Reliable Detection of LSB Steganography in Color and Grayscale Images

Jessica Fridrich
SUNY Binghamton
Department of SS&IE
Ph: +607 777 2577
fridrich@binghamton.edu

Miroslav Goljan
SUNY Binghamton
Department of EE
Ph: +607 777 2577
mgoljan@binghamton.edu

Rui Du
SUNY Binghamton
Department of EE
Ph: +607 777 2577
rdu@binghamton.edu

ABSTRACT

A large number of commercial steganographic programs use the Least Significant Bit embedding (LSB) as the method of choice for message hiding in 24-bit, 8-bit color images, and grayscale images. It is commonly believed that changes to the LSBs of colors cannot be detected due to noise that is always present in digital images. In this paper, we describe a new very accurate and reliable method that can detect LSB embedding in randomly scattered pixels in both 24-bit color images and 8-bit grayscale or color images. It is based on our previous work on lossless data embedding [1]. By inspecting the differences in the number of regular and singular groups for the LSB and the "shifted LSB plane", we can reliably detect messages as short as 0.03bpp.

Categories and Subject Descriptors

Algorithms, Design, Security

General Terms

Detection of LSB steganography – algorithms, results

Keywords

Steganalysis, steganography, LSB embedding, attacks

1. INTRODUCTION

Steganography is the art of secret communication. Its purpose is to hide the very presence of communication as opposed to cryptography whose goal is to make communication unintelligible to those who do not posses the right keys [2].

Digital images, videos, sound files, and other computer files that contain perceptually irrelevant or redundant information can be used as "covers" or carriers to hide secret messages. After embedding a secret message into the cover-image, a so-called stego-image is obtained. It is important that the stego-image does not contain any detectable artifacts due to message embedding. A third party could use such artifacts as an indication that a secret message is present. Once message detection can be reliably achieved, the steganographic tool becomes useless (in this paper, only the passive warden scenario is considered).

Obviously, the less information is embedded into the cover-image, the smaller the probability of introducing detectable artifacts by the embedding process. Another important factor is the choice of the cover-image. Images with a low number of colors, computer art, images with a unique semantic content, such as fonts, should be avoided as cover images. Some steganographic experts recommend grayscale images as the best cover-images [3]. Uncompressed scans of photographs or images obtained with a digital camera contain a high number of colors and are usually recommended and considered safe for steganography. Reliable detection of messages in this class of images and accurate message length estimation is the focus of the present paper. Before we outline the structure of this paper, we briefly summarize previously proposed steganalytic methods.

In our previous work [4], we have shown that images stored previously in the JPEG format are a very poor choice for cover images. This is because the quantization introduced by JPEG compression can serve as a "watermark" or a unique fingerprint, and one can detect even very small modifications of the cover image by inspecting the compatibility of the stego-image with the JPEG format.

In [5], we introduced the RQP method for detection of LSB embedding in 24-bit color images. It works reasonably well as long as the number of unique colors in the cover image is less than 30% of the number of pixels. Then the results become progressively unreliable. Also, it cannot be used for grayscale images.

Pfitzmann and Westfeld [6] proposed a method based on statistical analysis of Pairs of Values (PoVs) that are exchanged during message embedding. This method provides very reliable results when the message placement is known (e.g., sequential). However, randomly scattered messages can only be reliably detected with this method when the message length becomes comparable with the number of pixels in the image.

Johnson and Jajodia [7] introduced a steganalytic method that can be applied to stego-images in the palette format created by programs that preprocess the palette.

In this paper, we present a new, reliable and extremely accurate steganalytic method that can be applied to 24-bit color images as well as to 8-bit grayscale (or color) images with randomly scattered message bits embedded in the LSBs of colors or pointers to the palette. In the next section, we introduce the concepts and terminology necessary to explain our stego-detection technique. In Section 4, we present the new steganalytic method for grayscale images and study its accuracy. The results of our tests on images are presented in Section 5. Finally, the paper is concluded in Section 6.

2. TERMINOLOGY

Let us assume that we have a cover image with $M \times N$ pixels and with pixel values from the set P. For example, for an 8-bit grayscale image, $P = \{0, \ldots, 255\}$. The stego-detection method starts with dividing the image into disjoint groups of n adjacent pixels $(x_1, \ldots, x_n)$. For example, we can choose groups of n=4 consecutive pixels in a row. We define so called discrimination function f that assigns a real number $f(x_1, \ldots, x_n) \in \mathbf{R}$ to each pixel group $G = (x_1, \ldots, x_n)$. The purpose of the discrimination function is to quantify the smoothness or "regularity" of the group of pixels G. The noisier the group of pixels $G=(x_1, \ldots, x_n)$ is, the larger the value of the discrimination function becomes. For example, we can choose the 'variation' of the group of pixels $(x_1, \ldots, x_n)$ as the discrimination function f:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i| . \quad (1)$$

Finally, we define an invertible operation F on P called "flipping". Flipping is a permutation of gray levels that entirely consists of two-cycles. Thus, F^2 = Identity or $F(F(x)) = x$ for all $x \in P$. The permutation F_1: $0 \leftrightarrow 1$, $2 \leftrightarrow 3$, …, $254 \leftrightarrow 255$ corresponds to flipping (negating) the LSB of each gray level. We further define so called shifted LSB flipping F_{-1} as $-1 \leftrightarrow 0$, $1 \leftrightarrow 2$, $3 \leftrightarrow 4$, …, $253 \leftrightarrow 254$, $255 \leftrightarrow 256$, or

$$F_{-1}(x) = F_1(x+1) - 1 \text{ for all } x. \quad (2)$$

For completeness, we also define F_0 as the identity permutation $F(x)=x$ for all $x \in P$. We use the discrimination function f and the flipping operation F to define three types of pixel groups: R, S, and U

$$\begin{aligned} &\underline{\text{R}}\text{egular groups: } G \in R \;\Leftrightarrow\; f(F(G)) > f(G) \\ &\underline{\text{S}}\text{ingular groups: } G \in S \;\Leftrightarrow\; f(F(G)) < f(G) \\ &\underline{\text{U}}\text{nusable groups: } G \in U \;\Leftrightarrow\; f(F(G)) = f(G), \end{aligned}$$

where $F(G)= (F(x_1), \ldots, F(x_n))$. We may wish to apply different flipping to different pixels in the group G. The assignment of flipping to pixels can be captured with a mask M, which is a n-tuple with values −1, 0, and 1. The flipped group $F_M(G)$ is defined as $(F_{M(1)}(x_1), F_{M(2)}(x_2), \ldots, F_{M(n)}(x_n))$. Our stego-detection technique is based on analyzing how the number of regular and singular groups changes with the increased message length embedded in the LSB plane. This stego-detection technique is a result of pure serendipity stemming from our research on lossless data embedding [1] (for more details, see the journal version of this paper [9]).

3. STEGANALYTIC TECHNIQUE

Let us denote the number of regular groups for mask M as R_M (in percents of all groups). Similarly, S_M will denote the relative number of singular groups. We have $R_M + S_M \leq 1$ and $R_{-M} + S_{-M} \leq 1$, for the negative mask. The statistical hypothesis of our steganalytic method is that in a typical image, the expected value of R_M is equal to that of R_{-M}, and the same is true for S_M and S_{-M}:

$$R_M \cong R_{-M} \text{ and } S_M \cong S_{-M} . \quad (3)$$

This hypothesis can be heuristically justified by inspecting the expression (2). Applying the flipping operation F_{-1} is the same as applying F_1 to an image whose colors have been shifted by one. For a typical image, there is no a priori reason why the number of R and S groups should change significantly by shifting the colors by one. This assumption has been experimentally verified for images taken with a digital camera for both lossy and lossless formats. It also holds well for images processed with common image processing operations and for most scanned images. The relationship (2), however, is violated after randomizing the LSB plane, for example due to LSB steganography!

Randomization of the LSB plane forces the difference between R_M and S_M to zero as the length m of the embedded message increases. After flipping the LSB of 50% of pixels (which is what would happen after embedding a random message bit into every pixel), we obtain $R_M \cong S_M$. This is equivalent to saying that the lossless embedding capacity in the LSB plane is zero [1]. What is surprising is that the influence of randomizing the LSB plane has the *opposite* effect on R_{-M} and S_{-M}. Their difference *increases* with the length m of the embedded message. The graph that shows R_M, S_M, R_{-M}, and S_{-M} as functions of the number of pixels with flipped LSBs appears in Figure 1 (the RS diagram). The explanation of the peculiar increase in the difference between R_{-M} and S_{-M} is omitted for brevity (see the journal version of this paper [9]).

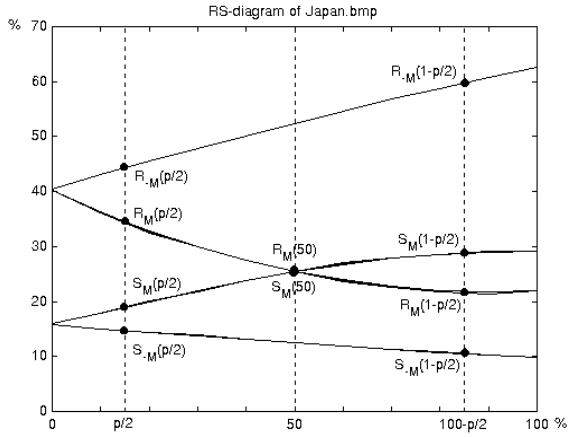

Figure 1. RS-diagram of a typical image. The *x*-axis is the relative number of pixels with flipped LSBs, the *y*-axis is the relative number of regular and singular groups with masks M and $-M$, M=[0 1 1 0]

The principle of our new steganalytic method (the RS method) is to estimate the four curves of the RS diagram and calculate their intersection using extrapolation. The general shape of the four curves in the diagram varies with the cover-image from almost perfectly linear to curved. We have collected experimental evidence that the R_{-M} and S_{-M} curves are well modeled with straight lines, while the "inner" curves R_M and S_M can be

reasonably well approximated with second degree polynomials. The parameters of the curves can be determined from the points marked in Figure 1.

If we have a stego-image with a message of an unknown length p (in percents of pixels) embedded in the LSBs of randomly scattered pixels, our initial measurements of the number of R and S groups correspond to the points $R_M(p/2)$, $S_M(p/2)$, $R_{-M}(p/2)$, and $S_{-M}(p/2)$ (see Figure 1). The factor of one half is due to the fact that, assuming the message is a random bit-stream, on average only one half of the pixels will be flipped. If we flip the LSBs of *all* pixels in the image and calculate the number of R and S groups, we will obtain the four points $R_M(1-p/2)$, $S_M(1-p/2)$, $R_{-M}(1-p/2)$, and $S_{-M}(1-p/2)$ (see Figure 1). By randomizing the LSB plane of the stego-image, we will obtain the middle points $R_M(1/2)$ and $S_M(1/2)$. We can fit straight lines through the points $R_{-M}(p/2)$ $R_{-M}(1-p/2)$ and $S_{-M}(p/2)$ $S_{-M}(1-p/2)$. The points $R_M(p/2)$, $R_M(1/2)$, $R_M(1-p/2)$, and $S_M(p/2)$, $S_M(1/2)$, $S_M(1-p/2)$ determine the two parabolas.

It is possible to avoid the time consuming statistical estimation of the middle points $R_M(1/2)$ and $S_M(1/2)$ and, at the same time, make the message length estimation much more elegant by accepting two additional (natural) conditions:

1. The point of intersection of the curves R_M and R_{-M} has the same x coordinate as the point of intersection for the curves S_M and S_{-M}. This is essentially a stronger version of our assumption (2).
2. The curves R_M and S_M intersect at m=50%, or $R_M(1/2) = S_M(1/2)$. This assumption is equivalent to saying that the lossless embedding capacity for a randomized LSB plane is zero [1].

These assumptions have been experimentally verified for a large database of images with unprocessed raw BMPs, JPEGs, and processed BMP images. They make it possible to derive a simple formula for the secret message length p. After linearly rescaling the x axis so that $p/2$ becomes 0 and $100-p/2$ becomes 1, the x-coordinate of the intersection point is a root of the following quadratic equation

$$2(d_1 + d_0)x^2 + (d_{-0} - d_{-1} - d_1 - 3d_0)x + d_0 - d_{-0} = 0, \text{ where}$$

$$d_0 = R_M(p/2) - S_M(p/2), \qquad d_1 = R_M(1-p/2) - S_M(1-p/2),$$
$$d_{-0} = R_{-M}(p/2) - S_{-M}(p/2), \quad d_{-1} = R_{-M}(1-p/2) - S_{-M}(1-p/2).$$

The message length p is calculated from the root x whose absolute value is smaller as,

$$p = x/(x-1/2).$$

Due to space limitations, we omit the derivation of these equations. Suffice it to say that the straight lines are defined by the number of R and S groups at $p/2$ and $1-p/2$, and the assumptions 1 and 2 provide enough constraints to uniquely determine the parabolas and their intersections.

3.1 Accuracy

There are several factors that influence the accuracy of the estimated message length.

Initial bias: Even original cover-images may indicate a small non-zero message length due to random variations. This initial non-zero bias could be both positive and negative and it puts a limit on the theoretical accuracy of our steganalytic method. We have tested this initial bias for a large database of 331 grayscale JPEG images and obtained a Gaussian distribution with a standard deviation of 0.5%. Smaller images tend to have higher variation in the initial bias due to smaller number of R and S groups. Scans of half-toned images and noisy images exhibit larger variations in the bias as well. On the other hand, the bias is typically very low for JPEG images, uncompressed images obtained by a digital camera, and high resolution scans. As another rule of thumb, we state that color images exhibit larger variation in the initial bias than grayscales.

Noise: For very noisy images, the difference between the number of regular and singular pixels in the cover image is small. Consequently, the lines in the RS diagram intersect at a small angle and the accuracy of the RS Steganalysis decreases.

Message placement: The RS Steganalysis is more accurate for messages that are randomly scattered in the stego-image than for messages concentrated in localized areas of the image. To address this issue, we can apply the same algorithm to a sliding rectangular region of the image. For sequentially embedded messages, the method described in [6] is also a good alternative.

4. EXPERIMENTAL RESULTS

In our first test, we used the Kodak DC260 digital camera and converted a color 1536×1024 image 'kyoto.bmp' to grayscale and down-sampled to 384×256 pixels. A series of stego-images was created from the original image by randomizing the LSBs of 0–100% pixels in 5% increments. We detected the number of pixels with flipped LSBs in each stego-image using our method. Groups of 2×2 pixels with the mask [1 0; 0 1] were used in our experiment.

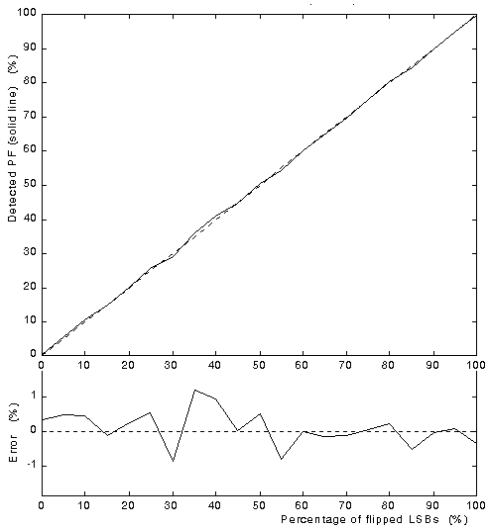

Figure 2. Estimated percentage of flipped pixels using the RS Steganalysis (solid line) vs. the actual number of flipped pixels for 'kyoto.bmp'. The bottom part of the figure shows the magnified detection error

The result, typical for images with an initial bias close to zero, is plotted in Figure 2. As can be seen from the chart, the error between the actual and estimated percentage of flipped pixels is almost always smaller than 1%.

The RS Steganalysis gave us wonderful and accurate detections for stego-images created using most commercial steganographic software products [9]. In all cases, stego-images were readily distinguished from original cover images and the estimated message length was within a few percent off the actual message length. The performance of the RS Steganalysis is demonstrated below on one relatively small image 'siesta.bmp' (24-bit color scan, 422×296, message=20% capacity, 100% = 3bpp) and a large image 'cat.bmp' (24-bit JPEG color image from Kodak DC260 cropped to 1024×744, message= 5%).

Table 1. Initial bias and estimated number of pixels with flipped LSBs for the test image siesta.bmp' and 'cat.bmp' (in parentheses). Note that the number of flipped pixels should be ½ of the message length

	'siesta.bmp'			'cat.bmp'		
	R	G	B	R	G	B
Bias	2.5	2.4	2.6	0.00	0.17	0.33
Steganos	10.6	13.3	12.4	2.41	2.70	2.78
S-Tools	13.4	11.4	10.3	2.45	2.62	2.75
Hide4PGP	12.9	13.8	13.0	2.44	2.62	2.85



'kyoto.bmp' 'siesta.bmp' 'cat.bmp'

The results shown in Table 2 demonstrate the extraordinary accuracy of the RS Steganalysis. Since the initial bias is about 2.5% in each color channel, as indicated in the first row of the table, the expected detected percentage of flipped pixels would be about 12.5%.

5. FUTURE DIRECTIONS

The subject of our future research will be focused on applying RS Steganalysis to palette images. One of the most common methods for GIF images is LSB embedding into the indices to a presorted palette.

We will also study the possibility of estimating the initial bias from stego images to improve the sensitivity of the RS detection method to short messages.

6. ACKNOWLEDGEMENT

The work on this paper was partially supported by Air Force Research Laboratory, Air Force Material Command, USAF, under a research grant number F30602-00-1-0521 and partially by the AFOSR grant No. F49620-01-1-0123. 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 Air Force Research Laboratory, the AFOSR, or the U. S. Government.

7. REFERENCES

[1] J. Fridrich, M. Goljan and R. Du, "Distortion-free Data Embedding for Images", *Proc. 4th Information Hiding Workshop*, Pittsburgh, Pennsylvania, April 25–27, 2001.

[2] R.J. Anderson and Petitcolas, F.A.P., "On the limits of steganography", *IEEE Journal of Selected Areas in Communications, Special Issue on Copyright and Privacy Protection* **16** No.4 (1998) 474–481.

[3] T. Aura, "Invisible communication", In *Proc. of the HUT Seminar on Network Security '95*, Espoo, Finland, November 1995.

[4] J. Fridrich, M. Goljan, and R. Du, "Steganalysis based on JPEG compatibility", submitted to *SPIE Multimedia Systems and Applications IV*, Denver, CO, August 20–24, 2001.

[5] J. Fridrich, R. Du, and L. Meng, "Steganalysis of LSB Encoding in Color Images", *ICME 2000*, New York City, July 31−August 2, New York.

[6] A. Westfeld and A. Pfitzmann, "Attacks on Steganographic Systems", *Proc. 3rd Information Hiding Workshop*, Dresden, Germany, September 28−October 1, 1999, pp. 61−75.

[7] N. F. Johnson and S. Jajodia, "Steganalysis of Images Created Using Current Steganography Software." *Proc. 2nd Information Hiding Workshop,* Portland, OR, April 1998.

[8] Steganography software for Windows, http://members.tripod.com/steganography/stego/software.html

[9] J. Fridrich and M. Goljan, "Steganalysis of LSB Embedding in Color and Grayscale Images", in preparation for the special issue on security in Magazine IEEE Multimedia.