

A Weighted Stego Image Detector for Sequential LSB Replacement

Andrew D. Ker

Oxford University Computing Laboratory
Parks Road, Oxford OX1 3QD, England
adk@comlab.ox.ac.uk

Abstract

We describe a simple modification of the so-called *Weighted Stego Image (WS) steganalysis method*, due to Fridrich and Goljan, to detect sequential replacement of least significant bits. The most sensitive methods for the detection of bit replacement do not work well when the payload is concentrated at the start of the cover instead of being spread evenly over it, so this detector fills a gap in the literature. Experimental results show that the adapted method gives sequentially-embedded payload estimation typically 10 times more accurate than the standard WS method.

1. Introduction

The aim of *steganography* is to embed a secret payload imperceptibly into an item of cover media: this hidden data is thus smuggled inside an apparently-innocent *stego object*. Steganography allows information security measures to be bypassed and this motivates the competing discipline *steganalysis*: the development of statistical tests to distinguish truly innocent cover objects from payload-carrying stego objects. In practice such detectors often take the form of payload size estimators, which may return a zero estimate.

Perhaps the simplest form of embedding is to take an item of digital media – in this paper we shall consider digital images – and replace the least significant bits (LSBs) of the data by the hidden payload. In images, at least, this produces quite invisible alterations yet is sufficient to hide as much as one payload bit in each cover sample (pixel). However it is now well-established that replacement of LSBs is an insecure embedding method because the changes it induces have a combinatorial structure. The first detector was the *chi-square* attack [9] but this venerable method is weak unless the hidden payload is practically the maximum possible; there are now much more sensitive detectors including [3–7]. The most effective make an explicit analysis of the effects of flipping LSBs in groups of pixels and are able to detect payloads of around 1-10% of the maximum.

Steganalysers with the best performance generally assume that the payload, embedded by replacing LSBs, is spread evenly over the cover (usually determined by a pseudorandom permutation of pixels); we will refer to this concisely as *spread embedding*. But early work on LSB replacement [9] also considered the possibility that the payload is embedded sequentially, scanning through the cover image in row order starting at the top left corner; we will refer to this as *sequential embedding*. Sequential embedding must be even less secure than spread embedding because the payload location is predictable, but it is an unfortunate fact that the very best detectors of LSB replacement do not work, or lose most of their sensitivity, when the payload is embedded sequentially; we shall see some examples in the following section.

Embedding sequentially is the type of error which an opponent foolish enough to use LSB replacement might plausibly make (indeed such embedding can be found in various layman-conceived software including [2] and the once-popular but now unavailable EzStego). Therefore there is a gap in the literature, for detectors using the principles of the very sensitive bit replacement detectors but applicable to sequential embedding, which we explore in this paper.

2. Structural methods fail on sequential embedding

Current state-of-the-art detectors for spread LSB replacement use its *structural* properties [3, 6, 7]. They are payload estimators which analyse the effect of bit replacement on certain features of an image called *trace subsets*, with a general framework described in [6]. Embedding, however, is modelled as random LSB changes occurring independently in each cover pixel: while appropriate for spread embedding, this model is not accurate for sequential embedding where changed pixels can occur at the start of the stego object but not at the end. For this simple reason the structural detectors, and many other detectors in the literature which implicitly make the same assumption, have poor performance against sequential embedding.

Estimator	RS	Triples	WS
<i>No embedding</i>			
Median	0.0054	0.0026	0.0118
Standard deviation	0.0267	0.0236	0.0544
Interquartile range	0.0215	0.0180	0.0426
<i>Spread embedding 30%</i>			
Median	0.3123	0.3049	0.3123
Standard deviation	0.0118	0.0301	0.0451
Interquartile range	0.0165	0.0184	0.0423
<i>Sequential embedding 30%</i>			
Median	0.2261	0.2261	0.3197
Standard deviation	0.0461	0.0979	0.0475
Interquartile range	0.0610	0.1545	0.0387

Table 1. Performance of some proportionate payload estimators.

Refer to Tab. 1 for some brief experimental verification. We have taken two structural payload estimators: the earliest such method known as *RS* [5], where structural analysis is implicit, and a newer structural detector *Triples* [6]. We also examine a different type of estimator, *WS*, of which more later.

Each estimator has been tested against a set of 3000 cover images (full colour, never compressed bitmaps, all sized approximately 0.3Mpixels), computing an estimated *proportionate* payload size, and then the same experiment repeated when the images have payloads of 30% of the maximum (approximately 90Kbits) embedded by both sequential and spread LSB replacement. We measure the accuracy of each estimator in each situation by the observed median estimate, and both standard deviation and interquartile range: robust measures of location and spread are included because [1] indicates that the results can be distorted by a very small number of extreme outliers.

Table 1 demonstrates that the structural detectors are very accurate estimates of spread payload, or no payload, but do not perform well on sequential embedding: they underestimate the payload size, and the estimates vary widely. It is unsatisfactory that detectors for sequential embedding are weaker than for spread embedding. Sequential embedding is highly predictable, and knowing where the payload is ought to make the steganalyst’s task much easier.

Macroscopic steganalysis methods which only use the image histogram, such as the old chi-square attack [9], work regardless of payload location. However their performance is extremely weak because they discard so much informa-

tion. There are also adaptations of the chi-square attack, including in [9], specialised to sequential embedding but they still lack sensitivity. We should be able to do better.

Evidence in the literature suggests, however, that adapting the framework of [6] for sequential embedding will have complications. Sequential embedding uses one part of the stego image to full capacity (one payload bit per cover sample) and leaves the rest unused. Unfortunately the structural detectors seem to exhibit a weakness for payloads at the maximum: this is perhaps exposed best in [8] where it is explained as the ill-conditioning of a system of linear equations which become singular at maximal embedding. Broadly speaking, maximal embedding destroys the key structural information. We suspect that structural sequential detectors may not perform well, if they can be constructed at all. Therefore we will look elsewhere for sensitive estimators of sequentially-embedded payloads.

3. The “weighted stego image” technique

Steganalysis using a *weighted stego image*, known by the acronym *WS*, was introduced by Fridrich and Goljan in [4]. It occupies an unusual position in steganalysis of LSB replacement, because it does not use the same pixel group analysis on which almost every other reasonably-accurate detector relies but it has fairly good accuracy. We give a brief exposition of the simplest *WS*-based detector of [4], modifying some of the notation to suit our purposes.

Suppose that a cover image consists of N samples (e.g. pixel intensities) c_1, \dots, c_N and that a payload of length $M \leq N$ bits is embedded by LSB replacement to form a stego image with corresponding samples s_1, \dots, s_N . The location of the M payload bits can be arbitrary – unlike the structural methods mentioned in the previous section it need not involve a random selection of cover samples – but we do require that the payload bits are uncorrelated with the cover samples. The *weighted stego image* $s_1^\alpha, \dots, s_N^\alpha$ is a real-valued series of samples formed by taking a weighted average between the stego image and the stego image with every sample’s LSB flipped¹, with the former given a weight of $1 - \alpha$ and the latter of α : $s_i^\alpha = \alpha \bar{s}_i + (1 - \alpha)s_i$. It is shown, in Theorem 1 of [4], that this weighted stego image is closest to the cover image, if difference between the two vectors c_1, \dots, c_N and $s_1^\alpha, \dots, s_N^\alpha$ is measured using the Euclidean L^2 -norm, when $\alpha = \frac{M}{2N}$. This corresponds to the expectation that proportion $\frac{M}{2N}$ of the cover pixels are flipped when embedding a payload of length M .

At the point of steganalysis, of course, the cover image is not known. But [4] proposes to predict the cover pixels c_i by

¹Throughout this paper we will write \bar{x} to mean the nonnegative integer x with the least significant bit flipped: $\bar{x} = x + (-1)^x$.

filtering the neighbourhood of each pixel in the stego image, forming an estimated cover image whose samples are \hat{c}_i . The filter used simply takes the mean of the immediately surrounding four stego pixels (it is explained, in [4], that each pixel's estimate must not depend on the corresponding stego pixel, only its neighbours).

Given a stego image, this basic WS method estimates the payload size by finding α to minimize the distance

$$E(\alpha) = \sum_{i=1}^N \left(((1-\alpha)s_i + \alpha\bar{s}_i) - \hat{c}_i \right)^2. \quad (1)$$

Differentiating, it is shown that the payload length estimator \hat{M} can be given a closed form:

$$\hat{M} = 2N \operatorname{argmin}_{\alpha} E(\alpha) = 2 \sum_{i=1}^N (s_i - \hat{c}_i)(s_i - \bar{s}_i). \quad (2)$$

Improved versions of this estimator are described later in [4], but we shall not use them here. These and other potential performance improvements are discussed briefly in Subsect. 4.2.

Experimental results (for example the very thorough series in [1]) demonstrate that the WS payload estimator can not quite match the accuracy of structural steganalysis when applied to spread LSB replacement. Indeed we can see this for ourselves in Tab. 1: the WS estimates have rather higher standard deviation and interquartile range. But the WS detector *does* work moderately well on sequential embedding, about as well as for spread embedding. We shall next modify the method to exploit the predictability of sequential embedding, making a much more sensitive detector.

3.1. Adapting the WS technique

Our novel detector requires only a small conceptual change to the weighted stego method. Since, under sequential embedding, we know that the first M pixels contain payload at every sample, and the rest none, we fix the weighting at $\alpha = \frac{1}{2}$ (embedding at every pixel flips half the LSBs) for the first M terms and $\alpha = 0$ for the rest. Then the measure of distance from a predicted cover becomes

$$F(M) = \sum_{i=1}^M \left(\left(\frac{1}{2}s_i + \frac{1}{2}\bar{s}_i \right) - \hat{c}_i \right)^2 + \sum_{i=M+1}^N (s_i - \hat{c}_i)^2 \quad (3)$$

where this time the unknown is M . Analogous to Theorem 1 of [4] it can be shown that $F(M)$ will indeed be minimized at the true value of M , if the cover predictors are exact, (we omit the precise statement of this result and comment that, strictly speaking, one must consider the expectation of $F(M)$ over random payloads). Then the payload size can be estimated by computing

$$\hat{M} = \operatorname{argmin}_M F(M).$$

Estimator	WS	Sequential WS
<i>No embedding</i>		
Median	0.0118	0.0000
Standard deviation	0.0544	0.0315
Interquartile range	0.0426	0.0017
<i>Sequential embedding 30%</i>		
Median	0.3197	0.3044
Standard deviation	0.0475	0.0320
Interquartile range	0.0387	0.0010

Table 2. The proposed sequential detector is more sensitive than the standard WS method.

We will call this detection method *Sequential WS*.

3.2. Efficient implementation

It is not possible to obtain a closed form for the payload estimate by differentiating (3); indeed the function might have multiple local minima. Therefore one must compute $F(M)$ for each M and find the minimum by brute force. Naive computation of (3) for each M would have quadratic time complexity but it is possible to find the minimum in linear time via the recurrence

$$f_0 = 0$$

$$f_j = f_{j-1} + \left(\frac{1}{2}(s_j + \bar{s}_j) - \hat{c}_j \right)^2 - (s_j - \hat{c}_j)^2$$

which, by elementary manipulation of (3), satisfies $F(j) = f_j + \sum_{i=1}^N (s_i - \hat{c}_i)^2$ and we may ignore the additive constant if we only require the location of the minimum.

Thus we require only one pass through the stego image to estimate M , which is exactly the same complexity needed to evaluate (2).

4. Results and Conclusions

4.1. Experimental Results

The Sequential WS estimator makes use of the additional predictability of sequential embedding: we now show that this has made it much more sensitive. We use the same set of 3000 cover images as in Sect. 2 and continue to express payload estimates as a proportion of the maximum for each cover image (i.e. we estimate M/N). The analogy to Tab. 1 is Tab. 2, in which we see that the Sequential WS detector is much more accurate as an estimate of payload size. Noting the difference between the standard deviation

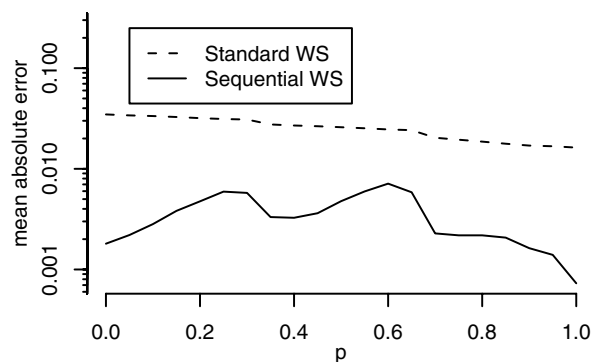


Figure 1. Comparing mean absolute error of proportionate payload estimates.

and interquartile range, we conclude that there are a small number of extreme outliers in the Sequential WS estimates.

A wider comparison of sequential payload estimation by WS and Sequential WS is shown in Fig. 1, where proportionate payloads of 0, 0.05, . . . , 1 are all tested. This figure uses an alternative measure of accuracy (observed mean absolute error) which gives a balance between robustness and sensitivity to outliers. It shows how the accuracy is dependent on the true payload (x -axis), with the y -axis displayed on a log scale. We see that the average size of error is approximately 10 times smaller for Sequential WS than standard WS. The uneven performance of the former is a mystery, probably due to the content of the covers, which we do not investigate here.

The extreme accuracy of the Sequential WS estimator is verified by other experiments, not reported here for reasons of space. We conclude with one final statistic: in each of 1000 grayscale covers, all 1.5Mpixels in size, exactly 500000 bits were embedded sequentially. Over 90% of the Sequential WS payload size estimates were within just 120 of the correct value 500000. This represents much higher accuracy than can be achieved by any currently-known detector for spread LSB embedding.

4.2. Conclusions and further directions

Sequential replacement of LSBs was one of the first steganographic embedding methods described, and is perhaps the worst: bit replacement has predictable combinatorial structure, and furthermore the payload is located predictably. We have developed a payload estimator with high accuracy, reflecting this weakness.

The idea presented here is very simple, and we have only given the most straightforward detection application. There are many possibilities for more advanced, presumably even more accurate, estimators. First, it is noted in [4] that the

WS estimates can be unsettled by a small number of rogue pixels in noisy areas, and proposes adding weighting factors to the L^2 -norm calculation (1). It is most likely that a parallel approach will improve the Sequential WS estimator, attenuating the rare outlier estimates. Second, there is the possibility of using a more sophisticated cover pixel predictor than that in [4]: in some unreported work this author has observed quite well-improved WS performance with filters based on local neighbourhoods of 8 pixels.

It seems certain that the extension of the WS method to multiple bit planes, found in [10], can be adapted for sequential embedding. Indeed the method also adapts for replacement of remainders at arbitrary modulus (it is not restricted to bit plane replacement).

Finally, it should also be possible to make a further modification to detect LSB-embedded payload placed sequentially at positions other than the beginning of the cover. There is no conceptual difficulty, but the minimization algorithm must be constructed carefully.

References

- [1] R. Böhme and A. Ker. A two-factor error model for quantitative steganalysis. In *Security, Steganography and Watermarking of Multimedia Contents VIII*, volume 6072 of *Proc. SPIE*, pages 59–74, 2006.
- [2] Camera/Shy. steganographic software. Downloaded from <http://sourceforge.net/projects/camerashy>, March 2007.
- [3] S. Dumitrescu, X. Wu, and Z. Wang. Detection of LSB steganography via sample pair analysis. *IEEE Transactions on Signal Processing*, 51(7):1995–2007, 2003.
- [4] 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*, volume 5306 of *Proc. SPIE*, pages 23–34, 2004.
- [5] J. Fridrich, M. Goljan, and R. Du. Detecting LSB steganography in color and grayscale images. *IEEE Multimedia, Special Issue on Security*, 8(4):22–28, 2001.
- [6] A. Ker. A general framework for the structural steganalysis of LSB replacement. In *Proc. 7th Information Hiding Workshop*, volume 3727 of *Springer LNCS*, pages 296–311, 2005.
- [7] A. Ker. Fourth-order structural steganalysis and analysis of cover assumptions. In *Security, Steganography and Watermarking of Multimedia Contents VIII*, volume 6072 of *Proc. SPIE*, pages 25–38, 2006.
- [8] A. Ker. Steganalysis of embedding in two least significant bits. *IEEE Trans. Information Forensics and Security*, 2(1):46–54, 2007.
- [9] A. Westfeld and A. Pfitzmann. Attacks on steganographic systems. In *Proc. 3rd Information Hiding Workshop*, volume 1768 of *Springer LNCS*, pages 61–76, 1999.
- [10] X. Yu, T. Tan, and Y. Wang. Extended optimization method of LSB steganalysis. In *Proc. IEEE International Conference on Image Processing*, volume 2, pages 1102–1105, 2005.