account their homogeneity. This allows to treat bits of similar pixels equivalently, but not the bits of pixels of di_erent homogeneity level. We focus in this paper in binary codes although any q-ary code can be used with the same techniques.

We use the following notations:

m: the (secret) message,

x: the cover,

y: the modi_ed cover,

E(): the embedding map,

R(): the retrieval map.

This paper is organized as follows. Section 2 describes how classical linear error correcting codes are applied in steganography. In Section 3 we introduce Linear error-block codes and recall the main tools to be used. Section 4 is our main contribution, it presents our steganographic scheme and motivates the use of Linear Error-Block Codes. The experimental results are given in Section 5.

Section 6 involves conclusion and perspective of this work.

2 The F5 algorithm

In 1998, Crandall was the _rst one to bring the idea of using error correcting codes in steganography [1]. Three years later,Westfeld designed a steganographic algorithm, called F5, that uses this idea [2]. Our proposal is a generalization of this algorithm to linear error block codes. Here is an overview of the F5 algo-rithm.

Let C be a linear error correcting code of parity check matrix H. The retrieval map requires simply computing the syndrome of the modi_ed cover $R(y) := S(y) = Hy^T$.

The embedding algorithm consists of three steps. First compute $u := S(x)-m$, then find e_u a word of smallest weight among all words of syndrome u. This is the syndrome decoding problem. It is equivalent to decode u in C and get the error vector eu. Finally compute $E(m; x) := x - e_u$.

For veri_cation we have

$R(E(m, x)) = S(x - e_u) = S(x) - S(e_u) = S(x) - u = m.$ (1)

Linear Error-Block Codes in Steganography 3

3 Linear error-block codes

Linear error-block codes (LEBC) are a generalization of linear error correcting codes. They were introduced in [3], and studied in several works [4{7]. The au-thors of [3] de_ned a special Hamming bound for LEBC with minimum distance even. The codes attaining this bound are thus considered perfect. There exist larger families of perfect linear error-block codes than the classical linear error correcting codes, but all of the known families are of minimum distance either 3 or 4 (in addition to the well known classical perfect codes which are also per-fect error-block codes) [5]. The existence of more perfect LEBC is still an open problem. In this section, we _rst recall preliminary notions to be used. Then we describe linear error-block codes. We discuss the metric used to deal with these codes.

A composition π of a positive integer n is given by $n = l_1m_1 + l_2m_2 + \ldots\ldots + l_rm_r$, where r, l_1, l_2, ….. , l_r,m_1,m_2, ….. ,m_r are integers ≥ 1, and is denoted

$\pi = [m1]^{l1} [m2]^{l2} \ldots\ldots. [mr]^{lr}$ (2)

If moreover $m1 > m2 > \ldots.. > mr \geq 1$ then π is called a partition.

Let q be a prime power and Fq be the _nite _eld with q elements. Let s, r, $l_1, l_2,\ldots., l_r, n_1, n_2,\ldots.., n_s$ be the non negative integers given by a partition π as

$s = l_1 + \ldots\ldots. + l_r;$

$n_1 = n_2 = \ldots\ldots. = n_{l1} = m_1$

$n_{l1}+_1 = n_{l1}+_2 = \ldots\ldots\ldots = n_{l1}+l_2 = m_2$
.
.
.
$n_{l1}+\ldots\ldots\ldots+l_{r-1}+1 = nl1+\ldots\ldots+l_{r-1}+2 = \ldots\ldots = n_s = m_r$
We can write
$\pi = [n_1][n_2] \ldots\ldots\ldots [n_s]$: (3)
Let $Vi = F_q^{ni}$ $(1 \leq i \leq s)$ and $v = v_1 \oplus v_2 \oplus \ldots\ldots \oplus v_s = F_q^n$. Each vector in V can be written uniquely as $v = (v_1, \ldots\ldots, v_s)$, $vi \in V_i$ $(1 \leq i \leq s)$. For any $u = (u_1, \ldots\ldots, u_s)$ and $v = (v_1, \ldots\ldots, v_s)$ in V , the π-weight $w_\pi(u)$ of u and the π -distance $d_\pi(u, v)$ of u and v are de_ned by
$w_\pi(u) = \#\{i/1 \leq i \leq s, u_i \neq 0 \in Vi\}$ and (4)
$d_\pi(u, v) = w_\pi(u - v) = \#\{i/1 \leq i \leq s, u_i \neq vi\}$. (5)
This means that a _xed vector can be of different π -weights if we change π. For example, consider the word $v = 1010001101$ of length 10 and the two partitions of the number 10: $\pi = [3][2]^3[1]$ and $\pi' = [3]^2[2][1]^2$. We have $w_\pi(v) = 4$ while

$w_\pi'(v) = 3$.
An Fq-linear subspace C of V is called an [n, k, d]q linear error-block code over Fq of type π, where $k = \dim_{Fq}(C)$ and $d = d_\pi(C)$ is the minimum π – distance of C, which is defined as
$d = \min\{d_\pi(c, c')/c, c' \in C; c \neq c'\} = \min\{w_\pi(c)/0 \neq c \in C\}$:
(6)
Remark 1. A classical linear error correcting code is a linear error-block code of type $\pi = [1]^n$.
Remark 2. A linear error-block code with a composition type is equivalent to some linear error-block code with a partition type.
The difference between decoding linear error block codes and decoding classical linear error correcting codes is the use of the π -distance instead of the Ham-ming distance. Therefore, coset leaders are words having minimum π -weight,although sometimes they are not of minimum Hamming weight.
It is well known that perfect codes have odd minimum distance. Nonetheless, the Hamming bound presented in [3] allows to construct perfect codes with even minimum distance. This is done by considering the sets
$B'_\pi(c,d/2) = b_\pi(c,d/2 - 1) \sqcup \{x \in v; d_\pi = d/2$ and $x_1 \neq c_1\}$
(7)
where $B_\pi(c, r) = \{x \in V ; d_\pi(x, c) \leq r\}$ is the ball of center a codeword c and radius r. The sets $B_\pi'(d/2)$ are pairwise disjoint. (And if the code is perfect, their union for all codewords c covers the space V). A word $x \in V$ is thus decoded as c if $x \in B_\pi'(c,d/2)$ Therefore, the decoding algorithm of a code with even minimum distance d corrects, further, error patterns with both π-weight d/2 and non null first block.
To construct a syndrome table, we add the following condition. If a coset of weight d/2 has more than one leader, we select among them a word which has a non null first block. Note that the coset leader has not to be unique unless the code is perfect. The maximum likelihood decoding (MLD) [8] is slightly modified.
Remark 3. This decoding technique can also be used with classical codes. Therefore, quasi-perfect codes [8] are considered perfect and can perform complete decoding. This makes them more effcient in steganography.
Example 1. Let $\pi = [3][2]^2[1]$, if we have to choose a coset leader between the words $e_1 = 000|01|00|1$ and $e_2 = 010|01|00|0$, we select the second one, since

they are both of π-weight 2 and the first block of e_2 is not null.
Note that this does not guaranty that every error pattern of π -weight d/2 can be corrected. We present below an example of syndrome decoding of a given linear error-block code.
Linear Error-Block Codes in Steganography 5
Example 2. Consider C the binary [7, 3] code of type $\pi = [3][2][1]^2$, defined with its parity check matrix
H =
1 1 1 0 0 0 1
0 1 1 1 1 0 0
1 0 0 1 0 1 0
1 0 1 0 0 1 0
Syndromes are computed by the classical formula $s = Hx^T$, $x \in F_2^7$. The words which have the same syndrome belong to the same coset. Coset leaders are words which have the minimum π -weight in their cosets. The syndrome table of the code C is presented in Table 1. We added two columns to mark, in the _rst one, the di_erence between the use of the π -distance and the Hamming distance, and in the second one, cases where π -metric decoding is not unique (since $d_\pi = 2$, correction capacity is $t_\pi = 1$). Note that in line 5, since the first block of the right side column word is null, it can not be chosen as a coset leader. In lines 6 and 8, the third column words have π -weight 2, they also can not be coset leaders. While in the last four lines, all of the right side column words can be a coset leader.

Table 1. Syndrome table of the error-block code C.

4 Linear error-block codes in steganography

By using the π -distance, when we correct a single error-block, all the bits in the block are being corrected at once [6], which means that we recover within one error correction not less than one message bit. The maximum recovered bits is

the size of the corrected block. In other words, assume that to embed a k-bit message we have to modify r bits in the n-bit cover. These r bits are located in r' blocks where $r' \leq r$. Thus we have to provoke r' errors, whilst with a classical code we provoke r errors.
However, it may happen, for some partitions, that alteration of a whole block is needed, while if we use a classical code we just need to ip a single bit to embed the same message. For instance, in Example 3 (Page 8), the coset leader for Hamming distance is e_u = 0001000 (Line 5 of Table 1), which looks more interesting to use. Nevertheless, the blocks are token from di_erent areas of the cover in order to give preferential treatment to each block of bits. The size and

the order of the blocks are considered as well. It is clear that the smaller is the block, the less is its probability of distortion. We show in the next subsection that it might be more suitable to ip a whole block rather than a single bit.
Although syndrome decoding using the π -distance returns coset leaders with more non null bits than Hamming distance, application of linear error-block codes in steganography is motivated by the following vision. In general, pictures
feature areas that can better hide distortion than other areas. The human vision
system is unable to detect changes in inhomogeneous areas of a digital media, due to the complexity of such areas. For example, if we modify the gray values
of pixels in smooth areas of a gray-scale image, they will be more easily noticed
by human eyes. In the other hand, the pixels in edged areas may tolerate larger
changes of pixel values without causing noticeable changes (Fig. 1). So, we can keep the changes in the modi_ed image unnoticeable by embedding more data in edged areas than in smooth areas.

Fig. 1. Embedding 16-valued pixels in two areas of Lena gray-scale image. Another vision consists of using one or two bits next to the least signi_cant ones. Actually, the least signi_cant bits (LSB) are the most suitable, but we can use further bit levels if this does not cause noticeable change to the image (Fig. 2). There are many embedding methods that uses bit signi_cation levels
in di_erent ways [10-13]. For our method, there are two possibilities. First one, we select a list $(p_i)_{i\in I}$ of pixels to be modi_ed, and embed the message within (some) bit levels of this list. The pixels must then be located in a su_ciently inhomogeneous area. The second way is by selecting independently, for each bit level j, a list $(p_i)_{i\in I_j}$ of pixels to be manipulated. Hence more bits are used, Linear Error-Block Codes in Steganography resulting in a larger cover size. If the pixels are strong enough against distortion,
higher bit levels can be used but with smaller and smaller block sizes.

Fig. 2. Lena image with different bit levels switched to 0 or 1.

We sketch our method in Figure 3. The cover bits are assorted by their ability to hide distortion. Thus, in the previous examples, _rst level bits refer to the most inhomogeneous areas or to the least signi_cant bits. We can also combine these methods by assorting bits starting from the LSB of the most inhomoge-neous area to the last signi_cant bit considered of the most homogeneous area.

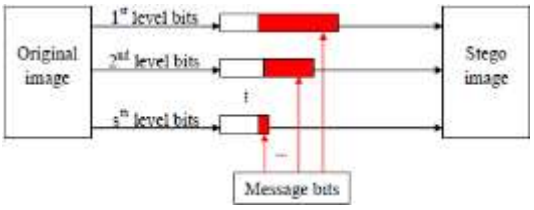

Fig. 3. Overview of our embedding method.

Now assume we have made s sequences of cover bits assorted by their abil-ity to support distortion in the decreasing sense (i.e. the _rst sequence bits are the less sensitive to distortion). We note the i^{th} sequence $(a^i_j)_{j=1,2,\ldots,m_i}$. In the next step, we reorder the bits such that the cover will be written as blocks $x = (x_1, x_2, \cdots, x_s)$ where $x_i \in V_i$ is a subsequence of n_i bits of α_i (we use the notation of Section 3). Considering these blocks, a partition $\pi = [n_1][n_2] \ldots [n_s]$

of the size of the cover is de_ned. And the cover is viewed as a list of vectors $x \in V$. Therefore, the distortion of the first level bits is more probable than the distortion of the second level bits, and so on. The sizes of the blocks are chosen such that the complete distortion of any block (the distortion of all of its bits) has an equal e_ect on the image change. In the next section we show how is a linear error-block code of type π used.

Remark 4. If the cover bits have equal inuence on image quality, for example only LSBs are used or all the bits are located in an area of a constant homo-geneity, then π turns to the classical partition $\pi = [1]^n$.

The steganographic protocol we propose follows the same steps as in Section 2. But with the π -distance, the syndrome decoding seeks coset leaders among vectors with non null first block. Therefore, we can have better image quality since first blocks contain the less sensitive bits. In the following we describe the steps of the embedding/retrieval algorithm and give a practical example.

Let C be an $[n, k]$ linear error-block code of type π and of parity check matrix H. The syndrome of a vector $v \in V$ is computed by the product $S(v) = Hv^T$. Hence the retrieval algorithm is de_ned by $R(y) := S(y) = Hy^T$.

The first step of the embedding algorithm is also simple, it consists of com-puting $u := S(x) - m$.

Now to accomplish the embedding operation we compute $E(m; x) := x - e_u$ where e_u is the coset leader of u. The vector e_u is also the error vector found by decoding u in C. We can compute it by the decoding algorithm presented in Section 3.

Remark 5. In our steganographic protocol, we do not need to perform unique decoding. We just choose a coset leader with the only condition that it has min-imum π-weight in order to get the least modi_cations on the cover image.

The final step is computing $y = x - e_u$.

Example 3. Assume we have a (part of a) cover $x = 0001101$, and that the first block of three bits is located in an area that can hide more distortion than the second block of two bits, and that the remaining two blocks of one bit are more sensitive to distortion. The cover is of size 7 and we just de_ned the partition $\pi = [3][2][1]^2$. We use the code C of Example 2. Let $m = 1100$ be the message to hide. We follow the steps:

Embedding

1. $u = S(x) - m = 0110$
2. $e_u = 1010000$ (found from the syndrome table)
3. $y = E(x;m) = x - e_u = 1011101$

Retrieval

R(y) = S(y) = 1100

Linear Error-Block Codes in Steganography 9

5 Results

The proposed method was tested with 3 different messages to be hidden within the famous Lena gray-scale image of size 256 × 256. For each message, we ap-plied several linear error-block codes with deferent partitions. The messages are divided to blocks which have the same size as the code length.

Table 2. Steganography performance of a [7,3] error-block code.

π	s	p_π	(n-k)/s	p_π/s
$[1]^7$	7	2	0.5714	0.2857
$[2][1]^5$	6	2	0.6666	0.3333
$[2]^2[1]^3$	5	2	0.8000	0.4000
$[3][2][1]^2$	4	2	1.0000	0.5000
[4][2][1]	3	2	1.3333	0.6666

Table 3. Steganography performance of a [6,3] error-block code.

π	s	p_π	(n-k)/s	p_π/s
$[1]^6$	6	2	0.5000	0.3333
$[2]\ [1]^4$	5	1	0.6000	0.2000
$[3]\ [1]^3$	4	1	0.7500	0.2500
[3][2][1]	3	1	1.0000	0.3333
[5] [1]	2	1	1.5000	0.5000

Table 4. Steganography performance of a [9,3] error-block code.

π	s	p_π	(n-k)/s	p_π/s
$[1]^9$	9	2	0.6666	0.2222
$[2]^2\ [1]^5$	7	2	0.8571	0.2857
$[3][2][1]^4$	6	2	1.0000	0.3333
$[3][2]^2[1]^2$	5	2	1.2000	0.4000
$[4]^2\ [1]$	3	2	2.0000	0.6666

Tables 2, 3 and 4 summarize the results for one block embedding. In order to compare the performance of different partitions, we used the following pa-rameters, (n-k)/s measures the block-embedding rate, p_π measures the maximum embeddable blocks, it is also the π-covering radius of the code defined by

$$p_\pi = \max\{d_\pi(x, C),\ x \in F_q^n\}. \quad (8)$$

10 R. DARITI, E. M. SOUIDI

And finally p_π/s measures the block embedding average distortion.

The results show that a careful selection of the code and the partition is critical. For a fixed code, a partition of a few number of blocks causes a big block-average distortion p_π/s (if the covering radius p_π remains unchanged), whilst a big number of blocks causes a small block-embedding rate (n-k)/s . For the [6, 3] code of Table 3, it is clear that the partition $[2][1]^4$ is the best, since it provides the largest block-embedding capacity and the smallest block-embedding distortion.

6 Conclusion and perspective

The steganographic protocol we introduce in this paper allows to handle different cover bits using linear error block codes. The choice of the code is avery critical especially for covers with few first level bits. Experimental results show that there exist linear error block codes which provide high quality images and good security parameters. The forthcoming work involves specifying the space of codes to use in order to automatically achieve acceptable quality and security objectives.

References

1. R. Crandall: Some Notes on Steganography. Posted on Steganography Mailing List, http://os.inf.tu-dresden.de/~westfeld/crandall.pdf (1998).
2. A. Westfeld : F5-A Steganographic Algorithm. IHW '01: Proceedings of the 4th International Workshop on Information Hiding, 289{302 (2001).
3. K. Feng, L. Xu, F.J. Hickernell: Linear Error-Block Codes. Finite Fields Appl. 12, 638{652 (2006).
4. S. Ling, F. Ozbudak: Constructions and Bounds on Linear Error-Block Codes. Designs, Codes and Cryptography. 45, 297{316 (2007).
5. R. Dariti, E.M. Souidi: New Families of Perfect Linear Error-Block Codes. Submitted.
6. R. Dariti, E.M. Souidi: Cyclicity and Decoding of Linear Error-Block Codes. Journal of Theoretical and Applied Information Technology, 25, No. 1, 39{42 (2011).
7. P. Udomkavanicha, S. Jitman: Bounds and Modifications on Linear Error-block Codes. International Mathematical Forum, 5, No. 1, 35{ 50 (2010).
8. J.H. van Lint: Introduction to Coding Theory, Third Edition. Graduate Texts in Mathematics, Vol. 86, Springer, Berlin (1999).
9. C. Munuera: Steganography and Error Correcting Codes. Signal Process., 87, 1528{1533 (2007).
10. L.H. Chen, Y.K Lee A High Capacity Image Steganographic Model. IEE Proceedings Vision, Image and Signal Processing, Vol. 147, No. 3, 288-294 (2000).
11. X. Liao, Q. Wen: Embedding in Two Least Signi_cant Bits with Wet Paper Coding. CSSE'08: Proceedings of the 2008 International Conference on Computer Science and Software Engineering, 555{558 (2008).
12. X. Zhang, W. Zhang, S. Wang: Effcient Double-Layered Steganographic Embed-ding. Electronics letters, 43, 482 (2007).