LSB STEGANOGRAPHY WITH IMPROVED EMBEDDING EFFICIENCY AND UNDETECTABILITY

Omed Khalind and Benjamin Aziz

School of Computing, University of Portsmouth, Portsmouth, United Kingdom
Omed.khalind@port.ac.uk, Benjamin.Aziz@port.ac.uk

ABSTRACT

In this paper, we propose a new method of non-adaptive LSB steganography in still images to improve the embedding efficiency from 2 to 8/3 random bits per one embedding change even for the embedding rate of 1 bit per pixel. The method takes 2-bits of the secret message at a time and compares them to the LSBs of the two chosen pixel values for embedding, it always assumes a single mismatch between the two and uses the second LSB of the first pixel value to hold the index of the mismatch. It is shown that the proposed method outperforms the security of LSB replacement, LSB matching, and LSB matching revisited by reducing the probability of detection with their current targeted steganalysis methods. Other advantages of the proposed method are reducing the overall bit-level changes to the cover image for the same amount of embedded data and avoiding complex calculations. Finally, the new method results in little additional distortion in the stego image, which could be tolerated.

KEYWORDS

Steganography, Embedding efficiency, Probability of detection, Single Mismatch, LSB matching, LSB replacement

1. INTRODUCTION

Steganography is the art and the science of keeping the existence of messages secret rather than only their contents, as it is the case with cryptography. Both steganography and digital watermarking belong to information hiding, but they differ in their purpose. Digital watermarking is intended to protect the cover, whereas steganography is used to protect the message. So, steganography is considered broken when the existence of the secret message is detected. Hence, the most important property for every steganographic method is undetectability by the existing steganalysis techniques.

LSB steganography is the most widely used embedding method in pixel domain, since it is easy to implement, has reasonable capacity, and is visually imperceptible. Unfortunately, both methods of LSB steganography (LSB replacement and LSB matching) are detectable by the current steganalysis approaches discussed in later sections.

There are some methods proposed to improve the capacity of LSB replacement like[1,2], or to avoid changing the histogram of the cover image like [3] which reduce the embedding capacity by 50%. As mentioned earlier, the undetectability, or the probability of detection is the most important property for any steganographic method. In this paper a new method of non-adaptive

David C. Wyld et al. (Eds) : ITCS, CST, JSE, SIP, ARIA, DMS - 2015
pp. 89–105, 2015. © CS & IT-CSCP 2015 DOI : 10.5121/csit.2015.50110

LSB steganography is proposed to reduce the probability of detection for the same amount of data embedded with LSB replacement, LSB matching, and LSB matching revisited [4]by the current detection methods. The proposed method also results in fewer ENMPP (Expected Number of Modifications Per Pixel) in both pixel and bit-level to the cover image, and changes the histogram of the cover image in a different way without any complex calculation.

The paper is organized like the following; it starts with clarifying adaptive and non-adaptive steganography and the related embedding methods in the literature. Then, it starts analysing both LSB replacement and LSB matching in grey-scale images from different perspectives such as the embedding efficiency, histogram changes, and bit-level ENMPP. Then, the proposed method is explained and followed by the same analysis process. After that, the experimental results are shown for the proposed method against both steganalysis methods; LSB replacement and LSB matching. Finally, the conclusion and future work are discussed in the last section.

2. ADAPTIVE AND NON-ADAPTIVE LSB STEGANOGRAPHY IN IMAGE

The embedding process of LSB steganography relies on some methods for selecting the location of the change. In general, there are three selection rules to follow in order to control the location of change, which are either sequential, random, or adaptive [5].

A sequential selection rule modifies the cover object elements individually by embedding the secret message bits in a sequential way. For example, it is possible to embed the secret message by starting from the top-left corner of an image to the bottom-right corner in a row-wise manner. This selection rule, sequential, is very easy to implement, but has a very low security against detection methods.

A pseudo-random selection rule modifies the cover object by embedding the secret message bits into a pseudo randomly chosen subset of the cover object, possibly by using a secret key as a pseudo-random number generator (PRNG). This type of selection rule gives a higher level of security than sequential methods.

An adaptive selection rule modifies the cover object by embedding the secret message bits in selected locations based on the characteristics of the cover object. For example, choosing noisy and high textured areas of the image, which are less detectable than smooth areas for hiding data. This selection rule, adaptive, gives a higher security than sequential and pseudo-random selection rules in terms of detection.

So, the non-adaptive image steganography techniques are modifying the cover image for message embedding without considering its features (content). For example LSB replacement and LSB matching with sequential or random selection of pixels are modifying the cover image according to the secret message and the key of random selection of pixels without taking the cover image properties into account. Whereas, adaptive image steganography techniques are modify the cover image in correlation with its features [6]. In other words, the selection of pixel positions for embedding is adaptive depending on the content of the cover image. The bit-plane complexity segmentation (BPCS) proposed by Kawguchi[7] is an early typical method of adaptive steganography.

As adaptive steganographic schemes embed data in specific regions (such as edges), the steganographic capacity of such method is highly depend on the cover image used for embedding. Therefore, in general it is expected to have less embedding rate than non-adaptive schemes. However, steganographers have to pay this price in order to have a better security or less detectable stego image.

3. RELATED WORKS

The undetectability is the most important requirement of any steganographic scheme, which is affected by the choice of the cover object, the type of embedding method, the selection rule of modifying places, and the number of embedding changes which is directly related to the length of secret message[8].

If two different embedding methods share the same source of cover objects, the same selection method of embedding place, and the same embedding operation, the one with less number of embedding changes will be more secure (less detectable). This is because the statistical property of the cover object is less likely to be disrupted by smaller number of embedding changes[8].
The concept of embedding efficiency is introduced by westfeld[9], and then considered as an important feature of steganographic schemes[10,11], which is the expected number of embedded random message bits per single embedding change[12].

Reducing the expected number of modifications per pixel (ENMPP) is well studied in the literature considering the embedding rate of less than 1 , like westfeld's F5-algorithm[13], which could increase the embedding efficiency only for short messages. However, short messages are already challenging to detect. Also, the source coding-based steganography (matrix embedding) proposed by Fridrich et al.[8,12], which are extensions of F5-algorithm improved the embedding efficiency for large payloads but still with embedding rate of less than 1. The stochastic modulation proposed by Fridrich and Goljan[14], is another method of improving the security for the embedding rate of up to 0.8 bits/ pixel.

For the embedding rate of 1, there have been some methods for improving the embedding efficiency of LSB matching like Mielikainen[4], which reduced the ENMPP with the same message length from 0.5 to 0.375. The choice of whether to add or subtract one to/from a pixel value of their method relies on both the original pixel values and a pair of two consecutive secret bits. However, this method of embedding cannot be applied on saturated pixels (i.e. pixels with values 0 and 255), which is one of the drawbacks of this method. Then, the generalization method of LSB matching is proposed by Li et al.[15] with the same ENMPP for the same embedding rate using sum and difference covering set (SDCS). Another method of improving the embedding efficiency of LSB matching is proposed by Zhang et al.[16], using a combination of binary codes and wet paper codes, The embedding efficiency of this method can achieve the upper bound of the generalized ±1 embedding schemes.

However, no method could be found in the literature to improve the embedding efficiency of non-adaptive LSB replacement, which is 2 bits per embedding change, for the embedding rate of 1. So, developing such a method could be more useful than other adaptive methods in reusability perspective. Moreover, the non-adaptive LSB embedding methods with higher embedding efficiency can be used by existing adapted embedding methods to improve the steganographic capacity and reduce the probability of detection. A good example is the LSB matching revisited[4], which has been extended by[17,19].

Also, moving from non-adaptive to adaptive LSB embedding method does not mean that improving the non-adaptive methods are impossible or useless, as we mentioned earlier, the LSB matching revisited[4] is a very good example to support this fact.

4. ANALYSIS OF LSB REPLACEMENT

In this section, LSB replacement is analysed in three perspectives; the embedding process itself (with its embedding efficiency), its effect on the intensity histogram after embedding process, and the bit-level ENMPP for each bit of the secret message. Also, the main weaknesses of this embedding method are highlighted with the steganalysis methods that can detect it.

LSB replacement steganography simply replaces the LSB of the cover image pixel value with the value of a single bit of the secret message. It leaves the pixel values unchanged when their LSB value matches the bit value of the secret message and changes the mismatched LSB by either incrementing or decrementing the even or odd pixel values by one respectively[4], as shown in Figure 1.

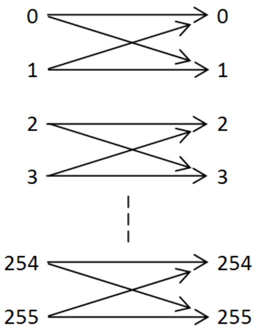

Figure1. Possible pixel value transitions with LSB replacement

The embedding algorithm of the LSB replacement can be formally described as follows:

$$P_s = \begin{cases} P_c + 1 & , if\ b \neq LSB(P_c)\ and\ P_c\ is\ even \\ P_c - 1 & , if\ b \neq LSB(P_c)\ and\ P_c\ is\ odd \\ P_c & , if\ b = LSB(P_c) \end{cases}$$

To analyse the influence of the LSB replacement on the cover image intensity histogram, we should consider that there is a probability of 50% for the LSB of the cover image pixel value to already have the desired bit value. Therefore, the probability of modified pixel values will be (P/2) for an embedding rate of P and the unmodified pixel values will be (1-P/2) after embedding process, which means that embedding each message bit needs 0.5 pixel values to be changed. In other words, it has an embedding efficiency of 2 bits of the secret message per one embedding change. Hence, the intensity histogram of the stego image could be estimated as follows:

$$\mathrm{h_s(n)} = \left(1 - \frac{\mathrm{P}}{2}\right)\mathrm{h_c(n)} + \frac{\mathrm{P}}{2}\begin{cases} \mathrm{h_c(n+1)} & \mathrm{, n\ is\ even} \\ \mathrm{h_c(n-1)} & \mathrm{, n\ is\ odd} \end{cases}$$

Where n is a greyscale level which ranges from 0 to 255, and h(n) indicates the number of pixels in the image with greyscale value of n.

This type of embedding, LSB replacement, leads to an imbalance distortion and produces ‘Pairs of Values’ on the intensity histogram of the stego image. Since LSB replacement is inherently asymmetric, current steganalysis methods can detect it easily[20], like: RS[21], SP[22], and WS[23,24].

Another way of analysing LSB embedding is the bit-level ENMPP, which is the expected number of bit modifications per pixel. This would be important too, as there are some steganalysis methods that can detect the existence of the secret message based on calculating several binary similarity measures between the 7th and 8th bit planes like[25]. Hence, an embedding process with less bit-level ENMPP would be better and less detectable by such detection methods.

The overall bit-level ENMPP for LSB replacement could be estimated by multiplying the probability of having mismatched LSBs, $\mathrm{P_r}(\overline{\mathrm{M}})$, which is 0.5 by the number of bits that needs to be changed in each case, as shown below.

$$\text{bit} - \text{level ENMPP} = \mathrm{P_r}(\overline{\mathrm{M}}) \times \text{no. of modified bits}$$
$$\text{bit} - \text{level ENMPP} = 0.5 \times 1 = 0.5 \quad \text{bits per message bits}$$

Hence, the overall bit-level ENMPP for LSB replacement is 0.5 bits for each bit of the secret message.

5. ANALYSIS OF LSB MATCHING

To analyse LSB matching steganography, we again consider the embedding process (with its embedding efficiency), its effect on the intensity histogram of the cover image, and bit-level ENMPP.

LSB matching or ±1 embedding is a modified version of LSB replacement. Instead of simply replacing the LSB of the cover image, it randomly either adds or subtracts 1 from the cover image pixel value that has mismatched LSB with the secret message bit[26]. The possible pixel value transitions of ±1 embedding are shown in Figure 2.

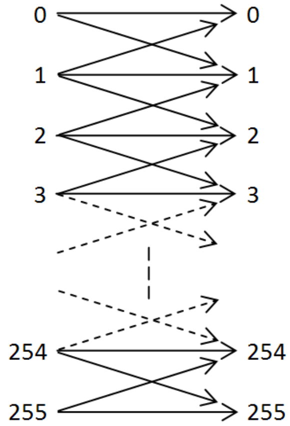

Figure 2. Possible pixel value transitions with LSB matching

The random increment or decrement in pixel values should maintain the boundary limitation and pixel values should always be between 0 and 255 [27]. In other words, the embedding process should neither subtract 1 from pixel values of 0 nor add 1 to the pixel values of 255.

This random ±1 change to the mismatched LSB pixel values avoids the asymmetry changes to the cover image, which is the case with LSB replacement. Hence, LSB matching is considered harder to detect than LSB replacement[4]. The embedding procedure of LSB matching can be formally represented as follows[28]:

$$\mathrm{P_s} = \begin{cases} \mathrm{P_c} + 1 & , \text{if } \mathrm{b} \neq \mathrm{LSB}(\mathrm{P_c}) \text{ and } (\mathrm{K} > 0 \ \mathit{or} \ \mathrm{P_c} = 0) \\ \mathrm{P_c} - 1 & , \text{if } \mathrm{b} \neq \mathrm{LSB}(\mathrm{P_c}) \text{ and } (\mathrm{K} < 0 \ \mathit{or} \ \mathrm{P_c} = 255) \\ \mathrm{P_c} & , \text{if } \mathrm{b} = \mathrm{LSB}(\mathrm{P_c}) \end{cases}$$

Where K is an independent and identically distributed random variable with uniform distribution on $\{-1, +1\}$.

For the intensity histogram we consider an embedding rate of P. There is a chance of 50% that the clean image pixel value contains the desired LSB, which means that $(\mathrm{P}/2)$ of the cover pixel values will change after the embedding process. Hence, the estimated unmodified pixel values will be $(1-\mathrm{P}/2)$, which means that embedding each message bit needs 0.5 pixel values to be changed. In other words, its embedding efficiency is 2 bits of the secret message per one embedding change. The intensity histogram of the stego image could be obtained as follows[28].

$$\mathrm{h_s(n)} = \left(1-\frac{\mathrm{P}}{2}\right)\mathrm{h_c(n)} + \frac{\mathrm{P}}{4}[\mathrm{h_c(n+1)} + \mathrm{h_c(n-1)}]$$

As mentioned earlier, the LSB matching will avoid the asymmetric property in modifying the cover image. However, as claimed by[29], ±1 embedding is reduced to a low pass filtering of the intensity histogram. This implies that the cover histogram contains more high-frequency power than the histogram of the stego image [28], which offers an opportunity to steganalyzers to detect the existence of the secret message embedded with LSB matching.

Apart from the supervised machine learning detectors of ±1 embedding like[30-33], which usually have problems in choosing an appropriate feature set and measuring classification error probabilities[34], the methods of detecting LSB matching steganography could be divided into two categories; the centre of mass of the histogram characteristic function (HCF) and the amplitude of local extrema (ALE)[35].

A number of detection methods have been proposed based on the centre of mass of the histogram characteristic function (HCF-COM) like Harmsen and Pearlman[36], which has better performance on RGB images than grey-scale. This method is modified and improved by Ker[27], who applied the HCF in two novel ways: using the down sampled image and computing the adjacency histogram.

Based on the amplitude of local extrema (ALE), Zhang et al.[29] considered the sum of the amplitudes of all local extrema in the histogram to distinguish between stego and clean images. This method is improved by Cancelli et al. [32] after reducing the border effects noise in the histogram and extending it to the amplitude of local extrema in the 2D adjacency histogram.
The bit-level ENMPP of LSB matching is also important and should be considered for the same reason, binary similarity measures. Since the probability of having mismatched LSB is also 50%, the bit-level ENMPP would be as follows:

$$\begin{aligned}&\mathrm{bit-level\ ENMPP} = \mathrm{P_r}\left(\overline{\mathrm{M}}\right) \times \mathrm{no.\,of\ modified\ bits}\\&\mathrm{bit-level\ ENMPP} = 0.5 \times (\geq 1)\\&\mathrm{bit-level\ ENMPP} \geq 0.5\ \mathrm{(bits\ per\ message\ bits)}\end{aligned}$$

Where $\mathrm{P_r}$ is the probability of having mismatched LSBs, which is 0.5. However, the number of modified bits would be more than 1, because of the random ±1 changes to the pixel values, as could be noted from the following examples:

$127\ (0111111)_2 + 1 = 128\ (10000000)_2$, 8-bits changed
$192\ (11000000)_2 - 1 = 191\ (10111111)_2$, 7-bits changed
$7\ (00000111)_2 + 1 = 8\ (00001000)_2$, 4-bits changed
$240\ (11110000)_2 - 1 = 239\ (11101111)_2$, 5-bits changed

Hence, the overall bit-level ENMPP for LSB matching is expected to be more than or equal to 0.5 bits for each bit of the secret message.

6. THE PROPOSED METHOD

Based on highlighting the weakest part of both LSB replacement and ±1 embedding, in this section we propose a new method of LSB embedding to improve the embedding efficiency and reduce the probability of detection by current steganalysis methods. Moreover, the new proposed method should also minimize the bit-level ENMPP to the cover image after embedding.

The new method, single mismatch LSB embedding (SMLSB), takes two bits of the secret message at a time and embeds them in a pair of selected pixel values of the cover image. The embedding method always assumes a single mismatch between the 2-bits of the secret message and the LSBs of the selected pair of pixel values. For each 2-bits of the secret message we consider two consecutive pixel values for simplicity. However, the selection could be based on other functions as well.

Since the proposed method embeds 2-bits at a time, there are four cases of having match (M) or mismatch ($\overline{M}$) between the LSBs of the selected two pixel values and the 2-bits of the secret message, as shown in Figure 3.

	LSB
Pixel value 1	M
Pixel value 2	M

	LSB
Pixel value 1	M
Pixel value 2	$\overline{M}$

	LSB
Pixel value 1	$\overline{M}$
Pixel value 2	M

	LSB
Pixel value 1	$\overline{M}$
Pixel value 2	$\overline{M}$

Figure 3. The possible cases of Match/ Mismatch

As the embedding method always assumes a single mismatch ($M\overline{M}$ or $\overline{M}M$) between pixel values and secret message bits, the 2nd LSB of the first pixel value should always refer to the index of the mismatch; 1 for $M\overline{M}$ and 0 for $\overline{M}M$. If the case is MM, then it changes one of the LSBs according to 2nd LSB of the first pixel value. If the 2nd LSB value was 0, then it flips the LSB of the first pixel value to create $\overline{M}M$. Otherwise, if it was 1, it flips the LSB of the second pixel value to create $M\overline{M}$. For the $\overline{MM}$ case, the embedding will also change one of the LSBs according to 2nd LSB of the first pixel value. But this time, if the 2nd LSB was 0, then it flips the LSB of the second pixel value to create $\overline{M}M$. Otherwise, if it was 1, it flips the LSB of the first pixel value to create $M\overline{M}$.

For the other two cases, $M\overline{M}$ and $\overline{M}M$, the embedding will be done by changing the 2nd LSB of the first pixel value based on the index of the mismatch. If it was $M\overline{M}$, then the 2nd LSB of the first pixel value will be set to 1. Otherwise, if it was $\overline{M}M$, then the 2nd LSB value of the first pixel value will be set to 0. Hence, after each embedding there is only $M\overline{M}$ or $\overline{M}M$ with the right index in the 2nd LSB of the first pixel value. The embedding algorithm is shown in Figure .

```
input: two cover pixel values $x_1, x_2$, and two message bits $b_1, b_2$
output: stego pixel values $y_1, y_2$
$y_1 = x_1$
$y_2 = x_2$
if $LSB(x_1) = b_1$ $AND$ $LSB(x_2) = b_2$
{
  if $2^{nd} LSB(x_1) = 0$
    $LSB(y_1) = \overline{b_1}$
  else
    $LSB(y_2) = \overline{b_2}$
}
else if $LSB(x_1) \neq b_1$ $AND$ $LSB(x_2) \neq b_2$
{
  if $2^{nd} LSB(x_1) = 0$
    $LSB(y_2) = \overline{b_2}$
  else
    $LSB(y_1) = \overline{b_1}$
}
else if $LSB(x_1) = b_1$ $AND$ $LSB(x_2) \neq b_2$
  $2^{nd} LSB(y_1) = 1$
else if $LSB(x_1) \neq b_1$ $AND$ $LSB(x_2) = b_2$
  $2^{nd} LSB(y_1) = 0$
end
```

Figure 4. The embedding algorithm of SMLSB embedding

Table 1, shows some examples of the embedding process by the proposed method.

Table 1. Examples of SMLSB embedding process.

Clean pair of pixels	Two message bits	Stego pair of pixels
xxxxxx01 xxxxxxx1	11	xxxxxx0**0** xxxxxxx1
xxxxxx11 xxxxxxx0	10	xxxxxx11 xxxxxxx**1**
xxxxxx01 xxxxxxx1	00	xxxxxx01 xxxxxxx**0**
xxxxxx11 xxxxxxx0	01	xxxxxx1**0** xxxxxxx0
xxxxxx11 xxxxxxx0	11	xxxxxx11 xxxxxxx0
xxxxxx01 xxxxxxx1	10	xxxxxx**1**1 xxxxxxx1
xxxxxx11 xxxxxxx0	01	xxxxxx**0**1 xxxxxxx0
xxxxxx00 xxxxxxx0	10	xxxxxx00 xxxxxxx0

7. Analysis of SMLSB Embedding

To analyse the proposed LSB embedding, just like other embedding methods mentioned earlier, we consider the embedding process itself (with its embedding efficiency), its effect on the intensity histogram of the image, and the bit-level ENMPP as well.

SMLSB embedding modifies the pixel values based on the match/mismatch cases between LSBs of the selected two pixel values and the 2-bits of the secret message. As it uses the 2nd LSB of the first selected pixel value to refer to the index of the mismatch, it modifies the first pixel value differently from the second one in the selected pair of pixels. The embedding algorithm could be formulated in two separate forms as follows.

$$p_s^{(2i)} = \begin{cases} p_c^{(2i)} + 2 & ,if\ b_{2i} = LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i)} - 2 & ,if\ b_{2i} \neq LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} = LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} + 1 & ,if\ b_{2i} = \left[LSB\left(p_c^{(2i)}\right) = 0\right] AND\ b_{2i+1} = LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ & \quad OR\ b_{2i} \neq \left[LSB\left(p_c^{(2i)}\right) = 0\right]\ AND\ b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} - 1 & ,if\ b_{2i} = \left[LSB\left(p_c^{(2i)}\right) = 1\right]\ AND\ b_{2i+1} = LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ & \quad OR\ b_{2i} \neq \left[LSB\left(p_c^{(2i)}\right) = 1\right]\ AND\ b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right)\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} & ,Otherwise \end{cases}$$

$$p_s^{(2i+1)} = \begin{cases} p_c^{(2i+1)} + 1 & ,if\ b_{2i} = LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} = \left[LSB\left(p_c^{(2i+1)}\right) = 0\right]\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ & \quad OR\ b_{2i} \neq LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} \neq \left[LSB\left(p_c^{(2i+1)}\right) = 0\right]\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i+1)} - 1 & ,if\ b_{2i} = LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} = \left[LSB\left(p_c^{(2i+1)}\right) = 1\right]\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ & \quad OR\ b_{2i} \neq LSB\left(p_c^{(2i)}\right)\ AND\ b_{2i+1} \neq \left[LSB\left(p_c^{(2i+1)}\right) = 1\right]\ AND\ 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i+1)} & ,Otherwise \end{cases}$$

Where i is the index of the secret message bit. The $p_s^{(2i)}$ and $p_c^{(2i)}$ refer to the stego and clean pixel values respectively for the 2ith secret message bit embedding. The $p_s^{(2i+1)}$ and $p_c^{(2i+1)}$ are again refer to the stego and clean pixel values used for embedding 2i+1th secret message bit.

The possible pixel value changes with SMLSB embedding could be simplified by separating the first $p_s^{(2i)}$ and second $p_s^{(2i+1)}$ pixel values from the selected pair, as shown in Figure 5 and Figure 6.

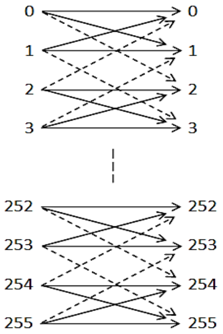

Figure 5. Possible pixel value transitions for $p_s^{(2i)}$ with SMLSB embedding

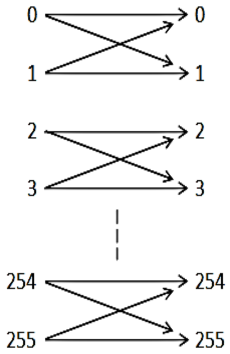

Figure 6. Possible pixel value transitions for $p_s^{(2i+1)}$ with SMLSB embedding

As could be noted from Figure and Figure , the pixel value transitions of $p_s^{(2i+1)}$ are like LSB replacement. While $p_s^{(2i)}$ is more complicated and has more transitions between clean and stego pixel values.

To analyse the impact of the SMLSB embedding on the intensity histogram, again we consider an embedding rate of P. Since the secret message is considered as a random sequence of 0 and 1, based on the fact that it will be close to its encrypted version [37], equal probabilities should be considered for match/mismatch cases. Hence, for each case of $(\mathrm{MM}, \mathrm{M}\overline{\mathrm{M}}, \overline{\mathrm{M}}\mathrm{M}, \overline{\mathrm{MM}})$ the probability of occurrence would be 0.25.

For MM and $\overline{\mathrm{MM}}$, the embedding process will change one of the two selected pixel values according to the 2nd LSB of the $p_c^{(2i)}$ to get either $\mathrm{M}\overline{\mathrm{M}}$ or $\overline{\mathrm{M}}\mathrm{M}$. The change will be -1 or +1 for the odd and the even pixel values respectively. So, $(\mathrm{P}/4)$ of the pixel values will be modified by adding or subtracting 1 according to their values, even or odd values respectively.

However, for $\mathrm{M}\overline{\mathrm{M}}$ and $\overline{\mathrm{M}}\mathrm{M}$ there is a probability of having 50% of the 2nd LSB of the $p_c^{(2i)}$ to have the desired value, which needs no change. The other 50% will be modified by flipping the 2nd LSB of the $p_c^{(2i)}$ only. In other word $(\mathrm{P}/8)$ of the pixel values will either incremented or decremented by 2 according to their 2nd LSB value. Hence, the remaining $(1-3\mathrm{P}/8)$ pixel values will stay unchanged after embedding the secret message with the embedding rate of P, which means that embedding each message bit needs 0.375 pixel values to be changed. This ENMPP, 0.375, is better than LSB replacement and LSB matching, which are 0.5 pixels per message bit. Hence, it improves the embedding efficiency from 2 to 8/3 bits per embedding change. The intensity histogram of the stego image could be estimated by the following:

$$\mathrm{h_s(n)} = \left(1-\frac{3\mathrm{P}}{8}\right)\mathrm{h_c(n)} + \frac{\mathrm{P}}{8}\begin{cases}\mathrm{h_c(n+2)} & \text{, if } 2^{\mathrm{nd}}\ \mathrm{LSB(n)}=0\\ \mathrm{h_c(n-2)} & \text{, if } 2^{\mathrm{nd}}\ \mathrm{LSB(n)}=1\end{cases} + \frac{\mathrm{P}}{4}\begin{cases}\mathrm{h_c(n+1)} & \text{, n is even}\\ \mathrm{h_c(n-1)} & \text{, n is odd}\end{cases}$$

Where, n is again the greys-cale level valued between 0 and 255. Both $\mathrm{h_s(n)}$ and $\mathrm{h_c(n)}$ refer to the number of pixels in the stego and clean image respectively with the greyscale value of n.

As only $(\mathrm{P}/4)$ of the pixel values are modified like LSB replacement, it is expected to effectively reduce the probability of detection with LSB replacement steganalysis methods. Also, it is expected to reduce the probability of detection by LSB matching steganalysis methods as well, based on the dissimilarity in pixel value transitions and its influence on the intensity histogram after embedding.

The bit-level ENMPP for the proposed method could be calculated based on the match/mismatch cases, in which equal probabilities are considered.

$$\mathrm{bit-level\ ENMPP} = \frac{\sum(\mathrm{P_r}[\text{each case}] \times \text{no. of modified bits})}{2}$$

$$\mathrm{bit-level\ ENMPP} = \frac{\mathrm{P_r}(\mathrm{MM}) \times 1 + \mathrm{P_r}\left(\mathrm{M}\overline{\mathrm{M}}\right) \times 0.5 + \mathrm{P_r}\left(\overline{\mathrm{M}}\mathrm{M}\right) \times 0.5 + \mathrm{P_r}\left(\overline{\mathrm{MM}}\right) \times 1}{2}$$

$$\mathrm{bit-level\ ENMPP} = \frac{0.25 \times 1 + 0.25 \times 0.5 + 0.25 \times 0.5 + 0.25 \times 1}{2}$$

$$\mathrm{bit-level\ ENMPP} = \frac{0.75}{2} = 0.375 \quad \text{bits per message bit}$$

The bit-level ENMPP is divided by two, as it embeds two bits of the secret message at a time. In this case the overall bit-level ENMPP for the proposed method will be 0.375 bits per message bit. Hence, the proposed method will result in fewer bit-level changes to the cover image after embedding the same amount of secret message.

8. EXPERIMENTAL RESULTS

To make the experimental results more reliable, two sets of images are considered. The first set is 3000 images from ASIRRA (Animal Species Image Recognition for Restricting Access) public corpus pet images from Microsoft research website[38], which are random with different sizes, compression rates, texture ...etc. The other group is a set of 3000 never compressed images from Sam Houston state university – Multimedia Forensics Group image database [39]. Both sets are used after converting them into grey-scale images.

To check the efficiency of the proposed LSB embedding, both detection methods are considered; the LSB replacement and LSB matching steganalysis methods. In all experiments, streams of pseudo random bits are considered as a secret message. This is due to the fact that it will have all statistical properties of encrypted version of the secret message according to[40]. Also, to eliminate the effect of choosing the embedding place (random or sequential embedding), the embedding rate of 1 bit per pixel (i.e. the images' total capacity) is considered. Then it is tested against both LSB replacement and matching steganalysis methods as shown in the following sections.

8.1 SMLSB against LSB replacement steganalysis methods

There are many methods for detecting LSB replacement steganography in the literature, this paper considers two structural steganalysis methods, the Sample Pair (SP) analysis[41] and Weighted Stego (WS)[24]. As mentioned earlier, for each case, the image is loaded with the maximum capacity of the random secret message twice; one with LSB replacement and the other with SMLSB embedding.

The experimental results showed that the proposed method effectively reduce the probability of detection for both detection methods over both sets of images compared to LSB replacement, as shown in Table 2.

Table 2. The overall reduction rates in probability of detection.

Image set	Detection method	The overall reduction in probability of detection
ASIRRA	WS	46.5%
Uncompressed	WS	48.4%
ASIRRA	SP	30.9%
Uncompressed	SP	39.8%

Also, there is a noticeable reduction in probability of detection for the threshold values that suits the LSB replacement by SMLSB embedding as shown in Figures 7-10.

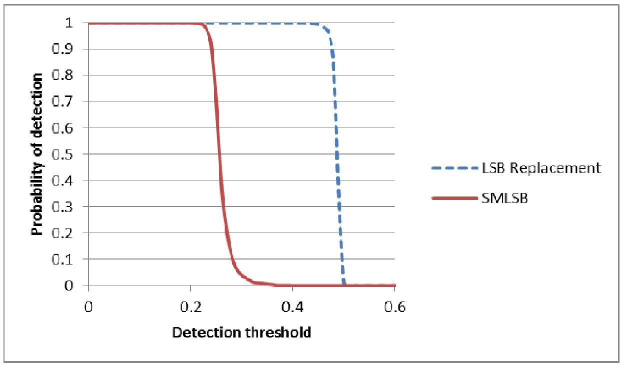

Figure 7. The probability of detection vs. detection threshold for ASIRRA images with WS.

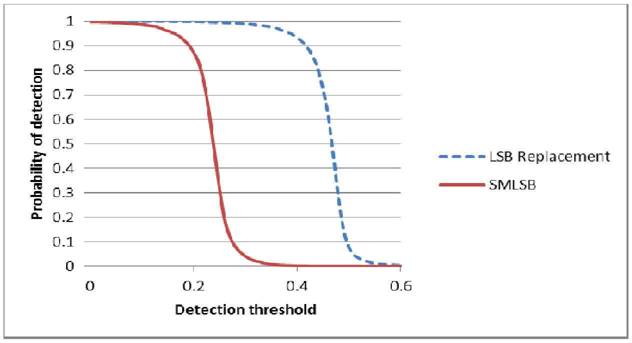

Figure 8. The probability of detection vs. detection threshold for uncompressed images with WS.

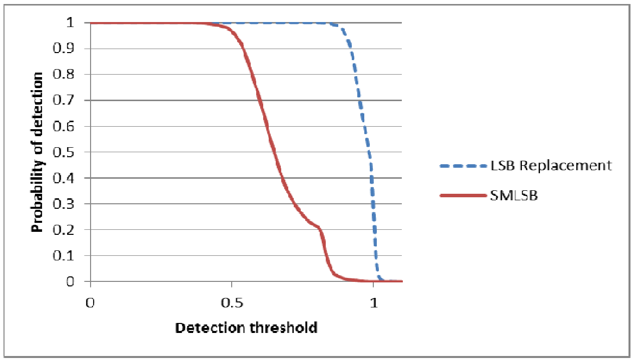

Figure 9. The probability of detection vs. detection threshold for ASIRRA images with SP.

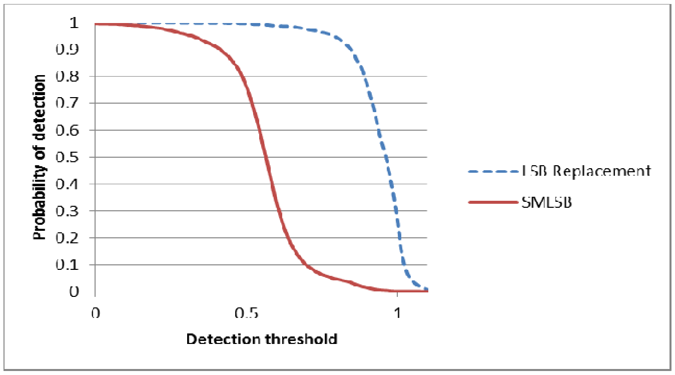

Figure 10. The probability of detection vs. detection threshold for uncompressed images with SP.

8.2 SMLSB against LSB matching steganalysis methods

As mentioned earlier, there are two main categories of LSB matching steganalysis methods. In this paper we use one detection method in each category. For the centre of mass of the histogram characteristic function (HCF-COM) we used Ker's method in[27], and for the amplitude of local extrema we used the method proposed by Zhang et al.[29].

The proposed method, SMLSB, outperforms both LSB matching and LSB matching revisited [4] embedding methods in terms of detection. Figures 11-14, show the ROC graph for each group of images with two different detection methods. As could be noticed from Figures 11 and 12, the ALE based steganalysis method is no more than a random classifier for the stego images embedded with SMLSB. Also, the performance of the HCF-COM based steganalysis method is considerably reduced by applying the SMLSB embedding method, as shown in Figures 13 and 14.

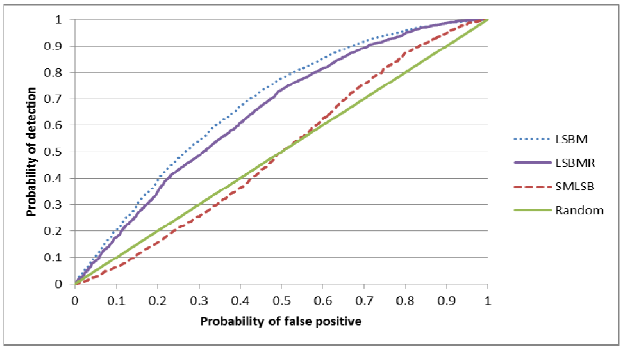

Figure 11. ROC graph of ALE steganalysis for LSB matching, LSB matching revisited, and SMLSB for ASIRRA images.

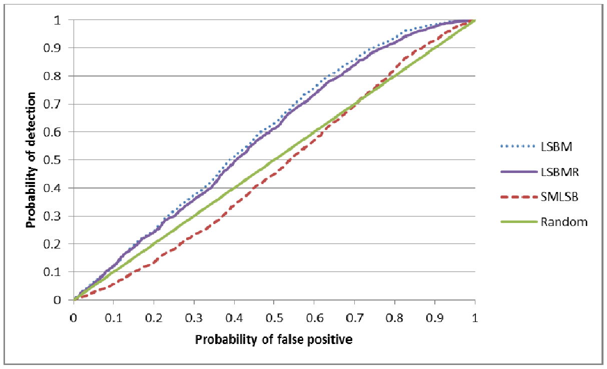

Figure 12. ROC graph of ALE steganalysis for LSB matching, LSB matching revisited, and SMLSB for Uncompressed images.

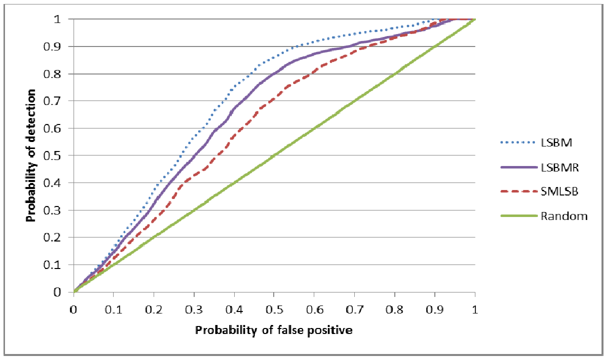

Figure 13. ROC graph of HCF-COM steganalysis for LSB matching, LSB matching revisited, and SMLSB for ASIRRA images.

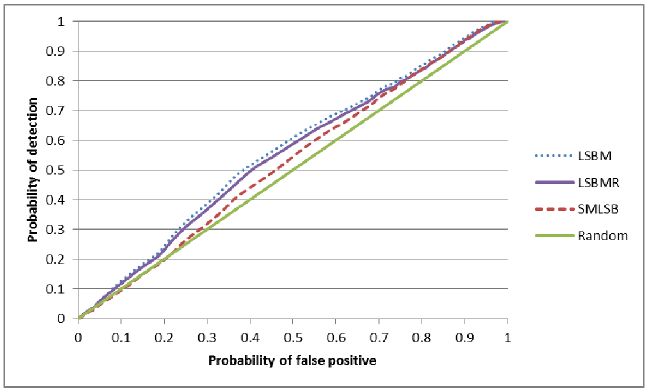

Figure 14. ROC graph of HCF-COM steganalysis for LSB matching, LSB matching revisited, and SMLSB for Uncompressed images.

Like any other steganography methods, the SMLSB cannot avoid all limitations and cannot totally defeat the detection methods. As could be noticed from Table 2 and Figures 7-14, it is not possible to entirely avoid the detection. Also, there is another weakness regarding the image quality measurement PSNR (Peak Signal to Noise Ratio) between the cover and a stego image. The proposed method results in a slightly lower PSNR than other methods; LSB replacement, LSB matching and LSB matching revisited, which is imperceptible and still very far from the lower limit value of PSNR (38 dB) according to [42, 43].

Table 3 shows the PSNR values for some standard images after embedding random binary streams with a maximum capacity using different embedding methods.

Table 13. PSNR values vs. embedding methods.

Images	LSB Replacement	LSB Matching	LSB Matching Revisited	SMLS B
Lena	50.88	50.88	52.13	49.12
Pepper	50.17	50.17	51.41	48.42
Baboon	50.28	50.28	51.53	48.52

9. EXTRACTION PROCESS

The extraction process is very simple, let s_1s_2 denote the least significant bits of the first and second selected pixel values respectively. It looks at the 2nd LSB of the first pixel value in the pair of pixels. If it is 0, then the LSBs of the pair of pixels would be extracted in the form of $\overline{s_1}s_2$ as two secret message bits, since, in this case, the mismatched LSB is in the first pixel value. If, on the other hand, it is 1, then it takes $s_1\overline{s_2}$ as an extracted message bits. Table 4, shows all different cases of extraction process.

Table 4. The extraction process.

The stego images pixel pair	Extracted message bits
$xxxxxx0s_1$ $xxxxxxxs_2$	$\overline{s_1}s_2$
$xxxxxx1s_1$ $xxxxxxxs_2$	$s_1\overline{s_2}$

Table 5, shows some examples of message bits extracted from stego pixel values.

Table 5. Examples of SMLSB extraction process.

The stego images pixel pair	Extracted message bits
xxxxxx01 xxxxxxx1	01
xxxxxx00 xxxxxxx1	11
xxxxxx11 xxxxxxx1	10
xxxxxx10 xxxxxxx1	00

10. CONCLUSION

In this study, we have shown that the proposed SMLSB method can improve the embedding efficiency in compare to LSB replacement and LSB matching from 2 to 8/3 and reduce the probability of detection by the two LSB steganalysis methods; LSB replacement and LSB matching. It also leaves a higher rate of pixel values unchanged for embedding the same amount of secret messages compared with other two LSB steganography methods. Moreover, the proposed method outperforms the LSB matching revisited, which has the same embedding efficiency, in terms of detection. Also, it can be applied to any pixel without restricting the saturated values (0 and 255). All embedding methods are analysed in detail including SMLSB and highlighted the cause of reducing the probability of detection. As could be noticed, the proposed method is very simple to implement with no complex calculation, less bit-level ENMPP on the cover image, and no reduction in the embedding capacity compared to other two LSB steganography methods, LSB replacement and LSB matching.

Finally, reducing the probability of detection by LSB replacement steganalysis methods is limited and the new method cannot totally avoid it. Also, it results in slightly more distortion in comparison to LSB replacement and LSB matching methods. As future work, it might be possible to modify the proposed method to give lower probability of detection and lower ENMPP for the same message length.

REFERENCES

[1] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," Pattern Recognition Letters, vol. 24, pp. 1613-1626, 2003.

[2] H.-C. Wu, N.-I. Wu, C.-S. Tsai, and M.-S. Hwang, "Image steganographic scheme based on pixel-value differencing and LSB replacement methods," IEE Proceedings-Vision, Image and Signal Processing, vol. 152, pp. 611-615, 2005.

[3] H.-M. Sun, Y.-H. Chen, and K.-H. Wang, "An image data hiding scheme being perfectly imperceptible to histogram attacks," Image and Vision Computing New Zealand IVCNZ, vol. 16, pp. 27-29, 2006.

[4] J. Mielikainen, "LSB matching revisited," Signal Processing Letters, IEEE, vol. 13, pp. 285-287, 2006.

[5] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, Digital Watermarking and Steganography: Morgan Kaufmann Publishers Inc., 2008.

[6] J. Fridrich and R. Du, "Secure Steganographic Methods for Palette Images," in Information Hiding. vol. 1768, A. Pfitzmann, Ed., ed: Springer Berlin Heidelberg, 2000, pp. 47-60.

[7] E. Kawaguchi and R. O. Eason, "Principles and applications of BPCS steganography," 1999, pp. 464-473.

[8] J. Fridrich and D. Soukal, "Matrix embedding for large payloads," 2006, pp. 60721W-60721W-12.

[9] A. Westfeld and A. Pfitzmann, "High capacity despite better steganalysis (F5–a steganographic algorithm)," in Information Hiding, 4th International Workshop, 2001, pp. 289-302.

[10] P. SALLEE, "MODEL-BASED METHODS FOR STEGANOGRAPHY AND STEGANALYSIS," International Journal of Image and Graphics, vol. 05, pp. 167-189, 2005.

[11] J. Fridrich, M. Goljan, and D. Soukal, "Steganography via codes for memory with defective cells," in 43rd Conference on Coding, Communication, and Control, 2005.

[12] J. Fridrich, P. Lisoněk, and D. Soukal, "On Steganographic Embedding Efficiency," in Information Hiding. vol. 4437, J. Camenisch, C. Collberg, N. Johnson, and P. Sallee, Eds., ed: Springer Berlin Heidelberg, 2007, pp. 282-296.

[13] A. Westfeld, "F5—A Steganographic Algorithm," in Information Hiding. vol. 2137, I. Moskowitz, Ed., ed: Springer Berlin Heidelberg, 2001, pp. 289-302.

[14] J. Fridrich and M. Goljan, "Digital image steganography using stochastic modulation," 2003, pp. 191-202.

[15] X. Li, B. Yang, D. Cheng, and T. Zeng, "A generalization of LSB matching," Signal Processing Letters, IEEE, vol. 16, pp. 69-72, 2009.

[16] Z. Weiming, Z. Xinpeng, and W. Shuozhong, "A Double Layered &#x201C;Plus-Minus One&#x201D; Data Embedding Scheme," Signal Processing Letters, IEEE, vol. 14, pp. 848-851, 2007.

[17] L. Weiqi, H. Fangjun, and H. Jiwu, "Edge Adaptive Image Steganography Based on LSB Matching Revisited," Information Forensics and Security, IEEE Transactions on, vol. 5, pp. 201-214, 2010.

[18] W. Huang, Y. Zhao, and R.-R. Ni, "Block Based Adaptive Image Steganography Using LSB Matching Revisited," Journal of Electronic Science and Technology, vol. 9, pp. 291-296, 2011.

[19] P. M. Kumar and K. L. Shunmuganathan, "Developing a Secure Image Steganographic System Using TPVD Adaptive LSB Matching Revisited Algorithm for Maximizing the Embedding Rate," Information Security Journal: A Global Perspective, vol. 21, pp. 65-70, 2012/01/01 2012.

[20] A. D. Ker, "A fusion of maximum likelihood and structural steganalysis," in Information Hiding, 2007, pp. 204-219.

[21] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color, and gray-scale images," Multimedia, IEEE, vol. 8, pp. 22-28, 2001.

[22] S. Dumitrescu, X. Wu, and Z. Wang, "Detection of LSB steganography via sample pair analysis," Signal Processing, IEEE Transactions on, vol. 51, pp. 1995-2007, 2003.

[23] J. Fridrich and M. Goljan, "On estimation of secret message length in LSB steganography in spatial domain," in Electronic Imaging 2004, 2004, pp. 23-34.

[24] A. D. Ker and R. Böhme, "Revisiting weighted stego-image steganalysis," in Electronic Imaging 2008, 2008, pp. 0501-0517.

[25] I. Avcibas, N. Memon, and B. Sankur, "Steganalysis using image quality metrics," Image Processing, IEEE Transactions on, vol. 12, pp. 221-229, 2003.

[26] T. Sharp, "An implementation of key-based digital signal steganography," in Information Hiding, 2001, pp. 13-26.

[27] A. D. Ker, "Steganalysis of LSB matching in grayscale images," Signal Processing Letters, IEEE, vol. 12, pp. 441-444, 2005.

[28] L. Xi, X. Ping, and T. Zhang, "Improved LSB matching steganography resisting histogram attacks," in Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on, 2010, pp. 203-206.

[29] J. Zhang, I. J. Cox, and G. Doërr, "Steganalysis for LSB matching in images with high-frequency noise," in Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on, 2007, pp. 385-388.

[30] J. Fridrich, D. Soukal, and M. Goljan, "Maximum likelihood estimation of length of secret message embedded using±k steganography in spatial domain," in Electronic Imaging 2005, 2005, pp. 595-606.

[31] M. Goljan, J. Fridrich, and T. Holotyak, "New blind steganalysis and its implications," in Electronic Imaging 2006, 2006, pp. 607201-607201-13.

[32] G. Cancelli, G. Doërr, I. J. Cox, and M. Barni, "Detection of±1 LSB steganography based on the amplitude of histogram local extrema," in Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, 2008, pp. 1288-1291.

[33] K. Sullivan, U. Madhow, S. Chandrasekaran, and B. Manjunath, "Steganalysis for Markov cover data with applications to images," Information Forensics and Security, IEEE Transactions on, vol. 1, pp. 275-287, 2006.

[34] R. Cogranne and F. Retraint, "An asymptotically uniformly most powerful test for LSB matching detection," 2013.

[35] G. Cancelli, G. Doerr, M. Barni, and I. J. Cox, "A comparative study of ±1 steganalyzers," in Multimedia Signal Processing, 2008 IEEE 10th Workshop on, 2008, pp. 791-796.

[36] J. J. Harmsen and W. A. Pearlman, "Steganalysis of additive-noise modelable information hiding," 2003, pp. 131-142.

[37] R. Chandramouli and N. Memon, "Analysis of LSB based image steganography techniques," in Image Processing, 2001. Proceedings. 2001 International Conference on, 2001, pp. 1019-1022.

[38] J. Douceur, J. Elson, and J. Howell. ASIRRA -- Public Corpus. Available: http://research.microsoft.com/en-us/projects/asirra/corpus.aspx

[39] Never-compressed image database. Available: http://www.shsu.edu/~qxl005/New/Downloads/index.html

[40] A. Westfeld and A. Pfitzmann, "Attacks on Steganographic Systems," in Information Hiding. vol. 1768, A. Pfitzmann, Ed., ed: Springer Berlin Heidelberg, 2000, pp. 61-76.

[41] S. Dumitrescu, X. Wu, and N. Memon, "On steganalysis of random LSB embedding in continuous-tone images," in Image Processing. 2002. Proceedings. 2002 International Conference on, 2002, pp. 641-644.

[42] K. Zhang, H.-Y. Gao, and W.-s. Bao, "Stegananlysis Method of Two Least-Significant Bits Steganography," in International Conference on Information Technology and Computer Science, 2009. ITCS 2009., 2009, pp. 350-353.

[43] F. A. P. Petitcolas and R. J. Anderson, "Evaluation of copyright marking systems," in Multimedia Computing and Systems, 1999. IEEE International Conference on, 1999, pp. 574-579 vol.1.