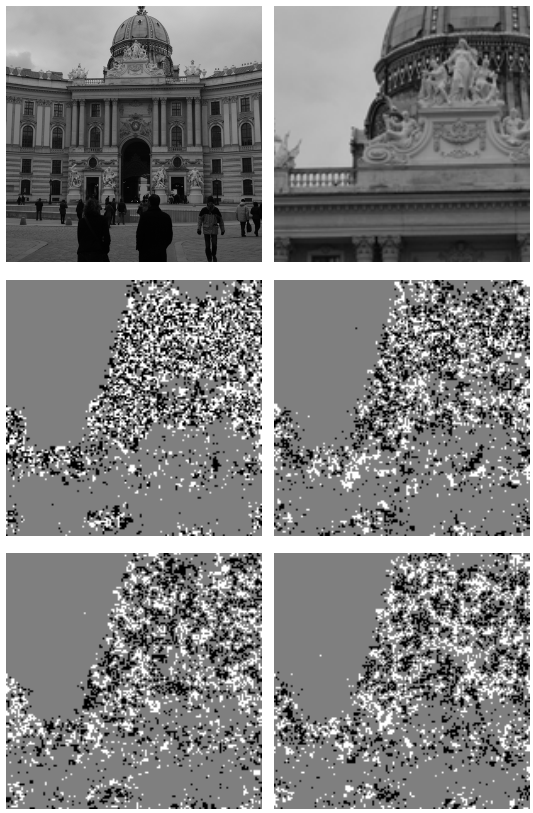

Figure 1: Actual embedding changes executed by Synch-HILL at 0.4 bpp after 10 sweeps of the Gibbs construction for a crop of BOSSbase image '1013.pgm'. White corresponds to changes by +1, black to -1, and medium gray is used for pixels that did not change. Top-left: original image, top-right: crop, middle-left: $\nu = 2$, middle-right: $\nu = 10$, bottom-left: $\nu = 100$, bottom-right: $\nu = 1000$.

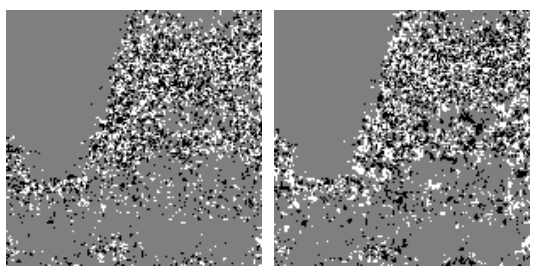

Figure 2: Actual embedding changes executed by Synch-HILL at 0.4 bpp for a crop of BOSSbase image '1013.pgm', $\nu = 5$. White corresponds to changes by +1, black to -1, and medium gray is used for pixels that did not change. Left: after 1 sweep, right: after 10 sweeps.

construction. The initial image was a stego image embedded with the additive scheme $\mathcal{A}$. During the first sweep, the distortion can dramatically decrease if $\mathcal{A}$ changes pixels adjacent to pixels with wet costs. After the first sweep, the distortion saturates (there is a small increase with sweeps) because the Gibbs construction does not exactly execute the Gibbs sampler, at least not for a small number of sweeps. The effect of the sweeps on the selection channel is shown in Figure 2.

There exists a transitional period when the Gibbs construction treats each sublattice slightly differently, which leads to different values of the parameter λ in each sublattice (see Figure 5). Another deviation of the Gibbs construction from the Gibbs sampler is that, asymptotically, it embeds the payload corresponding to the erasure entropy $H^{-}(\pi)$ [19] but introduces a larger distortion that corresponds to the larger entropy $H(\pi)$ [2]. This difference increases with stronger dependencies between the two sublattices, which makes the Gibbs construction suboptimal. Despite these limitations, the "Synched" algorithms still perform better than their additive versions.

Since in our experiments, we saw no advantage (or harm) when using more then one sweep, in Algorithm 1, we fixed the number of embedding sweeps to 1 when initializing the image with the stego image obtained by embedding the required payload with the additive scheme $\mathcal{A}$.

4. EXPERIMENTS AND COMPARISON TO PRIOR ART

In this section, we first describe the common core of all experiments. Then, we apply the proposed framework to two additive steganographic schemes that appear as the current state of the art, and we subject the proposed steganography to tests on a standard image database using two types of rich media models.

4.1 Cover source

All experiments were conducted on the BOSSbase database ver. 1.01 [4] containing 10,000 512×512 8-bit grayscale images coming from eight different cameras. The steganographic security was evaluated empirically using binary classifiers trained on the 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 customarily done in academic investigations of this type.

4.2 Features and machine learning

All steganographic methods will be analyzed using what became a standard today, the Spatial Rich Model [5] consisting of 39 symmetrized sub-models quantized with three different quantization factors with a total dimension of $34,671$. We will also use the maxSRMd2 model [1], which has the same dimension and is a selection-channel aware version of the SRM and a generalization of the tSRM introduced in [18]. The maxSRMd2 uses the approximate knowledge of the embedding change probabilities extracted from the stego image. We conservatively assume the worst case scenario in which the Warden knows the payload size but, of course, not the cover image. Moreover, as the original study reports [1], the decrease of detection power of the maxSRMd2 when steganalyzing with a mismatched payload is rather small.

All classifiers were implemented using the ensemble [13] with Fisher linear discriminant as the base learner. The security is quantified using the ensemble's minimal total testing error under equal priors,

$$P_{\mathrm{E}} = \min_{P_{\mathrm{FA}}} \frac{1}{2}(P_{\mathrm{FA}} + P_{\mathrm{MD}}), \tag{7}$$

when training on one randomly chosen half of the database and testing on the remaining half. Repeating ten times, we use the average of these ten testing errors, $\overline{P}_{\mathrm{E}}$, to quantify the algorithm's security.

To show how the statistical detectability increases with payload, we produce graphs showing $\overline{P}_{\mathrm{E}}$ as a function of the relative payload. With the feature dimensionality and the database size, the statistical scatter of $\overline{P}_{\mathrm{E}}$ 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`.

4.3 Tested steganographic schemes

We implemented the proposed Synch scheme for two adaptive steganographic algorithms that appear the current state of the art as of writing this paper (January 2015) and that work in entirely different fashion. They are the HIgh Low Low (HILL) algorithm [15] and the ternary version of the MVG [17], which is an abbreviation for an embedding scheme designed to minimize the power of optimal detector within the multivariate Gaussian cover model. HILL is a modification of the WOW algorithm [9] in which the three Daubechies directional kernels were replaced by one KB (Ker–Böhme) kernel [12] (this is the high-pass part of the algorithm). The KB residual is further low-pass filtered with a 3×3 averaging filter and used in the same manner as in WOW to compute the pixel costs. The resulting costs are again low-pass filtered with a quite large 15×15 kernel. The benefit of low-pass filtering the costs has also been demonstrated in [14]. A short explanation of why low-pass filtering the costs improves empirical security is because the costs are made more uniform, which increases the entropy of embedding changes in highly textured regions, which allows decreasing the distortion for the same payload. Additionally, the averaging spills large costs into the neighboring pixels, which makes the algorithm more conservative.

The MVG algorithm works in an entirely different manner. First, the cover is modeled as a sequence of independent but not identically distributed Gaussian random variables. The parameters of this model (the local pixel variances) are then estimated and the optimal embedding change rates are derived from the principle of minimizing the statistical detectability expressed in the form of the Kullback–Leibler divergence between the MVG cover distribution and the MVG stego mixture. Since the KL divergence can be analytically expressed using the estimated cover variances and the change rates, one can derive the change rates using the method of Lagrange multipliers. The main difference between the prior art [6] and the MVG algorithm as implemented in the current paper is the variance

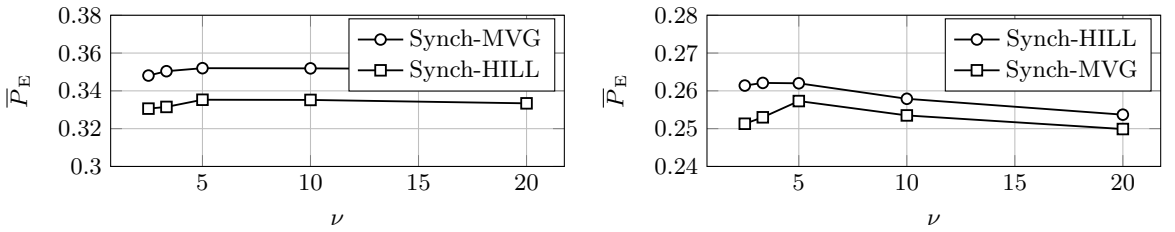

Figure 3: Search for the optimal value of ν with maxSRMd2 at 0.2 bpp (left) and 0.4 bpp (right).

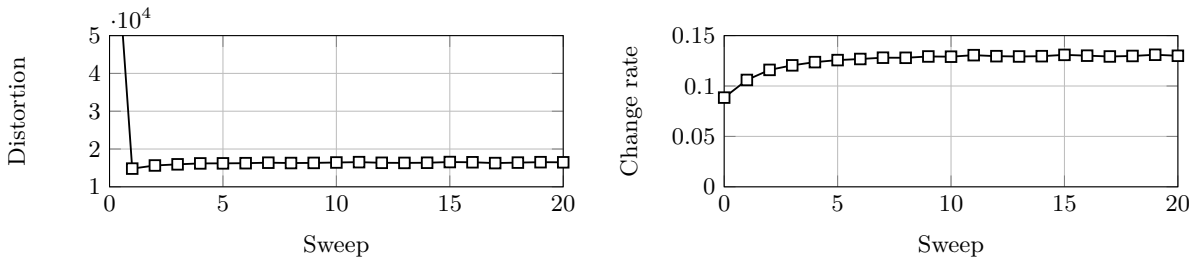

Figure 4: Distortion (1) **(left) and the embedding change rate (right) as a function of sweeps for BOSSbase image 1013.pgm when embedding with Synch-HILL at 0.4 bpp, $\nu = 5$.**

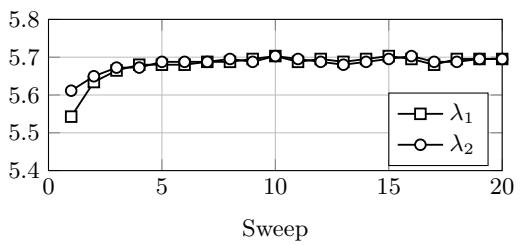

Figure 5: The parameters λ_1 and λ_2 in sublattices $\mathcal{L}_1$ and $\mathcal{L}_2$ as a function of the number of embedding sweeps when embedding with Synch-HILL at 0.4 bpp in BOSSbase image '1013.pgm', $\nu = 5$.

estimator. Instead of a very simple estimator used in [6], we estimate the local pixel variance as in [17]. First, the Wiener 2×2 denoising filter is used to extract a noise residual, which is subsequently locally fitted with DCT bases to reject more of the content. The variance estimator is described in detail in Section 5 in [17] available from `http://ws2.binghamton.edu/fridrich/publications.html`. Finally, the Fisher information estimated for each pixel is smoothed with a 7×7 averaging kernel. The reason for the smoothing is also explained in [17].

Before describing the experimental results, we elaborate on the important issue of how to attack non-additive steganographic schemes when using the selection-channel-aware maxSRMd2 features. These features require an estimation of the actual embedding change probabilities, which, however, strongly depend on the actual embedding changes in the pixel neighborhood and thus vary across the sweeps as well as different messages and steganographic keys. We study this in great detail in the next section, where we explain that the best the Warden can do is to use the embedding change probabilities computed from the additive scheme $\mathcal{A}$. For this, we grant the Warden the knowledge of the payload.

Figure 6 and Table 1 show that both algorithms, Synch-HILL and Synch-MVG achieve approximately the same level of security. Synch-HILL appears to be slightly more secure when steganalyzing with the SRM, while MVG is slightly more secure when the selection-channel-aware model, maxSRMd2, is used. The detection error of methods employing a synchronized selection channel is higher by approximately 2% (for payload 0.1 bpp) and 5% for the largest tested payload of 0.5 bpp.

The last two rows of the Table 1 show the detection errors when computing the maxSRMd2 features with the actual embedding change probabilities used during embedding within the Gibbs construction. Our intention is to show the lower bound on the security in the absolutely worst possible case for the sender. We stress that this case is completely unrealistic because in order to obtain these probabilities, the Warden would need to know the actual embedding changes in the first sublatice. This means that she would need to know the the corresponding portion of the secret message, the embedding costs obtained from the cover, and the permutation of the sublattice (the stego key).

5. FURTHER ANALYSIS

To better explain the increase in security, in this section we include further study of the impact of synchronizing the embedding changes on the distribution of noise residuals. Furthermore, we also study how feasible it is for the Warden to estimate the embedding change probabilities (and attack the proposed scheme using this knowledge of the selection channel) and explain that a good choice for the Warden is to use the probabilities of the additive embedding scheme $\mathcal{A}$.

5.1 Sign-changing kernels

The synchronizing of embedding changes in the proposed Synch-$\mathcal{A}$ algorithm is rather subtle, especially when embedding with a single sweep (see Figure 2). In this section, we

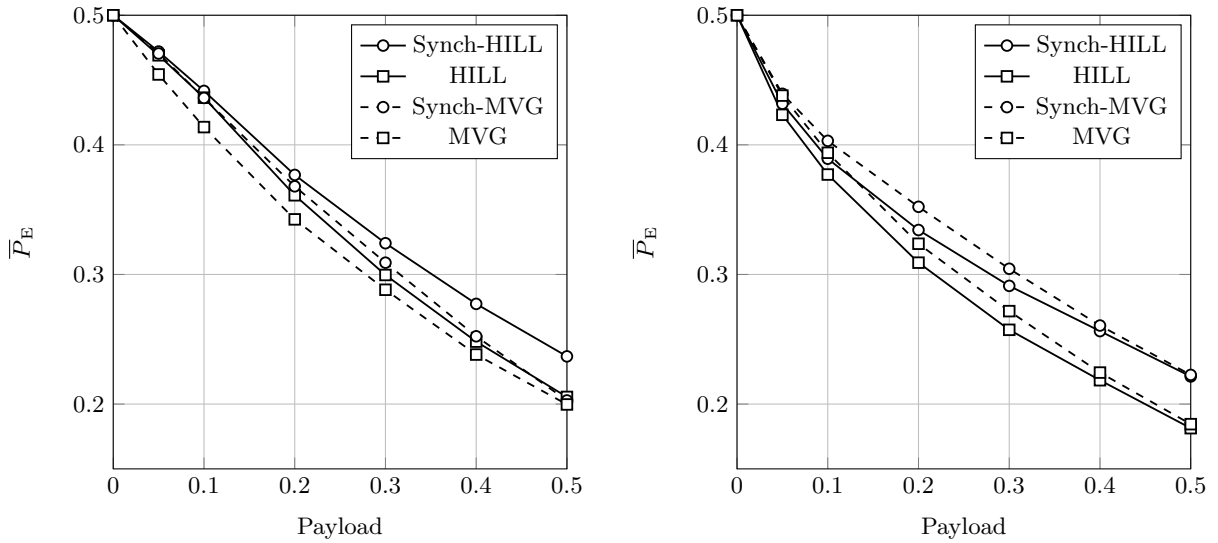

Figure 6: Average testing error $\overline{P}_{\mathrm{E}}$ versus payload for HILL and MVG and their synchronized version when steganalyzing with SRM (left) and maxSRMd2 (right).

Feature set	SRM						maxSRMd2					
Payload (bpp)	0.05	0.1	0.2	0.3	0.4	0.5	0.05	0.1	0.2	0.3	0.4	0.5
Synch-HILL	0.4720	0.4416	0.3768	0.3241	0.2773	0.2368	0.4317	0.3893	0.3343	0.2912	0.2563	0.2213
HILL	0.4691	0.4364	0.3611	0.2996	0.2482	0.2055	0.4232	0.3771	0.3091	0.2573	0.2184	0.1814
Synch-MVG	0.4705	0.4362	0.3680	0.3091	0.2523	0.2028	0.4394	0.4031	0.3522	0.3044	0.2606	0.2225
MVG	0.4543	0.4137	0.3425	0.2882	0.2382	0.1997	0.4380	0.3939	0.3237	0.2717	0.2243	0.1845
*Synch-HILL	x	x	x	x	x	x	0.4056	0.3640	0.3008	0.2553	0.2203	0.1855
*Synch-MVG	x	x	x	x	x	x	0.4189	0.3725	0.3289	0.2735	0.2236	0.1868

Table 1: Numerical values of testing error $\overline{P}_{\mathrm{E}}$ from Figure 6. The last two rows contain errors for the ideal case when the actual embedding probabilities are known to the Warden.

provide additional insight into why even subtle synchronization decreases the detectability using current steganalysis features.

Virtually all features for spatial domain steganalysis are constructed as higher-order statistical descriptors (either co-occurrences or histograms of projections in projection-type versions [10]) of noise residuals designed from local polynomial models of content. As such, all residuals used in the SRM model (and all other features based on SRM residuals) use kernels that *change signs*. For example, the first-, second-, and third-order residuals use kernels $[-1\ 1]$, $[1/2\ -1\ 1/2]$, and $[-1/3\ 1\ -1\ 1/3]$, respectively. This is true also for larger kernels, such as the 3×3 KB kernel and the 5×5 kernel used in the SRM for all 3×3 and 5×5 EDGE submodels as well as the SQUARE submodel [5]. Thus, a pair of adjacent synchronized embedding changes will disturb the residual less than a pair of desynchronized changes. This effect can be easily quantified.

Figure 7 (top) shows the relative change in the sample variance, v_i, $i=1,\ldots,100$, of the KB residual computed from 100 stego images randomly selected from BOSSbase and embedded with HILL and with Synch-HILL with one sweep with relative payload 0.4 bpp:

$$r_i = 100\times\frac{v_i^{(\mathrm{HILL})}-v_i^{(\mathrm{SYNCH})}}{v_i^{(\mathrm{HILL})}}. \tag{8}$$

The graph confirms that the variance of the KB residual of stego images produced with Synch-HILL is mostly lower than for HILL stego images. To further strengthen our interpretation of the results, we compute this ratio from the same 100 images but this time with a high-pass 3×3 kernel, which we denote KB^{++}, whose elements change the sign less frequently:

$$\mathrm{KB}^{++}=\begin{pmatrix}0.25 & -0.2 & 0.25\\ -0.2 & -0.2 & -0.2\\ 0.25 & -0.2 & 0.25\end{pmatrix}. \tag{9}$$

The result is shown in the bottom graph of Figure 7. Note that the change between the KB^{++} residual variances between the stego images of both schemes is now much smaller and one can even observe a very small bias towards larger variance for Synch-HILL. The detection performance of the KB^{++} kernel is, however, much lower than that of the KB kernel. Moreover, the KB^{++} kernel still detects HILL slightly better than Synch-HILL (Table 2).

A natural choice for high-pass filters that do not frequently change signs is the discrete cosine basis. Table 3 shows the ratio r_i when all 64 DCT bases are used for computing the residuals. As expected, the higher the spatial frequency of the DCT mode is, the more it changes the sign, and the higher the variance of HILL stego images becomes when compared to Synch-HILL images. The DCT modes (k,l),

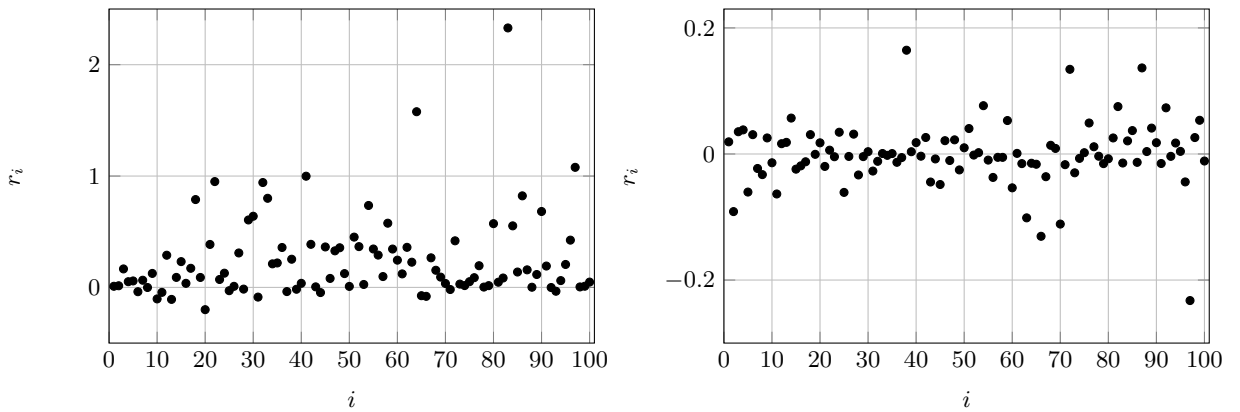

Figure 7: Relative change in the variance of the KB (left) and KB^{++} (right) residual across 100 randomly selected images from BOSSbase 1.01.

k/l	0	1	2	3	4	5	6	7
0	−0.0018	−0.0067	−0.0076	−0.0105	−0.0115	−0.0119	−0.0119	−0.0190
1	−0.0028	−0.0170	−0.0410	−0.0631	−0.0589	−0.0623	−0.0369	0.0015
2	−0.0064	−0.0313	−0.0739	−0.0926	−0.0815	−0.0779	−0.0176	0.1384
3	−0.0061	−0.0529	−0.1180	−0.1181	−0.0821	−0.0399	0.0599	0.2787
4	−0.0062	−0.0602	−0.1232	−0.0952	−0.0346	0.0497	0.1968	0.4314
5	−0.0044	−0.0470	−0.0747	−0.0065	0.0578	0.1660	0.3549	0.6533
6	0.0018	−0.0385	−0.0135	0.0866	0.2342	0.4214	0.6264	1.0071
7	0.0126	0.0086	0.0834	0.2008	0.4227	0.6242	0.8601	1.2453

Table 3: The ratio r_i averaged over all 100 images for DCT modes (k, l). Note the most promising DCT modes for attacking Synch-HILL are $(4, 2)$, $(3, 2)$, and $(3, 2)$. Predictably, with higher spatial frequency, the ratio becomes again positive – the kernels change the sign too frequently.

Stego algorithm	$\overline{P}_{\mathrm{E}}$
HILL	0.4410±0.0036
Synch-HILL	0.4523±0.0033

Table 2: Average testing detection error for the 6,084 DCT features on BOSSbase 1.01. The DCT features perform very poorly and still detect HILL better than Synch-HILL.

$0 \le k, l \le 7$, that exhibit a higher variance in Synch-HILL images are approximately those that satisfy $k + l \le 7$. To investigate their power for detecting Synch-HILL, we selected all such 36 DCT kernels, normalized each to an L_2 norm 1, computed from them 36 noise residuals, and finally formed 36 co-occurrence matrices after quantizing them with quantization step $q = 1$ and truncating with $T = 2$ as in the SRM residuals. The resulting feature vector of dimension $36 \times 169 = 6084$ was used to steganalyze both HILL and Synch-HILL. The detection error for Synch-HILL is, however, still higher than for HILL. Moreover, the overall detection using these features is very poor considering its dimensionality. Although these experiments point out a possible attack on Synch-$\mathcal{A}$ schemes, it appears unlikely that a reliable detection can be obtained by enriching the existing rich models by co-occurrences computed from residuals obtained using smoother kernels.

5.2 Estimating embedding change probabilities

Recently, steganalysis features built as co-occurrences of noise residuals have been made more powerful for detection of content-adaptive steganography by incorporating estimated embedding change probabilities (the Bayesian priors) into the feature construction [18, 1]. In particular, when building, e.g., the horizontal co-occurrence in the maxSRMd2 model [1] from a quantized and truncated noise residual z_{ij}, instead of adding 1 to the co-occurrence bin $(z_{ij}, z_{i,j+1}, z_{i+1,j+2}, z_{i+1,j+3})$, one adds to the bin the value of $\max\{p_{ij}, p_{i,j+1}, p_{i+1,j+2}, p_{i+1,j+3}\}$, where $p_{ij} = p_{ij}^{(+)} + p_{ij}^{(-)}$ is the probability of modifying pixel x_{ij} by either $+1$ or -1:

$$p_{ij} = \frac{\exp(-\lambda\rho_{ij}^{(+)}) + \exp(-\lambda\rho_{ij}^{(-)})}{\exp(-\lambda\rho_{ij}^{(0)}) + \exp(-\lambda\rho_{ij}^{(+)}) + \exp(-\lambda\rho_{ij}^{(-)})}. \quad (10)$$

While it is possible to relatively accurately estimate these probabilities from the stego image for additive schemes, it is much harder to estimate them for non-additive schemes of the type investigated in this report. This is because the cost of changing x_{ij} depends on the embedding changes executed in the previous sweep at all four neighboring pixels. As the pixels in both sublatices (4) and (5) change during the sweeps, the value of p_{ij} may change quite rapidly and unpredictably. Figure 8 shows p_{ij} versus the sweeps for five selected pixels in the cover image '1013.pgm' when embedding 0.4 bpp using Synch-HILL. The p_{ij} also depends on the stego key and the specific message that is being embedded.

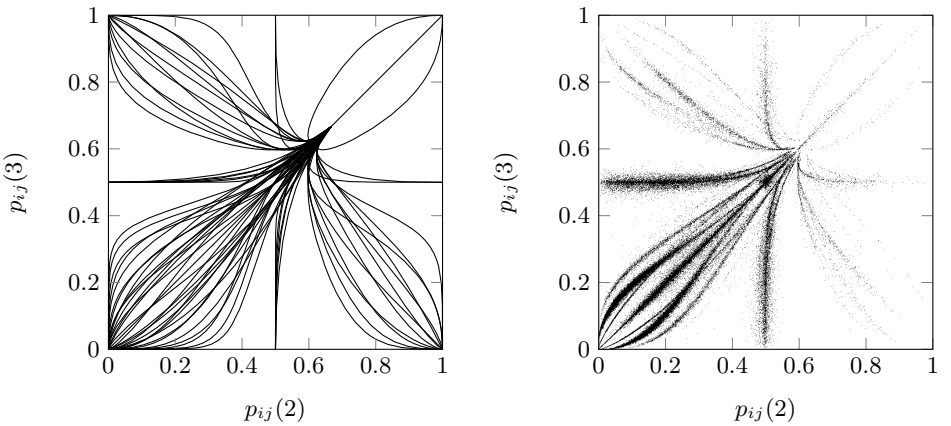

Figure 9: Theoretical (left) and real (right) total embedding change probabilities in Sweep 3 versus Sweep 2, $(p_{ij}(3), p_{ij}(2))$, for image '1013.pgm' when embedding 0.4 bpp with Synch-HILL ($\nu = 5$).

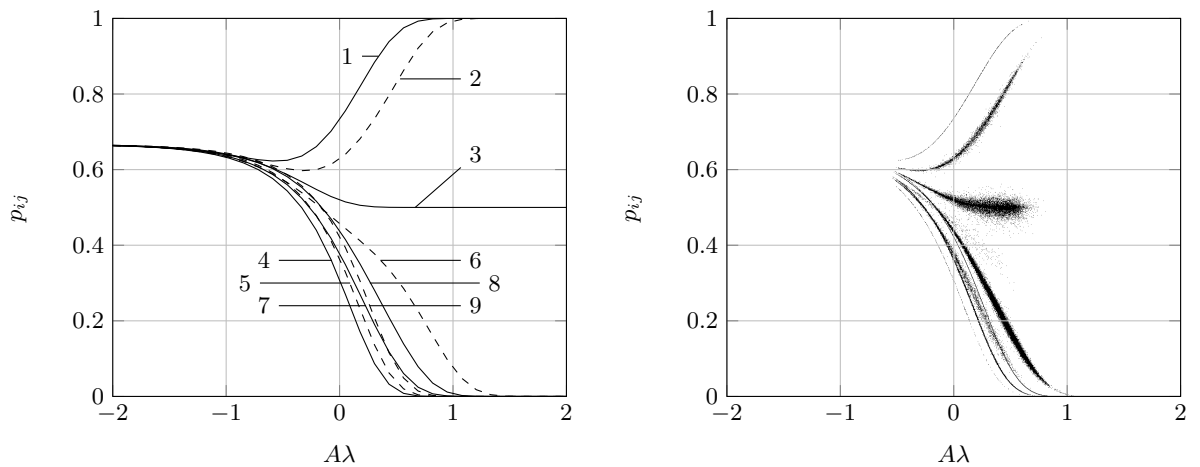

Figure 10: Theoretical (left) and real (right) total embedding change probabilities p_{ij} for image '1013.pgm' when embedding 0.4 bpp with Synch-HILL ($\nu = 5$).

To better understand what is happening during the sweeps, in Figure 9 (right) we plotted the points $(p_{ij}(3), p_{ij}(2))$ for all pixels (i, j) in the image '1013.pgm'. Here, $p_{ij}(k)$ stands for the total embedding change probability p_{ij} (10) at pixel (i, j) in kth sweep. The probabilities seem to approximately lie on a collection of well-defined smooth curves resembling an orchid. This can be explained by observing that the cost at pixel (i, j) depends on the actual embedding changes executed at its four neighboring pixels, namely those belonging to the four two-pixel cliques containing the pixel (i, j). For simplicity, let us assume that the costs ρ_{ij} of the additive scheme $\mathcal{A}$ are locally constant. Then, $A_{\mathcal{C}} \approx A$ for all four cliques, and there exist only nine types of neighborhood that lead to different costs of changing the pixel (i, j). The neighborhood types and the costs of changing the central pixel are all listed in Table 4. The last column is the frequency with which a given neighborhood type occurs if the embedding changes in the four neighboring pixels were executed equally likely. Only one representative example is listed for each neighborhood type. The rest is obtained by permutations. Also, some neighborhood types appear in two forms depending on the signs of the embedding changes. The listed costs correspond to the neighborhood types in the first column.

Therefore, when plotting $p_{ij}(k+1)$ (10) versus $p_{ij}(k)$, the points have to lie on 81 curves parametrized by λ. In fact, because the parameter A can be factored from the costs in the table, it is only the product $\lambda A = \lambda'$ that determines the value of $p_{ij}(k)$ (10) for each neighborhood type. Thus, when plotting the points $(p_{ij}(3), p_{ij}(2))$, they can lie on 81 curves depending on which type of neighborhood pixel (i, j) has in sweep $k + 1$ and k. For example, one of the curves corresponding to the case when the pixel (i, j) has a neighborhood of type 1 in sweep $k+1$ and type 2 in sweep k (both taken from the first column), the pair $(p_{ij}(k+1), p_{ij}(k))$ will lie on a curve given in its parametric form $(0 \le \lambda' < \infty)$:

$$p_{ij}(k+1) = \frac{\exp(-4\nu\lambda')}{\exp(-4\lambda') + \exp(-4\nu\lambda')}, \tag{11}$$

$$p_{ij}(k) = \frac{\exp(-\lambda') + \exp(-\lambda'(1+3\nu))}{\exp(-3\lambda') + \exp(-\lambda') + \exp(-\lambda'(1+3\nu))}. \tag{12}$$

When drawing all 81 possible curves in one graph, we

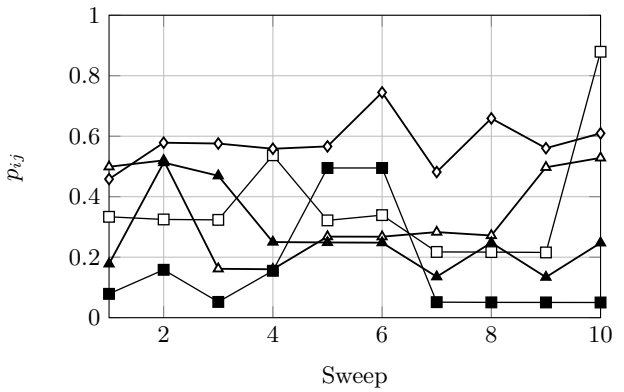

Figure 8: Embedding change probability p_{ij} for selected pixels as a function of sweeps. Note the rather unpredictable and rapid changes.

	Neighborhood type		$\rho^{(0)}$	$\rho^{(+)}$	$\rho^{(-)}$	Freq.
1	(1,1,1,1)	(-1,-1,-1,-1)	$4A$	0	$4\nu A$	2/81
2	(1,1,1,0)	(-1,-1,-1,0)	$3A$	A	$A+3\nu A$	8/81
3	(1,1,0,0)	(-1,-1,0,0)	$2A$	$2A$	$2A+2\nu A$	12/81
4	(1,1,-1,-1)		$4A$	$2\nu A$	$2\nu A$	6/81
5	(-1,1,0,0)		$2A$	$2A+\nu A$	$2A+\nu A$	12/81
6	(1,0,0,0)	(-1,0,0,0)	A	$3A$	$3A+\nu A$	8/81
7	(1,1,1,-1)	(-1,-1,-1,1)	$4A$	νA	$3\nu A$	8/81
8	(-1,1,1,0)	(1,-1,-1,0)	$3A$	$A+\nu A$	$A+2\nu A$	24/81
9	(0,0,0,0)		0	$4A$	$4A$	1/81

Table 4: Pixel costs for nine different types of changes to four neighboring pixels.

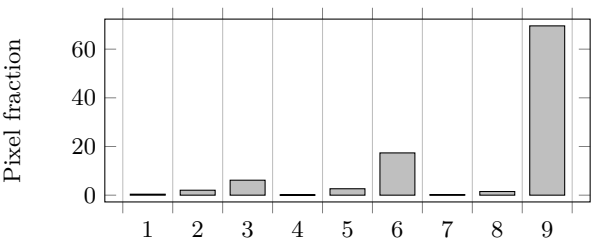

Figure 11: Percentage of pixels with embedding change probabilities lying on curves numbered 1–9 in Figure 10 (left). Curve no. 9 corresponds to the embedding change probabilities of the additive approximation to Synch-HILL.

obtain Figure 9 (left). The noise visible in Figure 9 (right) is due to the fact that the value of $A_{\mathcal{C}}$ depends on the clique and is in general not the same for all four two-pixel cliques surrounding the pixel (i,j).

An even better insight is obtained when plotting p_{ij} as a function of $\lambda' = \lambda A$ for all nine types of neighborhoods (Figure 10 left). Note that some pixels will be changed with an almost certainty since p_{ij} approaches 1 for neighborhood types 1 and 2. For type 3, the probability approaches 1/2 as $\lambda' \to \infty$. The probabilities of all types approach 2/3 when $\lambda' \to 0$, which is natural as this corresponds to a fully embedded image (maximal entropy at each pixel). Type 9 corresponds to the case when embedding with an additive approximation (6), whose pixel costs are just the slightly smoothed costs of the original additive scheme $\mathcal{A}$.

Figure 10 (right) shows the real embedding change probabilities computed from image '1013.pgm' for payload 0.4 bpp with Synch-HILL. Note that, with the exception of curves no. 1 and 9, there is some statistical spread of the values due to the fact that not all four values of $A_{\mathcal{C}}$ are the same (c.f. Table 4).

Figure 9 demonstrates the difficulty of estimating the embedding change probabilities for individual pixels. In fact, estimating on which curve the probability should lie is equivalent to estimating the *specific* embedding changes around each pixel! This indicates that attacking the Synch schemes using the selection channel will generally be much more difficult. The best choice of the embedding change probabilities for the Warden is thus to select the curve in Figure 10 (right) that has the majority of points (pixels), which is the curve corresponding to neighborhood type 9, namely its lower portion close to the x axis (for payload 0.4 bpp). This can be seen in Figure 11 where we plotted the percentage of pixels with neighborhood of type 1–9. Thus, considering the difficulty of estimating the pixel neighborhood type, the best the attacker can do is to use the embedding change probabilities of the additive scheme $\mathcal{A}$. Using the probabilities of the additive approximation (6) produced basically the same detection errors. This justifies the steganalysis carried out in Section 4.

6. CONCLUSION

This paper shows that the empirical security of additive spatial-domain steganographic schemes can be significantly improved when synchronizing adjacent embedding changes. The beneficial impact of the synchronization was linked to the fact that pixel prediction kernels frequently change sign as well as the fact that clustered changes disturb fewer noise residuals than scattered changes. Moreover, attacks that use an approximate knowledge of the selection channel are also less effective because it is significantly harder to estimate the embedding change probabilities of individual pixels as they strongly depend on the embedding changes in adjacent pixels.

The proposed approach is general and can be applied to any additive scheme. It uses the pixel costs of the additive scheme to construct a non-additive distortion function in which non-synchronized embedding changes made to adjacent pixels are penalized. The actual embedding is implemented with a single sweep of the Gibbs construction.

We subject the proposed scheme to steganalysis with rich models including their selection-aware versions. Detailed analysis of the embedding change probabilities in the synchronized schemes revealed that the embedding probabilities of individual pixels are of nine possible types depending on the actual embedding changes executed at the four neighboring pixels. Barring an accurate technique capable of estimating the individual embedding changes, the best option for the Warden is to steganalyze with the selection channel of the original additive scheme, which is how the steganalysis was executed in this paper.

There are a number of possible extensions of this work that may bring further improvement. First, one could consider larger neighborhoods than the four-pixel cross neighborhood to allow the embedding to "see" modifications along the di-

agonal direction. Second, it may be possible to extend the model-based approach to designing steganography (called MVG in this paper) to dependent pixels. The central problem we foresee with this direction is the estimation of the local model parameters.

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 Vahid Sedighi for useful discussions.

8. REFERENCES

[1] 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.

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

[3] 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.

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

[5] 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.

[6] J. Fridrich and J. Kodovský. Multivariate Gaussian model for designing additive distortion for steganography. In *Proc. IEEE ICASSP*, Vancouver, BC, May 26–31, 2013.

[7] 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.

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

[9] 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.

[10] V. Holub and J. Fridrich. Random projections of residuals for digital image steganalysis. *IEEE Transactions on Information Forensics and Security*, 8(12):1996–2006, December 2013.

[11] 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.

[12] A. D. Ker and R. Böhme. Revisiting weighted stego-image steganalysis. 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 6819, pages 5 1–17, San Jose, CA, January 27–31, 2008.

[13] 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.

[14] B. Li, S. Tan, M. Wang, and J. Huang. Investigation on cost assignment in spatial image steganography. *IEEE Transactions on Information Forensics and Security*, 9(8):1264–1277, August 2014.

[15] 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.

[16] 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.

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

[18] W. Tang, H. Li, W. Luo, and J. Huang. Adaptive steganalysis against WOW embedding algorithm. In A. Uhl, S. Katzenbeisser, R. Kwitt, and A. Piva, editors, *2nd ACM IH&MMSec. Workshop*, Salzburg, Austria, June 11–13, 2014.

[19] S. Verdú and T. Weissman. Erasure entropy. In *Proc. of ISIT*, Seattle, WA, July 9–14, 2006.

[20] 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.