Effect of image downsampling on steganographic security

Jan Kodovský, *Member, IEEE* and Jessica Fridrich, *Member, IEEE*

***Abstract*—The accuracy of steganalysis in digital images primarily depends on the statistical properties of neighboring pixels, which are strongly affected by the image acquisition pipeline as well as any processing applied to the image. In this paper, we study how the detectability of embedding changes is affected when the cover image is downsampled prior to embedding. This topic is important for practitioners because the vast majority of images posted on websites, image sharing portals, or attached to e-mails are downsampled. It is also relevant to researchers as the security of steganographic algorithms is commonly evaluated on databases of downsampled images. In the first part of this paper, we investigate empirically how the steganalysis results depend on the parameters of the resizing algorithm – the choice of the interpolation kernel, the scaling factor (resize ratio), anti-aliasing, and the downsampled pixel grid alignment. We report on several novel phenomena that appear valid universally across the tested cover sources, steganographic methods, and the steganalysis features. The paper continues with a theoretical analysis of the simplest interpolation kernel – the box kernel. By fitting a Markov chain model to pixel rows, we analytically compute the Fisher information rate for any mutually independent embedding operation and derive the proper scaling of the secure payload with resizing. For LSB matching and a limited range of downscaling, the theory fits experiments rather well, which indicates the existence of a new scaling law expressing the length of the secure payload when the cover size is modified by subsampling.**

I. Introduction

Steganography is the art of hiding secret messages in cover objects. When the object is a digital media file, the message is typically embedded by slightly changing the individual cover elements.[1]

The work on this paper was supported by the Air Force Office of Scientific Research under the research grant FA9550-12-1-0124. 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.

Jan Kodovský is currently with Facebook, Inc. His work on this paper has been done while he was with the Department of Electrical and Computer Engineering, Binghamton University, NY, USA. Email: jan@kodovsky.com.

Jessica Fridrich is with the Department of Electrical and Computer Engineering, Binghamton University, NY, USA. Email: fridrich@binghamton.edu.

[1]This embedding paradigm is known as steganography by cover modification (see, e.g., Chapter 4 in [6]).

In this paper, we deal with covers in the form of digital images represented in the spatial domain. The statistical detectability of steganographic embedding changes primarily depends on the strength and type of dependencies among neighboring pixels, which in turn depend on the image acquisition pipeline as well as any processing applied to the image prior to embedding. It is, for example, well recognized that embedding changes are very difficult to detect in scans of analogue photographs due to the strong noise typically present (see, e.g., the results on the NRCS database in [13]). On the other hand, previous JPEG compression (and low-pass filtering in general) removes the high spatial frequency components of the image content (noise), which allows for more accurate steganalysis [8], [2], [10], [17], [13], [21]. The final bit depth representation of the cover image also has a very strong effect on steganalysis [7]. Even simple point-wise operations, such as contrast/brightness adjustment and gamma correction can have a very strong impact because of their potential to introduce spikes in the first-order statistic of pixel values [3], [25]. The effect of local image variance and saturation on the error of structural steganalyzers appeared in [2]. Finally, the impact of the image size on steganalysis is addressed by the Square Root Law (SRL) of imperfect steganography [5], [15]. Note that this law pertains to the case when the cover size is changed by removing/adding cover elements from the same distribution (which is approximately valid when cropping or concatenating images when creating a panorama) and does not address image resizing, which changes the statistical properties of the cover source.

The main goal of this article is to study the effect of downsampling on the detectability of steganographic embedding changes. We consider this an important topic for several reasons. Full-resolution images are rarely used on the Internet, and image downscaling is commonly adopted by many popular high-traffic websites, including social networking websites (Facebook), on-line stores (Amazon, eBay), news websites (CNN, MSNBC), etc. Most of the image-sharing portals (Picasa Web Albums, Photobucket, Flickr) also utilize image downsampling and some of them allow downloading several different downscaled versions of a given image.[2] Email attachments and presentation slides are yet another two examples of communication channels where resized images are commonly used.

Additionally, for the purpose of benchmarking steganog-

[2]Many portals also apply lossy JPEG compression to the resized images. We note that in this paper we do not study the case when the cover images are resized and subsequently JPEG compressed.

raphy and steganalysis, the steganographic community adopted several image databases that contain resized images. Among the most often used databases are the BOSSbase[3] and BOWS2.[4] BOSSbase was originally used for the Break Our Steganographic System (BOSS) competition [1] aimed at attacking the content-adaptive embedding scheme called HUGO (Highly Undetectable steGO) [24]. Both BOSSbase and BOWS2 images are all *downsampled* (and cropped) versions of their RAW originals. As will be shown in this paper, the outcome of steganalysis can vary quite dramatically based on the downscaling algorithm and its parameters. Understanding these implications is important since practitioners often take the steganalysis results obtained on these databases as an *absolute measure* of security of a steganographic algorithm.

In summary, given the proliferation of imagery subjected to downscaling, it is rather surprising that, to the best knowledge of the authors, the effect of resizing on steganographic security has not yet been methodologically addressed. The only prior art the authors are aware of is the early conference version of this paper published at IEEE ICASSP in 2013 [18]. Here is the summary of the main differences between [18] and this paper:

1) This manuscript includes experiments on three different camera sources while in [18], only a single source was used. Furthermore, all images used in this manuscript are publicly available in their RAW format to facilitate reproducibility.
2) Instead of steganalysis features constructed using a single kernel originally proposed in [14], state-of-the-art rich features [9] and SPAM features [23] are used here.
3) Besides LSB matching used in [18], we added a state-of-the-art content-adaptive steganographic algorithm WOW [12].
4) We include a much more comprehensive study of the effects of anti-aliasing and kernel-shifting.

We start the next section with a motivational experiment showing strikingly different results of steganalysis of HUGO [24] depending on the choice of the resizing kernel used to downsample the original full-resolution images forming the BOSSbase. In Section III, we formally introduce the process of image downsampling and describe its parameters. We also introduce the common core of all subsequent experiments in this paper. The first part of the main results of this paper appears in Section IV, where we empirically study the effect of the interpolation kernel, downsampling factor, anti-aliasing, and the downsampled grid alignment on statistical detectability. We point out some interesting phenomena that appear to hold universally across the tested sources, steganography methods, and steganalysis features. In Section V, we provide a theoretical analysis of the impact of downsampling using the nearest neighbor resizing algorithm by adopting a Markov chain model for the cover source. For this type of the cover source and the resizing algorithm, there exists a closed-form expression for the steganographic Fisher information rate for any mutually independent embedding operation, which allows us to determine the size of the secure payload that leads to the same level of statistical detectability. The paper is concluded in Section VI.

[3]http://exile.felk.cvut.cz/boss

[4]http://bows2.ec-lille.fr

II. Illustrative experiment on BOSSbase

The BOSSbase image database (version 1.01) consists of 10,000 grayscale images of size 512×512 pixels obtained from full-resolution RAW images (coming from seven different cameras) by executing the following four-step procedure:

1) Image demosaicking (Color Filter Array interpolation);
2) Conversion to 8-bit grayscale;
3) Downsampling so that the smaller side is 512 pixels;
4) Central-cropping to 512×512 pixels.

Image demosaicking was performed using UFRaw,[5] while the remaining steps were carried out using the ImageMagick's `convert` command-line tool with all parameters kept at their default values. The actual script for creating BOSSbase images is available at the BOSS organizers' website [1].

In order to motivate our study, we modified the script and prepared four different versions of BOSSbase. Everything else being equal, the four databases differed only in the choice of the interpolation kernel in the `convert`'s image resizing algorithm: box, Lanczos [11] (default), triangle, and cubic.[6] Figure 1 (right) shows the four interpolation kernels.

Figure 1 (left) depicts the results of steganalyzing HUGO implemented with $\sigma = \gamma = 1$ and the threshold $T = 255$ on all four databases. For every payload, a steganalysis detector was constructed by training the ensemble classifier [19] when representing the images using the 12,753-dimensional spatial rich model SRMQ1 (called Q1 in [9]). Half of the images were used for training and the other half for testing, while the performance was measured in terms of the minimal total detection error under equal priors,

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

where P_{FA} and P_{MD} are the false-alarm and missed-detection rates achieved on the testing set. By $\overline{P}_{\mathrm{E}}$ we denote the testing error averaged over ten random splits of the database into two halves.

The differences between the error rates achieved on different versions of BOSSbase are rather striking. For example, at the relative payload 0.2 bpp (bits per pixel), the detection error dropped from 0.27 with the default

[5]http://ufraw.sourceforge.net/

[6]This was achieved by modifying a single line in the original resizing script.

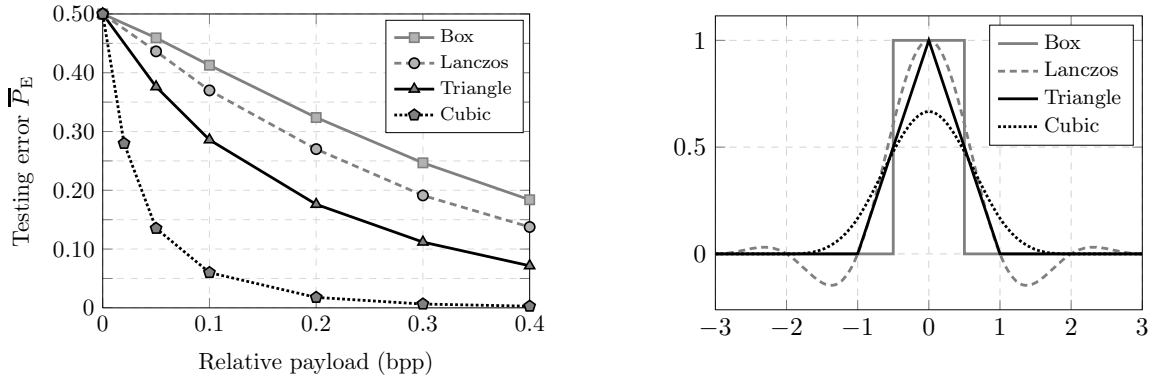

Figure 1. Steganalysis of HUGO on BOSSbase 1.01 images created with four different interpolation kernels implemented in ImageMagick's command-line tool `convert`. Left: Mean testing error $\overline{P}_E$; Right: Individual kernel functions.

Lanczos kernel to an almost perfect detectability (error 0.02) with bicubic interpolation.

In summary, the choice of the interpolation parameters significantly affects the steganographic security, and thus a deeper understanding of this phenomenon is of a great importance for steganalysis. In particular, the outcome of the BOSS competition and HUGO's security would be viewed in a very different light had the organizers inadvertently chosen a different interpolation algorithm for resizing. This experiment also points out the danger of interpreting the detection errors obtained on BOSSbase as an absolute measure of algorithm's security.

III. Notation, preliminaries, and experimental setup

In this section, we formalize the process of image downsampling and introduce the experimental setup for all subsequent experiments.

A. Image acquisition

A digital image $\mathbf{X}$ captured by an imaging sensor is a quantized sampled portion of the natural scene, which can be represented as a two-dimensional real function $f : \mathbb{R}^2 \to \mathbb{R}$ (the "reality"). Formally,

$$\mathbf{X}(x, y) = Q\left(C_\Theta(x, y) \cdot f(x, y)\right), \tag{2}$$

where Q denotes a scalar quantizer with a finite set of centroids $\mathcal{I}$ and $C_\Theta(x, y)$ is a discrete sampling function

$$C_\Theta(x, y) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} \delta(x - x_0 - k\Delta)\delta(y - y_0 - l\Delta) \tag{3}$$

parametrized by the vector $\Theta = (x_0, y_0, \Delta, M, N)$; $\delta(x)$ is defined as

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Equation (2) allows us to represent $\mathbf{X}$ as a matrix $\mathbf{X} \in \mathcal{I}^{M \times N}$, whose elements correspond to the natural scene f at $M \times N$ equally-spaced locations arranged in a rectangle that is uniquely defined by the parameter vector Θ. The set of quantizer centroids, $\mathcal{I}$, depends on the bit depth at which $\mathbf{X}$ is represented. For example, for 8-bit grayscale images, $\mathcal{I} = \{0, 1, \ldots, 255\}$.

B. Image resizing

The output of an image resizing algorithm, parametrized by the *downsampling factor*[7] k, is commonly defined as

$$\mathbf{X}^{(k)}(x, y) = Q\left(C_{\Theta^{(k)}}(x, y) \cdot \hat{f}(x, y)\right), \tag{5}$$

where

$$\Theta^{(k)} = (x_0^{(k)}, y_0^{(k)}, \Delta/k, \lfloor M/k \rfloor, \lfloor N/k \rfloor) \tag{6}$$

denotes the parameter vector of the resized image and

$$\hat{f}(x, y) = (\mathbf{X} * \varphi)(x, y) \tag{7}$$

is an approximation of reality obtained as a convolution of the original image $\mathbf{X}$ with an interpolation kernel $\varphi : \mathbb{R}^2 \to \mathbb{R}$. The kernel function φ needs to satisfy

$$\int_{\mathbb{R}^2} \varphi(x, y)\mathrm{d}x\mathrm{d}y = 1. \tag{8}$$

In general, the starting point of the grid of the resized image $(x_0^{(k)}, y_0^{(k)})$ can be different from the starting point of the original image (x_0, y_0), i.e., the first pixel of the resized image $\mathbf{X}^{(k)}$ does not have to coincide with the first pixel of the original image $\mathbf{X}$. In fact, many downsampling algorithms define the point $(x_0^{(k)}, y_0^{(k)})$ as

$$x_0^{(k)} = y_0^{(k)} = (k + 1)/2, \tag{9}$$

which corresponds to centering the sampling points of $\mathbf{X}^{(k)}$ within the grid of the original image $\mathbf{X}$. In Section IV-C, we will show that the position of the point $(x_0^{(k)}, y_0^{(k)})$ plays an important role in steganalysis.

For simplicity, in the rest of the paper we assume that $M = N$, $x_0^{(k)} = y_0^{(k)}$ for all k, and $\varphi(x, y) = \varphi(x)\varphi(y)$. The variable k will exclusively denote the resizing factor.

[7] Downsampling factor k corresponds to the image of relative size $1/k$ w.r.t. the original image size, e.g., $k = 2$ denotes downsampling by 50%.

Table I
List of individual camera models in BOSSbase 1.01 sorted by their native resolution.

Camera model	#	Full resolution	Mpix
Leica M9	2,758	3472 × 5216	18.1
Canon EOS 7D	1,354	5202 × 3465	18.0
Pentax K20D	1,398	3124 × 4688	14.6
Canon EOS Rebel XSi	2,042	2856 × 4290	12.3
Canon EOS 400D	1,354	2602 × 3906	10.2
Canon EOS 40D	61	2602 × 3908	10.2
Nikon D70	1,033	2014 × 3039	6.1

C. Controlled image database

BOSSbase is a collection of images coming from seven different cameras with different original resolutions, ranging from 6 to 18 megapixels (see Table I). Therefore, when downsampling to 512×512 pixels, individual cameras were resized with different resizing factors. For instance, Leica M9 was downsampled with $k = 6.78$, Pentax K20D with $k = 6.10$, and Canon EOS 400D with $k = 5.08$. Since different scaling factors introduce qualitatively different dependencies among pixels, in order to isolate the subtle effects of interpolation, we need to study individual camera models separately.

In the rest of the paper, we consider only the following three camera models: Leica M9 (LEI), Pentax K20D (PEN), and Canon EOS 400D (CAN). These three models (highlighted in Table I) were selected as examples of cameras coming from different camera manufacturers and are equipped with three different sensors.

From every camera model considered, we randomly selected $1,000$ raw images, demosaicked them using UFRaw (with the setup used during the BOSS competition), and converted to 8-bit grayscale. The resulting databases of never compressed (and not resized) images are the mother databases for all our subsequent experiments.[8]

D. Image downsampling in Matlab

The command-line utility `convert` is not transparent and its image resizing algorithm seems to incorporate several image-enhancing techniques (for example post-sharpening). Therefore, from now on we will solely use the Matlab's function `imresize`, as it follows the image-resizing procedure outlined in Section III-B *exactly*. Furthermore, it is easy to supply the function `imresize` with custom interpolation kernels, which will prove to be advantageous later in Section IV.

We consider the following built-in kernel functions of `imresize`:

[8]Fixing the image-processing pipeline, including the demosaicking algorithm, allowed us to isolate the effects of image downsampling and its parameters.

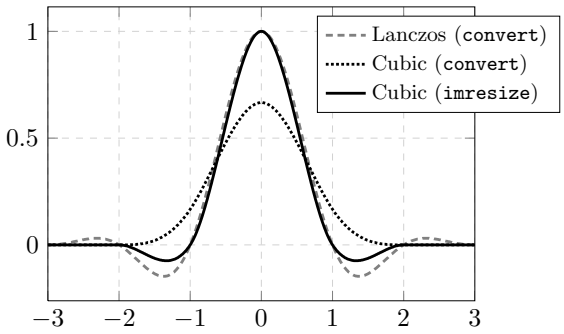

Figure 2. Cubic interpolation kernel as implemented in ImageMagick's `convert` and Matlab's `imresize`.

$$\varphi_{\mathrm{b}}(x) = \begin{cases} 1 & \text{if } -\frac{1}{2} \le x < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$\varphi_{\mathrm{t}}(x) = \begin{cases} 1 - |x| & \text{if } |x| \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\varphi_{\mathrm{c}}(x) = \begin{cases} \frac{3}{2}|x|^3 - \frac{5}{2}|x|^2 + 1 & \text{if } |x| \le 1 \\ -\frac{1}{2}|x|^3 + \frac{5}{2}|x|^2 - 4|x| + 2 & \text{if } 1 < |x| \le 2 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

which correspond to the nearest neighbor (box), bilinear (triangle), and bicubic (cubic [16]) interpolation, respectively. While the box and triangle kernels, φ_b and φ_t, are identical to the ones in `convert`, the bicubic interpolation is implemented differently. In Figure 2, we compare the cubic kernels of both software tools. We can see that the Matlab's implementation of the bicubic interpolation bears more similarity to the `convert`'s Lanczos kernel rather than its cubic counterpart. This observation will later explain the qualitatively different results of bicubic interpolation than the ones observed in Section II (Figure 1 left). It is also the reason why we omit the Matlab's Lanczos built-in kernels from our experiments.

E. Steganography and steganalysis

In the rest of the paper we attack the following two steganographic algorithms: LSB Matching (LSBM) and WOW [12]. The LSBM stego images were created by changing a pseudo-randomly and uniformly selected relative portion β of pixels (the change rate) by either increasing or decreasing their values by 1, equiprobably. For creating WOW stego images, we used the WOW embedding simulator.[9] Note that while LSBM spreads embedding changes uniformly over the image, WOW is a content-adaptive scheme whose changes are concentrated in noisy and textured areas. Therefore, these two choices cover two qualitatively different embedding paradigms.

[9]http://dde.binghamton.edu/download/

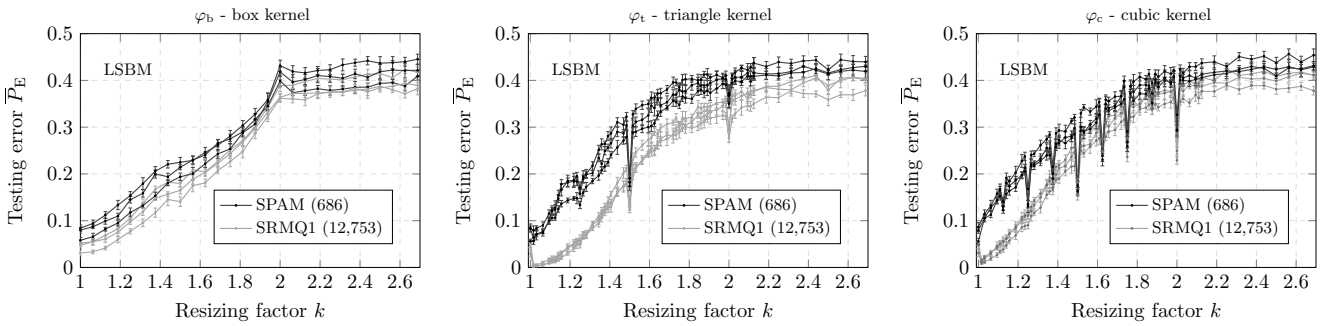

Figure 3. Steganalysis of LSBM with the change rate fixed at $\beta = 0.05$ in images from three cameras resized by three different interpolation kernels defined by formulas (10)–(12). Feature sets: SPAM (black), SRMQ1 (gray). The three different curves represent individual cameras (LEI, PEN, CAN).

Detection is executed using two different feature spaces: the 686-dimensional SPAM [23] and the 12,753-dimensional rich feature space SRMQ1 [9]. While the SRMQ1 represents state-of-the-art steganalysis, including the low-dimensional SPAM feature set brings a qualitatively different detector. For both feature-space representations, we always train the ensemble classifier [19] and evaluate the performance in the same way as described in Section II – half of the images are used for training and the other half for testing, and the performance is measured in terms of $\overline{P}_E$, the minimal total detection error defined by Equation 1 averaged over 10 different database splits.

In order to compare the steganographic security at different scaling factors, we always central-crop the resulting resized image $\mathbf{X}^{(k)}$ (the output of the interpolation formula (5)) to the 512×512 pixel region. This way, the statistical properties of pixels are preserved and the effect of the SRL on security is eliminated.

IV. Experimental results

In this section, we present a series of steganalysis experiments showing the effects of several different interpolation parameters on the steganographic security. In particular, we address the influence of the interpolation kernel, the downsampling factor k, anti-aliasing, and the spatial alignment of the resized grid of pixels. As will be shown, each of these factors can significantly affect the outcome of steganalysis.

A. Interpolation kernel

We start our investigation by fixing the change rate β of LSBM embedding and monitoring the change in the average testing error $\overline{P}_E$ with increasing downsampling factor k. The experiment was repeated for all three camera models (LEI, PEN, CAN), both feature spaces (SPAM, SRMQ1), and the three interpolation kernels defined by formulas (10)–(12). This can be achieved, for example, by calling the Matlab's function `imresize` with kernels 'box', 'triangle', and 'cubic', and turning the anti-aliasing off (the effect of anti-aliasing is studied in Section IV-B). The observed error rates are shown in Figure 3; the error bars correspond to the standard deviation of P_E over ten different database splits.

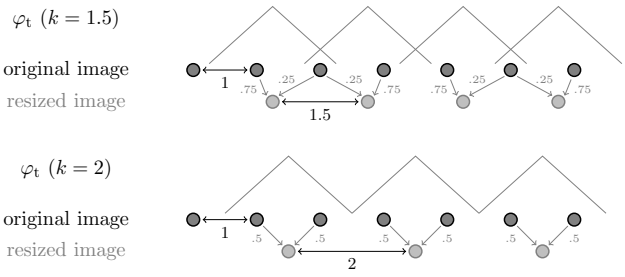

Figure 4. Explaining the sudden drops in $\overline{P}_E$: Downsampling with the triangle kernel φ_t and two different choices of k. Anti-aliasing turned off. See details in the text.

Generally, the testing error grows with increasing k. This is to be expected since the interpolation kernels have a fixed width while the spatial distance between the neighboring pixels of the resized image increases. Consequently, the strength of the dependencies among pixels decreases, which makes steganalysis more difficult.

There are differences among individual kernels, however. For the box kernel, the error monotonously increases as more and more pixels of the original image are being "skipped" during the process of downsampling – the box kernel with the anti-aliasing turned off is equivalent to the nearest-neighbor interpolation. This breaks after $k > 2$ when none of the pairs of neighbors from the original image are preserved any more and at least one pixel from the original grid is always skipped. While the rich feature space performs consistently better, both feature spaces exhibit qualitatively similar behavior.

The situation becomes more interesting (and less trivial) for the other two kernels whose width is greater than one and thus the pixel values in the resized image are interpolated as certain linear combinations of the original pixel values. While the testing error $\overline{P}_E$ generally still increases with growing k, its progress is not always monotonous. In particular, there are several noticeable sharp drops in the error rate for certain values of k. These sudden drops are

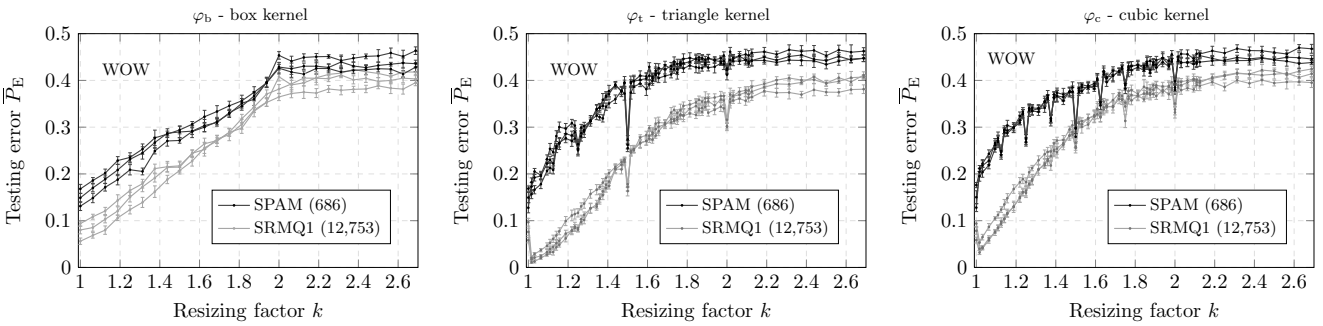

Figure 5. Steganalysis of WOW at a fixed payload of 0.3 bpp (bits per pixel) in images from three cameras resized by three different interpolation kernels defined by formulas (10)–(12). Feature sets: SPAM (black), SRMQ1 (gray). The three different curves represent individual cameras (LEI, PEN, CAN).

caused by a temporary synchronization of both pixel grids, which substantially strengthens pixel dependencies in the resized image.

Consider, for example, the case of downsampling with the triangle kernel and $k = 1.5$ (downsizing by 33%) where the testing error suddenly drops by 15% for the SPAM features and by 10% for the SRMQ1 features (see Figure 3 middle). This downsampling scenario is illustrated in full detail in the top diagram of Figure 4. As can be seen, for $k = 1.5$ the pixel positions in the resized image always fall between two neighboring pixels from the original grid in the way that the split ratio is 1:3. Consequently, *all* pixels of the resized image are formed as the *same* convex combination (i.e., $[1/4, 3/4]$) of two neighboring pixels from the original image. In other words, the interpolation formula stays the same across the whole image. This would not be true any more if the value of k was slightly increased (or decreased), in which case the coefficients of the linear combination change across the image. Consequently, any statistical features formed as sample joint probabilities of neighboring pixels (or their residuals) are essentially aggregates of a wide range of weaker statistics extracted from differently filtered image regions, which makes them less powerful for steganalysis.

A similar situation occurs for $k = 2$ shown in the bottom diagram of Figure 4. Now, the distance between two pixels of the resized image is 2 and they are always centered between two pixels from the original grid, which essentially amounts to pixel averaging.[10] As with $k = 1.5$, both grids of pixels are synchronized and the interpolation formula thus stays the same across the image, making the subsequently extracted features more sensitive to steganography.

Note that if the pixel grid of the resized image was slightly shifted, i.e., the position of the first resized pixel $(x_0^{(k)}, y_0^{(k)})$ was determined differently than through the formula (9), both diagrams in Figure 4 could have looked quite differently and the overall steganalysis results could have differed as well. We will inspect this in more detail in Section IV-C.

Before we proceed to the next section, we would like to make a few more comments. First, as can be seen in Figure 3 (right), the cubic kernel exhibits even more sudden drops in the testing error $\overline{P}_E$ for certain (rational) values of k when the grids exhibit a certain level of synchronization. This is likely because of the larger support of the cubic kernel which not only causes more pixel values to be combined together but it also creates a larger overlap between neighboring kernels and thus offers more space for synchronization.

Second, compared to the nearest neighbor interpolation (the box kernel), the difference between the performance of the SPAM features and the rich SRMQ1 features in case of the triangle and the cubic kernel seems to be larger. This is more pronounced for smaller values of k when the neighboring interpolation functions overlap and thus a single pixel from the original image can affect multiple pixels of the resized image (compare both diagrams in Figure 4 – for $k = 1.5$, one half of the original pixels contributes to two resized pixels). This creates complex dependencies among the pixels of the resized image that can be better exploited by the "richer" feature space SRMQ1 rather than the SPAM feature space. Another supporting evidence for this argument is the initial strong drop of error when k is only slightly larger than one, see Figure 3 (middle and right). This sudden *misalignment* of the pixel grids at both resolution levels causes some (if not all) pixels from the original image to start immediately contributing to more than one pixel of the resized image. While the SRMQ1 can utilize this strengthened dependence among pixels quite well, there is no drop of error for the SPAM features.

Finally, note that for larger values of k, the differences among all three considered kernels diminish and the error rates seem to saturate at similar values. This is likely because of the low-gradient portions of images (e.g., the sky) where pixel dependencies remain almost unaffected by downsampling. Furthermore, at the *odd* resolution levels $k = 3, 5, 7, \ldots$ and, trivially at $k = 1$, the pixel locations of the resized image always coincide with certain

[10] This can be seen as a simple denoising operator that increases local correlations among pixels and thus makes steganalysis more accurate.

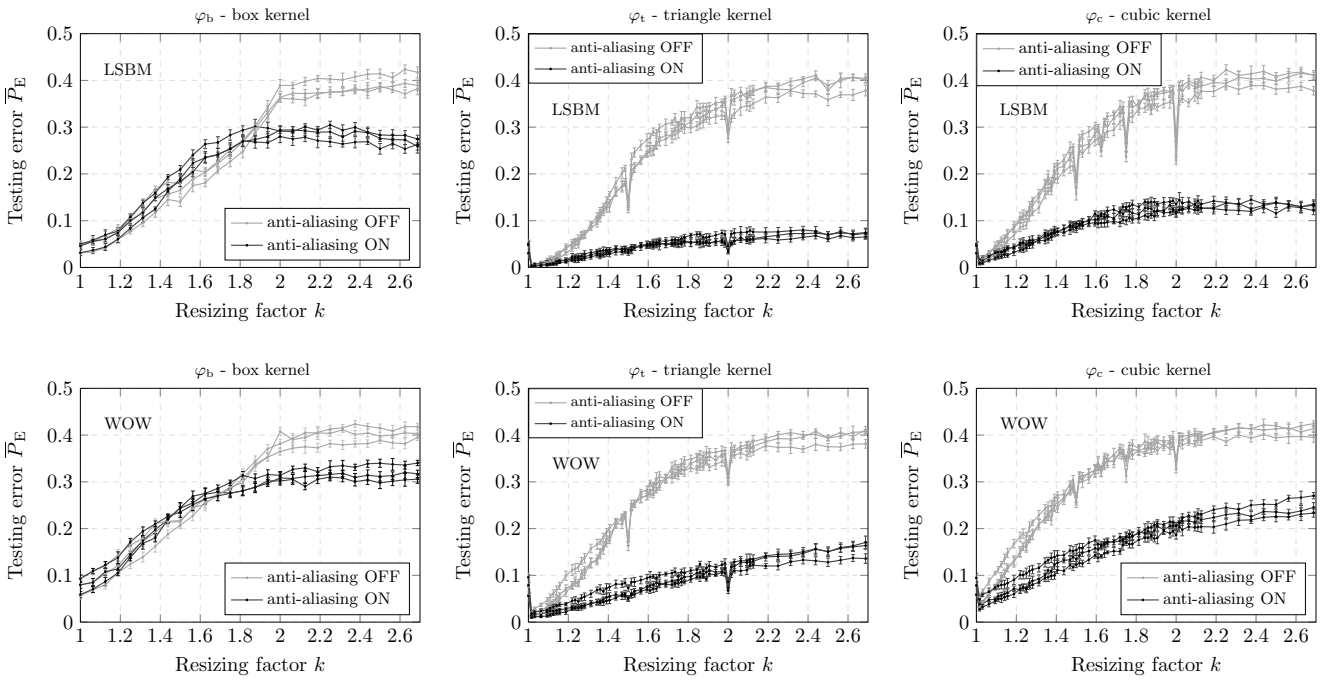

Figure 6. Steganalysis of LSBM (top) and WOW (bottom) using SRMQ1 features in images resized by factor k with anti-aliasing turned off (gray) versus on (black). The three different curves represent individual cameras (LEI, PEN, CAN).

pixels from the original grid. Since all three kernels are zero at integer values, this perfect alignment makes all the downsampled images identical.

The goal of the thorough discussion and interpretation of the results in this section was to bring more insight into the inner workings of image downsampling and its effects on steganalysis. To make the picture more complete, we repeated the same steganalysis experiment with the content-adaptive algorithm WOW. These results are shown in Figure 5. Even though WOW is based on a fundamentally different embedding paradigm, the progress of its error rates is qualitatively consistent with the non-adaptive LSB matching (compare to Figure 3).

B. Anti-aliasing

Anti-aliasing is a common image pre-processing technique whose goal is to suppress higher spatial frequencies in the image prior to resizing in order to eliminate disturbing visual artifacts around edges (for example Moiré patterns). In Matlab (and in other image processing tools as well), anti-aliasing is executed *at the same time* as resizing, simply by widening the interpolation kernel. Formally, the kernel function $\varphi(x)$ at resolution k is modified as follows

$$\varphi(x) \to \frac{1}{k}\varphi\left(\frac{x}{k}\right). \tag{13}$$

The larger the value of the resizing factor k is, the wider the support of the kernel becomes. Since this makes the resized image smoother, one can expect the steganalysis to be easier, compared to the situation with the anti-aliasing is turned off (all previous experiments). Note that it is far from obvious whether the overall effect of image *downsampling* will favor steganography or steganalysis – whether the smoothing effects will overcome the weaker dependencies due to content downsampling.

In Figure 6, we compare the results of steganalysis for the cases when the anti-aliasing is turned off and on. We performed the experiment for both the LSBM (change rate $\beta = 0.05$) and WOW (payload $\alpha = 0.3$ bpp), for the three considered kernels, and all three camera sources. This time, we opted only for the state-of-the-art steganalysis using SRMQ1 features rather than covering both feature spaces. Note that for a fixed kernel, the testing error $\overline{P}_E$ behaves consistently across both steganographic algorithms and all three cameras.

When the anti-aliasing is turned on, the width of the box kernel at the downscaling factor k is equal to k, which means that now all pixels from the original image are always utilized during interpolation. For small values of k, however, the situation is not much different from the case when the kernel width was fixed to one (anti-aliasing was turned off) because the majority of the resized pixels have the value equal to their nearest neighbor. But as the value of k increases, more and more resized pixels are created as an average of their neighbors and the obtained errors for both scenarios start to differ. For $k > 2$, *all* resized pixels are formed as averages of the neighboring pixels, which makes the downsized image smoother and consequently easier to steganalyze.

The situation is quite different for the other two kernels where the difference between both scenarios (anti-aliasing on/off) is much more profound. We attribute the stronger

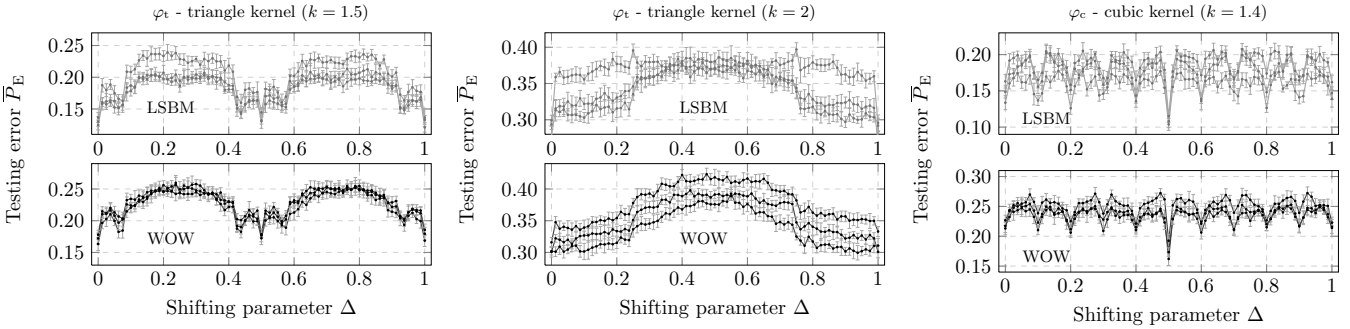

Figure 8. Steganalysis of LSBM at change rate $\beta = 0.05$ (top) and WOW at payload $\alpha = 0.3$ bpp (bottom) in resized images as a function of the grid-shifting parameter Δ. Anti-aliasing was turned off. Used feature space: SRMQ1. The three different curves represent the individual cameras (LEI, PEN, CAN).

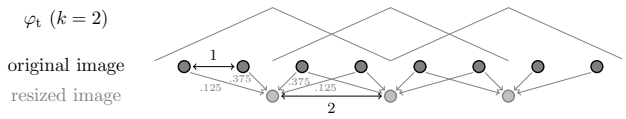

Figure 7. Effect of turning the anti-aliasing on with the triangle kernel φ_t and $k = 2$. Note the wider kernel width in comparison when the anti-aliasing is off (Fig. 4).

effect of anti-aliasing to the fact that both triangle and cubic kernel have their original support larger than one, hence with the increasing value of k the kernels get proportionally wider and their mutual overlap (not present for the box kernel) increases. We demonstrate this for the specific choice of the triangle kernel φ_t and $k = 2$ in Figure 7. As can be seen, not only is each resized pixel a combination of *four* neighboring pixels from the original resolution, but every pixel from the original image now affects *two* neighboring downsampled pixels – compare to the bottom diagram in Figure 4 where the same situation is depicted with the anti-aliasing turned off. Therefore, it should not be surprising that anti-aliasing affects wider interpolation kernels stronger.

Note that despite the strong smoothing effects of both kernels, downsampling still favors steganography even when the anti-aliasing is turned on, i.e., steganalysis gets more difficult as the value of k increases (apart from the initial drop of performance due to sudden desynchronization discussed in the previous section). In other words, for the considered range of k, the loss of interpixel dependencies due to resolution content is still stronger than the gain due to the smoothing effects. This conclusion holds for both steganographic methods and across all three tested camera sources.

C. Downsampled grid position (alignment)

For most interpolation algorithms, the position of the grid of the downsampled image is *centered* within the grid of the original image, which is achieved by setting the values of $(x_0^{(k)}, y_0^{(k)})$ according to Equation (9). As discussed in Section IV-A, this makes both grids at certain rational values of k synchronized, which strengthens the statistical value of extracted features and consequently aids steganalysis. However, some implementations of downsampling may position the downsized grid differently, for example by aligning the first pixels of both grids. As will become apparent in this section, this initial grid alignment may also have a strong effect on steganalysis.

The effect of the grid alignment depends on numerous factors, such as the downsampling factor k, the interpolation kernel, and likely even the in-camera processing, which prevent us from providing a truly comprehensive picture in this paper. Instead, we opted for demonstrating the effect of grid alignment on selected cases to give the reader an idea of the extent to which steganalysis performance can vary in practice.

We start by introducing a kernel shift parameter $0 \leq \Delta < 1$ into (9):

$$x_0^{(k)} = y_0^{(k)} = (k+1)/2 + \Delta, \tag{14}$$

Setting $\Delta = 0$ trivially leads to the original formula (9) while $\Delta = 1$ corresponds to the grid shifted by one pixel with respect to the original grid.

We start our investigation by steganalyzing LSBM and WOW in images resized by the triangle kernel φ_t at $k = 1.5$. Recall that this setup corresponds to a sharp drop in the testing error due to sudden synchronization of both grids when $\Delta = 0$, see Figure 3 (middle). Everything else being equal, we repeated this experiment with different choices of the shift Δ. The resulting error rates, again for all three camera sources, are shown in Figure 8 (left). Similarly as in the previous section, we restricted ourselves to the state-of-the-art rich features rather than covering both feature spaces.

First, note the two "lobes" symmetrical along $\Delta = 0.25$ and $\Delta = 0.75$. This is due to the fact that the shifts Δ and $\Delta' = 0.5 - \Delta$ result in the same convex combinations during the interpolation, which is caused by the fact that the distance between two neighboring pixels after downsizing is 1.5, see Figure 4 (top).

As $\Delta > 0$ increases, the images at both resolutions become desynchronized which weakens steganalysis performance and causes the detection error to increase. For the

mid-range shifts $0.1 \lesssim \Delta \lesssim 0.3$ (full desynchronization), the testing errors $\overline{P}_{\mathrm{E}}$ are back at the values as if there was no drop in Figure 3.

In the second experiment, we fix $k = 2$. The results are shown in Figure 8 (middle). Since the distance between two resized pixels is now 2, the symmetry is only along $\Delta = 0.5$, and the shifts Δ and $\Delta' = 1 - \Delta$ yield the same interpolation formulas. A quick glance at Figure 4 (bottom) reveals that for $k = 2$ changing the shift Δ does not affect the "degree of synchronization" between the two grids but rather changes the values of linear coefficients during interpolation (the weighted average) – it determines the weight of "odd pixels" and "even pixels" from the original image. For $\Delta = 0.5$, the triangle kernels are perfectly centered at even pixels of the original image and thus the downsampled image is just a sub-sampled version of the original image (and identical to the one we would obtain with the box kernel). Since every other pixel is skipped, the pixels of the downsampled image are less correlated, and we see higher error rates. For $\Delta = 0$, on the other hand, the kernels are positioned exactly in between of the pixels from the original grid and thus the downsampled pixels are averages of the corresponding neighboring pixels, which strengthens dependencies and helps steganalysis.

Finally, in addition to the previous two experiments, in Figure 8 (right) we show the results of steganalysis with the cubic kernel and the resizing factor fixed at $k = 1.4$. In this case, the interference between both grids oscillates with a higher frequency than in the previous two scenarios resulting in a periodic progress of the testing error as a function of the kernel shift Δ, with peaks corresponding to desynchronized grids and valleys to the (to the certain degree) synchronized grids. Similarly to the previous two cases, all three camera models exhibit qualitatively similar behavior, even though the actual error rates among them can differ by 10% or more.[11] The observed trends seem to be robust w.r.t. the choice of the steganographic method.

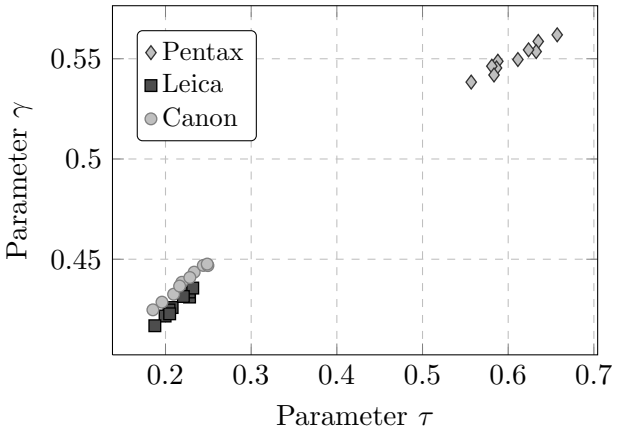

Figure 9. Parameters τ and γ estimated using 500 images from three camera models.

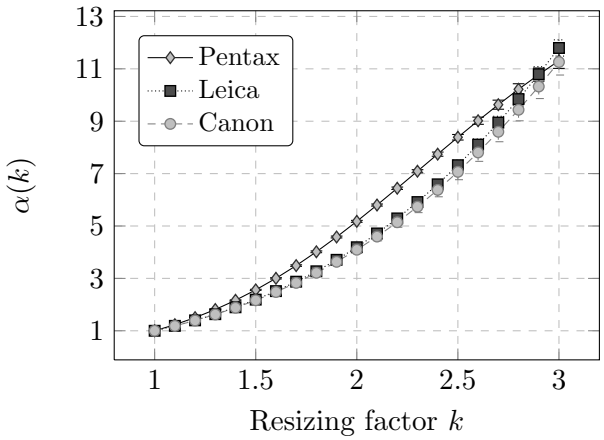

Figure 10. Experimental scaling law derived from the estimated parameters τ and γ. The error bars represent the standard deviation of the obtained values over the ten estimations of τ, γ shown in Figure 9.

V. Secure payload scaling w.r.t. image resolution

A. Image model

In this section we study the effects of image resizing on steganalysis analytically. To this end, we need to select a cover model that considers dependencies among neighboring pixels. One of the simplest models is the first-order Markov chain (MC). A substantial advantage of this model is that there exists a closed-form expression between the cover MC and the stego hidden MC for any mutually-independent embedding operation [4], which will allow us to compute the scaling of secure payload that leads to constant statistical detectability across the downsampling factor. The disadvantage of this cover model is that we are now limited to the box kernel only since downsampling with other kernels does not preserve the Markovian property.

The MC model is fully characterized by its transition probability matrix (TPM) $\mathbf{A} = (a_{ij})$, $i, j \in \mathcal{I} = \{0, \ldots, 255\}$. Following [4], we adopt the exponential cover model,

$$a_{ij} = 1/Z_i \exp(-(|i - j|/\tau))^{\gamma}, \tag{15}$$

with the parameters τ and γ estimated using the method of moments [22]. Figure 9 shows the results of the parameter estimation using 500 randomly selected images from individual camera models. To show the sensitivity to the image content, the estimation was repeated ten times for different sets of 500 images. The estimated parameters form three distinct clusters corresponding to the individual cameras, Leica and Canon being somewhat similar.

Resizing images with the box kernel by factor $k \geq 1$, $k \in \mathbb{R}$, changes the TPM $\mathbf{A} \to \mathbf{A}^k$, where the matrix

[11] This should not be surprising as the three sources of images differ in their original resolution and in-camera processing.

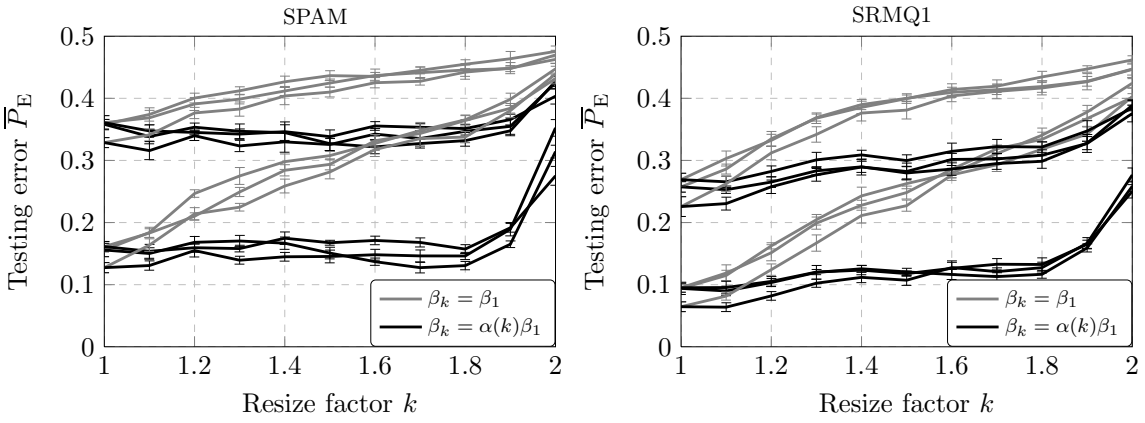

Figure 11. Scaling $\beta_k = \alpha(k)\beta_1$ (black) vs. constant change rate β_1 (gray). Algorithm: LSBM; Features: SPAM (left), SRMQ1 (right); features. Individual lines correspond to three different camera models.

power is defined in a generalized sense and can be evaluated via the eigenvalue decomposition for non-integer k. This allows us to study the scaling effects of image resizing on steganographic security solely based on the statistical properties of the original cover source (*non-resized* images).

B. Scaling factor $\alpha(k)$

It is well-known that the leading term of the KL divergence between cover and stego objects is quadratic in the change rate β [20]:

$$D(k;\beta) = \frac{1}{2}n\beta^2 I(k), \tag{16}$$

where n is the cover size and $I(k)$ is the steganographic Fisher information (FI) rate for the resizing factor k. For a fixed cover source described by its TPM and a fixed steganographic algorithm (LSBM in our case), the authors of [4] derived a closed-form expression for $I(k)$ (see Theorem 2 in [4]), from which one can obtain $D(k;\beta)$ at different resolutions (as a function of $\mathbf{A}^k$).

To obtain a constant level of statistical detectability (KL divergence D) after resizing by factor k, the change rate β needs to be scaled by $\alpha(k)$ satisfying

$$D(1;\beta) = D(k;\alpha(k)\beta). \tag{17}$$

Since we always central-crop the image after resizing to the same size n (and thus eliminate the effect of the SRL) it is easy to see that

$$\alpha(k) = \sqrt{I(1)/I(k)}. \tag{18}$$

In Figure 10, we show the computed values of the derived scaling factor $\alpha(k)$ for a range of resolutions $1 \le k \le 3$ and for all three considered camera models. The obtained experimental "scaling laws" are stable over different estimates of parameters τ and γ over all tested camera models and steganalysis features. At the same time, we can clearly see the differences among individual cameras which indicates that the scaling law sensitively depends on the cover source.

C. Experimental verification of the scaling law

The theoretically obtained results were verified in the following manner. For a fixed change rate β_1, we first steganalyzed LSBM[12]using both SPAM and SRMQ1 features across different scaling factors $1 \le k \le 2$. This is the situation when the change rate is *constant* and does *not* follow the scaling law. Next, according to the theory, in order to keep the same level of statistical detectability (the same error $\overline{P}_{\mathrm{E}}$), the change rate needs to be scaled as $\alpha(k)\beta_1$, provided the images at both resolutions are cropped to the same dimensions (otherwise, another change rate adjustment due to the SRL would be needed). We also steganalyzed LSBM across the same range of scaling factors, $1 \le k \le 2$, while following the derived law.

In Figure 11, we compare the progress of the error rate $\overline{P}_{\mathrm{E}}$ under both scenarios described in the above paragraph. When the change rate is kept constant, with an increasing value of k the testing error increases. When the change rate follows the derived scaling law, however, we can see that the testing error remains approximately constant up to the value $k \approx 1.8$ after which it starts deviating. This behavior is consistent across all tested camera models.

In Figure 12 we test the validity of the derived law from a different perspective. For a fixed resolution k, we now vary the change rate $\beta = \{0.005, \ldots, 0.035\}$, and compare the error obtained at the non-resized images with the error obtained at resolution k with the change rate scaled as $\alpha(k)\beta$. If the derived scaling law held perfectly, the points in the graph would lie on the diagonal. We can see that this is the case of resolutions $k = 1.2$, however, as k increases, the scaling law slowly ceases to hold which is more pronounced for the rich features. These results are consistent with the previous experiment and with the experiments presented in [18].

[12]Since WOW is a content-adaptive algorithm, its embedding operation is not applied to pixels in a mutually independent fashion, which makes our analysis of the scaling law inapplicable to such an algorithm.

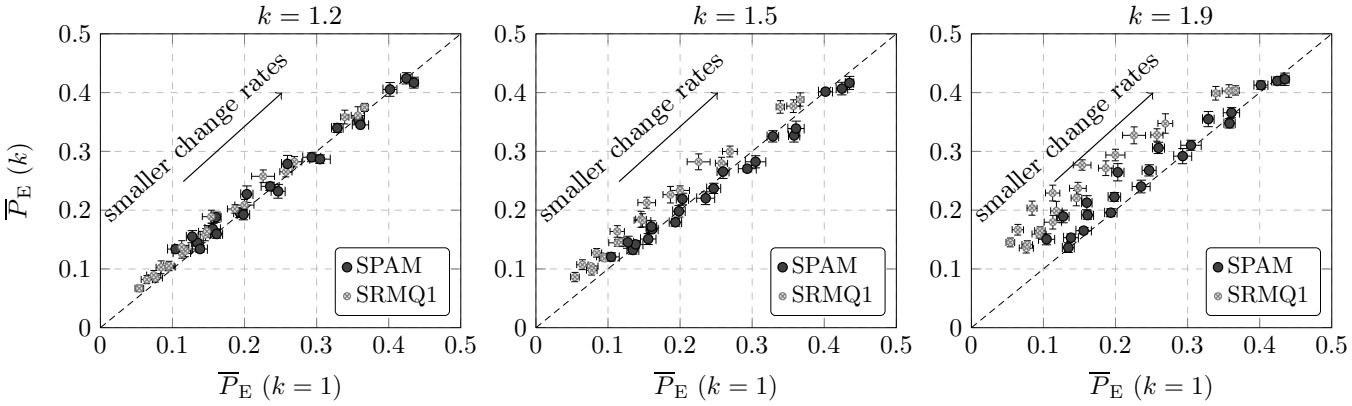

Figure 12. Predicted vs. measured error rate for three different resizing factors and two steganalysis feature spaces. The individual points come from three different camera models and a range of different change rates.

Despite the simplicity of the cover model (15) and the fact that we use a machine-learning based detector, a good match between theory and practice is observed for smaller values of k. The formula (18) does not seem to provide the correct scaling for larger k, especially when $k \gtrsim 1.8$. This is likely due to the limited extent the Markov chain model describes natural images. Moreover, we remind that the scaling was derived under the assumption of small change rates, which becomes violated for large values of k.

VI. Conclusions

Today, downscaled images are ubiquitous. They appear, for example, on image-sharing portals, social networking websites, in e-mail attachments, at news websites, and in on-line stores. It is thus natural to assume that such images will be used for steganography in practice. Since downsampling changes the strength and character of dependencies among adjacent image pixels, it also affects steganalysis. The lower image resolution *decreases* the strength of pixel dependencies due to more rapid changes in the image content. Depending on the image downsampling algorithm, on the other hand, the strength of pixel dependencies may *increase* due to interpolation (averaging). Consequently, it is generally rather difficult to predict whether steganalysis will be easier or more difficult in images created by a particular resizing algorithm and with a particular choice of its parameters. The main contribution of this paper is explaining how and why empirical steganographic security varies with the downsizing algorithm and its settings.

We study the effect of the downsampling factor, the interpolation kernel, anti-aliasing, and the position of the resampled grid on the empirical steganographic security. The universality of our conclusions is supported with experiments on three different camera sources, two qualitatively different steganographic algorithms, LSBM and WOW, and two different steganalyzers built using SPAM and the state-of-the-art spatial rich model (SRM).

We also describe a new form of a scaling law, which expresses how one should scale the payload with respect to image resolution to keep a constant statistical detectability. Note that while downsampling changes the number of pixels, it also changes their statistical properties, and thus the scaling of the secure payload w.r.t. downsampling is not covered by the square root law of imperfect steganography. To derive the proper scaling law analytically, we adopted the first-order Markov chain for image pixels and made an assumption that the embedding is realized using a mutually independent operation. This restricted our study to algorithms that are not content-adaptive (LSBM) and the box kernel, which is the only kernel that preserves the Markovian property of the cover source under resizing. The general validity of the results (for small values of the resize factor k) is supported with experiments on three different sources and two steganalysis features (SPAM and SRM).

The work in this paper is of interest to both practitioners and researchers. Practitioners need to be informed of the potential strong effect image downscaling may have on the security of their secretly embedded messages. Researchers need to be aware of this effect in order to supply all necessary details pertaining to preparing their image databases for tests. Last but not least, the detection accuracy on BOSSbase (or other databases containing resized images) should not be perceived as an *absolute measure* of a steganographic algorithm's security.

References

[1] P. Bas, T. Filler, and T. Pevný. Break our steganographic system – the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, volume 6958 of Lecture Notes in Computer Science, pages 59–70, Prague, Czech Republic, May 18–20, 2011.

[2] R. Böhme. Weighted stego-image steganalysis for JPEG covers. In K. Solanki, K. Sullivan, and U. Madhow, editors, *Information Hiding, 10th International Workshop*, volume 5284 of Lecture Notes in Computer Science, pages 178–194, Santa Barbara, CA, June 19–21, 2007. Springer-Verlag, New York.

[3] G. Cancelli, G. Doërr, I. J. Cox, and M. Barni. A comparative study of ±1 steganalyzers. In *Proceedings IEEE International Workshop on Multimedia Signal Processing*, pages 791–796, Cairns, Australia, October 8–10, 2008.

[4] 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 Workshop*, volume 5806 of Lecture Notes in Computer Science, pages 31–47, Darmstadt, Germany, June 7–10, 2009. Springer-Verlag, New York.

[5] 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, Media Forensics and Security*, volume 7254, pages 08 1–11, San Jose, CA, January 18–21, 2009.

[6] J. Fridrich. *Steganography in Digital Media: Principles, Algorithms, and Applications*. Cambridge University Press, 2009.

[7] J. Fridrich. Effect of cover quantization on steganographic Fisher information. *IEEE Transactions on Information Forensics and Security*, 8(2):361–373, February 2013.

[8] J. Fridrich, M. Goljan, and R. Du. Steganalysis based on JPEG compatibility. In A. G. Tescher, editor, *Special Session on Theoretical and Practical Issues in Digital Watermarking and Data Hiding, SPIE Multimedia Systems and Applications IV*, volume 4518, pages 275–280, Denver, CO, August 20–24, 2001.

[9] J. Fridrich and J. Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, June 2012.

[10] J. Fridrich and J. Kodovský. Steganalysis of LSB replacement using parity-aware features. In M. Kirchner and D. Ghosal, editors, *Information Hiding, 14th International Workshop*, volume 7692 of Lecture Notes in Computer Science, pages 31–45, Berkeley, CA, May 15–18, 2012.

[11] A. S. Glasser. *Graphics Gems*. Morgan Kaufman, 1990.

[12] 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.

[13] V. Holub and J. Fridrich. Optimizing pixel predictors for steganalysis. In A. Alattar, N. D. Memon, and E. J. Delp, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2012*, volume 8303, pages 09 1–13, San Francisco, CA, January 22–26, 2012.

[14] 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 1–5 17, San Jose, CA, January 27–31, 2008.

[15] 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.

[16] R. G. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29(6):1153–1160, December 1981.

[17] J. Kodovský and J. Fridrich. JPEG-compatibility steganalysis using block-histogram of recompression artifacts. In M. Kirchner and D. Ghosal, editors, *Information Hiding, 14th International Workshop*, volume 7692 of Lecture Notes in Computer Science, pages 78–93, Berkeley, CA, May 15–18, 2012.

[18] J. Kodovský and J. Fridrich. Steganalysis in resized images. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 26–31, 2013.

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

[20] S. Kullback. *Information theory and statistics*. John Wiley and Sons., New York, 1959.

[21] W. Luo, Y. Wang, and J. Huang. Security analysis on spatial ± 1 steganography for JPEG decompressed images. *IEEE Signal Processing Letters*, 18(1):39–42, 2011.

[22] S. Meignen and H. Meignen. On the modeling of small sample distributions with generalized gaussian density in a maximum likelihood framework. *IEEE Transactions on Image Processing*, 15(6):1647–1652, 2006.

[23] 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.

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

[25] M. C. Stamm. *Digital Multimedia Forensics and Anti-Forensics*. PhD thesis, Dept. of Electrical and Computer Engineering, University of Maryland, College Park, MD, 2012.

Jessica Fridrich holds the position of Professor of Electrical and Computer Engineering at Binghamton University (SUNY). She has received her PhD in Systems Science from Binghamton University in 1995 and MS in Applied Mathematics from Czech Technical University in Prague in 1987. Her main interests are in steganography, steganalysis, digital watermarking, and digital image forensic. Dr. Fridrich's research work has been generously supported by the US Air Force and AFOSR. Since 1995, she received 19 research grants totaling over $9 mil for projects on data embedding and steganalysis that lead to more than 160 papers and 7 US patents. Dr. Fridrich is a member of IEEE and ACM.

Jan Kodovský holds the position of Software Engineer at Facebook, Inc. He received his Ph.D. degree in Electrical Engineering from Binghamton University in 2012 and M.S. degree in Mathematical Modeling from the Czech Technical University in Prague in 2006. His professional interests include steganalysis, steganography, and applied machine learning. The work performed by Jan Kodovský on this paper has been done while he was a Postdoctoral Associate at Binghamton University.