Keywords Steganography, least-significant-bit (LSB)-based steganography, edge detection, Canny filter, security, syndrome trellis codes

Noname manuscript No.
(will be inserted by the editor)

STABYLO: STeganography with Adaptive, Bbs, and binarY embedding at LOw cost

Jean-François Couchot, Raphael Couturier, and Christophe Guyeux

Received: date / Accepted: date

Abstract A new steganographic method called STABYLO is introduced in this research work. Its main advantage is to be much lighter than the so-called HUGO, WOW, and UNIWARD schemes, the state of the art steganographic processes. To achieve the proposed goal, famous experimented components of signal processing, coding theory, and cryptography are combined together, leading to a scheme that can reasonably face up-to-date steganalysers.

1 Introduction

This research work takes place in the field of information hiding, considerably developed these last two decades. The proposed method for steganography considers digital images as covers. It belongs to the well-known large category of spatial least significant bits (LSBs) replacement schemes. Let us recall that, in this LSB replacement category, a subset of all the LSBs of the cover image is modified with a secret bit stream depending on: a secret key, the cover, and the message to embed. In this well-studied steganographic approach, if we consider that a LSB is the last bit of each pixel value, pixels with an even value (resp. an odd value) are never decreased (resp. increased), thus such schemes may break the structural symmetry of the host images. And these structural alterations can be detected by well-designed statistical investigations, leading to well-known steganalysis methods [5, 6, 10].

Let us recall too that this drawback can be fixed by considering the LSB matching (LSBM) subcategory, in which a $+1$ or -1 is randomly added to the cover pixel's LSB value only if this one does not correspond to the secret bit. By considering well-encrypted hidden messages, the probabilities of increasing or decreasing the value of pixels are equal. Then usual statistical approaches cannot be applied here to discover stego-contents in LSBM. The most accurate detectors for this matching are universal steganalysers such as [11, 15, 20], which classify images according to extracted features from neighboring elements of residual noise.

Finally, LSB matching revisited (LSBMR) has recently been introduced in [24]. It works as follows: for a given pair of pixels, the LSB of the first pixel carries a first bit of the secret message, while the parity relationship (odd/even combination) of the two pixel values carries a second bit of the message. By doing so, the modification rate of pixels can decrease from 0.5 to 0.375 bits/pixel (bpp) in the case of a maximum embedding rate, meaning fewer changes in the cover image at the same payload compared to both LSBR and LSBM. It is also shown in [24] that such a new scheme can avoid the LSB replacement style asymmetry, and thus it should make the detection slightly more difficult than in the LSBM approach.

Additionally to (efficiently) modifying LSBs, there is also a need to select pixels whose value modification minimizes a distortion function. This distortion may be computed thanks to feature vectors that are embedded for instance in the steganalysers referenced above. The Highly Undetectable steGO (HUGO) method [26], WOW [14], and UNIWARD [13] are some of the most efficient instances of such a scheme.

HUGO takes into account so-called SPAM features. Thus a distortion measure for each pixel is individually determined as the sum of the differences between the features of the SPAM computed from the cover and from the stego images. The features embedded in WOW and UNIWARD are based on Wavelet-based directional filter. Thus, similarly, the dis-

Authors in alphabetic order

Jean-François Couchot, Raphael Couturier, and Christophe Guyeux
FEMTO-ST Institute, UMR 6174 CNRS
Computer Science Laboratory DISC, University of Franche-Comté, Besançon, France. E-mail: {jean-francois.couchot, raphael.couturier, christophe.guyeux}@univ-fcomte.fr

tortion function is the sum of the differences between these wavelet coefficients computed from the cover and from the stego images. Due to this distortion measures, HUGO, WOW and UNIWARD allow to embed messages that are 7 times longer than the former ones with the same level of indetectability as LSB matching. However, this improvement has a larger computation cost, mainly due to the distortion function computation.

There remains a large place between random selection of LSB and feature based modification of pixel values. We argue that modifying edge pixels is an acceptable compromise. Edges form the outline of an object: they are the boundaries between overlapping objects or between an object and its background. When producing the stego-image, a small modification of some pixel values in such edges should not impact the image quality, which is a requirement when attempting to be undetectable. Indeed, in a cover image, edges already break the continuity of pixels' intensity map and thus already present large variations with their neighboring pixels. In other words, minor changes in regular areas are more dramatic than larger modifications in edge ones. Our first proposal is thus to embed message bits into edge shapes while preserving other smooth regions.

Edge based steganographic schemes have already been studied, the most interesting approaches being detailed in [23] and in [4]. In the former, the authors present the Edge Adaptive Image Steganography based on LSB matching revisited further denoted as EAISLSBMR. This approach selects sharper edge regions with respect to a given embedding rate: the larger the number of bits to be embedded is, the coarser the edge regions are. Then the data hiding algorithm is achieved by applying LSBMR on some of the pixels of these regions. The authors show that their proposed method is more efficient than all the LSB, LSBM, and LSBMR approaches through extensive experiments. However, it has been shown that the distinguishing error with LSB embedding is lower than the one with some binary embedding [7]. We thus propose to take advantage of this optimized embedding, provided they are not too time consuming. In the latter, an hybrid edge detector is presented followed by an ad hoc embedding. The Edge detection is computed by combining fuzzy logic [27] and Canny [3] approaches. The goal of this combination is to enlarge the set of modified bits to increase the payload of the data hiding scheme.

One can notice that all the previously referenced schemes [4, 23, 26] produce stego contents by only considering the payload, not the type of image signal: the higher the payload is, the better the approach is said to be. For instance, studied payloads range from 0.04 to 0.4 modified bits per pixel. Contrarily, we argue that some images should not be taken as a cover because of the nature of their signals. Consider for instance a uniformly black image: a very tiny modification of its pixels can be easily detected. Practically speaking, if Alice would send a hidden message to Bob, she would never consider such kind of image and a high embedding rate. *This desire to be adaptive has been studied too in [21], but in JPEG frequency domain*. The approach we propose here is thus to provide a small complexity self adaptive algorithm with an acceptable payload, which depends on the cover signal. The payload is further said to be acceptable if it allows to embed a sufficiently long message in the cover signal. Practically speaking, our approach is efficient enough for payloads close to 0.06 bit per pixel which allows to embed messages of length larger than 15,728 bits in an image of size 512×512 pixels.

Finally, even if the steganalysis discipline has known great innovations these last years, it is currently impossible to prove rigorously that a given hidden message cannot be recovered by an attacker. This is why we add to our scheme a reasonable message encryption stage, to be certain that, even in the worst case scenario, the attacker will not be able to obtain the original message content. Doing so makes our steganographic protocol, to a certain extend, an asymmetric one.

To sum up, well-studied and experimented techniques of signal processing (adaptive edges detection), coding theory (syndrome-trellis codes), and cryptography (Blum-Goldwasser encryption protocol) are combined in this research work. The objective is to compute an efficient steganographic scheme, whose principal characteristic is to take into consideration the cover image and to be compatible with small computation resources.

The remainder of this document is organized as follows. Section 2 presents the details of the proposed steganographic scheme and applies it on a running example. Among its technical description, its adaptive aspect is emphasized. Section 3 presents the overall complexity of our approach and compares it to HUGO, WOW, and UNIWARD. Section 4 shows experiments on image quality, steganalysis evaluation, and compares them to the state of the art steganographic schemes. Finally, concluding notes and future work are given in Section 5.

2 Presentation of the Proposed Approach

This section first presents the embedding scheme through its four main steps: the data encryption (Sect. 2.1), the cover pixel selection (Sect. 2.2), the adaptive payload considerations (Sect. 2.3), and how the distortion has been minimized (Sect. 2.4). The message extraction is then presented (Sect. 2.5) while a running example ends this section.

The flowcharts given in Fig. 1 summarize our steganography scheme denoted by STABYLO, which stands for STeganography with Adaptive, Bbs, binarY embedding at LOw cost. What follows are successively some details of the inner steps and the flows both inside the embedding stage (Fig. 1a) and inside the extraction one (Fig. 1b). Let us first focus on the data embedding.

2.1 Security considerations

To provide a self-contained article without any bias, we shortly present the selected encryption process. Among the methods of message encryption/decryption (see [9] for a survey) we implement the asymmetric Blum-Goldwasser cryptosystem [2] that is based on the Blum Blum Shub [1] pseudorandom number generator (PRNG) and the XOR binary function. The main justification of this choice is that it has been proven [1] that this PRNG has the property of cryptographical security, *i.e.*, for any sequence of L output bits x_i, x_{i+1}, $\ldots$, x_{i+L-1}, there is no algorithm, whose time complexity is polynomial in L, and which allows to find x_{i-1} or x_{i+L} with a probability greater than $1/2$. Equivalent formulations of such a property can be found. They all lead to the fact that, even if the encrypted message is extracted, it is impossible to retrieve the original one in polynomial time.

Starting thus with a key k and the message *mess* to hide, this step computes a message m, which is the encrypted version of *mess*.

2.2 Edge-based image steganography

The edge-based image steganography schemes already presented [4,23] differ in how carefully they select edge pixels, and how they modify them.

Many techniques have been proposed in the literature to detect edges in images (whose noise has been initially reduced). They can be separated in two categories: first and second order detection methods on the one hand, and fuzzy detectors on the other hand [17]. In first order methods like Sobel, Canny [3], and so on, a first-order derivative (gradient magnitude, etc.) is computed to search for local maxima, whereas in second order ones, zero crossings in a second-order derivative, like the Laplacian computed from the image, are searched in order to find edges. As far as fuzzy edge methods are concerned, they are obviously based on fuzzy logic to highlight edges.

Canny filters, on their parts, are an old family of algorithms still remaining a state of the art edge detector. They can be well-approximated by first-order derivatives of Gaussians. As the Canny algorithm is fast, well known, has been studied in depth, and is implementable on many kinds of architectures like FPGAs, smart phones, desktop machines, and GPUs, we have chosen this edge detector for illustrative purpose.

This edge detection is applied on a filtered version of the image given as input. More precisely, only b most significant bits are concerned by this step, where the parameter b is practically set with 6 or 7. Notice that only the 2 LSBs of pixels in the set of edges are returned if b is 6, and the LSB of pixels if b is 7. If set with the same value b, the edge detection returns thus the same set of pixels for both the cover and the stego image. Moreover, to provide edge gradient value, the Canny algorithm computes derivatives in the two directions with respect to a mask of size T. The higher T is, the coarse the approach is. Practically, T is set with 3, 5, or 7. In our flowcharts, this step is represented by "Edge Detection(b, T, X)".

Let x be the sequence of these bits. The next section presents how to adapt our scheme with respect to the size of the message m to embed and the size of the cover x.

2.3 Adaptive embedding rate

Two strategies have been developed in our approach, depending on the embedding rate that is either *Adaptive* or *Fixed*. In the former the embedding rate depends on the number of edge pixels. The higher it is, the larger the message length that can be inserted is. Practically, a set of edge pixels is computed according to the Canny algorithm with parameters $b = 7$ and $T = 3$. The message length is thus defined to be lesser than half of this set cardinality. If x is too short for m, the message is split into sufficient parts and a new cover image should be used for the remaining part of the message.

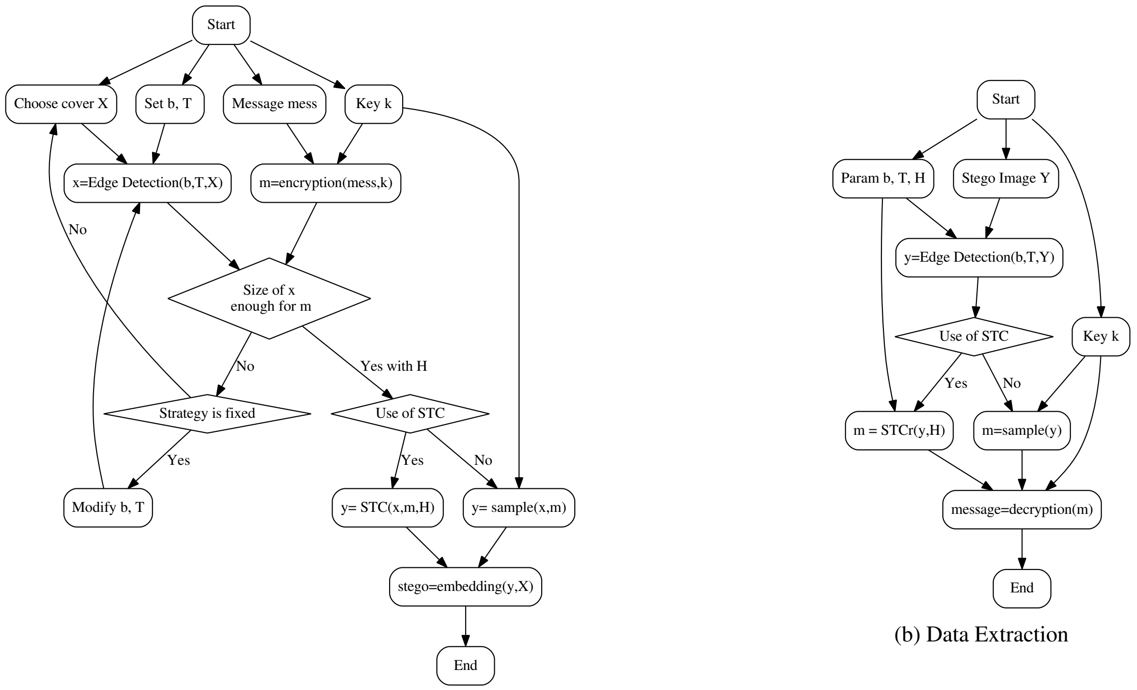
In the latter, the embedding rate is defined as a percentage between the number of modified pixels and the length of the bit message. This is the classical approach adopted in steganography. Practically, the Canny algorithm generates a set of edge pixels related to increasing values of T and until its cardinality is sufficient. Even in this situation, our scheme adapts its algorithm to meet all the user's requirements.

Once the map of possibly modified pixels is computed, two methods may further be applied to extract bits that are really changed. The first one randomly chooses the subset of pixels to modify by applying the BBS PRNG again. This method is further denoted as a *sample*. Once this set is selected, a classical LSB replacement is applied to embed the stego content. The second method considers the last significant bits of all the pixels inside the previous map. It next directly applies the STC algorithm [7]. It is further referred to as *STC* and is detailed in the next section.

2.4 Minimizing distortion with Syndrome-Trellis Codes

To make this article self-contained, this section recalls the basis of the Syndrome Treillis Codes (STC). A reader who is familar with syndrome coding can skip it.

Let $x = (x_1, \ldots, x_n)$ be the n-bits cover vector issued from an image X, m be the message to embed, and $y = (y_1, \ldots, y_n)$



(a) Data Embedding

(b) Data Extraction

Fig. 1: The STABYLO scheme

be the n-bits stego vector. The usual additive embedding impact of replacing x by y in X is given by a distortion function $D_X(x,y) = \Sigma_{i=1}^{n} \rho_X(i,x,y)$, where the function ρ_X expresses the cost of replacing x_i by y_i in X. The objective is thus to find y that minimizes $D_X(x,y)$.

Hamming embedding proposes a solution to this problem. This is why some steganographic schemes [12, 16, 28] are based on this binary embedding. Furthermore this code provides a vector y s.t. Hy is equal to m for a given binary matrix H.

Let us explain this embedding on a small illustrative example where m and x are respectively a 3 bits column vector and a 7 bits column vector, and where $\rho_X(i,x,y)$ is equal to 1 for any i, x, y (*i.e.*, $\rho_X(i,x,y) = 0$ if $x = y$ and 1 otherwise).

Let $\dot{H}$ be the binary Hamming matrix

$$\dot{H} = \begin{pmatrix} 0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1 \\ 1\,0\,1\,0\,1\,0\,1 \end{pmatrix}.$$

The objective is to modify x to get y s.t. $m = \dot{H}y$. In this algebra, the sum and the product respectively correspond to the exclusive *or* and to the *and* Boolean operators. If $\dot{H}x$ is already equal to m, nothing has to be changed and x can be sent. Otherwise we consider the difference $\delta = d(m, \dot{H}x)$, which is expressed as a vector :

$$\delta = \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} \text{ where } \delta_i \text{ is 0 if } m_i = Hx_i \text{ and 1 otherwise.}$$

Let us thus consider the $j-$th column of H, which is equal to δ. We denote by $\overline{x}^j$ the vector obtained by switching the $j-$th component of x, that is, $\overline{x}^j = (x_1, \ldots, \overline{x_j}, \ldots, x_n)$. It is not hard to see that if y is $\overline{x}^j$, then $m = \dot{H}y$. It is then possible to embed 3 bits in 7 LSBs of pixels by modifying at most 1 bit. In the general case, communicating a message of p bits in a cover of $n = 2^p - 1$ pixels needs $1 - 1/2^p$ average changes.

This Hamming embedding is really efficient to very small payload and is not well suited when the size of the message is larger, as in real situations. The matrix H should be changed to deal with higher payload. Moreover, for any given H, finding y that solves $Hy = m$ and that minimizes $D_X(x,y)$, has an exponential complexity with respect to n. The Syndrome-Trellis Codes presented by Filler *et al.* in [8, 22] is a practical solution to this complexity. Thanks to this contribution, the solving algorithm has a linear complexity with respect to n.

First of all, Filler *et al.* compute the matrix H by placing a small sub-matrix $\hat{H}$ next to each other and by shifting down by one row. Thanks to this special form of H, one can represent any solution of $m = Hy$ as a path through a trellis.

Next, the process of finding y consists in two stages: a forward and a backward part.

1. Forward construction of the trellis that depends on $\hat{H}$, on x, on m, and on ρ. This step is linear in n.
2. Backward determination of y that minimizes D, starting with the complete path having the minimal weight. This corresponds to traversing a graph and has a complexity which is linear in n.

For a given set of parameters, the Canny algorithm returns a numerical value and states whether a given pixel is an edge or not. In this article, in the Adaptive strategy we consider that all the edge pixels that have been selected by this algorithm have the same distortion cost, *i.e.*, ρ_X is always 1 for these bits. In the Fixed strategy, since pixels that are detected to be edge with small values of T (e.g., when $T = 3$) are more accurate than these with higher values of T, we give to STC the following distortion map of the corresponding bits

$$\rho_X = \begin{cases} 1 \text{ if an edge for } T = 3, \\ 10 \text{ if an edge for } T = 5, \\ 100 \text{ if an edge for } T = 7. \end{cases}$$

2.5 Data extraction

The message extraction summarized in Fig. 1b follows the data embedding approach since there exists a reverse function for all its steps.

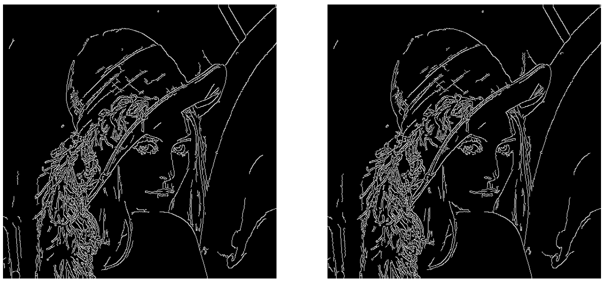
More precisely, let b be the most significant bits and T be the size of the Canny mask, both be given as a key. Thus, the same edge detection is applied on a stego content Y to produce the sequence y of LSBs. If the STC approach has been selected in embedding, the STC reverse algorithm is directly executed to retrieve the encrypted message. This inverse function takes the $\hat{H}$ matrix as a parameter. Otherwise, *i.e.*, if the *sample* strategy is retained, the same random bit selection than in the embedding step is executed with the same seed, given as a key. Finally, the Blum-Goldwasser decryption function is executed and the original message is extracted.

2.6 Running example

In this example, the cover image is Lena, which is a 512 × 512 image with 256 grayscale levels. The message is the poem Ulalume (E. A. Poe), which is constituted by 104 lines, 667 words, and 3,754 characters, *i.e.*, 30,032 bits. Lena and the first verses are given in Fig. 2.

The skies they were ashen and sober;
The leaves they were crisped and sere—
The leaves they were withering and sere;
It was night in the lonesome October
Of my most immemorial year;
It was hard by the dim lake of Auber,
In the misty mid region of Weir—
It was down by the dank tarn of Auber,
In the ghoul-haunted woodland of Weir.

Fig. 2: Cover and message examples



(a) b is 7.

(b) b is 6.

Fig. 3: Edge detection wrt b with $T = 3$

The edge detection returns 18,641 and 18,455 pixels when b is respectively 7 and 6 and $T = 3$. These edges are represented in Figure 3. When b is 7, it remains one bit per pixel to build the cover vector. This configuration leads to a cover vector of size 18,641 if b is 7 and 36,910 if b is 6.

The STC algorithm is optimized when the rate between message length and cover vector length is lower than 1/2. So, only 9,320 bits are available for embedding in the configuration where b is 7.

When b is 6, we could have considered 18,455 bits for the message. However, first experiments have shown that modifying this number of bits is too easily detectable. So, we choose to modify the same amount of bits (9,320) and keep STC optimizing which bits to change among the 36,910 ones.

In the two cases, about the third part of the poem is hidden into the cover. Results with Adaptive and STC strategies are presented in Fig. 4.

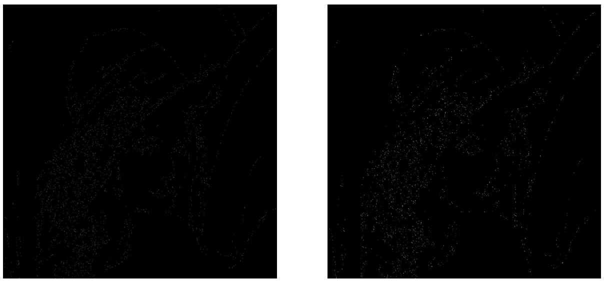
Finally, differences between the original cover and the stego images are presented in Fig. 5. For each pair of pixel X_{ij} and Y_{ij} (X and Y being the cover and the stego content respectively), the pixel value V_{ij} of the difference is defined with the following map

$$V_{ij} = \begin{cases} 0 \text{ if } X_{ij} = Y_{ij} \\ 75 \text{ if } |X_{ij} - Y_{ij}| = 1 \\ 150 \text{ if } |X_{ij} - Y_{ij}| = 2 \\ 225 \text{ if } |X_{ij} - Y_{ij}| = 3 \end{cases}$$

This function allows to emphasize differences between contents. Notice that since b is 7 in Fig. 5a, the embedding is

(a) b is 7. (b) b is 6.

Fig. 4: Stego images wrt b



(a) b is 7. (b) b is 6.

Fig. 5: Differences with Lena's cover wrt b

binary and this image only contains 0 and 75 values. Similarly, if b is 6 as in Fig. 5b, the embedding is ternary and the image contains all the values in $\{0,75,150,225\}$.

3 Complexity Analysis

This section aims at justifying the lightweight attribute of our approach. To be more precise, we compare the complexity of our schemes to some of current state of the art of steganographic schemes, namely HUGO [26], WOW [14], and UNIWARD [13]. Each of these schemes starts with the computation of the distortion cost for each pixel switch and is later followed by the STC algorithm. Since this last step is shared by all, we separately evaluate this complexity. In all the remainder of this section, we consider a $n \times n$ square image.

First of all, HUGO starts with computing the second order SPAM Features. This steps is in $\theta(n^2 + 2 \times 343^2)$ due to the computation of the difference arrays and next of the 686 features (of size 343). Next for each pixel, the distortion measure is calculated by +1/-1 modifying its value and computing again the SPAM features. Pixels are thus selected according to their ability to provide an image whose SPAM features are close to the original ones. The algorithm thus computes a distance between each feature and the original ones, which is at least in $\theta(343)$, and an overall distance between these metrics, which is in $\theta(686)$. Computing the distance is thus in $\theta(2 \times 343^2)$ and this modification is thus in $\theta(2 \times 343^2 \times n^2)$. Ranking these results may be achieved with a quick sort, which is in $\theta(2 \times n^2 \ln(n))$ for data of size n^2. The overall complexity of the pixel selection is finally $\theta(n^2 + 2 \times 343^2 + 2 \times 343^2 \times n^2 + 2 \times n^2 \ln(n))$, *i.e*, $\theta(2 \times n^2(343^2 + \ln(n)))$.

Let us focus now on WOW. This scheme starts to compute the residual of the cover as a convolution product which is in $\theta(n^2 \ln(n^2))$. The embedding suitability η_{ij} is then computed for each pixel $1 \leq i,j \leq n$ thanks to a convolution product again. We thus have a complexity of $\theta(n^2 \times n^2 \ln(n^2))$. Moreover the suitability is computed for each wavelet level detail (HH, HL, LL). This distortion computation step is thus in $\theta(6n^4 \ln(n))$. Finally a norm of these three values is computed for each pixel which adds to this complexity the complexity of $\theta(n^2)$. To summarize, the complixity is in $\theta(6n^4 \ln(n) + n^2)$

What follows details the complexity of the distortion evaluation of the UNIWARD scheme. This one is based to a convolution product W of two elements of size n and is again in $\theta(n^2 \times n^2 \ln(n^2))$, and a sum D of these W which is in $\theta(n^2)$. This distortion computation step is thus in $\theta(6n^4 \ln(n) + n^2)$.

Our edge selection is based on a Canny filter. When applied on a $n \times n$ square image, the noise reduction step is in $\theta(5^3 n^2)$. Next, let T be the size of the Canny mask. Computing gradients is in $\theta(4Tn^2)$ since derivatives of each direction (vertical or horizontal) are in $\theta(2Tn^2)$. Finally, thresholding with hysteresis is in $\theta(n^2)$. The overall complexity is thus in $\theta((5^3 + 4T + 1)n^2)$.

We are then left to express the complexity of the STC algorithm. According to [7], it is in $\theta(2^h.n)$ where h is the size of the duplicated matrix. Its complexity is thus negligible compared with the embedding map construction.

The Fig. 6 summarizes the complexity of the embedding map construction, for WOW/UNIWARD, HUGO, and STABYLO. It deals with square images of size $n \times n$ when n ranges from 512 to 4096. The y-coordinate is expressed in a logarithm scale. It shows that the complexity of all the algorithms is dramatically larger than the one of the STABYLO scheme. Thanks to these complexity results, we claim that our approach is lightweight.

4 Experiments

First of all, the whole code of STABYLO can be downloaded [1]. For all the experiments, the whole set of 10,000 images of the BOSS contest [25] database is taken. In this

[1] http://http://members.femto-st.fr/jf-couchot/en/stabylo

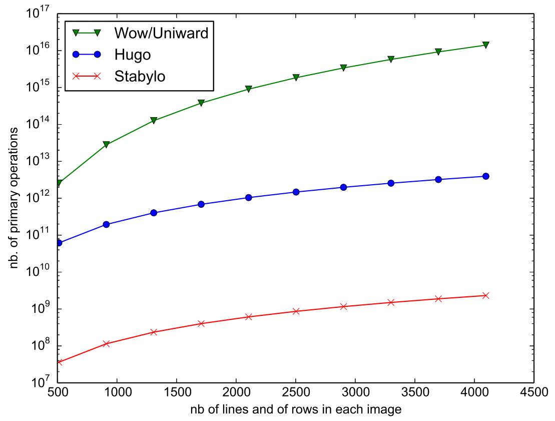


Fig. 6: Complexity evaluation of WOW/UNIWARD, HUGO, and STABYLO.

Rate	Matrix generators
1/2	$\{71,109\}$
1/3	$\{95,101,121\}$
1/4	$\{81,95,107,121\}$
1/5	$\{75,95,97,105,117\}$
1/6	$\{73,83,95,103,109,123\}$
1/7	$\{69,77,93,107,111,115,121\}$
1/8	$\{69,79,81,89,93,99,107,119\}$
1/9	$\{69,79,81,89,93,99,107,119,125\}$

Table 1: Matrix Generator for $\hat{H}$ in STC.

set, each cover is a 512×512 grayscale digital image in a RAW format. We restrict experiments to this set of cover images since this paper is more focused on the methodology than on benchmarks.

We use the matrices $\hat{H}$ generated by the integers given in Table 1 as introduced in [8], since these ones have experimentally be proven to have the strongest modification efficiency. For instance if the rate between the size of the message and the size of the cover vector is 1/4, each number in $\{81,95,107,121\}$ is translated into a binary number and each one constitutes thus a column of $\hat{H}$.

Our approach is always compared to HUGO, to EAISLSBMR, to WOW and to UNIWARD for the two strategies Fixed and Adaptive. For the former one, the payload has been set to 10%. For the latter one, the Canny parameter T has been set to 3. When b is 7, the average size of the message that can be embedded is 16,445 bits, that corresponds to an average payload of 6.35%. For each cover image the STABYLO's embedding rate with these two parameters is memorized. Next each steganographic scheme is executed to produce the stego content of this cover with respect to this embedding rate.

4.1 Steganalysis

The steganalysis quality of our approach has been evaluated through the Ensemble Classifier [18] based steganalyser. Its particularization to spatial domain is considered as state of the art steganalysers. Features that are embedded into this steganalysis process are CCPEV and SPAM features as described in [19]. They are extracted from the set of cover images and the set of training images. Next a small set of weak classifiers is randomly built, each one working on a subspace of all the features. The final classifier is constructed by a majority voting between the decisions of these individual classifiers.

Results of average testing errors are summarized in Table 2. First of all, STC outperforms the sample strategy as already noticed in the quality analysis presented in the previous section. Next, our approach is more easily detectable than HUGO, WOW and UNIWARD which are the most secure steganographic tool, as far as we know. However by combining Adaptive and STC strategies our approach obtains similar results than the ones of these schemes.

Compared to EAILSBMR, we obtain similar results when the strategy is Adaptive. However due to its huge number of integration features, it is not lightweight.

All these numerical experiments confirm the objective presented in the motivations: providing an efficient steganography approach in a lightweight manner for small payload.

In Figure 7, Ensemble Classifier has been used with all the previous steganographic schemes with 4 different payloads. It can be observed that face to high values of payload, STABYLO is definitely not secure enough. However thanks to an efficient very low-complexity (Fig.6), we argue that the user should embed tiny messages in many images than a larger message in only one image.

5 Conclusion

The STABYLO algorithm, whose acronym means STeganography with Adaptive, Bbs, and binarY embedding at LOw cost, has been introduced in this document as an efficient method having comparable, though somewhat smaller, security than well-known steganographic schemes HUGO, WOW, and UNIWARD. This edge-based steganographic approach embeds a Canny detection filter, the secure Blum-Blum-Shub cryptosystem with its pseudorandom number generator, together with Syndrome-Trellis Codes for minimizing distortion. The complexity study of our proposed method and of the state of the art steganographic tools has shown that our approach has the lowest computation cost among all. This

Schemes	STABYLO				HUGO		EAISLSBMR		WOW		UNIWARD	
Embedding	Fixed	Adaptive (about 6.35%)			Fixed	Adapt.	Fixed	Adapt.	Fixed	Adapt.	Fixed	Adapt.
Rate	10%	+ sample	+STC(7)	+STC(6)	10%	≈6.35%	10%	≈6.35%	10%	≈6.35%	10%	≈6.35%
Ensemble Classifier	0.35	0.44	0.47	0.47	0.48	0.49	0.43	0.47	0.48	0.49	0.46	0.49

Table 2: Steganalysing STABYLO.

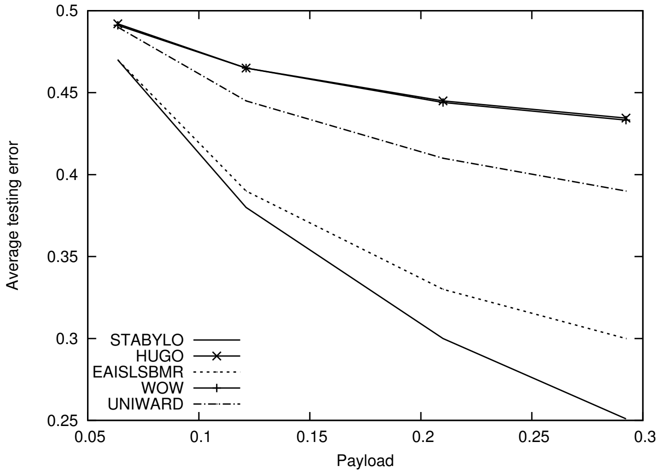


Fig. 7: Testing errors obtained by Ensemble classifier with WOW/UNIWARD, HUGO, and STABYLO w.r.t. payload.

justifies the lightweight attribute of our scheme. The evaluation of introduced noise and of its embedding through stegenalysers (namely Ensemble Classifier) have shown that STABYLO is efficient enough to produce qualitative images and to face steganalysers.

For future work, the authors' intention is to investigate systematically all the existing edge detection methods, to see if the STABYLO evaluation scores can be improved by replacing Canny with another edge filter. Moreover, we plan to improve the distortion function by integrating into a numerical cost the gradient value of this kind of algorithm. We could thus transmit this value to STC contrary to the current version where the distortion that is transmited is either 1 in the adaptive strategy or 1,10, 100 in the fixed strategy.

Other steganalysers than the ones used in this document will be examined for the sake of completeness. Finally, the systematic replacement of all the LSBs of edges by binary digits provided by the BBS generator will be investigated, and the consequences of such a replacement, in terms of security, will be discussed. Furthermore, we plan to investigate information hiding on other models, such as high frequency for JPEG encoding.

References

1. Blum, L., Blum, M., Shub, M.: Comparison of two pseudo-random number generators. In: D. Chaum, R.L. Rivest, A.T. Sherman (eds.) Advances in Cryptology: Proceedings of CRYPTO '82, Santa Barbara, California, USA, August 23-25, 1982, pp. 61–78. Plenum Press, New York, NY, USA (1983)
2. Blum, M., Goldwasser, S.: An efficient probabilistic public-key encryption scheme which hides all partial information. In: G.R. Blakley, D. Chaum (eds.) Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings, *Lecture Notes in Computer Science*, vol. 196, pp. 289–302. Springer, Berlin (1985)
3. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-8**(6), 679–698 (1986). DOI 10.1109/TPAMI.1986.4767851. URL `http://dx.doi.org/10.1109/TPAMI.1986.4767851`
4. Chen, W.J., Chang, C.C., Le, T.H.N.: High payload steganography mechanism using hybrid edge detector. Expert Systems with Applications **37**(4), 3292–3301 (2010). URL `http://dblp.uni-trier.de/db/journals/eswa/eswa37.html#ChenCL10`
5. Dumitrescu, S., Wu, X.: Lsb steganalysis based on high-order statistics. In: A.M. Eskicioglu, J.J. Fridrich, J. Dittmann (eds.) Proceedings of the 7th workshop on Multimedia & Security, MM&Sec 2005, New York, NY, USA, August 1-2, 2005, 2006, pp. 25–32. ACM, New York, NY, USA (2005)
6. Dumitrescu, S., Wu, X., Wang, Z.: Detection of lsb steganography via sample pair analysis. IEEE Transactions on Signal Processing **51**(7), 1995–2007 (2003). URL `http://dblp.uni-trier.de/db/journals/tsp/tsp51.html#DumitrescuWW03`
7. Filler, T., Judas, J., Fridrich, J.J.: Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Transactions on Information Forensics and Security **6**(3-2), 920–935 (2011). URL `http://dblp.uni-trier.de/db/journals/tifs/tifs6.html#FillerJF11`
8. Filler, T., Judas, J., Fridrich, J.J.: Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Transactions on Information Forensics and Security **6**(3-2), 920–935 (2011)
9. Fontaine, C., Galand, F.: A survey of homomorphic encryption for nonspecialists. EURASIP Journal on Information Security **2007**(1), 013,801 (2007)
10. Fridrich, J.J., Goljan, M., Du, R.: Reliable detection of lsb steganography in color and grayscale images. In: N.D. Georganas, R. Popescu-Zeletin (eds.) Proceedings of the 4th workshop on Multimedia & Security: New Challenges, MM&Sec 2001, Ottawa, Ontario, Canada, October 5, 2001, pp. 27–30. ACM, New York, NY, USA (2001)
11. Fridrich, J.J., Kodovský, J.: Steganalysis of lsb replacement using parity-aware features. In: M. Kirchner, D. Ghosal (eds.) Information Hiding - 14th International Conference, IH 2012, Berkeley, CA, USA, May 15-18, 2012, Revised Selected Papers, *Lecture Notes in Computer Science*, vol. 7692, pp. 31–45. Springer, Berlin (2012)
12. Fridrich, J.J., Pevný, T., Kodovský, J.: Statistically undetectable jpeg steganography: dead ends challenges, and opportunities. In: D. Kundur, B. Prabhakaran, J. Dittmann, J.J. Fridrich (eds.)

Proceedings of the 9th workshop on Multimedia & Security, MM&Sec 2007, Dallas, Texas, USA, September 20-21, 2007, pp. 3–14. ACM, New York, NY, USA (2007)

13. Holub, V., Fridrich, J., Denemark, T.: Universal distortion function for steganography in an arbitrary domain. EURASIP Journal on Information Security **2014**(1), 1 (2014). DOI 10.1186/1687-417X-2014-1. URL `http://dx.doi.org/10.1186/1687-417X-2014-1`
14. Holub, V., Fridrich, J.J.: Designing steganographic distortion using directional filters. In: 2012 IEEE International Workshop on Information Forensics and Security, Tenerife, Spain, pp. 234–239. IEEE (2012). URL `http://dblp.uni-trier.de/db/conf/wifs/wifs2012.html#HolubF12`
15. Ker, A.D.: A general framework for structural steganalysis of lsb replacement. In: M. Barni, J. Herrera-Joancomartí, S. Katzenbeisser, F. Pérez-González (eds.) Information Hiding, 7th International Workshop, IH 2005, Barcelona, Spain, June 6-8, 2005, Revised Selected Papers, *Lecture Notes in Computer Science*, vol. 3727, pp. 296–311. Springer, Berlin (2005)
16. Kim, Y., Duric, Z., Richards, D.: Modified matrix encoding technique for minimal distortion steganography. In: J. Camenisch, C.S. Collberg, N.F. Johnson, P. Sallee (eds.) Information Hiding, *Lecture Notes in Computer Science*, vol. 4437, pp. 314–327. Springer, Berlin (2006). URL `http://dblp.uni-trier.de/db/conf/ih/ih206.html#KimDR06`
17. Kodovský, J., Fridrich, J.: Steganalysis in high dimensions: Fusing classifiers built on random subspaces. In: Proc. SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics XIII, pp. 78,800L–78,800L–13. Society of Photo-Optical Instrumentation Engineers, Bellingham, WA (2011)
18. Kodovský, J., Fridrich, J.J., Holub, V.: Ensemble classifiers for steganalysis of digital media. IEEE Transactions on Information Forensics and Security **7**(2), 432–444 (2012)
19. Kodovský, J., Pevný, T., Fridrich, J.J.: Modern steganalysis can detect yass. In: Media Forensics and Security, p. 754102 (2010)
20. Li, B., Huang, J., Shi, Y.Q.: Textural features based universal steganalysis. In: Proc. SPIE 6819, p. 12. Society of Photo-Optical Instrumentation Engineers, Bellingham, WA (2008)
21. Li, F., Zhang, X., Yu, J., Shen, W.: Adaptive jpeg steganography with new distortion function. annals of telecommunications - annales des télécommunications **69**(7-8), 431–440 (2014). DOI 10.1007/s12243-013-0415-2. URL `http://dx.doi.org/10.1007/s12243-013-0415-2`
22. Liu, W., Liu, G., Dai, Y.: Syndrome trellis codes based on minimal span generator matrix. annals of telecommunications-annales des télécommunications **69**(7-8), 403–416 (2014)
23. Luo, W., Huang, F., Huang, J.: Edge adaptive image steganography based on lsb matching revisited. IEEE Transactions on Information Forensics and Security **5**(2), 201–214 (2010). DOI 10.1109/TIFS.2010.2041812. URL `http://dx.doi.org/10.1109/TIFS.2010.2041812`
24. Mielikainen, J.: Lsb matching revisited. IEEE Signal Processing Letters **13**(5), 285–287 (2006)
25. Pevný, T., Filler, T., Bas, P.: Break our steganographic system (2010). Available at `http://www.agents.cz/boss/`
26. Pevný, T., Filler, T., Bas, P.: Using high-dimensional image models to perform highly undetectable steganography. In: R. Böhme, P.W.L. Fong, R. Safavi-Naini (eds.) Information Hiding - 12th International Conference, IH 2010, Calgary, AB, Canada, June 28-30, 2010, Revised Selected Papers, *Lecture Notes in Computer Science*, vol. 6387, pp. 161–177. Springer, Berlin (2010)
27. Tyan, C.Y., Wang, P.P.: Image processing-enhancement, filtering and edge detection using the fuzzy logic approach. In: Proceedings of the Second IEEE International Conference on Fuzzy Systems, vol. 1, pp. 600–605. IEEE Computer Society, Washington, DC (1993)
28. Westfeld, A.: F5-a steganographic algorithm. In: I.S. Moskowitz (ed.) Information Hiding, 4th International Workshop, IHW 2001, Pittsburgh, PA, USA, April 25-27, 2001, Proceedings, *Lecture Notes in Computer Science*, vol. 2137, pp. 289–302. Springer, Berlin (2001)