STEGANOGRAPHIC SCHEME USING THE $\mathbb{Z}_4$-LINEAR GOETHALS CODES

Houda JOUHARI[1], EL Mamoun SOUIDI[2]
Laboratory of mathematics, informatics and applications
University Mohammed V Agdal
Faculty of Sciences BP 1014 Rabat Morocco
[1] jouharihouda1@gmail.com & [2] souidi@fsr.ac.ma

***Abstract*—In this paper, we show that certain families of non-linear codes can achieve better performance for application in steganography than simple linear codes currently in use, and that the theory of coverings functions should not be restricted to the binary case. The idea of this paper is to use the $\mathbb{Z}_4$-linearity of Goethals non-linear codes for the construction of a new steganographic scheme and to show that we can embed large amounts of information and can maintain good image quality when compared to the F5 method.**

***Index Terms*—Steganography, Galois Ring, Quaternary codes, Goethals codes.**

1 INTRODUCTION

The steganography is the science of hiding messages, where the sender communicates with the receiver by hiding her messages in generally trusted media, such as digital images, so that it is hard to distinguish between the original (cover) object and the objects carrying the message (stego objects). The message is typically hidden (embedded) in the cover image by slightly modifying individual elements of the cover (pixels, DCT coefficients, ect.).

An interesting steganographic method is known as the matrix encoding, introduced by Crandall[1] and analyzed by Bierbrauer [2] and independently discovered by van Dijk [3] and Galand [4]. Matrix encoding requires the sender and the receiver to agree in advance on a parity check matrix H, and the secret message is then extracted by the receiver as the syndrome (with respect to H) of the received object. This method was made popular by Westfeld [5], who made a specific implementation using Hamming codes in his F5 algorithm, which can embed t bits of message in (2^t-1) cover symbols by changing at most, one of them. Relations between covering codes [6, Section 14.2] and steganography were also used in large payload applications [7]. In [8], BCH codes were applied to achieve a tradeoff between embedding complexity and efficiency. An explicit description of the relationship between the maximum length embeddable (MLE) codes and perfect error correcting codes was presented by Zhang and Li in [9]. All the above-mentioned methods are applications of linear covering codes, which can be used in LSB steganography.

Let r be the embedded bit numbers of the secret messages and N the bit number of the cover-data and ρ the covering radius. To construct a steganographic scheme the following design problems arise:

- We want r/N, the embedding rate, to be large.
- The relative covering radius ρ/N, should be small to have good embedding efficiency.

We acknowledge, though, that the number of changes is not the only important factor influencing the security of the steganographic scheme but the choice of the cover object and the type of modifications play an equally important role.

In this paper, we describe a novel steganographic scheme based on the $\mathbb{Z}_4$-linearity of non-linear Goethals codes. The experimental results show that the proposed scheme can embed large amounts of information and can maintain good image quality as well.

The rest of the paper is organized as follows. In Section 2 we recall the relationship between information hiding and coding theory, and we give a brief introduction to the non-linear codes defined over $\mathbb{Z}_4$ and more especially Goethals codes. In Section 3 we discuss our proposed steganographic scheme. Experimental results are given in Section 4. Finally, Section 5 provides the conclusion.

2 CODING THEORY AND STEGANOGRAPHY

2.1 Notations

Let $\mathbb{F}_2$ denote the Galois fields $\{0,1\}$ and $\mathbb{F}_2^N$ denotes the vector space of all N-bit vectors $x = (x_1, \cdots, x_N)$. A linear $[N,k]$-code C of length N and dimension k is a k-dimensional vector subspace of $\mathbb{F}_2^N$, where the sum of two vectors and a multiplication of a vector by scalar are defined using the usual arithmetics in the finite field $GF(2)$. The $(N-k) \times N$ matrix is called a parity check matrix of C if $xH^T = 0$ for each $x \in C$, where H^T denotes the transpose of H. For $x \in \mathbb{F}_q^N$, the vector $s = xH^T \in \mathbb{F}_q^{N-k}$ is called the syndrome of x. For each syndrome $s \in \mathbb{F}_q^N$, the set $C(s) = \{x \in \mathbb{F}_q^N | xH^T = s\}$ is called a coset. Note that $C(0) = C$. Obviously, cosets associated with different syndromes are disjoint. We know also from elementary linear algebra, that every coset can be written as $C(s) = x + C$, where $x \in C(s)$ arbitrary. Thus, there are 2^{N-k} disjoint cosets, each consisting of 2^k vectors. Any member of the coset $C(s)$ with the smallest Hamming weight is called a coset leader and will be denoted as $e_L(s)$.
The Hamming weight ω_H of a vector x is defined as the number of ones in x. The distance between two vectors x and y is defined as the Hamming weight of their difference $d_H(x,y) = \omega_H(x-y)$.
The covering radius ρ of a code C is defined as

$$\rho = \max_{x \in \mathbb{F}_2^N} d_H(x, C)$$

where $d_H(x,C) = \min_{c \in C} d_H(x,c)$ is the distance between x and the code C.

2.2 Matrix Encoding

We now briefly review a few relevant known facts about embedding schemes and covering codes that appeared in [4]. Let $M \in \mathbb{F}_2^r$ be the secret messages and $x \in \mathbb{F}_2^N$ denote the cover-data. An embedding scheme on $\mathbb{F}_2^N$ with a distortion bound T is a pair of embedding and extraction functions defined as $Emb: \mathbb{F}_2^N \times \mathbb{F}_2^r \to \mathbb{F}_2^N$ and $Ext: \mathbb{F}_2^N \to \mathbb{F}_2^r$, such that

$$\forall (x, M) \in \mathbb{F}_2^N \times \mathbb{F}_2^r, Ext(Emb(x,M)) = M \quad (1)$$

$$\forall (x, M) \in \mathbb{F}_2^N \times \mathbb{F}_2^r, d_H(x, Emb(x,M)) \leq T \quad (2)$$

Equation (1) means that we can embed any message from $\mathbb{F}_2^r$ in any binary N-tuple and (2) states that we can do it using at most T changes.

Matrix encoding embeds data with the parity check matrix of a linear covering code. Let C be an $[N,k]$ code with a parity check matrix H and covering radius ρ. The embedding scheme defined by

$$Emb(x,M) = x + D(M - xH^T) = y$$

$$Ext(y) = yH^T$$

can hide $M \in \mathbb{F}_2^{N-k}$ of length $r = N-k$ in a sequence $x \in \mathbb{F}_2^N$ of length N using at most $T \leq \rho$ changes, where D is a decoding function.

Exemple 1 *(F5-Matrix Coding): F5 [5] is a LSB steganographic program that embeds binary message sequences into the LSBs of (pixels, DCT coefficients..) of images. F5 can embed k bits of message in* $2^k - 1$ *coefficients by changing at most one of them. The inputs are codewords (LSBs of pixels)* $x \in GF^{2^k-1}(2)$ *and the block of message* $M \in GF^k(2)$*. The coding function is defined as*

$$f(x) = \bigoplus_{i=1}^{2^k-1} x_i \cdot i$$

where, to do $\oplus$, the integer $x_i \cdot i$ is interpreted as a binary vector. And the encoding procedure is as follows: Compute the bit place that has to be changed as $s = M \oplus f(x)$ where the resulting binary vector s is interpreted as an integer. And then output the changed codeword

$$x' = \begin{cases} x & if s = 0 \\ (x_1, x_2, \cdots, x_s \oplus 1, \cdots, x_{2k+1}) & if s \neq 0 \end{cases}$$

which satisfies $M = f(x')$.

Steganography technique should generally have two important properties: good visual/statistical imperceptibility and a sufficient payload. The first is essential for the security of hidden communication and the second ensures that a large quantity of data can be conveyed. Two levels of protection can be done if the message is encrypted before hiding it [10].

2.3 Linear Codes Over $\mathbb{Z}_4$.

Let $\mathbb{Z}_4$ be the ring of integers modulo 4 and $\mathbb{Z}_4^n$ be the set of n-tuples over $\mathbb{Z}_4$.

Definition 1 *A* $\mathbb{Z}_4$*-linear code of length* n *is a submodule of* $\mathbb{Z}_4^n$.

By a quaternary code $\mathcal{C}$ of length n we shall mean a linear block code over $\mathbb{Z}_4$. We define the Lee distance for words over $\mathbb{Z}_4$ as follow.

Definition 2 *A Lee weigth* $\omega_L: \mathbb{Z}_4 \to \mathbb{Z}$ *of an element in* $\mathbb{Z}_4$ *is defined as*

$$\omega_L(0) = 0, \quad \omega_L(1) = \omega_L(3) = 1, \quad \omega_L(2) = 2$$

and a Lee weight of a vector $c \in \mathbb{Z}_4^n$ is naturally: $\omega_L(c) = \sum_{i=1}^n \omega_L(c_i)$. The Lee distance is defined as $d_L(x,y) = \omega_L(x-y)$.

We define an inner product on $\mathbb{Z}_4^n$ by $\langle a, b \rangle = a_1 b_1 + \cdots + a_n b_n \ (mod\ 4)$ where $a = (a_1, \cdots, a_n)$

and $b = (b_1, \cdots, b_n)$, then the notions of dual code ($\mathcal{C}^{\perp}$), self-orthogonal code ($\mathcal{C} \subseteq \mathcal{C}^{\perp}$), and self-dual code ($\mathcal{C} = \mathcal{C}^{\perp}$) are defined in the standard way.

Recently Hammons [11] showed that non-linear codes can be very simply constructed as binary images under a certain natural map, called the Gray map.

2.4 The Gray Map

For an element in $\mathbb{Z}_4$, the Gray map $\phi: \mathbb{Z}_4 \to \mathbb{F}_2^2$ is defined by $\phi(0) = 00$, $\phi(1) = 01$, $\phi(2) = 11$, and $\phi(3) = 10$.

The Gray map is then extended to a map, also denoted ϕ, from $\mathbb{Z}_4^n$ to $\mathbb{F}_2^{2n}$ in the obvious way (by concatenating the images of each component).

Formally, we define three maps from $\mathbb{Z}_4$ to $\mathbb{F}_2$ by:

c	$\alpha(c)$	$\beta(c)$	$\gamma(c)$
0	0	0	0
1	1	0	1
2	0	1	1
3	1	1	0

We construct binary codes from $\mathbb{Z}_4$-linear codes using the Gray map $\phi : \mathbb{Z}_4^n \to \mathbb{F}_2^{2n}$ given by

$$\phi(c) = (\beta(c), \gamma(c)) \ , \qquad c \in \mathbb{Z}_4^n.$$

Lemme 1 *The Gray map $\phi: (\mathbb{Z}_4^n, d_L) \to (\mathbb{F}_2^n, d_H)$ is an isometry of metric spaces, that is, ϕ is a bijection and $d_H(\phi(x), \phi(y)) = d_L(x, y)$ for all $x, y \in \mathbb{Z}_4^n$.*

If u and v are in $\mathbb{Z}_4^n$, then

$$\phi(u) + \phi(v) = \phi(u + v + 2(u * v))$$

and

$$\phi(u + v) = \phi(u) + \phi(v) + \phi(2\alpha(u) * \alpha(v))$$

where $u * v$ is the componentwise product of the two vectors u and v in $\mathbb{Z}_4^n$.

Knowing that all the digital files are binary, we simply use the inverse gray map for working on the support quaternary.

2.5 Goethals codes

The Goethals codes discovered by Goethals [12,13], of length 2^{m+1} has $2^{2^{m+1}-3m-2}$ codewords and minimum Lee distance 8 for any odd integer $m \geq 3$. The important property of Goethals codes is that codewords have low correlation values, which makes them useful in Code-Division-Multiple-Access (CDMA) communications systems, and they have excellent error-correcting properties.

We refer the reader to [14] for information concerning Galois rings and their use in the construction of codes over $\mathbb{Z}_4$.

The Galois ring $GR(4^m)$ is a finite ring of characteristic 4 with 4^m elements. In $GR(4^m)$ there exists a unique cyclic subgroup of order $(2^m - 1)$.

Let β be a generator of this subgroup, $\mathcal{T} = \{0,1,\beta,\cdots,\beta^{2^m-2}\}$ and $\mu: \mathbb{Z}_4 \to \mathbb{F}_2$ denote the modulo 2 reduction map. By extending μ to $GR(4^m)$ in a natural way, we show that $\mu(\mathcal{T}) = \mathcal{F}$, where $\mathcal{F}$ is the finite field of order 2^m.

Our Goethals code is thus defined as $C = \phi(\mathcal{C})$, where $\mathcal{C}$ is the quaternary code with parity-check matrix given by:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & \beta & \beta^2 & \cdots & \beta^{2^m-2} \\ 0 & 2 & 2\beta^3 & 2\beta^6 & \cdots & 2\beta^{3(2^m-2)} \end{bmatrix}$$

in which each β^j ($j \geq 0$) should be replaced by $(b_{0j}, b_{1j}, \cdots, b_{m-1,j})^T$, where $b_{ij} \in \mathbb{Z}_4$. Then H is a matrix of $2m + 1$ rows and 2^m colonnes over $\mathbb{Z}_4$.

In [11], it is shown that if m is odd, then $\mathcal{C}$ has minimum Lee distance 8 which is equal to the minimum Hamming distance of C. The binary $(2^{m+1}, 2^{2^{m+1}-3m-2}, 8)$ code C has parameters that are identical to the (extended) binary Goethals code. The quaternary Goethals code $\mathcal{C}$ is a $\mathbb{Z}_4$-linear code of length 2^m which has $2^{2^{m+1}-3m-2}$ codewords and minimum Lee distance 8 for any odd $m \geq 3$. The covering radius of the $\mathbb{Z}_4$-linear Goethals code is 6 [15].

3 THE PROPOSED STEGANOGRAPHIC SCHEME

Now taking a non-linear Goethals codes as example, we show how the performance of binary steganographic method can be improved by applying a non-linear covering codes on our information hiding method.

First we present a complete decoding algorithm for the $\mathbb{Z}_4$-linear Goethals codes of length 2^m.

3.1 Decoding of $\mathbb{Z}_4$.-Linear Goethals Codes

In this section, we describe a decoding algorithm for the $\mathbb{Z}_4$-linear Goethals code $\mathcal{C}$, i.e., an algorithm that for any received vector finds the closest codeword .

Let $r \in \mathbb{Z}_4^{2^m}$ be the received vector which differs from the original codeword c by an error word $e = (e_X)_{X \in \mathcal{T}} \in \mathbb{Z}_4^{2^m}$ i.e, $r = c + e$. We calculate the syndrome of the received vector

$$S = rH^T = eH^T = (t, A + 2B, 2C)$$

where $t \in \mathbb{Z}_4$ and $A, B, C \in \mathcal{T}$.

Now the task for the decoder is: for any syndrome give the minimum weight codeword e (leader) in the corresponding coset $(e + \mathcal{C})$. The syndrome equations that have to be solved are

$$\sum_{X\in\mathcal{T}} e_X = t, \quad t \in \mathbb{Z}_4$$
$$\sum_{X\in\mathcal{T}} e_X X = A + 2B, \quad A, B \in \mathcal{T}$$
$$2\sum_{X\in\mathcal{T}} e_X X^3 = 2C, \quad C \in \mathcal{T}$$

Let X, Y, A, B, etc. denote elements in $\mathcal{T}$ and x, y, a, b their respective projections modulo 2 in $\mathcal{F}$.

As an example the decoding of coset corresponding to syndromes with $t = 1$ are given below. The cases $t = 0,2$ and 3 are similar.

Let $S = (1, A + 2B, 2C)$ denote the syndrome of a coset, σ be the error locator polynomial and Tr denotes the generalized trace map from $GR(4^m)$ to $\mathbb{Z}_4$.

Algorithm 1 Decoding $\mathbb{Z}_4$-linear Goethals codes

1). If $b = 0$ and $c = a^3$, then the coset leader has Lee weight 1 and is uniquely determined by $x = a$ and $e_X = 1$.

2). If $b \neq 0$ and $c = a^3$, then the coset leader has Lee weight 3 and is uniquely determined by $x = a + b$, $e_X = 2$, $y = a$ and $e_Y = -1$.

3). If $b \neq 0$, $c \neq a^3$ and $Tr(b^3/(a^3 + c)) = 0$, then the coset leader has Lee weight 3. The coset leader is uniquely determined such that x and y are solutions of

$$b^2u^2 + (a^3 + c)u + a^4 + a^2b^2 + ac + b^4 = 0,$$

$$e_X = e_Y = 1,\ z = a + \frac{a^3 + c}{b^2} \text{ and } e_Z = -1.$$

4). If $\sigma(u) = u^3 + au^2 + (a^2 + b^2)u + ab^2 + c$ has three distinct zeros in $\mathcal{F}$ then a coset leader has Lee weight 3 and is uniquely determined such that x, y, z are the three distinct zeros in $\mathcal{F}$ of $\sigma(u)$ and $e_X = e_Y = e_Z = -1$.

5). If none of (1)-(4) hold, then any coset leader has Lee weight ≥ 5.

In the considerably more complicated cases when more than 3 errors occur it is shown in [15] how to construct a coset leader in any coset. In addition one gets the following results.

Theorem 1

1. For any coset with syndrome $S = (0, A + 2B, 2C)$ there exists a coset leader of Lee weight ≤ 6.
2. For any coset with syndrome $S = (t, A + 2B, 2C)$ where $t = 1$ or $t = 3$, there exists a coset leader of weight ≤ 5.
3. Let $m \geq 5$, then for any coset with syndrome $S = (2, A + 2B, 2C)$ there exists a coset leader of weight ≤ 4.

3.2 Embedding Process

Our proposed data hiding method is based on Goethals codes $C = \phi(\mathcal{C})$ of length 2^{m+1}, where m is odd and ≥ 3.

The embedding process consists of the following steps:

Algorithm 2 The Proposed Embedding Process

Input Let $x = (x_1, \cdots, x_{2^{m+1}})$ in $\mathbb{F}_2^{2^{m+1}}$ be a block of cover data,
$M = (M_1, \cdots, M_{4m+2})$ in $\mathbb{F}_2^{4m+2}$ the message to hide such that $\phi^{-1}(M)$ takes the form $(t, A + 2B, 2C)$.

Output $y = (y_1, \cdots, y_{2^{m+1}})$ in $\mathbb{F}_2^{2^{m+1}}$, stego-data such that: $d_H(x, y) \leq \rho$.

(a) We compute $a = \phi^{-1}(x)$ and $\mathcal{M} = \phi^{-1}(M)$
(b) Compute the syndrome: $S = \mathcal{M} - aH^T$ over $\mathbb{Z}_4$
(c) If $S = 0$, then $e = 0$ else we look for an e such that $eH^T = S$ and $\omega_L(e) \leq \rho$ by using Algorithm 1 ;
(d) Put: $b = (a + e)\ mod\ 4$
(e) [*Embedding / modification*] y is the stego object

$$y = Emb(M, x) = \phi(b) = x + \phi(e) + \phi(2\alpha(a) * \alpha(e))$$

(f) if we are at the end of the cover object, Stop; otherwise, go to 1.

In fact, this embedding process works because :

$$\begin{aligned} d_H(x, y) &= d_H(\phi(a), \phi(b)) \\ &= d_L(a, b) \\ &= \omega_L(a - b) \\ &= \omega_L(e) \leq \rho \end{aligned}$$

3.3 Extracting Process

The message embedded is retrieved from the stego-data by applying the proposed extracting function given as follows:

$$M = Ext(y) = \phi(\phi^{-1}(y).H^T)$$

M is the secret information which the receiver extract from the cover.

In fact:

$$\phi(\phi^{-1}(y).H^T) = \phi(b.H^T) = \phi((a+e).H^T)$$
$$= \phi(a.H^T + e.H^T)$$
$$= \phi(a.H^T + \mathcal{M} - a.H^T)$$
$$= \phi(\mathcal{M}) = \phi(\phi^{-1}(M)) = M$$

3.4 Evaluation of image quality

For comparing stego-image with cover results requires a measure of image quality, commonly used measures are Mean-Squared Error, Peak Signal-to-Noise Ratio.

Mean-Squared Error

The mean-squared error (MSE) between two images $I_1(i, j)$ and $I_2(i, j)$ is defined by

$$MSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(I_1(i, j) - I_2(i, j))^2}{m \times n} \quad (3)$$

m and n are the number of rows and columns in the input images, respectively.

Peak Signal-to-Noise Ratio

Peak Signal-to-Noise Ratio (PSNR) avoids this problem by scaling the MSE according to the image range:

$$PSNR = 10log_{10}\left(\frac{I_{max}^2}{MSE}\right) \quad (4)$$

where I_{max} is is the maximum possible pixel value of the image, which is equal to 255 for 8 bit gray-scale images, PSNR is measured in decibels (dB). PSNR is a good measure for comparing restoration results for the same image, but between image comparisons of PSNR are meaningless. Generally speaking, if the value of PSNR is more than 30 dB, then people have difficulty to notice the difference between the cover image and the stego image.

Histogram

The histogram is a function that counts the number of observations that fall into each of the disjoint categories (known as bins). The height of the bins represents the number of values that fall within each range. An image histogram is a chart that shows the distribution of intensities in an indexed or intensity image.

4 EXPERIMENTAL RESULTS

For concreteness, we assume that the cover object used for communication is a gray-scale digital image whose pixels are integers between 0 and 255, then we assign 8 bits to each pixel value.

Our steganographic scheme features two essential components. First, is the selection of places within the cover that might be modified and that used to hide the secret message. The second component is the steganographic protocol.

The best widely known steganographic algorithm to embed secret information in Spatial and Transform domain of images is based on modifying the least significant bit layer of images, hence known as the LSB technique. This technique makes use of the fact that the least significant bits in an image could be thought of random noise and changes to them would not have any effect on the image.

In our example for $m = 3$, let $\mathcal{G}_3$ be a $\mathbb{Z}_4$-linear Goethals code of length 2^3 and covering radius 6, and witch has the parity-check matrix described below:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 2 & 3 & 1 \\ 0 & 0 & 1 & 0 & 3 & 3 & 3 & 2 \\ 0 & 0 & 0 & 1 & 2 & 3 & 1 & 1 \\ 0 & 2 & 2 & 2 & 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 2 & 2 & 2 & 0 & 2 \end{bmatrix}$$

By applying the non-linear Goethals code $G_3 = \phi(\mathcal{G}_3)$, we can hide 14 bits in a sequence of 16 bits by changing at most 6 bits.

Thus, for our method we embed the message $(M_1, \cdots, M_{14})$ in the LSBs 2^4 pixel gray values $(p_1, \cdots, p_{16})$ by at most 6 changes in the following manner $(M_1, \cdots, M_{14}) = (x_1, \cdots, x_{16}).H^T$ where x_i denotes the LSB of p_i

4.1 Design Details

The proposed scheme are implemented to visualize the data-hiding effect. In the following we describe the steps of our embedding algorithm implemented under Matlab:

1. Read the host gray-scaling image A, which is to be modified and to embed data.
2. The host image is partitioned into groups of 16 pixels.
3. The size of the image and the number of bits to be embedded in each group of 16 pixels together determine the capacity of embedding.
4. If the message size fits to the estimated ca-pacity, the embedding proceeds (go to step 5), otherwise an error message showing the maximal possible length is displayed.
5. The texte message to be embedded is divided into segments of 14 bits that are embedded into a groups of 16 pixels along the embedding process.
6. For each group of 16 pixels, do the following:

- Extract the cover-data $x = (x_1, \cdots, x_{16})$ of 16 bits from the group by concatenation the LSB of each pixel

value;

- Hide the secret message $M = (M_1, \cdots, M_{14})$ of 14 bits into the cover-data such that

$$\phi(M) = (\phi(M_1 M_2), \cdots, \phi(M_{13} M_{14}))$$

has the form $(t, A+2B, 2C)$. If the last 6 bits of the message not verified

$$(\phi(M_9 M_{10}), \cdots, \phi(M_{13} M_{14})) = 2C$$

they are replaced by zero, and the same 6 message bits are re-embedded in the next group of pixels, then Go to step (7).

7. Store the resulting image as Stego Image (S).

In this present implementation Lena gray-scale image of 512×512 pixels and Baboon grayscale image of 298×298 pixels, has been taken as cover images as shown in Figures 1(a) and 2(a). For each image, we applied our method and we present a comparative study in Figures 1 and 2 of the proposed method with the optimal $F5$ algorithm.

(a) original host image.

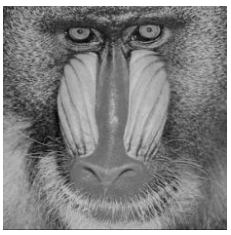
(b) After embedding 10240 bytes by the proposed scheme (PSNR= $44{,}53$).

(c) After embedding 5120 bytes by the $F5$ method (PSNR= $44{,}56$)

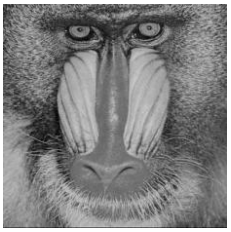
(d) After embedding 8730 bytes by the proposed scheme (PSNR= $44{,}54$).

(e) After embedding 4365 bytes by the $F5$ method (PSNR= $44{,}55$)

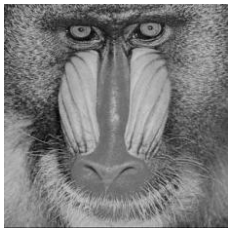
Figure 1: Embedding effect on Lena image.



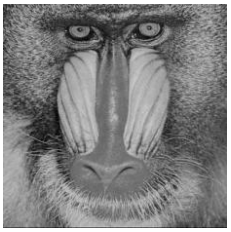
(a) original host image.



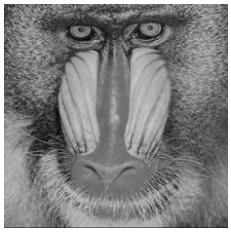
(b) After embedding 5914 bytes by the proposed scheme (PSNR= $58{,}79$).



(c) After embedding 2957 bytes by the $F5$ method (PSNR= $58{,}81$).



(d) After embedding 3468 bytes by the proposed scheme (PSNR= $58{,}78$).



(e) After embedding 1369 bytes by the $F5$ method (PSNR= $58{,}78$)

Figure 2: Embedding effect on Baboon image.

4.2 Embedding Capacity of the proposed scheme

In the proposed scheme the bit numbers of the secret messages that carried in sequence of length N bits by using the $\mathbb{Z}_4$-linearity of Goethals codes is up to

$$log_2\left[\binom{N}{0} + \binom{N}{1} + \cdots + \binom{N}{6}\right]$$

In our example, using $F5$ method with $k = 4$ to carry secret messages in 512×512 pixels of Lena grayscale image, its rate of the embedding capacity is

$$\frac{log_2\left[\binom{15}{0} + \binom{15}{1}\right]}{15} = \frac{4}{15} = 0{,}2667$$

The rate of the embedding capacity using the proposed method with $m = 3$ is

$$\frac{log_2\left[\binom{16}{0} + \cdots + \binom{16}{6}\right]}{16} = \frac{13.8623}{16} = 0.8664$$

Therefore, about $230K$ bits of secret messages can be embedded into the 512×512 image applying the proposed method compared to $70K$ bits applying the

$F5$ method.

The proposed scheme has the following features:

• Applying the proposed method much amount of data could be embedded in the image. Therefore, a part of the image remains unused resulting in a distortion less image.

• The amount of data embedded using the proposed scheme leads indeed to a good results compared to the $F5$ algorithm for embedding the secret data into a cover image, see Table 1.

• The effectiveness of the embedding process has been studied by calculating PSNR for the two digital images using the proposed method and the $F5$ algorithm as given in Table 2.

Table 1: Comparaison of amount of embedded data between the proposed method and the $F5$ method.

Host Image	Amount of embedded data applying (in bytes)	
	The proposed scheme	$F5$
Lena	28.672	8.738
Baboon	9.712	2.960

Table 2: Comparaison on PSNR values between the proposed method and the $F5$ method after embedding 2.957 bytes.

Host Image	PSNR (dB)	
	The Proposed scheme	$F5$
Lena	44,5565	44,5638
Baboon	58.8065	58,8156

By comparing the histograms (See Figures 3, 4, 5 and 6) of the Lena and Baboon images before and after the embedding, higher security performance was inferred applying our of the Lena and Baboon images before and after the embedding, higher security method. This improves the imperceptibility and enhances the embedding capacity.

5 CONCLUSION

In this paper, we have presented a novel and adaptive method to embed the secret data in the cover image with a good imperceptibility and a high embedding capacity. The receiver does not need the original image to extract the information. Our testing results have shown that our method based on Goethals codes leads indeed to good results compared to the $F5$ method and can maintain a good image quality which is seen in the PSNR value.

This paper has presented a novel steganography scheme capable of concealing a large amount of data in a binary image when compared to the $F5$ method. one future research direction is to account for human visual effects during the data embedding process.

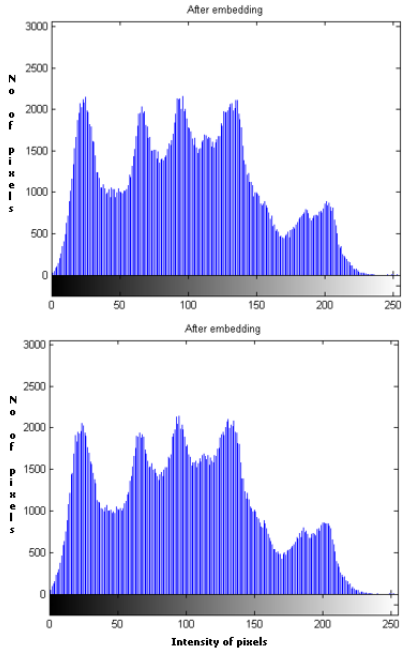


Figure 3: Histogram of Lena for Our proposed method.

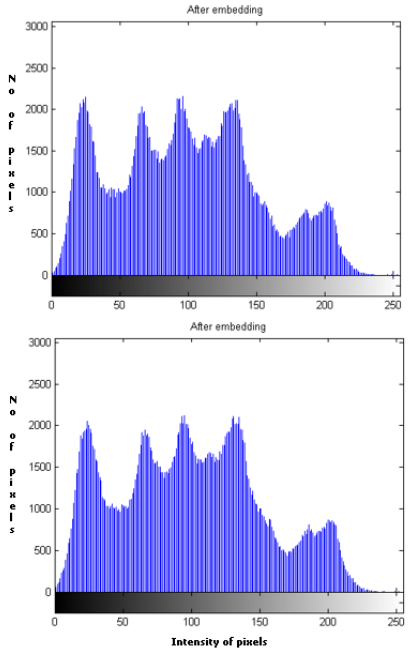


Figure 4: Histogram of Lena for $F5$ method.

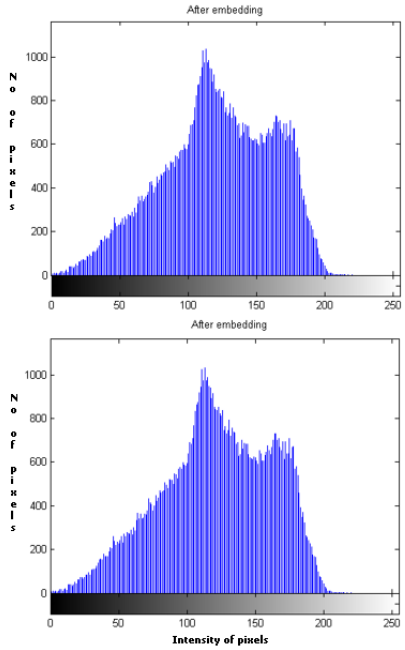


Figure 5: Histogram of Baboon for Our proposed method.

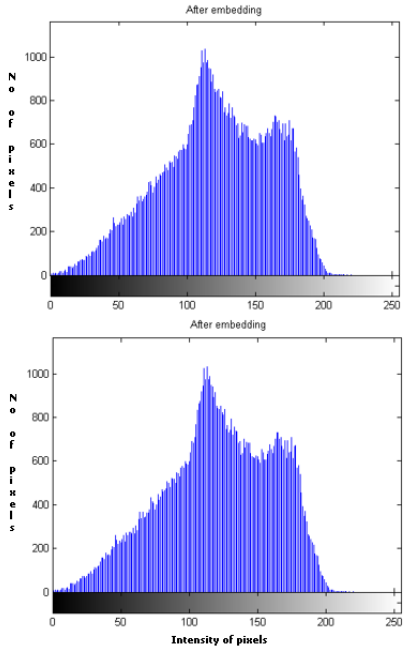


Figure 6: Histogram of Baboon for $F5$ method.

REFERENCES

[1] Crandall, R.: Some notes on steganography. Posted on steganography mailing list (1998). http://os.inf.tu-dresden.de/ westfeld/crandall.pdf.

[2] Bierbrauer, J.: On Crandall's Problem. Personal Communication(1998). http://www.ws.binghamton.edu/fridrich/covcodes.pdf.

[3] Van Dijk, M., Willems, F.: Embedding information in grayscale images. In: 22nd Symp. Information and Communication Theory Benelux, Enschede, The Netherlands, pp.147–154 (2001).

[4] Galand, F., Kabatiansky, G.:Information hiding by coverings. In: Proc. ITW, Paris, France (2003), pp. 151–154.

[5] Westfeld, A.: F5-A steganographic algorithm: High capacity despite better steganalysis. In: Moskowitz, I.S. (ed) IH 2001. LNCS, vol. 2137, pp. 289–302. Springer, Heidelberg (2001).

[6] Bierbrauer, J.: Introduction to Coding Theory. Chapman and Hall, CRC Press (2005).

[7] Fridrich, J., Soukal, D.: Matrix embedding for large payloads. In: IEEE Transactions on Inf. Security and Forensics, vol. 1, Issue. 3, pp. 390–395, Sept (2006).

[8] Schönfeld, D., Winkler, A.: Embedding with syndrome coding based on BCH codes. In: workshop on Multimedia and security, pp. 214–223 (2006).

[9] Zhang, W., Li, S.: A Coding Problem in Steganography. Designs, Codes and Cryptography, Vol. 46, Issue 1, pp. 67–81 (2008).

[10] Stalling, W.: Cryptography and Network Security. Englewood Cliffs, NJ: Prentice-Hall, (1999).

[11] Hammons, R., Kumar, P. V., Calderbank, A. R., Sloane, N. J. A., Solé, P.: The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes. In: IEEE Transactions on Information Theory, vol. 40, Issue. 2, pp. 301–319, Mar (1994).

[12] Goethals, J. M.: Two dual families of nonlinear binary codes. In: Electronics Letters, vol. 10, Issue. 23, pp. 471–472 (1974).

[13] Goethals, J. M.: Nonlinear codes defined by quadratic forms over GF(2). Information and Control, vol. 31, Issue. 1, pp. 43–74 (1976).

[14] Huffman, W. C., Pless, V.: Fundamentals of error-correcting codes. Cambridge Univ (2003).

[15] Helleseth Tor, Vijay Kumar, P.: The Algebraic Decoding of the $\mathbb{Z}_4$-Linear Goethals Code. In: IEEE Transactions on Information Theory, vol. 41, Issue. 6, pp. 2040–2048 (1995).