

AN EMBEDDING MECHANISM FOR NATURAL STEGANOGRAPHY AFTER DOWN-SAMPLING

Patrick Bas

Univ. Lille, CNRS, Centrale Lille, UMR 9189
CRISTAL, F-59000 Lille, France

ABSTRACT

Natural Steganography (NS) uses the concept of cover-source switching to provide good undetectability performances [1]. The sensor noise of the source (camera) for a given ISO sensitivity ISO_1 is first modeled as an independent Gaussian distribution for each photo-site, then the embedding mimics a switch to another sensitivity $ISO_2 (> ISO_1)$. Because the embedding has to be performed on developed images, we investigate in this paper how to generate a stego-signal once the image is down-sampled. By studying different down-sampling mechanisms (sub-sampling, box down-sampling, tent down-sampling) applied on RAW images generated from a monochrome sensor, we show that the use of a convolution kernel with overlapping boundaries implies a synchronization mechanism similar to the one used by the CMD and Synch embedding schemes, but motivated here by statistical foundations. For each mechanism and scaling factor, we also compute the associated embedding rates and show that our results are in-line with experimental results previously highlighted for other steganographic schemes.

Index Terms— Steganography, model-based, sensor noise, down-sampling

1. INTRODUCTION

Contrary to distortion-based steganography [2, 3, 4] where the embedding is governed by the computation of a cost function computed from the activity of the cover image, model-based steganography relies on performing an embedding that mimics the statistical distribution of the cover content. It was proposed by Sallee [5] for JPEG-image by fitting the Laplacian model of DCT coefficients, and more recently it was the starting point of the Mipod [6] scheme proposed by Sedighi *et al.* whose embedding minimizes the distortion w.r.t. a Gaussian distribution. Another embedding strategy, related to model-based steganography, relies on mimicking image-noise and the idea here is to generate a stego-signal mimicking the properties of the image self-noise, it was proposed by Franz and Pfizmann [7, 8] as one of the first embedding strategy for digital images in 1999.

Other ingredients, shared with the scheme proposed in this paper, enable to increase the undetectability of the scheme: the use of the of side-information of the pre-cover proposed in [9] is one of them; and the use of synchronization mechanisms between neighboring pixels as recently proposed by the Synch and CMD strategies [10, 11] is another one. Moreover, the current paper is also linked with an interesting analysis on the impact of image down-scaling scaling laws [12] which shows that for a given embedding

rate, the detectability of a steganographic scheme relies on both the sub-sampling kernel and the scaling factor.

We recently proposed a steganographic scheme called *Natural Steganography* (NS) [1] relying on both the concept of model-based steganography and sensor noise mimicking. As recalled in sections 2 and 3 the embedding uses the concept of cover-source switching to mimic a picture captured at a higher ISO sensitivity than the sensitivity of the Cover image. We show in [1] that high undetectability is possible when the side-information offered by the pre-cover (here the RAW image) is used.

This paper studies how to generate a stego-signal for NS after image down-sampling and when considering different sampling strategies. Section 2 presents the concept of cover-source switching and describes the model for the cover-source based on the sensor noise. Section 3 recalls the embedding mechanism associated to NS. Section 4 explores the different sampling strategies of the stego-signal for different down-sampling kernels and section 5 presents the detectability performances and the embedding rates of NS before and after downscaling.

2. COVER SOURCE SWITCHING AND SENSOR NOISE MODEL

The key idea of this paper is to propose a steganographic scheme where the message embedding will be equivalent to switching from one source¹ S_1 to another source S_2 ; this practically can be done by designing an embedding that, when applied on S_1 , mimics the statistical properties of S_2 . More specifically we have decided to use the sensor noise to model a given source because its statistical model is rather simple, and we perform the embedding in such a way that the statistical properties of the stego images mimic the sensor noise of source S_2 . Another requirement in order to achieve practical embedding is to be able to compute the probability of embedding changes π_k in the developed domain², this in order to perform the practical embedding but also in order to compute the embedding rate. Because the embedding scheme relies on the natural statistical noise of the sensor, we decided to call this steganographic scheme “Natural Steganography” (NS).

For a given ISO setting ISO_1 , the sensor noise $N_{i,j}^{(1)}$ can be approximated as normally and independently distributed (see [1]). We have also a linear relation between the sensor noise variance and the photo-site³ expectation μ :

$$N_{i,j}^{(1)} \sim \mathcal{N}(0, a_1 \mu_{i,j} + b_1). \quad (1)$$

¹A source is defined here as a camera sensor at a given ISO sensitivity

²Contrary to the raw domain, the developed domain represents the domain where live the images once all the processes are done.

³A photo-site represents the basic cell of a CDD or CMOS sensor.

Thanks to the CNRS cyber-security action for funding.

The acquired photo-site sample $x_{i,j}^{(1)}$ is given by:

$$x_{i,j}^{(1)} = \mu_{i,j} + n_{i,j}^{(1)}, \quad (2)$$

and $X \sim \mathcal{N}(\mu, a_1\mu_{i,j} + b_1)$.

In order to estimate the parameters (a_i, b_i) we follow the protocol explained in [1] which is similar to the one proposed by Foi *et al.* [13] and summarized below.

We first capture a set of N_a raw images of a printed photo picturing a rectangular gradient going from full black to white. The camera is mounted on a tripod and the light is controlled using a led lighting system in a dark room. The raw images are then converted to PPM format (for color sensor) or to PGM format (for B&W sensor) using the dcrw open-source software [14] using the command:

```
dcrw -k 0 -4 file_name
```

which means that the dark signal is not automatically removed (option $-k=0$), and that the captured photo-sites are not post-processed and plainly converted to 16-bit (option -4). The empirical mean $\hat{\mu}$ and variance $\hat{\sigma}_N^2$ are after estimated by grouping each pixel location w.r.t. the empirical expectation over the N_a acquisitions and the (a_i, b_i) are coefficients of the linear regression $\hat{\sigma}_N^2 = f(\hat{\mu}) = \hat{a}\hat{\mu} + \hat{b}$.

3. PRINCIPLES OF NATURAL STEGANOGRAPHY

We first propose in this section a steganographic system that can be used with a monochrome sensor where nor demosaicking neither color transform is possible. Note that this setting is realistic since such sensors are available on the market (see 5.1), moreover preliminary works [15] also use the same methodology, but for color images.

We also assume that the developed images do not undergo further processing and only suffer 8-bit quantization. We can call this type of images “Out Of Camera” (OOC) Pictures. In [1] we have presented how NS has to be adapted to handle gamma correction, and in section 4 of this paper, we show how to deal with sub-sampling operations.

We propose to model the stego signal $S_{i,j}$ in such a way that it mimics the model of images captured at $ISO_2 > ISO_1$.

The equivalent of (1) and (2) for a camera sensitivity parameter ISO_2 are $N_{i,j}^{(2)} \sim \mathcal{N}(0, a_2\mu_{i,j} + b_2)$ and $x_{i,j}^{(2)} = \mu_{i,j} + n_{i,j}^{(2)}$.

Since the sum of two independent noises normally distributed is normal with the variances summing up, we can write that $x_{i,j}^{(2)} = \mu_{i,j} + n_{i,j}^{(1)} + s'_{i,j} = x_{i,j}^{(1)} + s'_{i,j}$ with $s'_{i,j} \sim \mathcal{N}(0, (a_2 - a_1)\mu_{i,j} + b_2 - b_1)$ representing the signal necessary to mimic image captured at ISO_2 . Assuming that the observed photo-site is very close to its practical expectation, i.e. that $\mu_{i,j} \simeq x_{i,j}^{(1)}, x_{i,j}^{(2)}$ can be approximated by:

$$x_{i,j}^{(2)} \simeq x_{i,j}^{(1)} + s_{i,j} \triangleq y_{i,j}, \quad (3)$$

with:

$$S_{i,j} \sim \mathcal{N}(0, (a_2 - a_1)x_{i,j}^{(1)} + b_2 - b_1), \quad (4)$$

adopting the following notations $a' \triangleq a_2 - a_1, b' \triangleq b_2 - b_1$, $\sigma_S^2 \triangleq a'x_{i,j}^{(1)} + b'$, and the photo-site of the stego image is distributed as:

$$Y_{i,j} \sim \mathcal{N}(x_{i,j}^{(1)}, \sigma_S^2). \quad (5)$$

Note that equation (3) shows explicitly the principle of cover-source switching which is simply represented in this case by adding an independent signal on each image photo-site to generate the stego photo-site $y_{i,j}$. The distribution of the stego signal in the continuous domain (see (4)) takes into account the statistical model of the sensor noises estimated for two ISO settings.

For OOC images, the only developing process lies in the 8-bit quantization function, consequently the goal here is to compute the embedding changes probabilities $\pi_{i,j}(k) = \Pr[\bar{S}_{i,j} = k]$ after this process, where \bar{s} represents the developed pixel. These probabilities can be either used to simulate optimal embedding, or to compute additive costs $\rho_{i,j}$ that feed a multilayered Syndrome Trellis Code using the “flipping lemma” [16] (see also section VI of [16] for Q-ary embedding and multi-layered constructions).

We compute the discretized probability mass function after a quantization step of size Δ (typically $\Delta = 256$ by quantizing from 16-bit resolution to 8-bit resolution). The embedding probabilities are directly linked to the 8 bits quantized value $x_{8B} = Q_\Delta(x_{16B})$ - where $Q_\Delta(\cdot)$ denotes the quantization function - and the pdf of the Normal distribution:

$$\begin{aligned} \pi(k) &= \int_{u_k}^{u_{k+1}} f(y|x = x_{16B})dy, \\ &= \frac{1}{2} \left(\text{erf} \left(\frac{u_{k+1} - x_{16B}}{2\sigma_S} \right) - \text{erf} \left(\frac{u_k - x_{16B}}{2\sigma_S} \right) \right), \end{aligned} \quad (6)$$

with $u_k = x_{8B} - (0.5 - k)\Delta$.

Once the embedding probabilities are computed for each pixel, it's possible to derive the payload size using the entropy formula:

$$H(\bar{S}) = - \sum_{i,j,k} \pi_{i,j}(k) \log_2 \pi_{i,j}(k). \quad (7)$$

4. STEGO-SIGNAL GENERATION AFTER DOWN-SAMPLING

We propose embedding strategies to deal with image down-sampling and we restrict our analysis to integer down-scaling factors $c \in \mathbb{N}$. We distinguish three strategies: sub-sampling, box down-sampling and down-sampling using convolutional kernels such as tent down-sampling.

4.1. Sub-sampling

Sub-sampling consists in selecting pixels distant by kc pixels ($k \in \mathbb{N}$) on each column and row of the image. For a stationary image, sub-sampling consequently does not modify the average embedding rate, but this sub-sampling method is rarely used in practice since it creates aliasing.

4.2. Box down-sampling

Box down-sampling consists in computing the averages of disjoint blocs of size $c \times c$ to compute down-sampled values (see Figure (2)(a)). The stego signal is consequently averaged on c^2 pixels is distributed as:

$$S_{box} \sim \mathcal{N}(0, \sigma_{box}^2), \quad (8)$$

with $\sigma_{box}^2 = \left[\sum_{i=1}^{c^2} \sigma_S^2(i) \right] / c^4 = (a'\bar{x} + b') / c^2$.

This means that on the developed image, the variance of the stego signal is divided by c^2 . As it will be analyzed in section 5.3, we can already notice that the embedding rate is a decreasing function of the scaling factor in this case.

4.3. Tent down-sampling

We now investigate more elaborated filters and particularly the problem of tent down-sampling (aka triangle down-sampling or bilinear down-sampling). This analysis enables to understand how it is possible to embed a message in the downscaled image using this particular filter, but it can also be adapted to all the class of linear filters, including for example the Gaussian kernel or the Lanczos kernel [17].

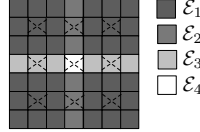


Fig. 1: Embedding steps used for tent-down sampling: dark grey photo-sites are sampled during the first step, light grey photo-sites during the second and third steps and white photo-sites during the last step.

Without loss of generality the principle of the embedding scheme is explained for $c = 2$ and in this case the tent filter and down-sampling process is illustrated on Figure 2(b).

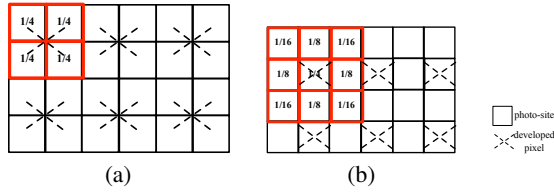


Fig. 2: Neighborhood and filters for box down-sampling (a) and tent-down-sampling (b) for a factor 2, crosses represent the location of the predicted pixel.

This method starts by decomposing the down-scaled pixels into four disjoint lattices and associated subsets. This decomposition is done in order to obtain developed pixels for which the stego signal is independently distributed conditionally to the neighborhood. On figure 2(b) we can see that the first and second developed pixels (represented by crosses) are not independent since 3 photo-sites contribute to the computation of both pixels, on the contrary the first and third pixels of the first row are independent.

The embedding procedure can be decomposed into four steps, each of them associated to a given subset of photo-sites.

In this section we adopt the following notations: indexes i, j are centered on the pixel to develop ($\{-1, 0, 1\}$ represents respectively $\{1, 2, 3\}$ rows or columns), and the tent filter is denoted as a 3×3 matrix with coefficient $c_{i,j}$. The four different subsets $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$ of photo-sites are represented on Figure 1. Moreover $\uparrow, \downarrow, \leftarrow, \rightarrow$ denotes the stego signal added on the photo-sites related to neighboring developed pixels according to the $\uparrow, \downarrow, \leftarrow, \rightarrow$ directions. As an example it means that $s_{-1,0} = \uparrow s_{1,0}$.

The embedding is sequentially performed in 4 steps:

1) The first step embeds part of the message (or generate the stego signal) into pixels belonging to \mathcal{E}_1 . Because the subset \mathcal{E}_1 generates independent pixels, the stego signal in the developed domain is distributed as:

$$\mathcal{N}(0, \sigma_{S1}^2), \quad (9)$$

with $\sigma_{S1}^2 = \sum_{i,j=-1}^1 c_{i,j}^2 \sigma_S^2(i, j)$. One can compute the embedding probabilities, and the associated payload length associated to the pixels belonging to \mathcal{E}_1 . In order to be able to sample the neighboring pixels, once an embedding change is done, we draw realizations of the 9 underlying photo-sites. This can be done by computing conditional probabilities (see section 4.3.1)

2) Developed pixels belonging to \mathcal{E}_2 have a sensor noise distributed according to the conditional density

$$f(\bar{s} | \bar{s}_{i-1,1}, \bar{s}_{i,1}, \bar{s}_{i+1,1}, \bar{s}_{i-1,1}, \bar{s}_{i,1}, \bar{s}_{i+1,1})$$

and consequently can be expressed as:

$$\mathcal{N}(\mu_{S2}, \sigma_{S2}^2), \quad (10)$$

with $\mu_{S2} = \sum_{i=-1}^1 c_{i,1} \bar{s}_{i,1} + \sum_{i=-1}^1 c_{i,-1} \bar{s}_{i,-1}$ and $\sigma_{S2}^2 = \sum_{i=-1}^1 c_{i,0}^2 \sigma_S^2(i, 0)$. As for the first step, we can compute embedding probabilities and payload length for this subset. We can also draw the realizations of stego signal related to the 3 photo-sites belonging to this subset. Note that the same applies for steps 3 and 4.

3) Similarly pixels belonging to \mathcal{E}_3 have a sensor noise are distributed as:

$$\mathcal{N}(\mu_{S3}, \sigma_{S3}^2), \quad (11)$$

with $\mu_{S3} = \sum_{j=-1}^1 c_{1,j} \bar{s}_{1,j} + \sum_{j=-1}^1 c_{-1,j} \bar{s}_{-1,j}$ and $\sigma_{S3}^2 = \sum_{j=-1}^1 c_{0,j}^2 \sigma_S^2(0, j)$, and as for step 2, it is possible to draw realizations of the stego signal.

4) An finally, pixels belonging to \mathcal{E}_4 have a sensor noise distributed as:

$$\mathcal{N}(\mu_{S4}, \sigma_{S4}^2), \quad (12)$$

with $\mu_{S4} = \sum_{j=-1}^1 c_{1,j} \bar{s}_{1,j} + \sum_{j=-1}^1 c_{-1,j} \bar{s}_{-1,j} + c_{0,1} \bar{s}_{0,1} + c_{0,-1} \bar{s}_{0,-1}$ and $\sigma_{S4}^2 = c_{0,0}^2 \sigma_S^2(0, 0)$. For this last step, notice that only one photo-site is drawn.

Note that since

$$H(\bar{S} | \bar{s}_{i-1,1}, \bar{s}_{i,1}, \bar{s}_{i+1,1}, \bar{s}_{i-1,1}, \bar{s}_{i,1}, \bar{s}_{i+1,1}) \leq H(\bar{S}),$$

the payload length embedded during steps 4 is smaller than the payload length embedded during 2 and 3, which is in turn smaller than the payload length embedded during step 1.

4.3.1. Embedding algorithm:

We describe here a sketch of the embedding algorithm, and we highlight how the sampling of the stego signal has to be conducted:

Step 1: Sample the stego signal \bar{s} in the developed domain on lattice \mathcal{E}_1 according to $\mathcal{N}(0, \sigma_{S1}^2)$.

Step 2: Sample the the related stego-signal on each photo-site $S_{i,j}$ belonging to \mathcal{E}_1 according to the conditional distribution of normal random variables [18]:

$$S_{i,j} | \bar{s}, \{s_{\text{drawn}}\} \sim \mathcal{N}\left(\left(\bar{s} - s_d\right) \frac{c_{i,j} \sigma_{i,j}^2}{\text{var}_{nd}}, \sigma_{i,j}^2 - \frac{c_{i,j}^2 \sigma_{i,j}^4}{\text{var}_{nd}}\right), \quad (13)$$

where $\{s_{\text{drawn}}\}$ and $\{s_{\text{not drawn}}\}$ represent respectively the sets of photo-sites already drawn and not drawn, $s_d \triangleq \sum_{\text{drawn}} c_{i',j'} s_{i',j'}$,

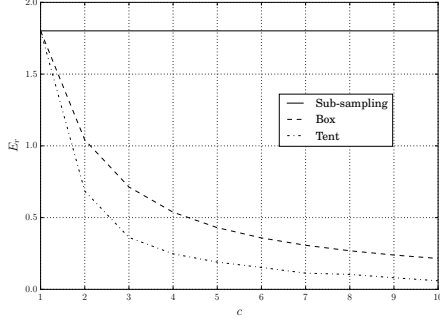


Fig. 3: Embedding rate vs scaling factor, 1000 ISO toward 1250 ISO embedding for a cover image uniformly distributed.

$\text{var}_{nd} \triangleq (C_{\text{not drawn}}^2 \otimes \sigma^2)(i, j)$, and \otimes denotes the convolution operator. Note that the implementation is faster if $(C_{\text{not drawn}}^2 \otimes \sigma^2)(i, j)$, $\sum_{\text{drawn}} c_{i',j'} s_{i',j'} = (C \otimes s)(i, j) = s_d$ and $c_{i,j} \sigma_{i,j}^2$ are computed in advance.

Step 3: On lattices \mathcal{E}_2 , \mathcal{E}_3 and \mathcal{E}_4 , sequentially sample alternatively as in steps 1 and 2: (i) the stego signal \bar{s} in the developed domain according to respectively $\mathcal{N}(\mu_{S2}, \sigma_{S2}^2)$, $\mathcal{N}(\mu_{S3}, \sigma_{S3}^2)$ and $\mathcal{N}(\mu_{S4}, \sigma_{S4}^2)$ and (ii) the stego photo-sites $S_{i,j}$ in the raw domain using formula (13).

5. RESULTS

5.1. Generation of “MonoBase”

In order to benchmark the concept of embedding using cover-source switching, we needed to acquire different sources providing OOC images. To do so we conducted the following experiment: using a Leica M Monochrome Type 230 camera, we captured two sets of 172 pictures taken at 1000 ISO or 1250 ISO. The acquisition protocol is detailed in [1] and MonoBase can be downloaded here [19].

5.2. Benchmark settings

We adopt the Spatial Rich Model feature sets [20] combined with the Ensemble Classifier (EC) [21] and we report the average total error $P_E = \min((P_{FA} + P_{MD})/2)$ obtained after training the EC on 10 different training/testing sets divided in 50/50. The stego database consists of images captured at 1000 ISO perturbed with an stego signal mimicking 1250 ISO, and the cover database consists of images captured at 1250 ISO. In order to have an effect equivalent with the principle of training using pairs of cover and stego images, the pairs are constructed using one couple of images capturing the same scene. We used the values $a' = 2.1 \cdot 10^{-5}$ and $b' = 8.4 \cdot 10^{-7}$ to mimic a transition from 1000 ISO to 1250 ISO and $N_b = 16$. For NS and NS after downscaling, the embedding rate is computed after 8 bits quantization using formula (6).

5.3. Detectability before and after down-sampling

Using this setting the average embedding rate of NS is equal to 1.24 bpp before downscaling. Table 1 presents the detectability results before and after down-sampling the images of a factor 2. We can notice the detectability is smaller after down-sampling, which is due to

the square root law [22]. Note also that contrary to [23] which performs the embedding directly on 12bits images, here the embedding is directly performed on the 8-bit version of the image.

	NS $E_r = 1.24$	NS Sub-sampling	NS $c = 2$, Box	NS $c = 2$, Tent
P_E	42.8%	48%	47.7%	48.0%

Table 1: Detectability after x2 down-sampling.

	SUni-SI $E_r = 1.24$	SUni $E_r = 1.24$	NS $c=2$, indep. Tent
P_E	18.2%	12.3%	22.6%

Table 2: Comparison w.r.t more detectable implementations.

Table 2 compare the detectability of NS w.r.t. other steganographic implementations which are S-Uniward [3], S-Uniward using side informed embedding as proposed in [9] and a generation of the stego-signal where each sample is drawn independently using for each pixel the sampling mechanism devoted to lattice \mathcal{E}_1 . The S-Uniward implementations uses a content embedding rate of 1.24 bpp. Note that all these implementations are more detectable than NS, and we highlight that it is important to sample correctly the signal in the developed domain by introducing the synchronization mechanism modeled by sampling on conditional distributions.

Figure 3 presents the evolution of the embedding rates computed using the densities of the stego signal for the three down-sampling methods presented in section 4 for an image having photo-sites uniformly and independently distributed between 0 and $2^{16} - 1$. We can notice that the embedding rates rapidly decrease w.r.t. the scaling factor for Box or Tent downscaling. The rate is constant for classical sub-sampling but this method generates aliasing and is never used in practice. If we for example look at the typical down-scaling operation used in BOSS-Base, a 18MP image (3840 x 2592) was downsampled with $c=5$, which lead in these case to $E_r \approx 0.4$ pbb for Box down-sampling and $E_r \approx 0.2$ pbb for Tent down-sampling. Compared with the initial embedding rate of 1.8 pbb, the reduction is rather important. Note that however for a given detectability constraint, the embedder can always increase the values of a' and b' to increase the payload, or change the cover-source switching setup by using $ISO_1 < 1000$ and/or $ISO_2 > 1250$.

6. CONCLUSION AND PERSPECTIVES

We have investigated in this paper how to adapt NS in order to handle down-sampling operations and we have proposed an embedding method to deal with pixels generated from overlapping kernels such as the tent filter. By comparing with the literature on steganography, we can draw two remarks:

1) The evolution of the detectability is perfectly in-line with the analyzed effect of rescaling in steganalysis [12] (see for example Figure 1 of [12]) which outlines that the Tent kernel is more detectable than the Box kernel.

2) The generation of the stego signal after down-sampling implies computation of conditional probabilities which force the embedding scheme to correlate the neighboring embedding changes, which is also inline with the Synch or CMD strategies presented in [10, 11].

7. REFERENCES

- [1] P. Bas, "Steganography via Cover-Source Switching," 2016, IEEE Workshop on Information Forensics and Security (WIFS). [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01360024>
- [2] T. Pevny, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Information Hiding 2010*, 2010.
- [3] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP Journal on Information Security*, vol. 2014, no. 1, pp. 1–13, 2014.
- [4] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4206–4210.
- [5] P. Sallee, "Model-based steganography," in *International Workshop on Digital Watermarking (IWDW), LNCS*, vol. 2, 2003.
- [6] V. Sedighi, R. Cogranne, and J. Fridrich, "Content-adaptive steganography by minimizing statistical detectability," *Information Forensics and Security, IEEE Transactions on*, vol. 11, no. 2, pp. 221–234, 2016.
- [7] E. Franz and A. Pfitzmann, "Steganography secure against cover-stego-attacks," in *Information Hiding*. Springer, 1999, pp. 29–46.
- [8] E. Franz and A. Schneidewind, "Pre-processing for adding noise steganography," in *Information Hiding, 7th International Workshop*, 2005, pp. 189–203.
- [9] T. Denemark and J. Fridrich, "Side-informed steganography with additive distortion," in *Information Forensics and Security (WIFS), 2015 IEEE International Workshop on*. IEEE, 2015, pp. 1–6.
- [10] T. Denemark, V. Sedighi, V. Holub, R. Cogranne, and J. Fridrich, "Selection-channel-aware rich model for steganalysis of digital images," in *IEEE Workshop on Information Forensic and Security, Atlanta, GA*, 2014.
- [11] B. Li, M. Wang, X. Li, S. Tan, and J. Huang, "A strategy of clustering modification directions in spatial image steganography," *Information Forensics and Security, IEEE Transactions on*, vol. 10, no. 9, pp. 1905–1917, 2015.
- [12] J. Kodovsky and J. Fridrich, "Steganalysis in resized images," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2857–2861.
- [13] A. Foi, S. Alenius, V. Katkovnik, and K. Egiazarian, "Noise measurement for raw-data of digital imaging sensors by automatic segmentation of nonuniform targets," *IEEE Sensors Journal*, vol. 7, no. 10, pp. 1456–1461, 2007.
- [14] "<http://www.cybercom.net/dcoffin/dcraw/>."
- [15] P. Bas, "Natural Steganography: Cover-source Switching For Better Steganography," Jul. 2016, working paper or preprint. [Online]. Available: <https://arxiv.org/abs/1607.07824>
- [16] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *Information Forensics and Security, IEEE Transactions on*, vol. 6, no. 3, pp. 920–935, 2011.
- [17] K. Turkowski, "Filters for common resampling tasks," in *Graphics gems*. Academic Press Professional, Inc., 1990, pp. 147–165.
- [18] A. Papoulis and U. Pillai, *Probability, Random Variables and Stochastic Processes*. Mac Graw Hill, 2002.
- [19] P. Bas, "Monobase," <http://patrickbas.ec-lille.fr/MonoBase/>, July 2016.
- [20] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *Information Forensics and Security, IEEE Transactions on*, vol. 7, no. 3, pp. 868–882, 2012.
- [21] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *Information Forensics and Security, IEEE Transactions on*, vol. 7, no. 2, pp. 432–444, 2012.
- [22] A. D. Ker, T. Pevný, J. Kodovsky, and J. Fridrich, "The square root law of steganographic capacity," in *Proceedings of the 10th ACM workshop on Multimedia and security*. ACM, 2008, pp. 107–116.
- [23] J. Fridrich, "Effect of cover quantization on steganographic fisher information," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 2, pp. 361–373, 2013.