Lecturas Matemáticas
Volumen **41** (2) (2020), páginas 89-110
ISSN 0120-1980

A mixed steganographic and fragile watermarking algorithm for the authentication of secret message in a digital image

Un algoritmo esteganográfico y de marca de agua frágil mixto para la autenticación del mensaje secreto en una imagen digital

Erodis Pérez Michel, Yuniel Guzman Bazan and Yisel de los Angeles González Pompa

University of Granma, Cuba

ABSTRACT. This paper proposes a steganographic method which uses fragile watermarking techniques proposed in the literature. In embedding phase, the cover image is divided into 32×32 non-overlapped blocks and each block are selected according to discrete fractional-order logistic map position generator. Then, each block is divided in 16 sub-block of 8×8, from them only four sub-blocks are used to embed the watermark and four to embed the secret bits. By using SHA-256 hash function is generated the watermark. The generated 256-bit binary watermark is embedded into two 8×8 sub-block with LSB substitution. The secret bits are embedded into LSB in Spatial or Frequency domain respectively. This research aims to implement a robust steganographic algorithm with the ability to detect if the authenticity of the secret message has not been compromised. In order to ensure authentication of the secret message and rate of tamper detection, this article proposes an efficient mixed steganographic and fragile watermarking scheme based on discrete fractional-order Logistic map. As a result, the proposed method produces stego images with a high visual quality and good PSNR values, which is in correspondence with the heuristic values of PSNR. Moreover, it detects the tampered regions of the stego image, which ensures the authentication of the secret message. Finally, generated SHA-256 hash values and discrete fractional-order logistic map have been altogether used to improve the reliability of our method.

***Key words*:** Image steganographic algorithm, Fragile watermarking, Tamper detection, Secret message authentication.

RESUMEN. En este artículo se propone un método esteganográfico el cual utiliza técnicas de marca de agua frágiles propuestas en la literatura. En la fase de incrustación, la imagen de portada se divide en bloques no superpuestos de 32×32 y cada bloque se selecciona de acuerdo con generador de posición basado en el mapa logístico de orden fraccionario. Luego, cada bloque se divide en 16 subbloques de 8×8, de ellos solo se utilizan cuatro subbloques para incrustar la marca de agua y cuatro para incrustar los bits secretos. La marca de agua se genera utilizando la función hash SHA-256. La marca de agua binaria de 256 bits generada se incrusta en dos subbloques de 8×8 con sustitución LSB. Los bits secretos se incrustan en el dominio espacial(LSB) o en el dominio de la frecuencia respectivamente. Esta investigación tiene como objetivo implementar un algoritmo esteganográfico robusto con la habilidad de detectar si la autenticidad del mensaje secreto no ha sido comprometida. Para garantizar la autenticación del mensaje secreto y la tasa de detección de manipulación, se propuso un algoritmo esteganográfico mixto que utiliza marcado de agua frágil eficiente basado en un mapa logístico de orden fraccionario discreto. Como resultado, el método propuesto produjo stego imágenes con una alta calidad visual y un buenos valores de PSNR, que está en correspondencia con los valores heurísticos de PSNR. Además, detecta las regiones manipuladas de la imagen stego, lo que garantiza la autenticación del mensaje secreto. Finalmente, los valores generados por la función hash SHA-256 y el mapa logístico discreto de orden fraccional se han utilizado para mejorar la fiabilidad del método propuesto.

***Palabras clave*:** Algoritmo esteganográfico de imagen, Autenticación de mensaje secreto, Detección de manipulación, Marca de agua frágil.

2010 AMS Mathematics Subject Classification. Primary 65G20; Secondary 94A08, Third 68P30.

1 Introduction

The development of computer technologies and communications and the rapid growth of the Internet have made it a widely used medium to transmit a large amount of information in a diverse digital form such as audio, video, image and text. This information can be public or private; in case of private information; the risk of being revealed by unauthorized external circumstances is high. Therefore, it is necessary to implement new methods to protect confidential information against illegal access and manipulation. Currently, watermark and steganography techniques offer a certain level of security through private communication against unauthorized human interception.

In this paper, steganography is defined as the practice of imperceptibly altering a digital medium to embed a secret message and watermarking as the practice of imperceptibility altering a digital medium to embed information about that digital medium [8]. While the objectives of the watermarking and steganography are different, both applications share certain high-level elements. Both systems consist of an embedder and a detector. The embedder takes two inputs. One is the payload that it is wanted to embed (for example, the watermark or secret message), and the other is the cover.

In recent years, digital watermarking has attracted important research interests and several techniques have been developed. In case of fragile watermark, it is intended to

verify the authentication of digital content [1]. When a part of the original content is replaced with false information, it is convenient to be able to locate the tampered areas. Some fragile watermark schemes divide a host image into small blocks and embed the mark in each block. Also, fragile watermarks are designed to detect minimal changes in an image; it is also possible to identify where an image has been altered.

1.1 Literature survey

Several techniques of digital steganography and fragile watermarking have been proposed lately. All of them have a fundamental premise of inserting secret information in a cover medium to generate a stego output. The aforementioned techniques can be achieved in two ways: spatial domain and frequency domain techniques. Several methods have been proposed for steganography, each with its advantages and disadvantages. Hussain et al. [12] presents a literature review of image steganography techniques in the spatial domain for last five years. Moreover, the authors highlight that the existing embedding techniques may not be perfect, they propose as objective to provide a comprehensive survey and to highlight the pros and cons of existing up-to-date techniques for researchers that are involved in the designing of image steganographic system. Also, the general structure of the steganographic system and classifications of image steganographic techniques with its properties in spatial domain are exploited. Furthermore, different performance matrices and steganalysis detection attacks are also discussed. The paper concludes with recommendations and good practices drawn from the reviewed techniques.

Recently, in [23], the authors proposed a steganography method based on DCT and entropy thresholding technique. This steganographic algorithm uses a random function in order to select the positions of image blocks where the elements of the binary sequence of a secret message will be inserted. Also, Chowdhuri and other proposed a weighted matrix based steganographic scheme that has been designed for highly compressed color image through discrete cosine transform (DCT) to maintain a good equilibrium between payload and imperceptibility. Here, the AC components are collected from (8×8) quantized DCT coefficient matrices of YCbCr channel. Then, a series of (3×3) original matrices are formed to hide secret data. The gathering of AC components is controlled by 128-bit shared secret key. Finally, the author demonstrated that the proposed scheme provides good embedding capacity with the high visual quality compared to existing state-of-the-art methods [7].

In [11] new algorithms are developed for strengthening the existing cybersecurity frameworks, ensure security, privacy, copyright protection and authentication of data. In the paper, a new technique for copyright protection, data security and content authentication of multimedia images is presented. The copyright protection of the media is taken care of by embedding a robust watermark using an efficient inter-block coefficient differencing algorithm and is proposed as Scheme I. Scheme II has been utilized to ensure both copyright protection, and content authentication. The authentication of the content has been ensured by embedding a fragile watermark in spatial domain while as copyright protection has been taken care of utilizing a robust watermark. In order to thwart an adversary and

ensure that it has no access to actual embedded data, the authors make use of a novel encryption algorithm in conjunction with Arnold transform to encrypt data prior to its embedding. The experimental results reveal that the proposed framework offers high degree of robustness against single/dual/triple attacks; with Normalized Correlation (NCC), more than or equal to 0.95. Besides, the fragile watermark embedding makes the system capable of tamper detection and localization with average BER more than 45% for all signal processing/geometric attacks. The average PSNR achieved for both schemes is greater than 41 dB. A comparison of the proposed framework with various state-of-the-art techniques demonstrate its effectiveness and superiority.

A reversible image authentication method that can improve the accuracy of tamper detection and the quality of watermarking image is presented in [19]. In the proposed method, reversible data hiding is implemented with two identical host images, where the secret information is embedded in one host image while the distortion information is embedded in the other image. In the authentication process, the information extracted from the image is compared with the original authentication information to judge whether the image was tampered with. Experimental results show that the algorithm can improve the accuracy of tamper detection and guarantee the image quality while a large number of authentication information is embedded.

In [10], by using SHA-256 hash function, a novel block based fragile watermark embedding and tamper detection method is proposed. In watermark embedding phase, host image is divided into 32×32 non-overlapped blocks. Each 32×32 block is then divided into four 16×16 non-overlapped sub-blocks. The entire hash value of the first three sub-blocks is generated as a watermark using SHA-256 hash function. The generated 256-bit binary watermark is embedded into the least significant bits (LSBs) of the fourth sub-block and watermarked image is obtained. In tamper detection phase, the detection of tampered block has been performed by comparing the hash value obtained from the three sub-blocks with the extracted watermark from the fourth sub-block of the watermarked image. The performance of the proposed method has been evaluated by applying linear and nonlinear attacks to the different regions of the watermarked images. Experimental results show that the proposed method detects all the tampered regions of the attacked images and high visual quality of watermarked images has been obtained.

The research presented in this article aims to implement a robust steganographic algorithm with the ability to detect if the authenticity of the secret message has not been compromised. In order to ensure authentication of the secret message and rate of tamper detection, we propose a mixed steganographic and fragile watermarking scheme based on discrete fractional-order Logistic map. The discrete fractional-order Logistic map [13] is used to generate the chaotic positions of the image blocks that will be marked. Following [16], in this paper digital content authentication is defined as the fact of determining whether a multimedia object has been maliciously tampered. Taking into account the previous definition, authentication of the secret message is defined as the ability of our steganographic algorithm to ensure that the embedded message is authentic and that the communication channel is secure. As a main result, the proposed method produced stego images with a high visual quality and good PSNR values, which is in correspondence with

the heuristic values of PSNR [20, 24]. In addition, it detects the tampered regions of the stego image, which ensures authenticity of the secret message.

Finally, the paper is organized as follows: materials and methods are introduced in Section 2. Section 3 describes the proposed method, including the steganographic and the tamper detection algorithm. Experimental results are demonstrated in Section 4. The conclusions are given in Section 5.

2 Materials and methods

2.1 Discrete Cosine Transform

Let $\mathcal{C}$ be the cover image (gray scale or RGB images) and let $\mathcal{K}$ be the set of all the non-overlapping 8×8 bytes blocks of $\mathcal{C}$, such that

$$\mathcal{C} = \bigcup_{k \in \mathcal{K}} B^k. \tag{1}$$

Moreover, let $\mathcal{B}^k$ be the two dimensional discrete cosine transform. The relationship between $\mathcal{B}^k \equiv$ DCT and its inverse $B^k \equiv$ IDCT (Inverse Discrete Cosine Transform) is given by

$$\mathcal{B}^k_{u,v} = \sum_{0 \leq i,j \leq N-1} \mathbb{K}_{u,v}(i,j) B^k_{i,j}, \quad 0 \leq u, v \leq 7, \tag{2}$$

$$B^k_{i,j} = \sum_{0 \leq u,v \leq N-1} \mathcal{B}^k_{u,v} \mathbb{K}_{u,v}(i,j), \quad 0 \leq i, j \leq 7, \tag{3}$$

where $\mathbb{K}_{m,n}(x,y) = \mathcal{P}_m(x)\mathcal{P}_n(y)$ with

$$\mathcal{P}_n(x) = \sigma(n) \cos\left(\frac{\pi n(2x+1)}{2N}\right),$$

and

$$\sigma(n) = \begin{cases} \sqrt{1/N} & \text{if} \quad n = 0, \\ \sqrt{2/N} & \text{otherwise.} \end{cases}$$

2.2 Quantification procedure and zigzag scan order

From (2) each integer block B^k is transformed into a real block $\mathcal{B}^k$, which is scaled according to the quantification matrix Q^μ by a compression quality factor μ

$$Q^\mu = \chi(\mu) \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix} \tag{4}$$

where $\chi(\mu) = \dfrac{100 - \mu}{50}$, with $50 < \mu < 100$.

The process of quantization is accomplished by dividing each element $\mathcal{B}^k$ by the corresponding element in the quantification matrix Q^μ, and then rounding to the nearest integer, i.e., the coefficients of the quantized blocks $\Theta^k_{u,v}$ are computed by

$$\Theta^k_{u,v} = \text{round}\left(\frac{\mathcal{B}^k_{u,v}}{Q^\mu_{u,v}}\right), \quad 0 \leq u, v \leq 7. \tag{5}$$

Next, it applies the zigzag scan to the matrix of the quantized coefficients $\left(\Theta^k_{u,v}\right)_{0\leq u,v\leq 7}$, see Figure 1, with the purpose of aligning frequency coefficients in ascending order, starting from frequency zero (DC coefficient) to high frequency components (AC coefficients), see [24, 31].

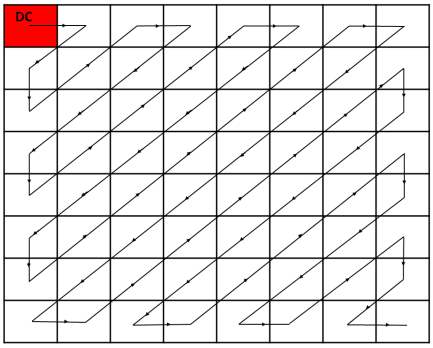

Figure 1. Zigzag order scan

On the other hand, the unquantization process for $\overline{\Theta}^k_{u,v}$ is carried out by

$$\overline{\mathcal{B}}^k_{u,v} = \text{round}\left(\overline{\Theta}^k_{u,v} Q^\mu_{u,v}\right), \quad 0 \leq u, v \leq 7. \tag{6}$$

2.3 Chaotic permutation. Fractional chaotic map

Recently, several authors [26, 27, 29] have proposed embedding schemes based on chaotic maps, which are nonlinear systems, characterized by a pseudo random behavior and a high sensitivity to initial conditions and control, unpredictability, ergodicity, etc [17]. In this research, is used the discrete fractional-order Logistic map proposed in [13], which is defined as follows

$$x_k = (ax_{k-1} - bx^2_{k-1})h^p - \sum_{1\leq j\leq k} C^{(p)}_j x_{k-j}, \quad k = 1, 2, \ldots,$$

where $p \in (0, 2)$, $a = (\gamma - 1)/\delta$, $b = \gamma/\delta$, in which, δ is a small constant, $\gamma > 0$ is the variable parameter and $h \cong \delta$. Moreover,

$$C^{(p)}_j = \begin{cases} 1 & \text{if} \quad j = 0, \\ \left(1 - \dfrac{1+p}{j}\right) C^{(p)}_{j-1} & \text{otherwise.} \end{cases}$$

Thus, the chaotic permutation of $\varrho = \{\varrho_1, \ldots, \varrho_n\}$ is determined by $(\varrho_j)_{j\in\mathscr{P}}$ where

$$\mathscr{P} = \left\{ \lfloor x_k 10^{14} \text{ mod. } n \rfloor, \quad 1 \leq k \leq n \right\}. \tag{7}$$

2.4 Notations

In this work, the following notations are taken into account:

- ✓ $\Delta(\eta)$ denote the function that reorganizes the vector η of length 64 to a matrix of order 8, taking into account the zigzag scan order 1.
- ✓ $\mathrm{R}(x, \beta)$ denote the function that replaces the Least Significant Bit (LSB) of $x \in \mathbb{N}$ by $\beta \in \{0, 1\}$, see [24].
- ✓ $\mathrm{R}^{-1}(x)$ denote the function that extracts LSB of $x \in \mathbb{N}$, see [24].
- ✓ $\oplus$ is the exclusive OR operation, defined by:

$$0 \oplus 0 = 1 \oplus 1 = 0 \quad \text{and} \quad 0 \oplus 1 = 1 \oplus 0 = 1.$$

2.5 Programming language and image dataset

The proposed algorithm is implemented in Python 3.7. For the experimental analysis several color images with size (512×512) were collected from two different datasets: image dataset of 1500 RGB-BMP images, transformed from Caltech birds' dataset in JPEGC format [2] and image dataset of 1500 RGB-BMP images, transformed from NRC dataset in TIFF format [2]. For the experimental results several different randomly generated keys were used. The proposed algorithm has been tested by embedding a message of 65536 and 8192 bits on the spatial and frequency domain, respectively. Moreover, the generated 256-bit binary watermark from the cover image is embedded into each 32×32 block.

3 Proposed method

In this section a mixed steganographic and fragile watermarking algorithm is presented, which uses a 128-bit private key and a 128-bit public key. In addition, it is assumed that the emitter as well as the receiver hold the same system of 128-bit private keys. The secret message is inserted into the cover image $\mathcal{C}$ and is extracted from the stego image $\mathcal{S}$ by the embedding and extracting algorithm, respectively.

3.1 Embedding algorithm

Input: Secret message $\mathbb{M} = \{m_\ell \in \{0,1\} \,:\, 1 \leq \ell \leq |\mathcal{M}|\}$, cover image $\mathcal{C}$, quality factor μ (for the frequency domain), 128-bit private key $\kappa^{(1)}$, 128-bit public key $\kappa^{(2)}$ and the initial condition x_0, which can also be adopted as a private key jointly with the control parameters p, γ, δ.

Output: Stego image $\mathcal{S}$.

Procedure: Algorithm 1 shows the step by step embedding process.

Firstly, vectors η and ϑ are created with the positions of the blocks where the fragile watermark and the secret message respectively will be embedded. Then, from the private key and the public key the sha256$^{(1)}$ of the result of the logical operation $\kappa^{(1)} \oplus \kappa^{(2)}$ is determined.

The next step gets the sha256$^{(2)}$ of the private key only. The sha256$^{(1)}$ is then divided into a logical sequence of length 64; i.e., sha256$^{(1)} \leftarrow$ sha256$_1^{(1)} \cup$ sha256$_2^{(1)} \cup$ sha256$_3^{(1)} \cup$ sha256$_4^{(1)}$. The Hash sha256$^{(2)}$ is also divided into a sequence of equal length as the previous one. In the next step, the cover image $\mathcal{C}$ is divided into Λ non-overlapping blocks B_k^{32} of 32×32 bytes, with $k \in \{1, \ldots, \Lambda\}$.

From the initial seed x_0 and the control parameters p, δ and γ the Λ chaotic positions $\mathscr{P}$ of $\{1, \ldots, \Lambda\}$ are determined according to (7). The following step of the algorithm divides each block B_τ^{32} into 16 non-overlapping blocks B_j^8 of 8×8 bytes, with $1 \leq j \leq 16$. Each block is selected taking into account $\tau \in \mathscr{P}$ chaotic position determined previously. Then, from step 15 to 21 the watermark of each block B^8 is generated in the order provided by the vectors η and ϑ.

Next, steps 24 to 27 describe the embedding process of the watermarking into the least significant bits of each $B_{\eta_i}^8$ blocks, being r the control variable of each portion of the embedded watermark. The process of embedding the secret bits in each element of the block $B_{\vartheta_i}^8$ is shown in steps 28 through 35. This process occurs in two ways: if the value of *method* parameter is set to LSB, the secret bits are embedded in the least significant bits of each element of the $B_{\vartheta_i}^8$ block, see steps 29 through 32; on the other hand, if the value of parameter is not set to LSB the secret bits are embedded in the least significant bits of the first eight AC coefficients as described in step 34, see Algorithm 3. In both cases, ℓ is the control variable of the current portion of the secret message bit sequence. Finally, in step 37 all the marked blocks are collected and then the stego image is reconstructed (see step 40).

3.2 Extracting algorithm

Input: Stego image $\mathcal{S}$, quality factor μ (for the frequency domain), 128-bit private key $\kappa^{(1)}$, 128-bit public key $\kappa^{(2)}$ and the initial condition x_0, which can also be adopted as a private key jointly with the control parameters p, γ, δ.

Output: Secrete message $\mathbb{M}$ in case of that integrity and authenticity of stego image is verified.

Procedure: The Algorithm 2 show the step by step extracting process.

Firstly, vectors η and ϑ are created with the positions of the blocks where the fragile watermark and the secret message respectively will be extracted. Then from the private key and the public key the sha256$^{(1)}$ of the result of the logical operation $\kappa^{(1)} \oplus \kappa^{(2)}$ is determined.

Algorithm 1 Embedding algorithm

Input: $\mathbb{M}$, $\mathcal{C}$, $\kappa^{(1)}$, $\kappa^{(2)}$, x_0, p, γ, δ, method=$\{LSB, DCT\}$
Output: $\mathcal{S}$
$\ell = 1$
$\eta \leftarrow \{1, 4, 13, 16\}$ **/* Blocks to embed the fragile watermarking */**
$\vartheta \leftarrow \{6, 7, 10, 11\}$ **/* Blocks to embed the secret bits */**
sha256$^{(1)}$ $\leftarrow$ hash function sha256 of $\kappa^{(1)} \oplus \kappa^{(2)}$
sha256$^{(2)}$ $\leftarrow$ hash function sha256 of $\kappa^{(2)}$
Divide the sha256$^{(1)}$ into a logical sequence of length 64; i.e., sha256$^{(1)}$ $\leftarrow$ sha256$_1^{(1)}$ $\cup$ sha256$_2^{(1)}$ $\cup$ sha256$_3^{(1)}$ $\cup$ sha256$_4^{(1)}$
Divide the sha256$^{(2)}$ into a logical sequence of length 64; i.e., sha256$^{(2)}$ $\leftarrow$ sha256$_1^{(2)}$ $\cup$ sha256$_2^{(2)}$ $\cup$ sha256$_3^{(2)}$ $\cup$ sha256$_4^{(2)}$
Divide $\mathcal{C}$ into Λ non-overlapping blocks B_k^{32} of 32×32 bytes, with $k \in \{1, \ldots, \Lambda\}$
From x_0, p, γ and δ, to determine the Λ chaotic positions $\mathscr{P}$ of $\{1, \ldots, \Lambda\}$ according to (7)
for each $\tau \in \mathscr{P}$ **do**
    Divide B_τ^{32} into 16 non-overlapping blocks B_j^8 of 8×8 bytes, with $1 \leq j \leq 16$
    $\overline{B}^8 \leftarrow B^8$ **/* Temporary copy of B^8 to $\overline{B}^8$ */**
    /* Create watermarking (from step 15 to 21) */
    for each $\eta_i \in \eta$ **do**
        $\overline{B}_{\eta_i}^8 \leftarrow$sha256$_i^{(1)}$
    end for
    for each $\vartheta_i \in \vartheta$ **do**
        $\overline{B}_{\vartheta_i}^8 \leftarrow$sha256$_i^{(2)}$
    end for
    sha256$^{(3)}$ $\leftarrow$ hash function sha256 of $\overline{B}^8$
    $r = 1$
    /* Watermarking embedding process (from step 24 to 27) */
    /* Secret bits embedding process (from step 28 to 35) */
    for each $i \in \{1, \ldots, 4\}$ **do**
        for each $x \in B_{\eta_i}^8$ **do**
            $x \leftarrow \mathrm{R}(x, \text{sha256}_r^{(3)})$
            $r \leftarrow r + 1$
        end for
        if method $==$ LSB **then**
            for each $x \in B_{\vartheta_i}^8$ **do**
                $x \leftarrow \mathrm{R}(x, m_\ell)$
                $\ell \leftarrow \ell + 1$
            end for
        else
            $B_{\vartheta_i}^8, \ell \leftarrow \mathcal{R}_{DCT}(B_{\vartheta_i}^8, \mathbb{M}, \ell)$
        end if
    end for
    $\overline{B}_\tau^{32} \leftarrow \cup_j B_j^8, 1 \leq j \leq 16$
end for
$\mathcal{S} \leftarrow \cup_\tau \overline{B}_\tau^{32}, 1 \leq \tau \leq \Lambda$
return $\mathcal{S}$

Algorithm 2 Extracting algorithm

Input: $\mathcal{S}$, $\kappa^{(1)}$, $\kappa^{(2)}$, x_0, ξ
Output: $\mathbb{M}$, $\mathcal{A}$
$\ell = 1$, $\varpi = \mathcal{A} = \emptyset$
$\eta \leftarrow \{1, 4, 13, 16\}$ **/* Blocks to extract the fragile watermarking */**
$\vartheta \leftarrow \{6, 7, 10, 11\}$ **/* Blocks to extract the secrete bits */**
sha256$^{(1)}$ $\leftarrow$ hash function sha256 of $\kappa^{(1)} \oplus \kappa^{(2)}$
sha256$^{(2)}$ $\leftarrow$ hash function sha256 of $\kappa^{(2)}$
Divide the sha256$^{(1)}$ into a logical sequence of length 64; i.e., sha256$^{(1)}$ $\leftarrow$ sha256$_1^{(1)}$ $\cup$ sha256$_2^{(1)}$ $\cup$ sha256$_3^{(1)}$ $\cup$ sha256$_4^{(1)}$
Divide the sha256$^{(2)}$ into a logical sequence of length 64; i.e., sha256$^{(2)}$ $\leftarrow$ sha256$_1^{(2)}$ $\cup$ sha256$_2^{(2)}$ $\cup$ sha256$_3^{(2)}$ $\cup$ sha256$_4^{(2)}$
Divide $\mathcal{S}$ into Λ non-overlapping blocks B_k^{32} of 32×32 bytes, with $k \in \{1, \ldots, \Lambda\}$
From x_0, p, γ and δ, to determine the Λ chaotic positions $\mathscr{P}$ of $\{1, \ldots, \Lambda\}$ according to (7)
for each $\tau \in \mathscr{P}$ **do**
    Divide B_τ^{32} into 16 non-overlapping blocks B_j^8 of 8×8 bytes, with $1 \leq j \leq 16$
    $\overline{B}^8 \leftarrow B^8$ **/* Temporary copy of B^8 to $\overline{B}^8$ */**
    /* Recreate the watermark (from step 15 to 21) */
    for each $\eta_i \in \eta$ **do**
        $\overline{B}_{\eta_i}^8 \leftarrow$sha256$_i^{(1)}$
    end for
    for each $\vartheta_i \in \vartheta$ **do**
        $\overline{B}_{\vartheta_i}^8 \leftarrow$sha256$_i^{(2)}$
    end for
    sha256$^{(3)}$ $\leftarrow$ hash function sha256 of $\overline{B}^8$
    /* Watermarking extraction process (from step 26 to 33) */
    /* Secret bits extraction process (from step 23 to 25) */
    for each $i \in \{1, \ldots, 4\}$ **do**
        for each $x \in B_{\eta_i}^8$ **do**
            $\varpi \leftarrow \varpi \cup \mathrm{R}^{-1}(x)$
        end for
        if method $==$ 'LSB' **then**
            for each $x \in B_{\vartheta_i}^8$ **do**
                $m_\ell \leftarrow \mathrm{R}^{-1}(x)$
                $\ell \leftarrow \ell + 1$
            end for
        else
            $m_\ell, \ell \leftarrow \mathcal{R}_{DCT}^{-1}(B_{\vartheta_i}^8, \ell)$
        end if
    end for
    /* Tamper detection */
    if *isTampered*(sha256$^{(3)}$, ϖ) **then**
        $\mathcal{A} \leftarrow \mathcal{A} \cup \tau$
    end if
end for
return $\mathcal{A}$, $\mathbb{M}$

The next step gets the sha256$^{(2)}$ of the private key only. The sha256$^{(1)}$ is then divided into a logical sequence of length 64; i.e., sha256$^{(1)} \leftarrow$ sha256$_1^{(1)} \cup$ sha256$_2^{(1)} \cup$ sha256$_3^{(1)} \cup$ sha256$_4^{(1)}$. The Hash sha256$^{(2)}$ is also divided into a sequence of equal length as the previous one.

In the next step, the stego image $\mathcal{S}$ is divided into Λ non-overlapping blocks B_k^{32} of 32×32 bytes, with $k \in \{1, \ldots, \Lambda\}$. From the initial seed x_0 and the control parameters p, δ and γ the Λ chaotic positions $\mathscr{P}$ of $\{1, \ldots, \Lambda\}$ are determined according to (7).

The following step of the algorithm divides each block B_τ^{32} into 16 non-overlapping blocks B_j^8 of 8×8 bytes, with $1 \leq j \leq 16$. Each block are selected taking into account $\tau \in \mathscr{P}$ chaotic position determined previously. Then, from step 15 to 21 the watermark of each block B^8 is recreated in the order provided by the vectors η and ϑ.

Next, the watermark is recovered from steps 23 to 25. The process of extracting the secret bits in each element of the block $B_{\vartheta_i}^8$ is shown in steps 26 through 33. This process can be in two ways: if the value of *method* parameter is set to LSB, the secret bits are extracted from the least significant bits of each element of the $B_{\vartheta_i}^8$ block, see steps 26 through 31. The other way is in the case that the parameter value is not set to LSB; in this case the secret bits are extracted from the least significant bits of the first eight AC coefficients as described in step 32, see Algorithm 4. In both cases, ℓ is the control variable of the current portion of the secret message bit sequence. After the previous step, it is determined if the image has been tampered. If B^8 was tampered, the position of the current block is collected in a list called $\mathcal{A}$. Finally, the extracted secret bits and the list with the positions of the tampered blocks are returned.

Algorithm 3 $\mathcal{R}_{DCT}(B^8, \mathbb{M}, \ell)$

1: **Input:** Block of 8×8 (B^8), the message $\mathbb{M}$, ℓ
2: **Output:** Modified block $\overline{B}^8$, ℓ
/* Transforming the block */
3: $\mathcal{B}^8 \leftarrow B^8$: DCT(B^8) according to (2)
4: $\Theta \leftarrow \mathcal{B}^8$: Quantify $\mathcal{B}^8$ according to (5)
/* Determine a sequence ν of length 64 */
5: $(\nu_i) \leftarrow \Theta$: Apply the zigzag scan, see Figure 1
/* Embedding the secret bits in the block */
6: **for** $i \in \{2, 9\}$ **do**
7: $\quad \nu_i \leftarrow \mathrm{R}(\nu_i, m_\ell)$
8: $\quad \ell \leftarrow \ell + 1$
9: **end for**
/* Rebuilding the block */
10: $\overline{\Theta} \leftarrow \Delta(\nu_i)$
11: $\overline{\mathcal{B}}^8 \leftarrow \overline{\Theta}$: Apply the operation (6)
12: $\overline{B}^8 \leftarrow \overline{\mathcal{B}}^8$: IDCT($\overline{\mathcal{B}}^8$) according to (3)
13: **return** $\overline{B}^8$, ℓ

Algorithm 4 $\mathcal{R}_{DCT}^{-1}(B^8, \ell)$

Input: Block of 8×8 (B^8), the message $\mathbb{M}$, ℓ
Output: $\mathbb{M}$, ℓ
/* Transforming the block */
$\mathcal{B}^8 \leftarrow B^8$: DCT(B^8) according to (2)
$\Theta \leftarrow \mathcal{B}^8$: Quantify $\mathcal{B}^8$ according to (5)
/* Determine a sequence ν of length 64 */
$(\nu_i) \leftarrow \Theta$: Apply the zigzag scan, see Figure 1
/* Extracting the secret bits in the block */
for $i \in \{2, 9\}$ **do**
    $m_\ell \leftarrow \mathrm{R}^{-1}(\nu_i)$
    $\ell \leftarrow \ell + 1$
end for
return m_ℓ, ℓ

4 Results and discussion

In this section the experimental results of the proposed algorithm are presented. The performance of the proposed approach has been studied using different types of statistical measures.

4.1 Imperceptibility test

As the cover image is altered to embed the secret data, there will be changes in the cover image pixel values. Thus, the changes need to be analyzed since this directly affects the imperceptibility of the output stego image. Peak Signal to Noise Ratio (PSNR), is one of the popular and top notch metric used to measure the quality of the stego image by analyzing the mean squared error value between the cover and the stego image [14]. Indeed, this measure is used to evaluate the invisibility of a secret message [3] as well as the imperceptibility [4] and the visual quality [21, 25] of the stego image compared to the cover image, with decibel (db) as measurement unit. Higher PSNR indicates that the reconstruction of the image is of higher quality, [9]. The PSNR is given by [24]

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{\Xi^2}{\mathrm{MSE}} \right),$$

where

$$\mathrm{MSE} = (mn\rho)^{-1} \sum_{\gamma \in \Gamma} \left\| \mathcal{C}(\gamma) - \mathcal{S}(\gamma) \right\|^2,$$

and $\mathcal{C}$ and $\mathcal{S}$ are the cover image and the stego image respectively, of size $m \times n \times \rho$, with $\mathcal{C}, \mathcal{S} \in \{0, 1, \ldots, \Xi\}$, and $\Xi = \max(\max(\mathcal{C}), \max(\mathcal{S}))$.

The index set $\gamma = (\ell_1, \ell_2, \ell_3)$ sums over the set

$$\Gamma = \{1, \ldots, m\} \times \{1, \ldots, n\} \times \{1, \ldots, \rho\},$$

where $\rho = 1$ for gray scale images and $\rho = 3$ for 24-bit color images.

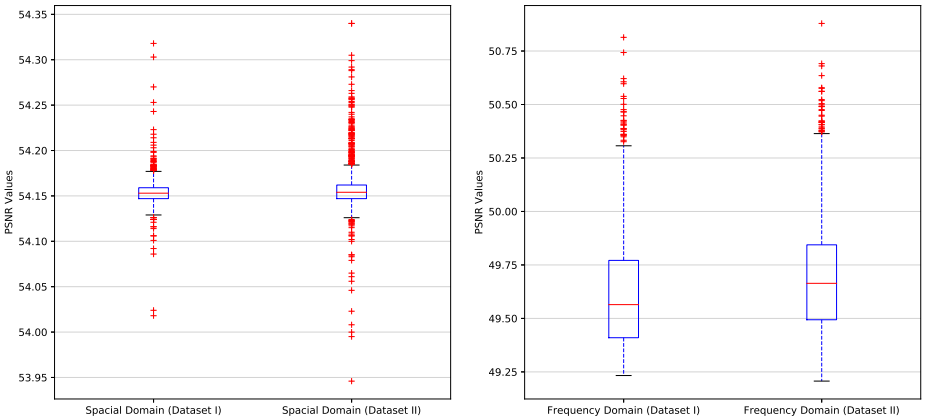

Figure 2. PSNR values.

In the boxplots drawn in Figure 2, the horizontal axis represents the different methods that are compared, and the vertical axis represents the PSNR values. The upper and lower limit of the rectangle are the upper and lower quartiles (Q_1 and Q_3) of test results separately, and the difference between the upper and lower quartile is the quartile difference IQR. The red line in the rectangle is the median. The two black horizontal lines at $Q_3 + 1.5$IQR and $Q_1 - 1.5$IQR are the cut-off points for abnormal values, known as the internal limit. The data outside the internal limits is outliers and is represented by the red ?+?. From Figure 2 it is inferred that for the two datasets the median of PSNR values corresponding to proposed method is greater in value than 54.15 and 49.50 db for both domains. In this experiment, the experimental results showed that the proposed algorithm produced good quality stego images with good PSNR values, which is in correspondence with the heuristic values of PSNR [20, 24]

4.2 Quality test

Usually the image quality based on the Human Visual System (HVS) is measured by the Universal Image Quality Index (UIQI), which was proposed by Wang and Bovik in [28]. This measure is universal in the sense that it does not take the viewing conditions or the individual observer into account [5]. Moreover, it does not use traditional error summation methods [30]. The dynamic range of UIQI is between -1 and 1. For identical images its value will be 1.

$$\text{UIQI} = \frac{4\sigma_{\mathcal{CS}}}{\sigma_{\mathcal{C}}^2 + \sigma_{\mathcal{S}}^2} \frac{\overline{\mathcal{C}}\,\overline{\mathcal{S}}}{\overline{\mathcal{C}}^2 + \overline{\mathcal{S}}^2},$$

where

$$\overline{\mathcal{C}} = (mn\rho)^{-1} \sum_{\gamma \in \Gamma} \mathcal{C}(\gamma),$$

$$\overline{\mathcal{S}} = (mn\rho)^{-1} \sum_{\gamma \in \Gamma} \mathcal{S}(\gamma),$$

$$\sigma_{\mathcal{C}}^2 = (mn\rho - 1)^{-1} \sum_{\gamma \in \Gamma} \left(\mathcal{C}(\gamma) - \overline{\mathcal{C}}\right)^2,$$

$$\sigma_{\mathcal{S}}^2 = (mn\rho - 1)^{-1} \sum_{\gamma \in \Gamma} \left(\mathcal{S}(\gamma) - \overline{\mathcal{S}}\right)^2,$$

$$\sigma_{\mathcal{CS}} = (mn\rho - 1)^{-1} \sum_{\gamma \in \Gamma} \left[\left(\mathcal{C}(\gamma) - \overline{\mathcal{C}}\right)\left(\mathcal{S}(\gamma) - \overline{\mathcal{S}}\right)\right]$$

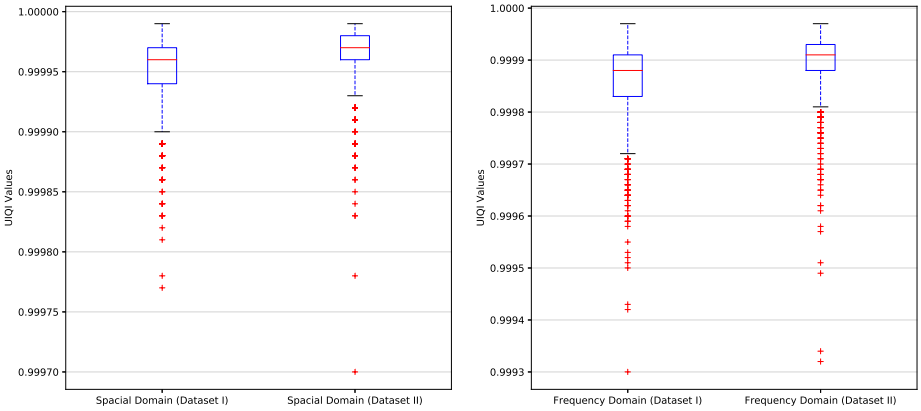

Figure 3. UIQI values

The second experiment shows that there are no significant differences between the cover and stego images, since for the two datasets the median of UIQI values is close to the unit, see Figure 3.

4.3 Similarity test

Image fidelity is a measure that shows a consistent relationship with the quality perceived by the human visual perception. Moreover, it is a metric that measures the similarity between the cover image $\mathcal{C}$ and the stego image $\mathcal{S}$ after insertion of the message without any visible distortion or information loss [24]. It is defined by [15, 22, 24]

$$\text{IF} = 1 - \sum_{\gamma \in \Gamma} \left(\mathcal{C}(\gamma) - \mathcal{S}(\gamma)\right)^2 \; / \; \sum_{\gamma \in \Gamma} \mathcal{C}(\gamma)^2.$$

Figure 4 shows that the boxplots corresponding to the proposed method are comparatively smaller and less dispersed than rest of the cases, which suggests that the most values of IF have a high level of agreement with each other. Thus, we conclude that the embedding process was visually lossless.

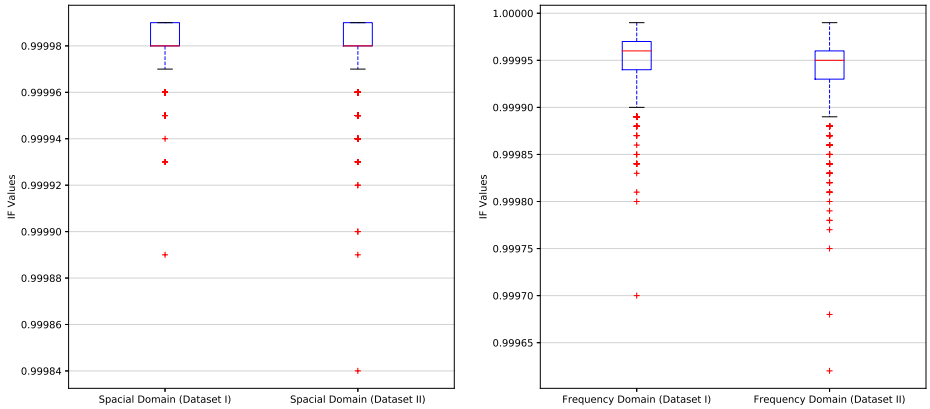

Figure 4. IF values

4.4 Security test

The security of a steganographic system is defined in terms of the relative entropy

$$\mathrm{RE}\left(P_{\mathcal{C}}||P_{\mathcal{S}}\right) = \sum P_{\mathcal{C}} \left|\log \frac{P_{\mathcal{C}}}{P_{\mathcal{S}}}\right|,$$

where $P_{\mathcal{C}}$ and $P_{\mathcal{S}}$ represent the distribution of cover and stego image, respectively. This statistical measure was proposed by Cachin in [6]. Moreover, a steganographic system is said to be

- ε-secure if $\mathrm{RE}\left(P_{\mathcal{C}}||P_{\mathcal{S}}\right) \leq \varepsilon$,
- perfectly secure if $\mathrm{RE}\left(P_{\mathcal{C}}||P_{\mathcal{S}}\right) = 0$.

Summing up, for the $\mathrm{RE}\left(P_{\mathcal{C}}||P_{\mathcal{S}}\right)$, the closer the value is to 0, the higher the level of security.

In this experiment we observe that the values of the relative entropy are close to zero, which affirms that the steganographic system obtained from the proposed algorithm is sufficiently secure, see Figure 5.

4.5 Image tamper detection

Tampering is an intentional modification of documents in a way that would make them harmful for end users. So it is essential to reveal the watermark as well as cover image during extraction process [18]. The performance of the proposed tamper detection method is conducted by applying different image processing operations such as Blurring noise, Cropping noise, Salt and Pepper noise and Gaussian noise to the stego images in Figure 6. Random sized attacks are applied to the different regions of the watermarked images, see Figures 7-9-11-13. By using the proposed tamper detection method, tampered regions are marked, see Figures 8-10-12-14.

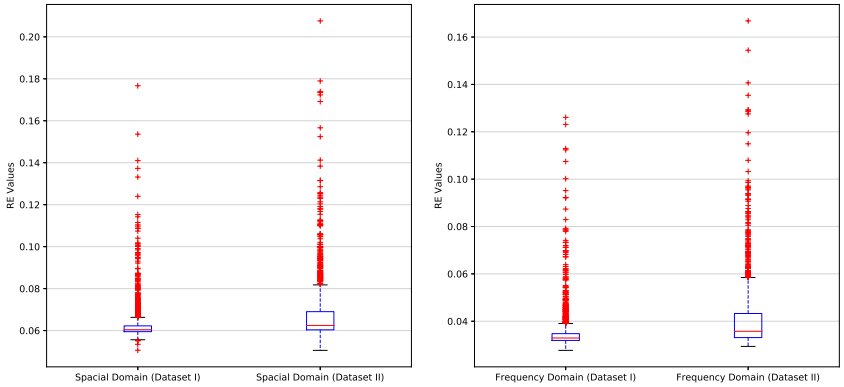

Figure 5. RE values

Figure 6. Stego images analyzed in the tamper detection

Figure 7. Implementation of random sized attacks to the different regions of the stego images corresponding to the Blurring noise

Figure 8. Detection of the tampered regions of the stego images corresponding to the Blurring noise

Figure 9. Implementation of random sized attacks to the different regions of the stego images corresponding to the Cropping noise

Figure 10. Detection of the tampered regions of the stego images corresponding to the Cropping noise

Figure 11. Implementation of random sized attacks to the different regions of the stego images corresponding to the Gaussian noise

Figure 12. Detection of the tampered regions of the stego images corresponding to the Gaussian noise

Figure 13. Implementation of random sized attacks to the different regions of the stego images corresponding to the Salt and Pepper noise

Figure 14. Detection of the tampered regions of the stego images corresponding to the Salt and Pepper noise

Conclusions

In this research, an efficient mixed steganographic and fragile watermarking method has been proposed for the authentication of secret message in a digital image and the tamper detection. The performance of the method has been tested using four types of attacks to different regions of the images with watermark. In the tamper detection phase, the tampered regions of the stego image were detected by comparing the value of the calculated hash function with the extracted watermark. Experimental results illustrate that the proposed method produced stego images with a high visual quality and good PSNR values, which is in correspondence with the heuristic values of PSNR. In addition, it detects the tampered regions of the stego image, which ensures the integrity and authenticity of the secret message. Finally, generated SHA-256 hash values and discrete fractional-order logistic map have been altogether used to improve the reliability of our method.

Acknowledgments

The authors thank the reviewers for their valuable comments, which helped improve this article.

References

[1] J. Abraham & V. Paul. *An imperceptible spatial domain color image watermarking scheme*. J. King Saud Univ., Comp. & Info. Sci. **31**(1) (2019), 125-133.

[2] M. Al-Jarrah. *RGB-BMP Steganalysis Dataset*, Mendeley Data (2018), v1. 10.17632/sp4g8h7v8k.1

[3] R. Atta & M. Ghanbari. *A high payload steganography mechanism based on wavelet packet transformation and neutrosophic set*, J. Vis. Commun. Image R. (2018). 10.1016/j.jvcir.2018.03.009

[4] A. Awad, M. Mursi & A. Alsammak. *Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3*, Ain Shams Engineering Journal (2017). 10.1016/j.asej.2017.02.003

[5] B. Bayraktar, T. Bernas, J. Robinson & B. Rajwa. *A numerical recipe for accurate image reconstruction from discrete orthogonal moments*, Pattern Recognition **40** (2007), 659-669.

[6] C. Cachin. *An information-theoretic model for steganography*, Lecture Notes in Computer Science **1525** (1998), 306-318.

[7] P. Chowdhuri, B. Jana & D. Giri. *Secured steganographic scheme for highly compressed color image using weighted matrix through DCT*, J. Inf. Secur. Appl. (2018) 1-12.

[8] I. Cox, M. Miller, J. Bloom, J. Fridrich & T. Kalker. Digital watermarking and steganography. *The Morgan Kaufmann series in multimedia information and systems*, Elsevier/Morgan Kaufmann, Amsterdam, 2ed edition, (2008).

[9] B. Datta, U. Mukherjee & S. Kumar. *Lsb layer independent robust steganography using binary addition*, Procedia Computer Science **85** (2016), 425-432.

[10] E. Gul & S. Ozturk. *A novel hash function based fragile watermarking method for image integrity*, Multimedia Tools and Applications (2019).

[11] N. N. Hurrah, S. A. Parah, N. A. Loan, J. A. Sheikh, M. Elhoseny & K. Muhammad. *Dual watermarking framework for privacy protection and content authentication of multimedia*, Future Generation Computer Systems **94** (2019), 654-673.

[12] M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. Ho & K.-H. Jung. *Image steganography in spatial domain: A survey*, Signal Processing: Image Communication, **65**, (2018) 46-66.

[13] Y. Ji, L. Lai, S. Zhong & L. Zhang. *Bifurcation and chaos of a new discrete fractional-order logistic map*, Comm. Nonlinear Sci. Numer. Simulat. **57** (2018), 352-358.

[14] I. J. Kadhim, P. Premaratne, P. J. Vial & B. Halloran. *Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research*, Neurocomputing (2018).

[15] A. Khamruia & J. K. Mandal. *A genetic algorithm based steganography using discrete cosine transformation (GASDCT)*, Procedia Technology **10** (2013), 105-111.

[16] S. Lian & Y. Zhang. *Handbook of Research on Secure Multimedia Distribution*, Information Science Reference-Imprint of: IGI Publishing, Hershey, PA, 1st edition, (2009).

[17] R. F. Martínez-González, J. A. Díaz-Méndez, L. Palacios-Luengas, J. López-Hernández & R. Vázquez-Medina. *A steganographic method using bernoulli's chaotic maps*, Computers & Electrical Engineering **54** (2016), 435-449.

[18] P. Pal, B. Jana & J. Bhaumik. *Watermarking scheme using local binary pattern for image authentication and tamper detection through dual image*, Security and Privacy **2**(2) (2019), 1-16.

[19] Y. Peng, X. Niu, L. Fu & Z. Yin. *Image authentication scheme based on reversible fragile watermarking with two images*, J. Inf. Secur. Appl. **40**, (2018), 236-246.

[20] R. Roy, A. Sarkar & S. Changder. *Chaos based edge adaptive image steganography*, Procedia Technology **10** (2013), 138-146.

[21] A. K. Sahu & G. Swain. *A novel n-rightmost bit replacement image steganography technique*, 3D Research **10**(1) (2019), 2.

[22] M. Sengupta, P. Mandal, T. Das & A. Dey. *A novel hash based technique for thermal image authentication*, Procedia Technology **10** (2013), 147-156.

[23] P. Singh, S. Singh & S. Rani. *Efficient steganography algorithm based on DCT and entropy thresholding technique*, Int. J. Adv. Res. Comput. Sci. Softw. Eng. **8**(1) (2018), 45-50.

[24] A. Soria-Lorente & S. Berres. *A secure steganographic algorithm based on frequency domain for the transmission of hidden information*, Security and Communication Networks 2017, 1-14, (2017). https://doi.org/10.1155/2017/5397082

[25] M. Y. Valandar, P. Ayubi & M. J. Barani. *A new transform domain steganography based on modified logistic chaotic map for color images*, J. Inf. Secur. Appl. **34** (2017), 142-151.

[26] M. Y. Valandar, M. J. Barani, P. Ayubi & M. Aghazadeh. *An integer wavelet transform image steganography method based on 3d sine chaotic map*, Multimedia Tools and Applications (2018), 1-19.

[27] G. S. Walia, S. Makhija, K. Singh & K. Sharma. *Robust stego-key directed lsb substitution scheme based upon cuckoo search and chaotic map*, Optik **170** (2018), 106-124.

[28] Z. Wang and A. Bovik. *A universal image quality index*, IEEE Signal Processing Letters **9**(3) (2002), 81-84.

[29] G. S. Yadav & A. Ojha. *Chaotic system-based secure data hiding scheme with high embedding capacity*, Computers & Electrical Engineering **69** July (2018), 447-460.

[30] Y. Zheng & Q. Zheng. *Objective image fusion quality evaluation using structural similarity.*, Tsinghua Science and Technology **14**(9) (2009), 703-709.

[31] H. Zhu. *Image representation using separable two-dimensional continuous and discrete orthogonal moments*, Pattern Recognition **45** (2012), 1540-1558.

Received in July 2020. Accepted for publication in November 2020.

ERODIS PÉREZ MICHEL
TECHNOLOGY DEPARTMENT
UNIVERSITY OF GRANMA
BAYAMO, CUBA
e-mail: eperezm@udg.co.cu

YUNIEL GUZMÁN BAZÁN
TECHNOLOGY DEPARTMENT
UNIVERSITY OF GRANMA
BAYAMO, CUBA
e-mail: yguzmanb@udg.co.cu

YISEL DE LOS ANGELES GONZÁLEZ POMPA
TECHNOLOGY DEPARTMENT
UNIVERSITY OF GRANMA
BAYAMO, CUBA
e-mail: ypompa@udg.co.cu