Revisiting Weighted Stego-Image Steganalysis

Andrew D. Ker[a] and Rainer Böhme[b]

[a]Oxford University Computing Laboratory, Parks Road, Oxford OX1 3QD, England;
[b]Technische Universität Dresden, Institute for System Architecture, 01062 Dresden, Germany

ABSTRACT

This paper revisits the steganalysis method involving a Weighted Stego-Image (WS) for estimating LSB replacement payload sizes in digital images. It suggests new WS estimators, upgrading the method's three components: cover pixel prediction, least-squares weighting, and bias correction. Wide-ranging experimental results (over two million total attacks) based on images from multiple sources and pre-processing histories show that the new methods produce greatly improved accuracy, to the extent that they outperform even the best of the structural detectors, while avoiding their high complexity. Furthermore, specialised WS estimators can be derived for detection of sequentially-placed payload: they offer levels of accuracy orders of magnitude better than their competitors.

Keywords: Steganalysis, Weighted Stego-Image, LSB Replacement, Benchmarking, Sequential Embedding

1. INTRODUCTION

One amongst many steganalysers for Least Significant Bit (LSB) replacement in digital images, the method involving a Weighted Stego-Image,[1] commonly identified by the acronym WS, has received little attention beyond a simple extension to multiple bit replacement.[2] This is probably because its performance is weaker than the state-of-the-art detectors based on analysis of structure.[3–7] Consequently, WS has not profited from as many refinements as the class of structural detectors. The standard WS method is not without advantages: it does not use the same pixel group analysis as the structural detectors and its performance is best precisely when the structural steganalysers are weakest (estimation of payload size for near-maximal messages[8]). Furthermore, unlike the structural detectors it retains its estimation accuracy when embedding changes are not distributed evenly over the cover.[9]

The aim of this paper is to reconsider the WS method, first recapitulating the method of Ref. 1 (Sect. 2) then offering improvements to each component of the WS procedure (Sect. 3), and finally extending the same techniques to produce specialised detection and size estimation of sequentially- instead of randomly-located payload (Sect. 4). The improved methods are benchmarked thoroughly (Sect. 5), first to determine which of the novel variants is the best performer, and then to compare with their leading payload estimation competitors. Improvements to the basic WS method show notable performance gains, outperforming the best structural detectors in a domain where they were previously believed more reliable, and sequential WS-based payload estimators will be demonstrated to be spectacularly accurate.

2. WEIGHTED STEGO-IMAGE DETECTORS

We begin with a brief exposition of the WS method described in Ref. 1, simplifying some of the terminology, followed by a short examination of its performance when compared with structural steganalysis methods.

The aim of the WS method is to estimate the size of payload, possibly zero, embedded by LSB replacement in a digital image. In this it is similar to most of the other sensitive detectors of LSB replacement.[3–7] Suppose that a cover image consists of a vector of N samples (e.g. pixel intensities in a single-channel image) $\boldsymbol{c} = (c_1, \ldots, c_N)$ and that a payload of length $M \leq N$ bits is embedded by LSB replacement. We suppose that the payload

Further author information:
A. D. Ker: E-mail: adk@comlab.ox.ac.uk, Telephone: +44 1865 283530
R. Böhme: E-mail: rainer.boehme@tu-dresden.de, Telephone: +49 351 463 38370

is uncorrelated with the cover (if the embedding locations are generated pseudorandomly, or if the payload is compressed or encrypted before embedding, this is likely to be the case). We write $\boldsymbol{s} = (s_1, \ldots, s_N)$ for the stego image and, for $\alpha \in [0, 1]$, $\boldsymbol{s}^\alpha$ for the real-valued sequence formed by taking a weighted average between the stego image and the stego image with every sample's LSB flipped:

$$s_i^\alpha = \alpha \overline{s_i} + (1 - \alpha) s_i, \tag{1}$$

where $\overline{x}$ denotes the nonnegative integer x with the least significant bit flipped, $\overline{x} = x + (-1)^x$. The sequence $\boldsymbol{s}^\alpha$ is called the *weighted stego-image* and the key to the WS payload estimator is Theorem 1 of Ref. 1, which demonstrates that the weighted stego image is closest to the cover image (if difference between the two vectors $\boldsymbol{c}$ and $\boldsymbol{s}^\alpha$ is measured using the Euclidean L^2-norm) when $\alpha = M/2N$. This corresponds to the expectation that proportion $M/2N$ of the cover pixels are flipped when embedding a payload of length M.

Of course, in steganalysis we do not have access to the cover as well as the stego object, so we cannot find the value of α which exactly minimizes distance from $\boldsymbol{c}$, but the WS method attempts to *estimate* the cover image by filtering the stego image. We will write $\mathcal{F}(\boldsymbol{s})$ for the filtered stego image: in Ref. 1 each cover sample is estimated by taking the mean of the surrounding four stego pixels. It is demonstrated that, for correctness of the WS method, each cover pixel estimate $\mathcal{F}(\boldsymbol{s})_i$ must not depend on the corresponding stego pixel s_i, only on other stego pixels.

Thus we can describe the standard WS method. Given an image which might be a cover or stego image, the payload size (possibly zero) is estimated by finding α to minimize the distance

$$D(\alpha) = \sum_{i=1}^{N} \big(s_i^\alpha - \mathcal{F}(\boldsymbol{s})_i\big)^2.$$

Differentiating D, it is shown that the proportionate payload length $p = M/N$ is estimated by

$$\hat{p} = 2 \operatorname*{argmin}_\alpha D(\alpha) = \frac{2}{N} \sum_{i=1}^{N} \big(s_i - \mathcal{F}(\boldsymbol{s})_i\big)\big(s_i - \overline{s_i}\big).$$

This can be computed in $O(N)$ operations, i.e. the estimation complexity is linear in the image size.

In Ref. 1, two enhancements are included. First, the distance between weighted stego and cover can itself be weighted so that pixels in noisy areas – for which prediction of the cover is more difficult – are given less weight than those in flatter areas. The distance minimized is $\sum w_i (s_i^\alpha - \mathcal{F}(\boldsymbol{s})_i)^2$. It is suggested that the weights should be proportional to $1/(1 + \sigma_i^2)$, where σ_i^2 is the local variance of the 4 pixels surrounding stego pixel i. The estimator still has a closed form, now

$$\hat{p} = 2 \operatorname*{argmin}_\alpha \sum_{i=1}^{N} w_i \big(s_i^\alpha - \mathcal{F}(\boldsymbol{s})_i\big)^2 = 2 \sum_{i=1}^{N} w_i (s_i - \mathcal{F}(\boldsymbol{s})_i)(s_i - \overline{s_i}), \tag{2}$$

where the weight vector $\boldsymbol{w}$ is scaled so that $\sum w_i = 1$. (No weighting corresponds to $w_i = 1/N$ for each i.) Ref. 1 demonstrates that this improves accuracy in some test images. The estimate is still computed in linear time.

Second, some outliers are analysed and it is demonstrated that "flat pixels" (areas of uniformity) in the cover are to blame: such areas receive a high weighting, and furthermore they create a positive bias in the estimator. A "flat pixel correction" term is derived, with the aim of reducing outliers. We will not repeat its derivation.

2.1. Performance

As a starting point and motivation for the modifications presented in this paper, we include a summary of performance, comparing the WS methods in Ref. 1 with one of the earliest structural estimators, *SPA*,[3] and the most recent refinement, *Couples/ML*.[7] For proportionate payload size $p = 0, 0.05, \ldots, 1$, we simulated randomly-spread LSB replacement in a set of 3,000 scanned greyscale cover images and measured the accuracy

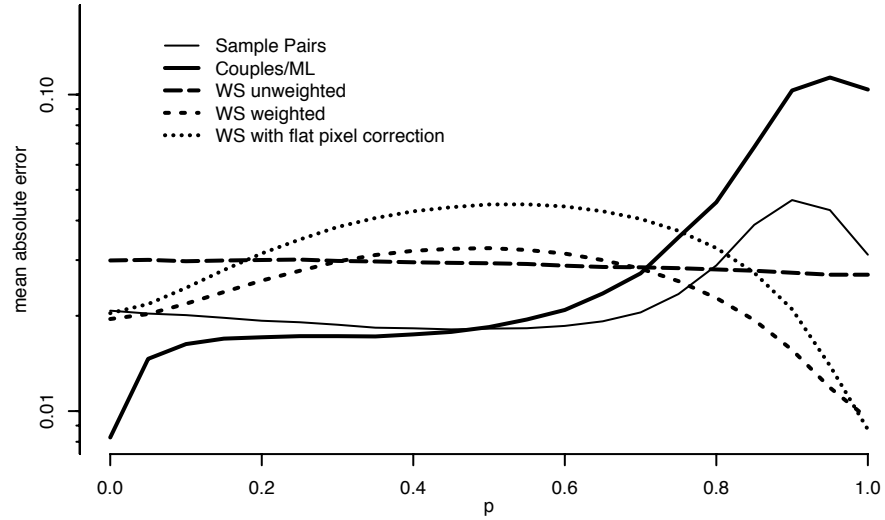


Figure 1. Mean absolute error of structural and WS estimators described in Ref. 1, observed in 3,000 scanned greyscale images, as a function of the true embedding rate. The y-axis is on a log scale. Lower values denote better performance.

of the ensuing estimates. There is no uniformly best measure of estimator performance,[8] but we display mean absolute error (MAE) – a reasonable overall measure – in Fig. 1.*

Observe that weighting usually improves the performance of the WS method, and that flat pixel correction only reduces performance. The structural detectors appear generally superior, but their performance is weaker for larger payloads. (This seems to be a feature of structural detectors, for reasons explained in Ref. 10.) The weighted WS method is the most accurate estimator for proportionate payloads of over about 75 % and its performance is also quite good for very small payloads, although not enough to match the very sensitive (and computationally expensive) Couples/ML method. Therefore we conclude that the WS method does have a place in steganalysis, and it is valuable to study it further.

3. IMPROVED WS DETECTORS

We propose improvements to all parts of the WS method – the cover pixel estimation, the weighting factors for WS image fitting – as well as a substantially more general bias correction to replace the "flat pixel correction" term from Ref. 1. Each component will be considered separately.

3.1. Enhanced Pixel Predictors

First, can we make a better predictor for the cover pixels than the very simple one in Ref. 1? Taking the average of four surrounding pixels amounts to computing the convolution of the stego image with the two-dimensional filter in (3), so it is natural to consider other filters, for example (4). However, this predictor based on eight neighbours has poor experimental performance.

$$\begin{matrix} 0 & \frac{1}{4} & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{matrix} \quad (3) \qquad \begin{matrix} \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{8} & 0 & \frac{1}{8} \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \end{matrix} \quad (4) \qquad \begin{matrix} -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ \frac{1}{2} & 0 & \frac{1}{2} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \end{matrix} \quad (5) \qquad \begin{matrix} b & a & b \\ a & 0 & a \\ b & a & b \end{matrix} \quad (6) \qquad \begin{matrix} e & d & c & d & e \\ d & b & a & b & d \\ c & a & 0 & a & c \\ d & b & a & b & d \\ e & d & c & d & e \end{matrix} \quad (7)$$

*An important factor, not visible in the figure, is that the SPA estimator can fail to produce an estimate at all. Such failures were excluded from the MAE calculation, so the figure flatters the SPA detector a little. However the occurrence of such failures is limited to large payloads: 50 % of estimates fail for maximal payloads $p = 1$, but the rate reduces rapidly for smaller payloads. In all our tests, there were zero failures for $p \leq 0.6$ and only a 2 % failure rate at $p = 0.8$.

In order to find a more accurate predictor, we first observed that the ideal filter should have symmetry, to reflect invariance of natural image properties, and then tested those of the form (6) on a library of cover images. We found the values of a and b to minimize the L^2 distance between the predicted cover and the true cover: rounding the numbers slightly, the optimum mask of this shape appears to be given by (5). All four filters: (3), (4), (5), and also a 5×5 convolution filter determined using the same method (not displayed) will be tested in Sect. 5.

But the optimal predictor might depend on the particular image being analysed so, inspired by the technique used to find a good fixed filter, we propose an adaptive method. For a given stego image, a filter of the form (6) can be applied to determine the values of a and b which best predicts its pixels; it is sensible to use a weighted least-squares regression here. For that image, this gives an optimal predictor of *stego image* pixels, and we expect that it is also a good predictor of cover image pixels because the LSB replacement process should cause errors of ± 1, in the pixels used for prediction as well as that predicted, approximately equally often. Then, the standard (weighted or unweighted) WS method is applied with prediction by the learned filter. A similar technique can be applied to 5×5 filters, using the symmetrical pattern in (7). With an adaptive filter the computation time is approximately doubled, as an initial pass through the image is required to determine the filter. WS based on adaptive 3×3 and 5×5 filters will be benchmarked in Sect. 5.

3.2. Enhanced Weighting Factors

Weighting the distance between stego and cover improves the accuracy of the detector considerably. However, the choice of good weights is not very well understood and we offer only a heuristic improvement. Ref. 1 proposes that the weights should be of the form (8).

$$w_i \propto \frac{1}{1+\sigma_i^2} \qquad (8) \qquad\qquad w_i \propto \frac{1}{5+\sigma_i^2} \qquad (9)$$

Our experiments suggested that, although these weights produce a low weighted MSE between predicted cover and true cover, they over-emphasise flatter areas in the image too much: in areas of low noise, the stego noise forms a significant part of the predictor error. Instead, w_i should be moderated so that higher weights are reduced: weights of the form (9) give better payload estimators, and will be benchmarked in Sect. 5.

3.3. Bias Correction

We have demonstrated that the "flat pixel correction", derived in Ref. 1, does not improve performance in (some) images. However systematic bias is a genuine feature of the WS method in some covers, and we seek a better way to correct it. Our analysis begins like that in Ref. 1 but then diverges, producing a bias correction term which is both simpler and better justified than the original.

To reason about bias, let us consider the expected payload estimate under the (weighted) WS method:

$$\mathrm{E}[\hat{p}] = 2\sum_{i=1}^{N} w_i \mathrm{E}\Big[\big(s_i - \mathcal{F}(\boldsymbol{s})_i\big)(s_i - \overline{s_i})\Big] = 2\sum_{i=1}^{N} w_i \mathrm{E}\Big[\Big(\big(s_i - c_i\big) + \big(c_i - \mathcal{F}(\boldsymbol{c})_i\big) + \big(\mathcal{F}(\boldsymbol{c})_i - \mathcal{F}(\boldsymbol{s})_i\big)\Big)(s_i - \overline{s_i})\Big]. \quad (10)$$

This breaks down into three parts: the true relative payload size, bias due to inaccurate prediction, and bias arising from the filtered stego noise.

In the first term of (10), $\mathrm{E}\big[(s_i - c_i)(s_i - \overline{s_i})\big] = p/2$ because proportion $p/2$ of c_i are equal to $\overline{s_i}$ and the rest are equal to s_i, so the contribution of $2\sum w_i \mathrm{E}\big[(s_i - c_i)(s_i - \overline{s_i})\big]$ is $p\sum w_i = p$, the correct estimate. In the second term, $\mathrm{E}\Big[\big(c_i - \mathcal{F}(\boldsymbol{c})_i\big)(s_i - \overline{s_i})\Big] = 0$ if the predictor error (in covers) is uncorrelated with the parity of the stego pixels (a reasonable assumption), so this contributes no bias. But in the third term

$$\mathrm{E}\Big[\big(\mathcal{F}(\boldsymbol{c})_i - \mathcal{F}(\boldsymbol{s})_i\big)(s_i - \overline{s_i})\Big] = \mathrm{E}\Big[\big(\mathcal{F}(\boldsymbol{c} - \boldsymbol{s})_i\big)(s_i - \overline{s_i})\Big]$$

(assuming a linear filter) is only zero if the filtered added stego signal $\boldsymbol{s} - \boldsymbol{c}$ is uncorrelated with the corresponding stego pixel. This is false if there is parity co-occurrence between neighbours in the cover: suppose that the pixel

c_i is even, and so are many of its neighbours. Then $\boldsymbol{c}-\boldsymbol{s}$ will expect more negative values than positive values near pixel i, so the same will be true for $\mathcal{F}(\boldsymbol{c}-\boldsymbol{s})_i$.

To quantify this bias term, we imagine that it is the *stego* image which is fixed and the *cover* which was generated by randomly flipping proportion $p/2$ of LSBs. (This disregards certain conditional probabilities in the structure of the cover, but that is not very significant.) Then we can simplify the expected bias, the third part of (10), to

$$b = 2\sum w_i(s_i - \overline{s_i})\ \mathrm{E}\big[\mathcal{F}(\boldsymbol{c}-\boldsymbol{s})_i\big] = p\sum w_i(s_i-\overline{s_i})(\mathcal{F}(\overline{\boldsymbol{s}}-\boldsymbol{s})_i) \tag{11}$$

since the filter is linear, and $(\boldsymbol{c}-\boldsymbol{s})_i$ is $(\overline{\boldsymbol{s}}-\boldsymbol{s})_i$ with probability $p/2$ and zero otherwise. Given an initial estimate of p, one can subtract the expected bias b to make the estimate more accurate. We will see, in Sect. 5, that this makes a substantial improvement to the accuracy of the estimator in covers where there is strong parity co-occurrence between neighbouring pixels in the cover image.

4. SPECIALIZING THE WS METHOD FOR SEQUENTIAL EMBEDDING

Most of the sensitive detectors for LSB replacement assume that the embedding changes are spread uniformly through the cover. When this is not the case, they fail. The WS method has two advantages here: first, it works with approximately equal accuracy on sequential embedding as spread embedding; second, it can easily be adapted to specialised and very highly sensitive detection of sequential embedding. Unlike in prior art,[9] we consider two types of sequential embedding: payload overwriting LSBs at the start of the cover which we call *initial sequential embedding*, and payload overwriting a sequence of LSBs starting at some other point, *arbitrary sequential embedding*.

4.1. Initial Sequential Embedding

Consider the weighted stego image (1). When the payload is located in the first M samples of the stego object we should fix $\alpha = \frac{1}{2}$ for the first M samples and $\alpha = 0$ for the rest. As with Theorem 1 of Ref. 1, the function

$$\sum_{i=1}^{m}\big(\tfrac{1}{2}(s_i+\overline{s_i})-c_i\big)^2 + \sum_{i=m+1}^{N}(s_i-c_i)^2$$

is minimized in expectation at $m=M$. Following the WS method, we can estimate the cover image by filtering the stego image and finding m to minimize

$$E(m) = \sum_{i=1}^{m}\big(\tfrac{1}{2}(s_i+\overline{s_i})-\mathcal{F}(\boldsymbol{s})_i\big)^2 + \sum_{i=m+1}^{N}\big(s_i-\mathcal{F}(\boldsymbol{s})_i\big)^2. \tag{12}$$

However, we cannot continue by differentiating E, since its derivative has no closed form and in any case the function can have multiple minima. All we can do is find the location of the minimum by trying all the values of m. Naively, computing the sum (12) for each $m = 0,\ldots,N$ would require $O(N^2)$ operations, but we can still achieve a linear time algorithm by noting that the linear recurrence

$$e_0 = 0, \quad e_m = e_{m-1} + \big(\tfrac{1}{2}(s_m+\overline{s_m})-\mathcal{F}(\boldsymbol{s})_m\big)^2 - \big(s_m-\mathcal{F}(\boldsymbol{s})_m\big)^2$$

generates $e_m = E(m) - \sum_{i=1}^{N}\big(s_i-\mathcal{F}(\boldsymbol{s})_i\big)^2$; the minimum term of e_m therefore gives the minimum of $E(m)$, and we require only linear time to generate and examine the sequence e_m. We can also apply the new pixel predictors and weighting (but not bias correction) suggested in Sect. 3 to improve the accuracy of the estimate.

4.2. Arbitrary Sequential Embedding

Finally, suppose payload embedded in consecutive samples $L, L+1, \ldots, M$. Again the WS method can be adapted, fixing $\alpha = \frac{1}{2}$ for these samples and $\alpha = 0$ elsewhere. We have that the function of l and m

$$\sum_{i=1}^{l-1}(s_i - c_i)^2 + \sum_{i=l}^{m}\left(\tfrac{1}{2}(s_i + \overline{s_i}) - c_i\right)^2 + \sum_{i=m+1}^{N}(s_i - c_i)^2$$

is minimized at $l = L, m = M$. Similarly to initial sequence embedding, we can proceed to an estimator for L and M by finding the location of the minimum of

$$E(l,m) = \sum_{i=1}^{l-1}\left(s_i - \mathcal{F}(\boldsymbol{s})_i\right)^2 + \sum_{i=l}^{m}\left(\tfrac{1}{2}(s_i + \overline{s_i}) - \mathcal{F}(\boldsymbol{s})_i\right)^2 + \sum_{i=m+1}^{N}\left(s_i - \mathcal{F}(\boldsymbol{s})_i\right)^2,$$

after predicting the cover in the usual way. However this appears to be an $O(N^3)$, or at least $O(N^2)$ problem. In fact, it is possible to achieve a linear time algorithm by considering the sequence

$$e_i = \left(\tfrac{1}{2}(s_i + \overline{s_i}) - \mathcal{F}(\boldsymbol{s})_i\right)^2 - \left(s_i - \mathcal{F}(\boldsymbol{s})_i\right)^2,$$

which is related to E by $E(l,m) = \sum_l^m e_i + \sum_1^N \left(s_i - \mathcal{F}(\boldsymbol{s})_i\right)^2$. Minimizing E is achieved by finding the subsequence of $e_1, \ldots, e_N$ with minimum sum. Finding the minimum subsequence sum is a standard problem with a simple linear time solution.[11] Finally, we can again improve the method by using our novel cover predictors, and weighting the distance calculations as in Subsect. 3.2.

5. EXPERIMENTAL RESULTS

Recent experience with steganalysis benchmarking has drawn attention to the significance of cover set selection and the need for results to be replicated in images from different sources. Accordingly, the results in this paper are drawn from three different sets of cover images, to exclude the possibility that any performance improvements are specific to one particular image source as well as providing some comparability with prior research.

1. Our primary set is a completely new library of 1,600 never-compressed digital camera images, taken by the first author with the Minolta DiMAGE A1 camera in a raw format. Crucially, the raw images were extracted (using the Minolta DiMAGE Viewer version 2.37) directly to 12-bit greyscale bitmaps, avoiding any colour filter array interpolation. All image denoising was disabled. After slight cropping, to avoid any possibility of vignette artefacts, the images were all exactly 2000×1500 pixels in size. This set of images is our "gold standard" as we controlled the entire acquisition and pre-processing chain.[†] This set will be referred to as "RAW camera images" throughout this paper.
2. A second set is of 3,000 images downloaded from the NRCS website.[12] Apparently scanned from film in full colour, these images vary slightly in size around approximately 2100×1500 pixels. We shall refer to this image set as "scanned images".
3. A final database of images was supplied by the researchers at Binghamton University. Their full set was made up of pictures from many different digital cameras and numbers several thousand; we selected 1040 images with the same size of 1504×1000 pixels, in which sixteen different camera models are represented. The images were supplied as 24-bit colour PNGs. We will call these images "alternative RAW images".

The results for RAW camera images and scanned images are broadly similar, but the alternative RAW images' performance profile is markedly different. Surprised by these anomalous results, we guessed that the supposedly RAW alternative images had been subject to some type of image processing operation, perhaps during the conversion from internal camera RAW format to PNG. Certainly the images are much less noisy than those in other sets. After a few attempts, we were able to mimic the behaviour of the alternative RAW images by

[†]All images derived from this set can be made available to other researchers on request.

applying a denoising filter to the primary set of RAW camera images (Softwhile's DeNoise plug-in for Photoshop CS, version 1.0 release 29). In the experiments, we will concentrate on the two classes of cover image: those not subjected to denoising (the primary RAW camera images and the scanned images) and those subject to denoising (the alternative RAW images and also the primary RAW images after the denoise filter was applied). The two classes exhibit quite different behaviour, while behaviour within each class is similar.

To standardize the size of the images, all were reduced to smaller images of width 640: in the case of the RAW camera images, to 640×480, in the case of the other two sets to 640×457 (the slight discrepancy due to aspect ratio). Experiments were also performed on the full-size images, but the results will not be reported because of comparability across image sets. We used four different size reduction methods to control for possible artefacts of common downsampling algorithms.

1. Downsampling using bilinear interpolation (the standard bilinear resampling algorithm in Photoshop CS).
2. Downsampling using bicubic interpolation (ditto).
3. Downsampling using nearest neighbour interpolation. Lacking an interpolation filter, this produces images more noisy than the above two methods. Noisy images are known to be particularly hard for steganalysis.
4. Cropping random regions. The aim is to preserve the original pixel neighbourhood characteristics; a drawback is that the image content changes when it is cropped.

The steganalysis estimators apply to greyscale images, which are presumed to be of 8-bit precision. In the case of the primary RAW camera images, downsampling was performed on the 12-bit images, with conversion to standard 8-bit greyscale after all other transformations. In the case of the scanned images, and also the alternative RAW images, downsampling was performed on the colour originals, with conversion to greyscale (performed by selecting the luminance component) after all other transformations.

5.1. Performance of WS Estimators for Spread Embedding

The modifications to the WS method, proposed in Sect. 3, give rise to very many variations. Estimators can be constructed using any of six pixel predictors (fixed filters (3), (4), (5), and one using 24 pixels, plus adaptive filters with 8 (6) and 24 (7)) taps, any of three weighting coefficients (no weights, standard weights (8), and the new moderated weights (9)), and with or without subtracting the estimated bias (11) derived in Subsect. 3.3. We would like to know which combinations leads to improved performance, and then to test the best of the WS estimators against their structural competitors.

It is impossible to display benchmarks for all 36 WS variants in every cover set. Instead, we will demonstrate that the optimal choices for predictor, weighting method, and bias correction, can be made one-by-one, and display enough experimental results to justify each choice.

First, let us select the cover prediction filter. In Fig. 2 we show how the mean absolute error (MAE) of the relative payload estimator depends on the choice of filter. The results displayed are for unweighted WS variants, without bias correction, in two of the cover sets. Although the denoised images have a clearly different profile, in both cases the adaptive 24 pixel filter has the best performance (gives the smallest errors). We observed similar behaviour with weighted variants, and when bias correction was included, in almost every cover set, and so from this point onwards we will fix the cover predictor to be of this type. Occasionally, in the scanned images, we did observe that the 8 pixel filter (5) produced slightly better performance, but the extent of the difference was not large enough to vary our choice of predictor.

Second, we consider the choice of weight coefficients in (2). Figure 3 benchmarks versions of the WS estimators, using the chosen pixel predictor, with each of the weighting options. Again we see very different results in RAW images and denoised images (and, again, comparable results were observed in scanned images, regardless of downsampling choices, and the alternative RAW images, respectively). For images that have not been denoised, the new moderated weights (9) give the best performance, the standard weights (8) the next best, and the unweighted detector gives the worst performance (largest errors). But in denoised images this is reversed. We conclude that we should use either moderated weights or no weights at all, and that choice will depend on the nature of the image under analysis.

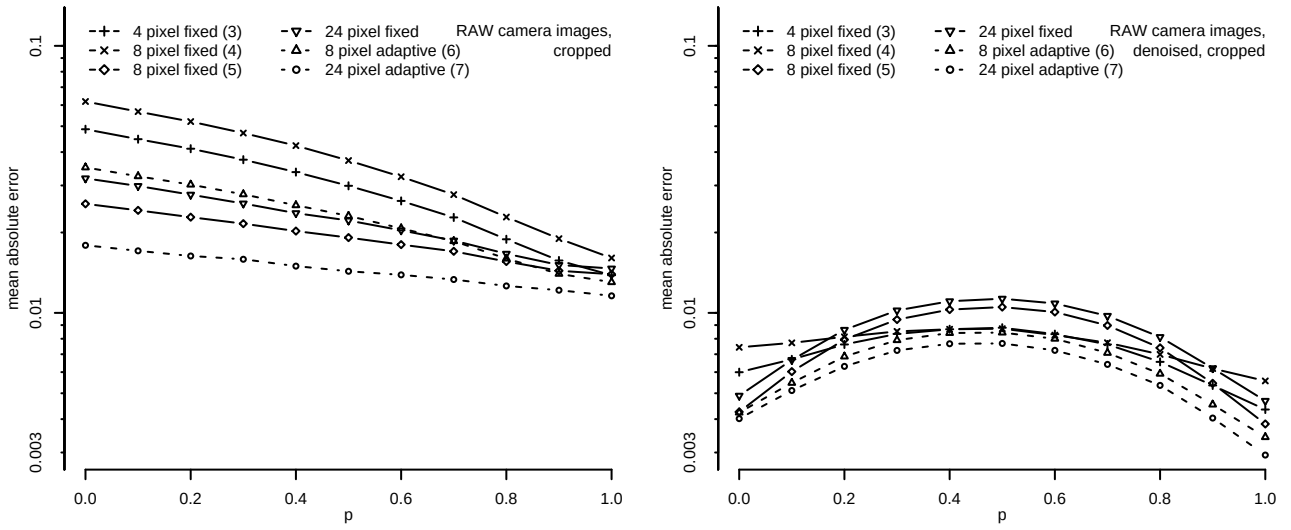


Figure 2. Mean absolute error (log scale) of unweighted WS estimators with different pixel prediction filters, in RAW camera images and denoised versions. The adaptive 24 pixel filter gives the best results in these covers.

The same figure also benchmarks estimators with and without the bias correction derived in Subsect. 3.3. In the RAW camera images the results are slightly in favour of bias correction for smaller payloads but the effect is negligible. In the denoised images we see a very different picture, with bias correction making a substantial improvement no matter which weighting method is chosen except for payloads greater than about 80 %. Similarly results were observed in the alternative RAW images. We conclude that bias correction should be used unless the uncorrected payload estimate is over about 0.8.

Thus we have reduced the WS variants to a shortlist of two: the 24 pixel adaptive filter, bias correction (disabled for initial estimates more than 0.8), and either no weights at all, or the new moderated weights (9). It seems that we should prefer the unweighted version only for images which show signs of denoising.

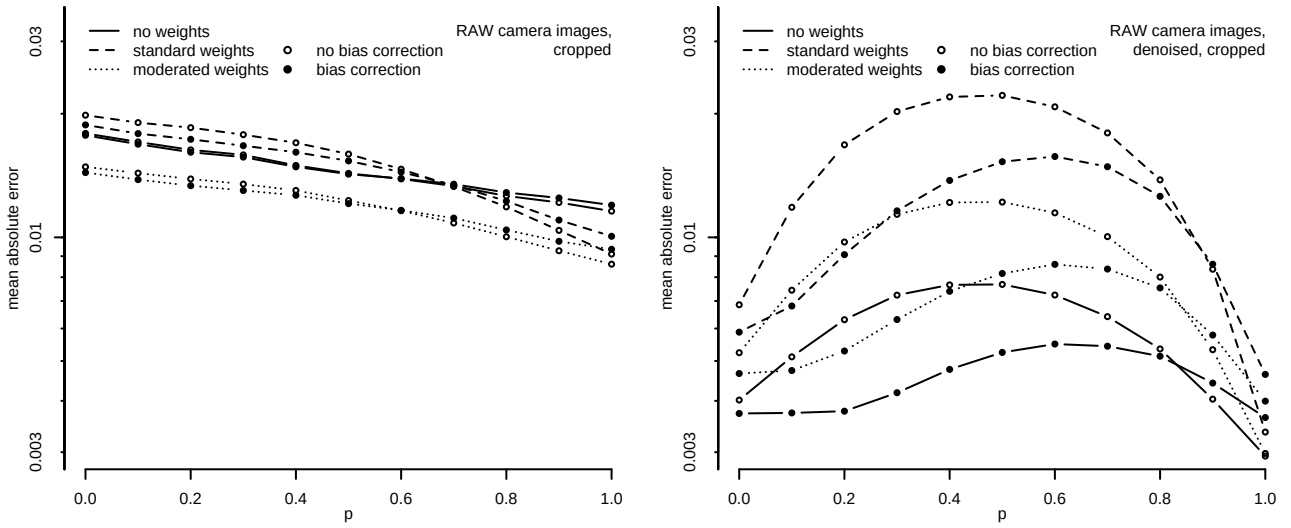


Figure 3. Mean absolute error (log scale) of WS estimators with the adaptive 24 pixel filter, unweighted or weighted according to standard or moderated weights, and with or without a bias correction term. In almost all cover sets the moderated weights are superior, but in denoised images unweighted WS has the better performance. Bias correction makes but a small difference to ordinary images, but can give a large advantage on denoised images.

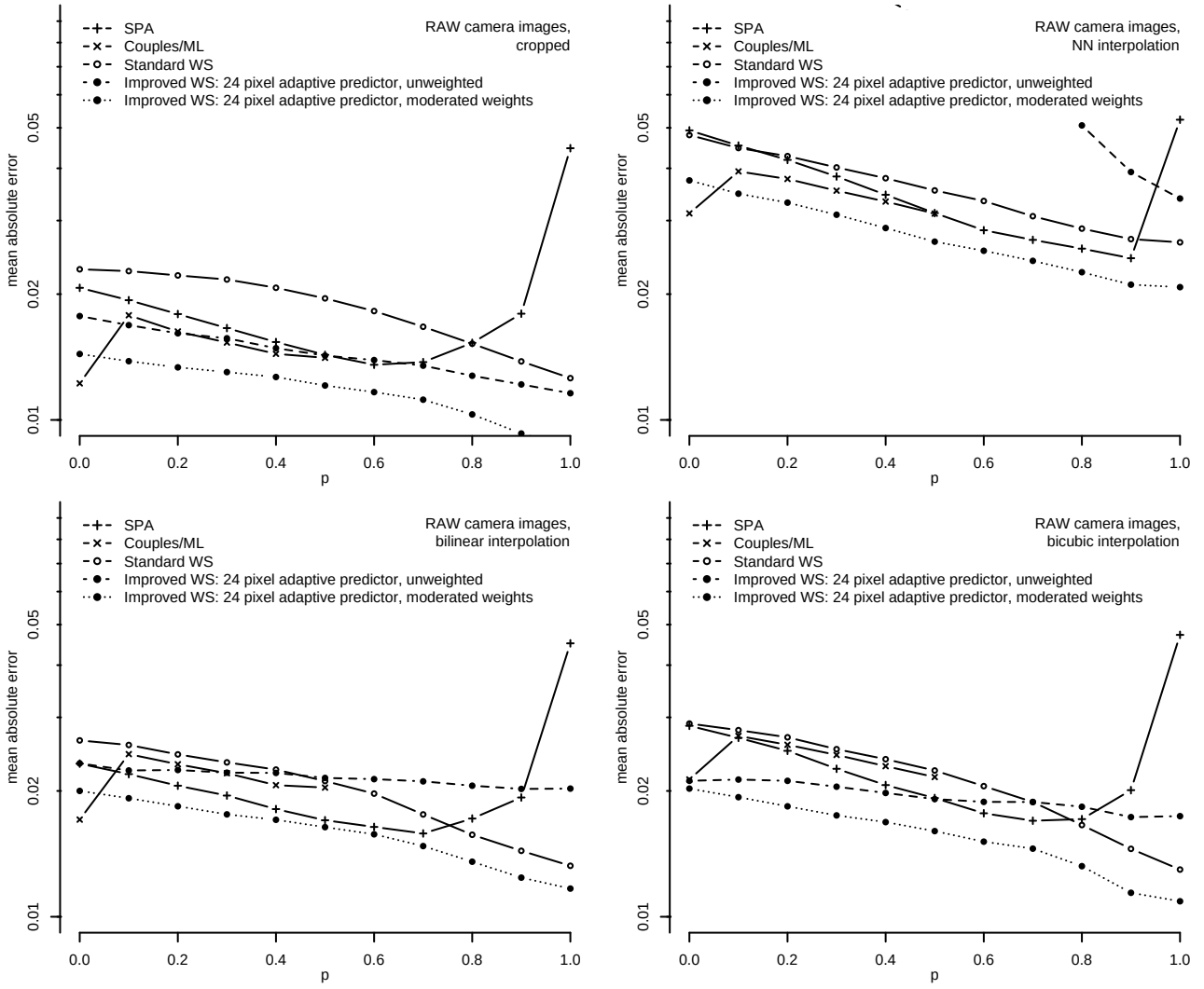


Figure 4. Comparison of the best structural payload estimators, standard WS, and the improved WS estimators proposed here, in four different sets of cover images derived from the raw camera images. The WS variant with a 24 pixel adaptive filter, moderated weights, and bias correction for $p < 0.8$, has the best overall performance.

We are now in a position to benchmark the new WS variants, against their competitors from the literature. The leading payload estimators and detectors of LSB replacement in digital images are the structural detectors, which were provided with a common framework in Ref. 5. They include estimators known as *RS*,[13] *SPA*,[3] *SPA-LSM*,[4] *Triples*,[5] *Quadruples*,[6] and *Couples/ML*.[7] Performance of the first four is evaluated thoroughly in Ref. 8: it is demonstrated that, for never-compressed greyscale images, there is not a great deal of difference but the SPA estimator is marginally most accurate, despite the more sophisticated methods used in SPA-LSM and Triples. As demonstrated in Ref. 7 (and confirmed here), the Couples/ML estimator is more accurate than SPA, but only for small payloads. The same is true, to a lesser extent, of the Quadruples estimator.

Since we consider only greyscale never-compressed images here, the leading competitors for WS are SPA and Couples/ML: the latter for small payloads only, and the former in other situations.[‡] We will now compare their performance, as well as standard WS as described in Ref. 1, with the best two improved WS estimators

[‡]The same is not true if the cover images are colour, or have previously been JPEG compressed; the results in Refs. 5 and 8 demonstrate that Triples is the most accurate estimator for payloads less than about half the maximum, SPA-LSM for payloads between about half and three-quarters, and standard WS for payloads near the maximum. The newer

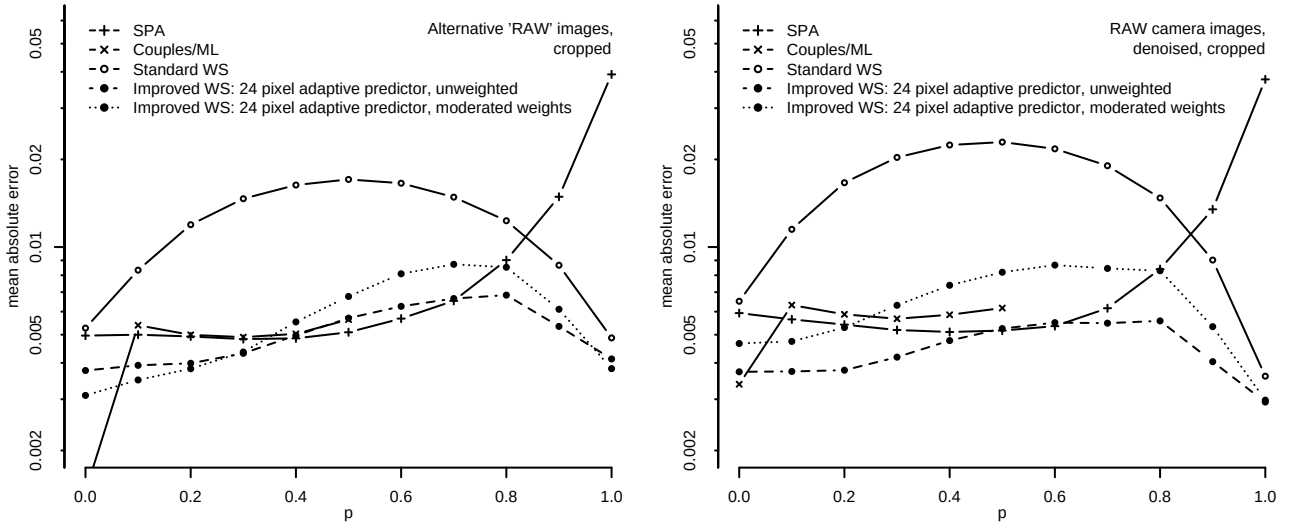


Figure 5. The relative performance is different when the cover images have been subject to denoising. Results here from the alternative RAW images and the primary set of RAW camera images which was subject to denoising before embedding. In this case, it is generally best to choose an unweighted WS estimator. For a small range of payloads, the SPA estimator remains most accurate, but the difference is very small.

proposed here. Because Couples/ML performs very poorly on large payloads, and is orders of magnitude slower to compute than the others, we will only include its results for payloads up to half maximum.

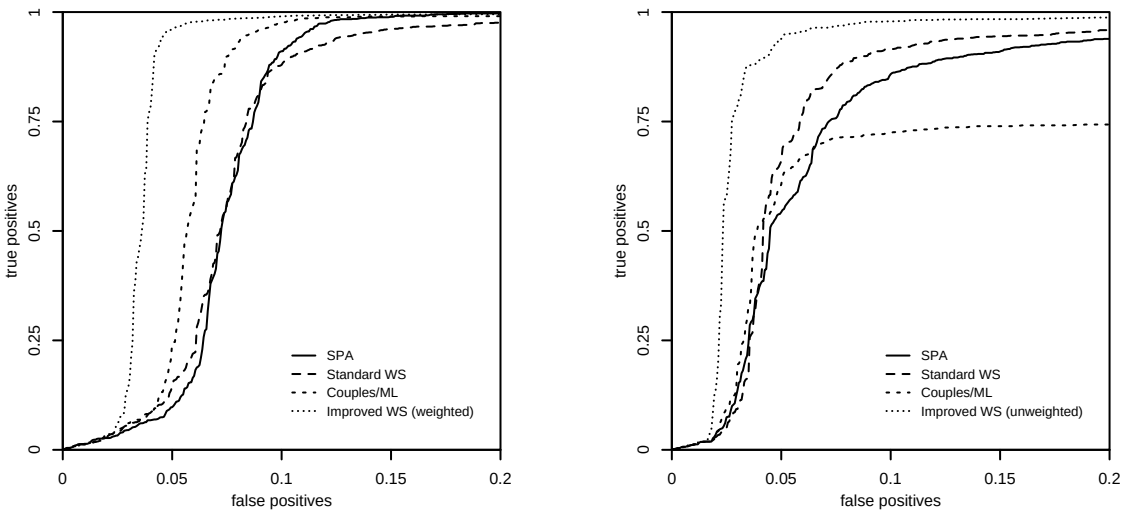
We display results from the RAW camera images, including the different downsampling options, in Fig. 4. Observe that the detectors are all much less accurate in nearest-neighbour downsampled images: this interpolation algorithm reduces neighbour correlation, causing the cover predictors to have larger errors. Aside from this difference, rather similar features are seen in each case: of the detectors in prior literature, SPA is generally the best performer but Couples/ML is better for very small payloads and standard WS for very large payloads. In almost all cases, however, the new WS method (with moderated weights) is the best performer. The only exception is that Couples/ML exhibits better performance for zero payloads (and one should balance this slight performance advtange against the computational costs).

We performed all the same experiments with the scanned images. The results were quite similar so we will not display more charts, but we did observe that the nearest-neighbour downsampled set produced slightly different results: the fixed 8-pixel filter (5) gave rather better results than the adaptive filter, and the Couples/ML performance was substantially better than improved WS, for a wider range of small payloads.

Now we turn to images which have been subject to denoising: they are displayed in Fig. 5. The general performance of the estimators is good (all error magnitudes are lower) which is to be expected for images where the cover can be predicted with high accuracy. The shape of the detectors' accuracy is different to the other image sets, and it is the unweighted variant of the improved WS estimator which performs best. In these respects the results for denoised images is quite different to that for those not subject to denoising. In the alternative RAW images there is a small range of payloads for which SPA remained the best performer, and a small range of payloads for which the weighted WS variant was more accurate than unweighted. However the differences, in such situations, are not very large. Overall, the unweighted WS variant is superior in denoised images.

Finally, *receiver operating characteristics* (ROC) curves plotted in Fig. 6 demonstrate that the lower error rates for our improved WS estimators translate to better performance (lower false positive rate at any given detection rate) when the methods are employed as discriminator between covers and stego objects with very small embedding rates. Results are given for the RAW camera images in their cropped versions with $p = 0.05$

Quadruples and Couples/ML estimators do not alter this conclusion.



(a) RAW camera images, cropped, $p = 0.05$ (b) RAW camera images, denoised, cropped, $p = 0.01$

Figure 6. Receiver operating characteristics (ROC): comparison of discriminatory power of selected WS and structural estimators. Improved WS variants outperform structural detectors by a fair margin.

using moderated weights (left chart, compare MAEs in the top-left chart of Fig. 4) and their denoised and cropped versions with $p = 0.01$ using unweighted WS (right chart, compare MAEs in the right chart of Fig. 5).

5.1.1. Sensitivity to Cover Properties

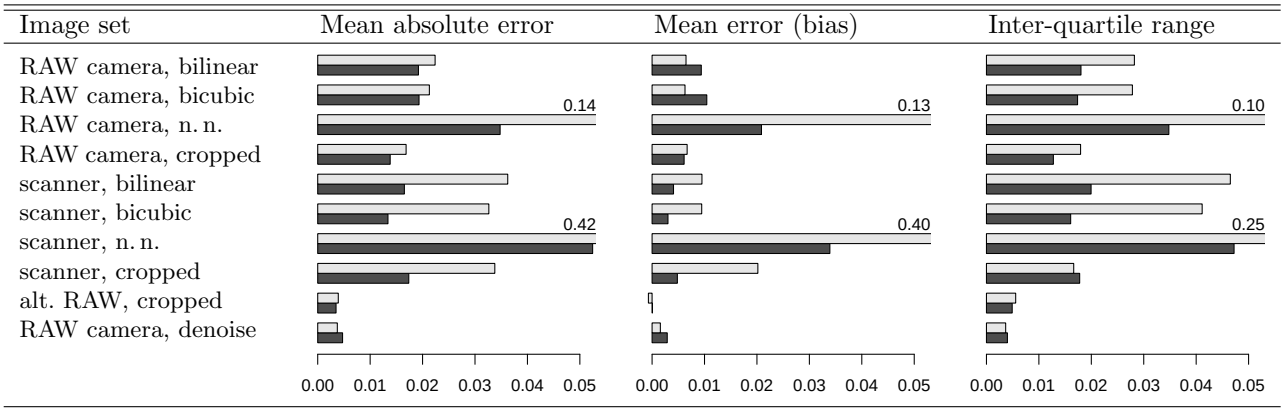
These experimental results have demonstrated that the nature of the cover images can have a large effect on the performance of the WS estimator. It is difficult to gauge whether such sensitivity is typical in steganalysis, because it is common practice in the literature to test against only one set of images. This could lead to misleading results: had we tested our WS improvements on only the alternative images (and some other steganalysis literature does exactly this) then our conclusions about weighting would have been completely different.

An advantage of WS is its conceptional and computational simplicity which allows us to run extensive experiments and dig a bit deeper into the dependencies between the components of the detection method and its performance. As a starting point, Tab. 1 reports the performance differences of two WS variants across all image sets considered in this study for $p = 0.1$. We present various measures of estimation performance, most prominently MAE (as in all previous charts), mean error as a measure of bias (to see whether a method over- or underestimates systematically), and the inter-quartile range as a robust measure of variation.

All measures show substantial performance differences across image sets and pre-processing methods. WS steganalysis is consistently least accurate for covers downsampled with nearest neighbour interpolation. Other downsampling methods somewhat increase error rates for our RAW camera images relative to the cropped images, but not consistently so for the scanned images.

Comparing the WS variants, it becomes apparent that the weighted variant (dark bars) performs considerably better in most image sets except for the bias measure in interpolated RAW camera images and for the alternative RAW images, where unweighted WS (bright bars) is slightly more accurate (by all measures). Hence we can see again that the alternative RAW set, as well as the images derived by denoising the primary RAW camera images, give very different results to those with images not subject to denoising (cf. Fig. 5). The variance of performance, between images sets, calls for extensive testing, with multiple datasets. (It also makes it difficult to draw general conclusions about which method or variant performs best, as every conclusion is conditional on a number of factors.) But no set of benchmarks will include the unimaginably high number of different image origins and pre-processing histories found in the wild. Hence, it would be beneficial to identify properties of images which influence the performance differences, and search for ways to adapt detection methods to the particular properties of each image to be steganalysed.

Table 1. Performance differences across image sets. Summary statistics for embedding rate $p = 0.1$. Dark bars are WS estimates using the adaptive filter, moderated weights and bias correction; bright bars are WS estimates using the unweighted adaptive predictor and bias correction (bars for nearest neighbour images exceed the scale and are censored).



5.1.2. Connection between Prediction Error and Estimator Performance

A crucial part of the WS method is the filter $\mathcal{F}(\boldsymbol{s})$ which predicts cover pixels, so we will study the relation between the performance of $\mathcal{F}(\boldsymbol{s})$ in predicting a known cover $\boldsymbol{c}$ and the secret message length estimation performance. For each image we compute the root mean square error (RMSE) of the predictor

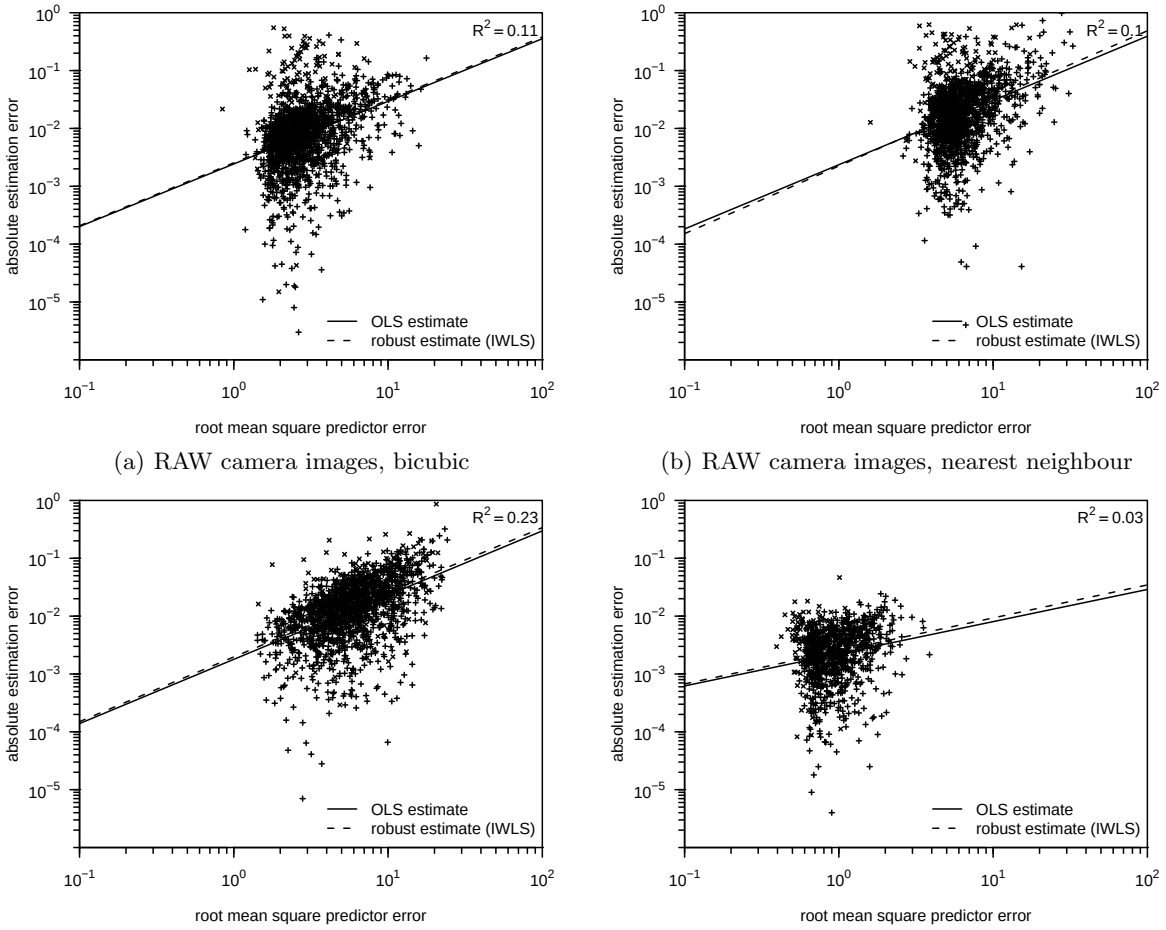
$$\mathrm{RMSE} = \left[\sum_{i=1}^{N} w_i \big(\mathcal{F}(\boldsymbol{c})_i - \boldsymbol{c}_i\big)^2\right]^{\frac{1}{2}}$$

and plot them against the absolute estimation error $|\hat{p} - p|$ for plain covers ($p = 0$). The RMSE calculation is weighted using the same weights as the WS method being studied. Selected results are displayed in Fig. 7. Both axes are in logs, and units are intensity differences on the x-axis and fraction of the embedding capacity on the y-axis. As the point clouds appear well-behaved we deem it justified to estimate regression lines, on log errors, to assess the slope and strength of the relation. The R^2 goodness of fit is based on the ordinary least squares estimate (OLS) which we complement, wary of misleading inference from outliers, with a robust regression using iterated least squares (IWLS) and the common Huber method.[14] In all cases, the two estimates are very close, so outliers are not an issue here.

Comparing Fig. 7(a) and Fig. 7(b), we see that differences in the linear predictability of pixel intensities can explain only about 10 % of the variation in the payload estimation *within* a given image set and pre-processing method. However, as bicubic and nearest neighbour covers lie on the same regression line, and the centre of mass for nearest neighbour images is shifted towards the upper-right corner (higher predictor RMSE, lower payload estimation accuracy), we conclude that differences in linear predictability may well explain much of the differences *between* image sets of different pre-processing. We do not possess enough different image sets to estimate the portion of variation thus explained.

Fig. 7(c) displays the relation for unweighted WS. The variation of the prediction error between images is higher than in the weighted case (cf. Fig. 7(a)), because weighting helps more for some pictures than for others. This explains why R^2 is higher. Put another way, weighting improves the predictor RMSE but the better cover estimate does not fully translate in better estimation performance because factors other than linear predictability (e.g. the weights' basis) also have influence. Fig. 7(d) completes this analysis with data from the alternative RAW images. It is very visible that the higher detection performance stems from much better cover predictability.

Given the importance of linear predictability for the detection performance, one may ask where differences in linear predictability come from. One source is clearly pre-processing with linear filters; another option is saturation, which has been found as influencing factor is previous work.[8, 15] Saturated pixels often appear in



(a) RAW camera images, bicubic

(b) RAW camera images, nearest neighbour

(c) RAW camera images, bicubic, unweighted

(d) alternative RAW images, cropped

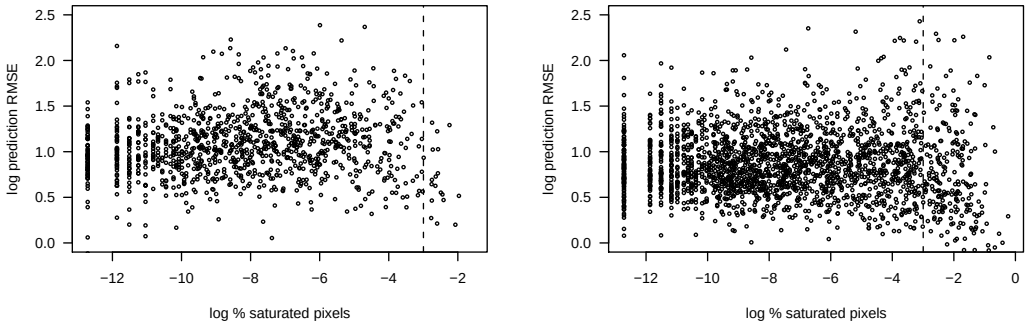
Figure 7. Relation between predictor accuracy (measured in RMSE for the improved predictor on cover images) and estimation performance (absolute errors for $p = 0$, no bias correction). Moderated weights unless otherwise stated.

spatial proximity; those regions are well predictable, its pixels get high weight, but do not fulfil some crucial assumptions for the correctness of the WS estimator (unlike other estimators, such as RS analysis,[13] which benefits from saturation[15]). Fig. 8 plots the log of the share of saturated pixels against the log predictor RMSE. It is very visible that only few of the RAW camera images possess significant portions of saturated pixels, and outliers with low RMSE appear on both sides of the vertical line denoting a share of 5 % of saturated images. For illustration, we have also printed the same chart for the cropped scanned images, where saturation is more frequently observed, and one can see the trend towards lower RMSE well beyond the 5 % line. Nevertheless, saturation as an explanatory factor can be ruled out for our set of RAW camera images.

If predictability, as a result of different image acquisition or pre-processing, influences the detection performance between image sets, the question remains what measurable image properties can explain (the still high) performance difference within a set of images from the same source. One indication that there are other factors is that, among the prediction methods offered in Sect. 3, the one that gives the lowest RMSE is rarely the best stego estimator: Tab. 2 shows that the adaptive predictor with standard weights produces consistently better RMSE than moderated weights, but the latter has overall higher performance when plugged into the WS method.

5.1.3. Parity Co-occurrence as Determinant for Estimation Performance

Another property relevant to WS and orthogonal to linear predictability is the distribution of the parity of intensity values in the local neighbourhood of a pixel. We shall call this property *parity co-occurrence* and,



(a) RAW camera images, nearest neighbour (b) scanned images, cropped

Figure 8. The share of saturated pixels (log percentage) can hardly explain the bulk of variation in predictor accuracy (log RMSE for the improved predictor with moderated weights for $p = 0$). Vertical lines correspond to 5 % saturation.

to measure it, we count the number of pixels with equal parity among a centre pixel's eight neighbours. We aggregate mean parity co-occurrence by building the average across all (non-edge) pixels in an image.

Going back to Fig. 7, one can see that most data points are plotted as '+' but some as '×' symbols. The latter correspond to the top 10 % quantile of mean parity co-occurrence in each image set. It is visible that for our RAW camera images, '×' cluster well above the regression lines, which indicates that high parity co-occurrence yields relatively poor detection performance for a given level of linear predictability. Given our comments that parity co-occurrence can lead to bias, in Subsect. 3.3, this is to be expected.

Note that 70 % of the alternative RAW images have an exceptionally high level of parity co-occurrence: it is plausible that such a feature could arise from denoising operations, which tend to smooth out image pixel values. This explains why bias correction yields substantial performance improvement in the alternative RAW images (see for example the right chart of Fig. 3). The distribution of average parity co-occurrence for selected image sets is depicted in Fig. 9. Again, it is visible that only our denoised RAW camera images (approximately) match the profile of the alternative RAW images.

5.2. Performance of WS Estimators for Sequential Embedding

Now we consider sequential LSB replacement, where both initial and arbitrary sequences of samples have their LSBs replaced. It is known that structural methods do not work well in this case (random location of cover changes is essential for their correctness) but we will test the SPA method anyway, as well as an old LSB replacement steganalysis method *Chi-Square*[16] which is adapted for initial sequential embedding. Of the WS methods, we can apply the standard version (for which random location of cover changes is not a requirement) or our newer versions, as well the those adapted specifically for sequential embedding. We do not include all the intermediate options, focusing only on chi-square, structural, standard WS, sequential WS with the standard mask and no weights, and sequential WS with the 8-pixel mask (5) and improved weights (9). (We did not test

Table 2. Median cover predictor RMSE of image sets used in our experiments

Image set	Predictor			
	standard	adaptive		
	unweighted	unweighted	std. weights	mod. weights
RAW images, bilinear	8.0	6.9	3.0	3.4
RAW images, bicubic	7.9	6.0	2.6	2.9
RAW images, nearest neighbour	12.1	11.7	5.3	5.8
RAW images, cropped	4.7	4.0	2.3	2.5
RAW images, denoised, cropped	1.3	0.7	0.5	0.5
alternative RAW images, cropped	1.7	1.2	0.8	0.9

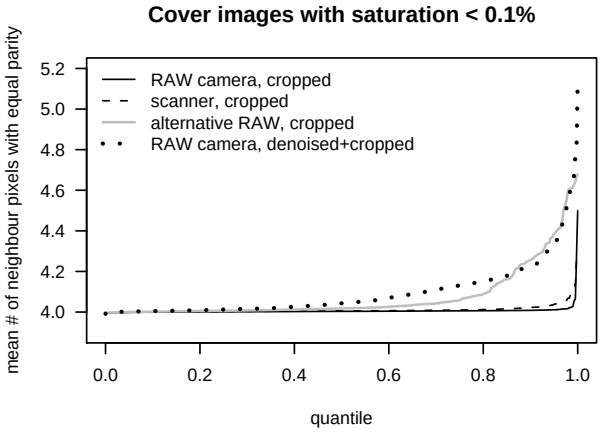


Figure 9. Neighbour pixel parity statistics. The high parity co-occurrence in the alternative RAW images could be successfully duplicated by denoising the RAW camera images. Saturated images are excluded to avoid interference from different amounts of saturated images in the sets (saturated areas cause strong parity co-occurrence).

the sequential WS method with adaptive predictors, because of minor complications with training the mask at the image edges: sequential WS is so accurate that there is a significant penalty if the edges are not included.)

Figure 10 displays the MAE of proportionate payload sizes for these detectors of initial sequence embedding. The chi-square estimator does not perform well except for full payload, SPA does better than one might expect, similar to standard WS. The sequential WS estimator is at least an order of magnitude more accurate: curiously, the weighted version works better in both RAW and denoised images.

Figure 11 shows analogous results for arbitrary sequential embedding: the chi-square estimator is not immediately applicable here. Since there is less certainty about the location of the payload we would expect that the accuracy is slightly lower, and that does turn out to be the case. But the sequential WS estimates of payload size are still at least an order of magnitude more accurate than their competitors. The method provides explicit estimates for the start and end locations of the payload: in practice, the error in size estimation is fairly evenly distributed between error estimating these two points.

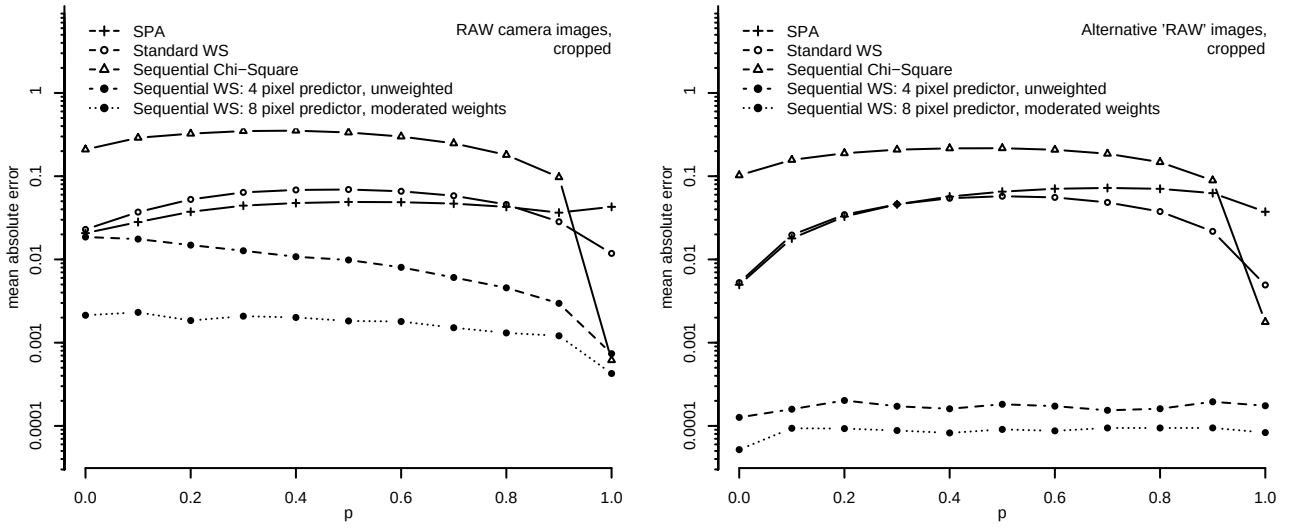


Figure 10. Mean absolute error (log scale) of structural, sequential chi-square, standard WS, and sequential WS estimators, when the payload is embedded as an initial sequence. Results for both raw cameras images (not subject to denoising) and the alternative raw images (which appear to have been denoised) are displayed.

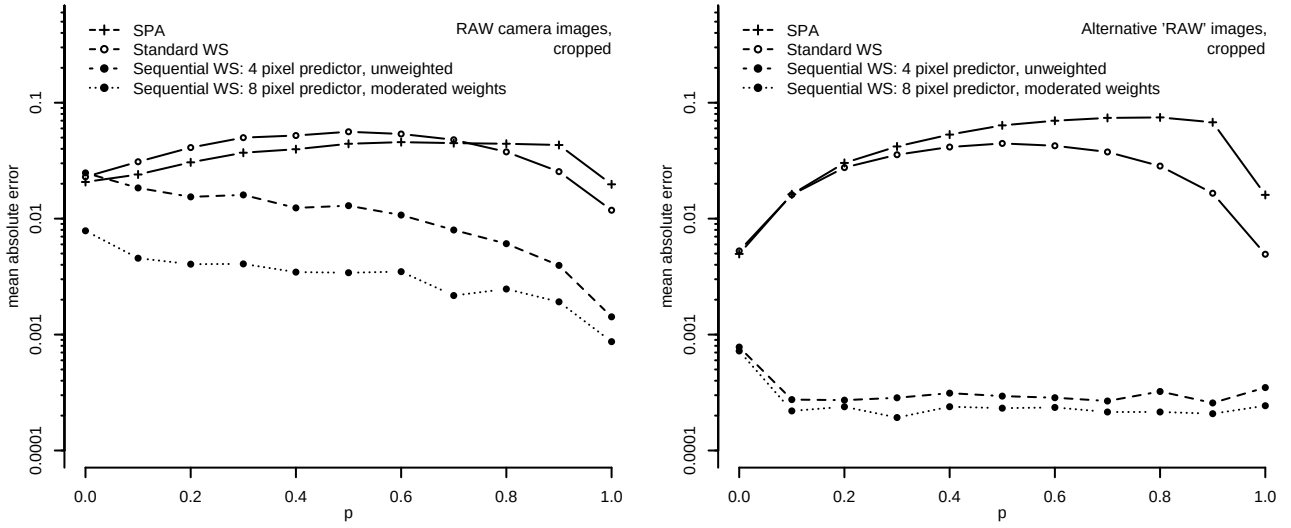


Figure 11. Mean absolute error (log scale) of structural, standard WS, and sequential WS estimators, when the payload is embedded as a segment with arbitrary start position. Results for both raw cameras images (not subject to denoising) and the alternative raw images (which appear to have been denoised) are displayed.

These estimators are highly accurate: mean absolute relative error of the order of 10^{-4} corresponds to payload size errors of around 30 bits, in an image of 0.3 megapixels. Such accuracy is literally unachievable in detection of spread embedding, because even a detector which detects every cover change perfectly cannot detect those pixel locations which convey payload but were not changed.

6. CONCLUSIONS

In a research effort aimed at exploring the behaviour of Weighted Stego-Image steganalysis under conditions where it has been known to work well (large payloads[8] and sequential embedding[9]), we created improvements to all of the method's three components. We found that an upgraded WS method outperforms even the best structural detectors in domains where they were previously believed more reliable. Extensive and robust experimental results on a variety of image sources confirm our findings and make the re-discovered WS approach the first choice quantitative detection method for LSB replacement in never-compressed cover images. More precisely, the WS variant using an adaptive 24 pixel filter with moderated weights has turned out to be the most accurate method in general, with the exception that equal weights are preferable if images are known to have been denoised. In either case, improved bias correction is recommended for initial payload estimates below 0.8. Fig. 12 illustrates these conclusions on the preferred detection strategy for LSB replacement in form of a decision tree.

We also performed some experiments in an attempt to identify the image properties which are responsible for the large observed performance discrepancies between different image sets. Predictor accuracy and cover parity co-occurrence are significant factors. Further, we have been able to modify the method for specialised detection of sequential LSB replacement, where it displays a very high level of performance, unmatched in the literature.

It is important to mention that the WS estimators rely on the cover images being natural images not subject to lossy compression. If the covers had been stored as JPEGs, prior to spatial-domain LSB replacement, the WS method would lose a lot of performance. Some of the structural methods are not badly affected by this phenomenon. Optimizing the WS method to specific cover types, such as previous JPEG compression or nearest neighbour downsampling, as well as further specialising the sequential detection to cases where embedding changes are introduced sparsely (e. g. through matrix embedding) are topics of further study.

ACKNOWLEDGMENTS

The first author is a Royal Society University Research Fellow. Thanks are due to Jessica Fridrich and Tomáš Pevný, who supplied the set of "alternative RAW images".

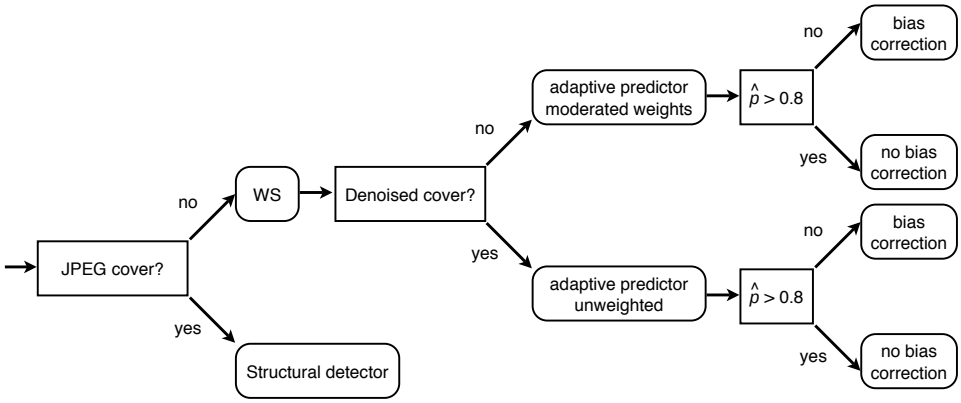


Figure 12. Decision tree: choosing the best LSB replacement estimator conditional on cover properties.

REFERENCES

1. J. Fridrich and M. Goljan, "On estimation of secret message length in LSB steganography in spatial domain," in *Security, Steganography, and Watermarking of Multimedia Contents VI*, E. J. Delp III and P. W. Wong, eds., *Proc. SPIE* **5306**, pp. 23–34, 2004.
2. X. Yu, T. Tan, and Y. Wang, "Extended optimization method of LSB steganalysis," in *Proc. IEEE International Conference on Image Processing*, **2**, pp. 1102–1105, 2005.
3. S. Dumitrescu, X. Wu, and Z. Wang, "Detection of LSB steganography via sample pair analysis," *IEEE Transactions on Signal Processing* **51**(7), pp. 1995–2007, 2003.
4. P. Lu, X. Luo, Q. Tang, and L. Shen, "An improved sample pairs method for detection of LSB embedding," in *Proc. 6th Information Hiding Workshop*, *Springer LNCS* **3200**, pp. 116–127, 2004.
5. A. Ker, "A general framework for the structural steganalysis of LSB replacement," in *Proc. 7th Information Hiding Workshop*, *Springer LNCS* **3727**, pp. 296–311, 2005.
6. A. Ker, "Fourth-order structural steganalysis and analysis of cover assumptions," in *Security, Steganography and Watermarking of Multimedia Contents VIII*, E. J. Delp III and P. W. Wong, eds., *Proc. SPIE* **6072**, pp. 25–38, 2006.
7. A. Ker, "A fusion of maximum likelihood and structural steganalysis." To appear in *Proc. 9th Information Hiding Workshop*, 2007.
8. R. Böhme and A. Ker, "A two-factor error model for quantitative steganalysis," in *Security, Steganography and Watermarking of Multimedia Contents VIII*, E. J. Delp III and P. W. Wong, eds., *Proc. SPIE* **6072**, pp. 59–74, 2006.
9. A. Ker, "A weighted stego image detector for sequential LSB replacement," in *Proc. Third International Symposium on Information Assurance and Security (IAS)*, pp. 453–456, IEEE Press, 2007.
10. A. Ker, "Steganalysis of embedding in two least significant bits.," *IEEE Transactions on Information Forensics and Security* **2**(1), pp. 46–54, 2007.
11. D. Gries, "The maximum-segment-sum problem," in *Formal Development Programs and Proofs*, E. W. Dijkstra, ed., pp. 33–36, Addison-Wesley, 1990.
12. "NRCS photo gallery." http://photogallery.nrcs.usda.gov/, accessed April 2004.
13. J. Fridrich, M. Goljan, and R. Du, "Reliable detection of LSB steganography in color and grayscale images," *Proc. ACM Workshop on Multimedia and Security*, pp. 27–30, 2001.
14. P. J. Huber, "Robust estimation of a location parameter," *Annals of Math. Statistics* **35**, pp. 73–101, 1964.
15. R. Böhme, "Assessment of steganalytic methods using multiple regression models," in *Proc. 7th Information Hiding Workshop*, M. Barni et al., ed., *Springer LNCS* **3727**, pp. 278–295, 2005.
16. A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," in *Proc. 3rd Information Hiding Workshop*, *Springer LNCS* **1768**, pp. 61–76, 1999.