Hybrid Watermarking Scheme for Copyright Protection using Chaotic Maps Cryptography

M.A. Mohamed
Faculty of Engineering-Mansoura University
Mansoura-Egypt

H.M. Abdel-Atty A.M Aboutaleb
Faculty of Engineering-PortSaed University
PortSaed-Egypt

M.G. Abdel-Fattah, A.S. Samrah
Faculty of Engineering-Mansoura University
Mansoura-Egypt

ABSTRACT

Nowadays, there is an explosive growth in the digital multimedia creation, capturing, processing and distribution. Protecting the multimedia contents from copyright infringement has become a major concern. Encryption and watermarking are two complementary techniques that are used for protecting the multimedia data. In this paper, a proposed hybrid encryption-watermarking algorithm for copyright protection is proposed. The watermarking phase of this proposed algorithm is based on combining the discrete wavelet transform (DWT), the discrete cosine transform (DCT), and the singular value decomposition (SVD), while the encryption phase is based on using four chaotic maps with different dimensions. The proposed watermarking scheme uses a new PN-codes embedding strategy of the watermark into the cover image. This strategy allows decreasing the embedding strength factor of the scheme to a value that maximizes imperceptibility performance while maintaining acceptable robustness of the watermarking scheme. The performance of the proposed watermarking scheme is evaluated individually based on the robustness and the imperceptibility measures. This scheme is compared with some recent existing algorithms and experimental results show the improvements of the proposed algorithm over these algorithms. On the other hand, the proposed chaos-based encryption algorithm used four chaotic maps of different dimensions and it has two diffusion stages rather than one to improve the algorithm efficiency. The proposed encryption algorithm is tested using different experiments. The experimental results demonstrate that the proposed encryption algorithm shows advantages of large key space, high resistance against differential attacks and high security analysis such as statistical analysis, and sensitivity analysis. Compared to some traditional and recent encryption algorithms, the proposed encryption algorithm is much more secure. Finally experimental tests demonstrate that the proposed hybrid encryption-watermarking algorithm introduces high degree of efficiency, robustness, and security.

General Terms

Hybrid Image watermarking schemes, Chaotic maps based cryptography, Statistical attacks, and Differential attacks.

Keywords

Chaotic maps, Encryption, Digital Watermarking, Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Singular Value Decomposition (SVD).

1. INTRODUCTION

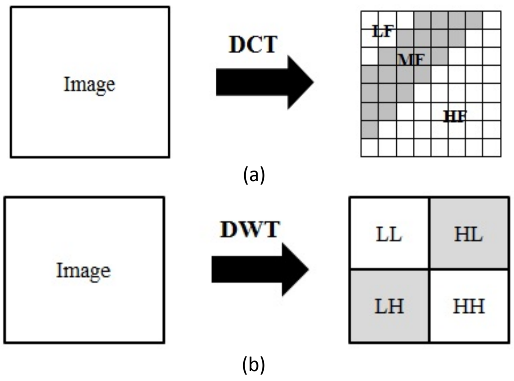
Fast development of Internet technology in recent years has improved the ways to distribute and exchange digital multimedia with less time, lower complexities and better efficiency than ever. Digital multimedia can be manipulated or reproduced easily without the loss of information by using powerful multimedia processing tools that are widely available. In particular, safe distribution and management of multimedia content have become the real challenge. To satisfy these needs, several techniques have been developed, among them digital watermarking and encryption. Digital watermarking and Encryption are two complementary techniques that are used for protecting the multimedia content. Encryption provides a means for secure delivery of multimedia content to the consumer. Legitimate consumers are provided with a key to decrypt the content in order to view or listen to it. After the decryption phase, an untrustworthy consumer may alter or copy the decrypted content in a manner that is not permitted by the content owner; hence encrypted content need an additional security level in order to keep control on them. On the other hand, one may want to check the presence of a watermark without deciphering the data. Therefore, the challenging goal seems to be the achievement of both levels of protection simultaneously in order to allow jointly exploiting the benefits of the two mechanisms [1].

Digital watermarking schemes can be classified based on the watermarking domain into two categories: spatial domain and transform domain watermarking schemes. In spatial domain watermarking schemes, the watermark is embedded by directly modifying the pixel values of the image [2]-[4]. In transform domain watermarking schemes, the transformation technique, such as the discrete cosine transform (DCT) [5]-[6], discrete wavelet transform (DWT) [7]-[9], and singular value decomposition (SVD) [10]-[11] is applied to an image and then the watermark is embedded by modifying the transform domain coefficients. DCT is a technique for converting a signal into elementary frequency components [12]. It decomposes an image into (i) low frequency (LF), (ii) medium frequency (MF), and (iii) high frequency (HF) sub-bands as shown in Fig.1(a). It has a strong "energy compaction" property that most of the image energy tends to be concentrated in the low frequency sub-band of the transformed image. DWT provides multi resolution representation of an image and can be efficiently implemented using digital filters [13]. An image can be decomposed using 1-level DWT into four sub-bands: (i) low frequency sub-band (LL), (ii) horizontal sub-band (HL), (iii) vertical sub-band (LH), and (iv) diagonal sub-band (HH) as shown in Fig.1(b). The high frequency sub-bands HL, LH, and HH are good regions for embedding the watermark because human naked eyes are less sensitive to modification in these sub-bands than the LL sub-band [26, 27]. SVD for any image ‘A’ of size N×N is a factorization of the form given by: $A = U \times S \times V^T$, where U and V are orthogonal matrices and S is a diagonal matrix of singular values in decreasing order. The main properties of SVD are (i) A small agitation added in the image does not cause large variation in its singular values (SVs), and (ii) the singular values represent algebraic image properties which are intrinsic and not visible [14]. These transformation domain techniques show good robustness and security against various attacks as compared to spatial domain techniques. In the last few years, researchers start to combine between these techniques such as combining between SVD-DCT [15]-[16], DWT-SVD [17]-[18], DWT-DCT [19]-[20], and DWT-DCT-SVD [21] in order to improve the performance of watermarking process.

Traditional encryption algorithms such as Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), and Advanced Encryption Standard (AES) etc. are not suitable for real time image encryption as these ciphers require a large computational time and high computing power. The chaos-based encryption has suggested new and efficient ways to deal with the intractable problem of fast and highly secure image encryption. It provides a good combination of speed, high security, complexity, reasonable computational overheads and computational power. After Matthews proposed the chaotic encryption algorithm in 1989 [22], increasing researches of image encryption technology are based on chaotic systems [23]-[25]. Chaotic systems have many important properties, such as the sensitive dependence on initial conditions and system parameters, pseudorandom property, non-periodicity and topological transitivity, etc. Most properties meet some requirements such as diffusion and mixing in the sense of cryptography [23]. Therefore, chaotic cryptosystems have more useful and practical applications. Chaos-based image encryption schemes are usually composed of two processes generally: chaotic confusion of pixel positions by permutation process and diffusion of pixel grey values by diffusion process. A typical architecture of existing chaos-based image cryptosystems is shown in Fig.2.

Encryption and watermarking can be combined in many different ways. Bas et al. [26] give an overview on the possible scenarios where the combination of both level of protection can be exploited, while Merhav [27] presents a theoretical analysis of this problem. The watermarking is suitable for copyright protection purposes where the invisible embedded watermarks carries some secret information that may be considered attributes of the cover host image such as copyright. To enhance the security of the user specific secret information in the networked multimedia system the watermark can be first encrypted using a secret key. The encrypted watermark is then embedded in the host image and transmitted to the intended user.

In this paper, a hybrid DWT-DCT-SVD watermarking scheme combined with a chaos-based encryption algorithm is proposed for copyright protection. The multi resolution representation that DWT provided, the strong energy compaction property of DCT and the stability of SVs of an image are combined in the proposed watermarking scheme to improve the scheme imperceptibility and robustness. The proposed watermarking scheme uses a new PN-codes embedding strategy of the watermark bits that highly improved the performance of the watermarking scheme. Each of PN-Codes was embedded into the singular values (SVs) of a sixteen elements that were chosen from the mid-frequency sub-band elements of a DCT block which also was selected from either the High-Low (HL) or Low-High (LH) sub-bands of DWT domain of the cover image depending on a shared secret Pseudorandom Noise (PN) code to increase the security level of the watermarking scheme. This code select between embedding the watermark bit into HL or LH block. The proposed encryption algorithm is based on combining four different dimensions chaotic maps (1D, 2D, 3D logistic map and 2D Henon map) to improve the algorithm security. An additional diffusion stage is used before the confusion stage to maximize the encryption speed, through minimizing the number of algorithm iterations. This diffusion stage contains a new shuffling function that shuffles the image bit panes depending on a round key which is sequentially updated for every pixel. This diffusion stage improves also the algorithm resistance against differential attacks. The remainder of this paper is organized as follows. Details and simulation results of the proposed watermarking scheme and are described in Section 2. In Section 3 Details and simulation results of the proposed encryption algorithm are introduced. In Section 4, the experimental results are presented to demonstrate the effectiveness of the proposed hybrid encryption-watermarking scheme. Conclusions are finally introduced in Section 5.



(a)

(b)

Fig.1 (a) The DCT of an image, (b) The DWT of an image

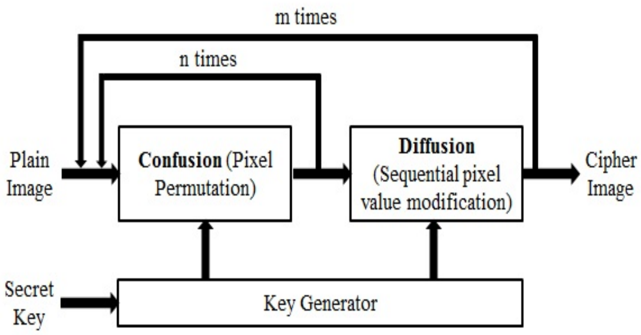

m times
n times
Plain Image
Confusion (Pixel Permutation)
Diffusion (Sequential pixel value modification)
Cipher Image
Secret Key
Key Generator

Fig.2 Typical architecture of chaos-based image cryptosystems

2. PROPOSED DWT-DCT-SVD COMBINED WATERMARKING SCHEME

In the last few years a number of watermarking algorithms were proposed based on combining between DWT, DCT, and SVD in various alternations. The proposed algorithm uses the same basic idea of combining between these watermarking domains but with different watermark embedding and extracting strategies. The watermark embedding is done using PN-codes, but unlike traditional correlation-based watermarking schemes, the proposed scheme uses a new PN-codes embedding strategy. This strategy allows decreasing the embedding strength factor of the scheme to a value that maximizes the degree of imperceptibility and robustness against different types of attacks. An additional PN-Code was used as a shared secret key to increase the security level of the watermarking scheme. This code select between embedding the watermark bit into HL or LH block of the wavelet domain. Block diagrams of the embedding and the extracting processes are shown in Fig.3 and Fig.4, respectively.

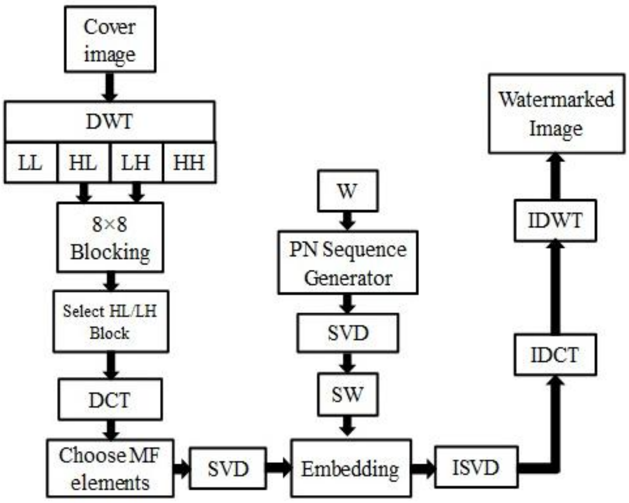


Fig.3 Block diagram of the embedding process of the proposed DWT-DCT-SVD watermarking scheme

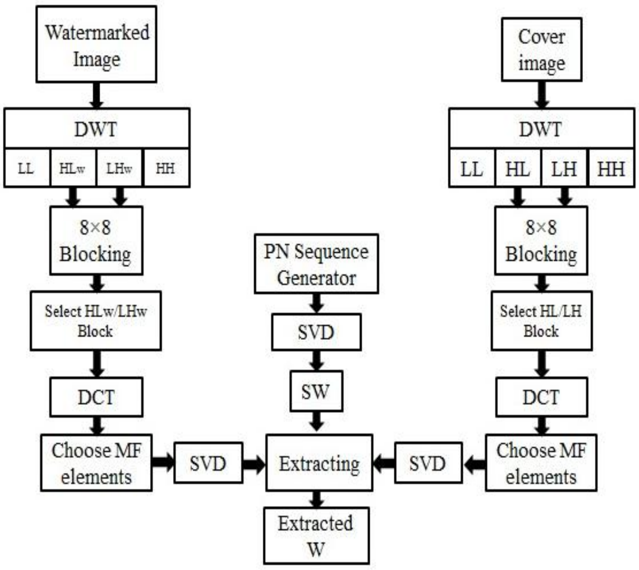


Fig.4 Block diagram of the extracting process of the proposed DWT-DCT-SVD watermarking scheme

2.1 Watermark Embedding Process

Step 1: The cover image 'I' is decomposed using one-level DWT into four sub bands (LL, HL, LH and HH).

Step 2: The watermark image is reformulated to be a binary vector 'W'.

Step 3: HL and LH sub-bands were chosen for embedding the watermark 'W'. The two sub-bands were divided into 8×8 blocks. In order to increase the watermarking security level a secret PN-code 'R' of length "n" equal to the W length was generated to select between embedding in the HL or LH sub-band block depending on 'R'. The selected block is called 'A'.

$$R = \begin{cases} 1; \text{Choose HL Block} \\ 0; \text{Choose LH Block} \end{cases} \quad (1)$$

Step 4: DCT was applied to the selected block 'A' and sixteen elements were selected from the MF elements. These elements were selected after the first five AC elements (coefficients) in a zigzag manner.

Step 5: The 16 elements were reshaped into a 4×4, called 'B', and the SVD of this matrix was computed.

$$[\mathrm{UB}, \mathrm{SB}, \mathrm{VB}] = \mathrm{SVD}(\mathrm{B}) \quad (2)$$

Step 6: A definite PN code 'PN_1' was generated to embed all watermark bits equal to "1", and every time a watermark bit equal to "0" needed to be embedded, a random PN code was generated which is restricted to have a possible minimum correlation with 'PN_1'. All of these PN codes, 'PN_1' and 'PN_0s', are of size 4×4.

Step 7: SVD was applied to the current PN code, and its singular values, 'S0' or 'S1', was embedded into the singular values 'SB' of the block 'B':

$$\mathrm{SN} = \begin{cases} \mathrm{SB} + \mathrm{K} \times \mathrm{S1}; \text{if } \mathrm{W} = 1 \\ \mathrm{SB} + \mathrm{K} \times \mathrm{S0}; \text{if } \mathrm{W} = 0 \end{cases} \quad (3)$$

where 'K' is the embedding strength factor.

Step 8: Inverse SVD (ISVD) was applied to the 'SN' block:

$$\mathrm{B_W} = \mathrm{UB} \times \mathrm{SN} \times \mathrm{VB}' \quad (4)$$

Step 9: Inverse DCT was applied to the modified block after replacing the sixteen elements in step-5 with 'BW', and then the selected block in step-4 was replaced with the watermarked block 'AW' to generate the watermarked HLW and LHW sub-bands.

Step 10: Inverse DWT (IDWT) was applied to the LL, HLW, LHW and HH sub-bands to obtain the watermarked image 'IW'.

2.2 Watermark Extracting Process

Step 1: The watermarked image 'IW' was decomposed using one-level DWT into four sub bands (LL, HLW, LHW and HH).

Step 2: The watermarked HLW and LHW sub-bands were divided into 8×8 blocks, and we use the shared secret PN code 'R' to select the watermarked sub-band block as in equation 1.

Step 3: DCT was applied to the selected block 'AW' and then the sixteen watermarked elements were selected from the MF elements.

Step 4: These sixteen elements were reshaped into a 4×4 matrix 'BW', and then SVD of this matrix was computed.

$$[\mathrm{UB}, \mathrm{SN}, \mathrm{VB}] = \mathrm{SVD}(\mathrm{B_W}) \quad (5)$$

Step 5: The first five steps in the embedding process were performed on the original cover image 'I' to find the original 'SB' and then the singular values difference 'DB' was computed:

$$\mathrm{DB} = \mathrm{SN} - \mathrm{SB} \qquad (6)$$

Step 6: The definite PN code ‘PN_1’ and the PN codes ‘PN_0s’ were regenerated by the same seed used in the embedding process.

Step 7: SVD was applied to the PN sequences, and the correlation between the singular values difference DB and the singular values of ‘PN_1’ and the current ‘PN_0’ was computed. If the correlation of ‘DB’ with the ‘S1’ was higher than the correlation with the current ‘S0’, then the extracted watermark bit is considered “1”, otherwise the extracted watermark is considered “0”.

Step 8: Reconstruct the watermark image using the extracted watermark bits, and compute the similarity between the original and extracted watermark.

The watermark embedding and extracting processes are described above for Greyscale images. This watermarking scheme can be extended for color images. The color image is represented by Red (R), Green (G) and Blue (B) channels. Out of these three channels, change in the intensity of R channel is the most sensitive to human eyes whereas for B channel it is least sensitive [28]. Hence, for our proposed scheme the blue channel is considered for watermark embedding and extracting using the same steps that described above.

2.3 Experimental Results of the Proposed Watermarking Scheme

In our experiments, twelve digital images are used as shown in Fig.5. The first six images are Greyscale images of different sizes in BMP image format, uncompressed, and of 8 bits per pixel depth. The other six images are color images of different sizes in Tiff format, uncompressed, and each color component of 8 bits per pixel depth. Performance measurements are evaluated to assess the efficiency and the effectiveness of the proposed DWT-DCT-SVD based watermarking scheme.

2.4 Imperceptibility Performance

Imperceptibility means that the perceived quality of the image should not be distorted by the presence of the watermark. The peak signal to noise ratio (PSNR) is typically used to measure the degradation between original image and watermarked image. Assume I is the cover image, and $\mathrm{I_w}$is the watermarked image, and each of dimensions $\mathrm{M} \times \mathrm{N} \times \mathrm{f}$, where M is the number of rows, N is the number of columns, and f is the number of image frames, i.e. f =3 for RGB image and f = 1 for Greyscale image.

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\mathrm{Max_I^2}}{\mathrm{MSE}}\right)\mathrm{dB} \qquad (7)$$

where $\mathrm{Max_I}$is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per pixel, this is 255. MSE is the mean squared error between the original and the watermarked image, given by:

$$\mathrm{MSE} = \frac{1}{\mathrm{M{\times}N{\times}f}}\sum_{\mathrm{k=1}}^{\mathrm{f}}\sum_{\mathrm{i=1}}^{\mathrm{M}}\sum_{\mathrm{j=1}}^{\mathrm{N}}[\mathrm{I(i,j,k)} - \mathrm{I_w(i,j,k)}]^2 \qquad (8)$$

2.5 Robustness Performance

Robustness of a watermarking algorithm is that the embedded data should survive any signal processing operation the host signal goes through and preserve its fidelity. The similarity between the original watermark and the extracted watermark from the attacked watermarked image can be measured by the normalized correlation ‘NC’ factor, which is defined as:

$$\mathrm{NC} = \frac{\sum_{\mathrm{i=1}}^{\mathrm{l}}\mathrm{W(i).W'(i)}}{\sqrt{\sum_{\mathrm{i=1}}^{\mathrm{l}}\mathrm{W^2(i)}}.\sqrt{\sum_{\mathrm{i=1}}^{\mathrm{l}}\mathrm{W'^2(i)}}} \qquad (9)$$

where W is the original watermark and $\mathrm{W'}$is the extracted watermark.

The twelve images of Fig.5 are used as cover images for the proposed watermarking scheme. A 20×50 BMP copyright watermark image is embedded into the twelve cover images. The original watermark, the resulting watermarked images, and the extracted watermark are shown in Fig.6.

It can be concluded that the visual appearance of the watermarked images is good with PSNR values (about 60 dB for the Greyscale images and about 72 dB for the color images), showing that no significant artifacts or distortions because of the process of watermarking. The NC values between the original and the extracted watermarks for the twelve images are ‘1’ (NC = 1) which mean that the original and the extracted watermarks are identical.

The proposed watermarking scheme will be now compared to four existing watermarking schemes that are based also on combining between DWT, DCT, and SVD transforms. Table.1 and Table.2 show the comparison of the proposed watermarking scheme with P. S. Murty et al. [21] and M. M. Rahman [29] schemes in terms of PSNR and NC using Greyscale images, respectively. The proposed scheme is also compared with J. Kaur et al. [30] scheme in terms of PSNR and with K. Chaitanya et al. [31] scheme in terms of PSNR and NC using color images as shown in Table.3 and Table.4, respectively. The results of Table.1, Table.2, Table.3 and Table.4 show the improvement of the proposed watermarking scheme over the four existing schemes with respect to the imperceptibility measure in terms of PSNR and the robustness measure in terms of NC.

Table.1 The PSNR (in dB) values of the P. S. Murty et al. [21], M. M. Rahman [29], and proposed watermarking schemes for Greyscale images.

Image	Murty et al. [21]	Rahman [29]	Proposed Scheme
Lena	47.42	40.5935	60.21
Boat	44.41	34.3189	60.19
Peppers	44.20	40.6685	59.31

Table. 2 The NC values of the P. S. Murty et al. [21], M. M. Rahman [29], and proposed watermarking schemes for the Boat image under different attacks.

Attack	Murty et al. [21]	Rahman [29]	Proposed Scheme
Avg. Filtering [13×13]	-0.0928	0.9589	0.96985
Med. Filtering [13×13]	-0.0852	0.9358	0.95273
AWGN	0.6749	0.4255	0.99666
JPEG [80:1]	0.9751	0.9997	0.97011
Cropping [25%]	0.6120	0.9530	0.98685
Resizing [512→128→512]	0.2570	0.7391	0.95225
Rotation [50˚]	0.8846	0.9338	0.96757
Histogram Equalization	0.9182	0.8416	0.996112
Motion Blurring	- 0.0363	0.9729	0.98662
Sharpening	0.7500	0.7727	0.986319

Table. 3 The PSNR (dB) values of the J. Kaur et al. [30] scheme and the proposed watermarking scheme for color images.

Image	Kaur et al. [30] Scheme	Proposed Scheme
Color Lena	46.76	73.11496
Color Peppers	46.77	72.9982
Color Baboon	46.76	71.4847

Table. 4 The PSNR (dB) and NC values of the K. Chaitanya et al. [31] scheme and the proposed watermarking scheme for color images.

Image	Chaitanya et al. [31] Scheme		Proposed Scheme	
	PSNR	NC	PSNR	NC
Flower 1	35.4944	0.9993	73.2182	1.00
Flower 2	34.7802	0.9991	73.3804	1.00
Flower 3	35.0686	0.9996	73.2973	1.00

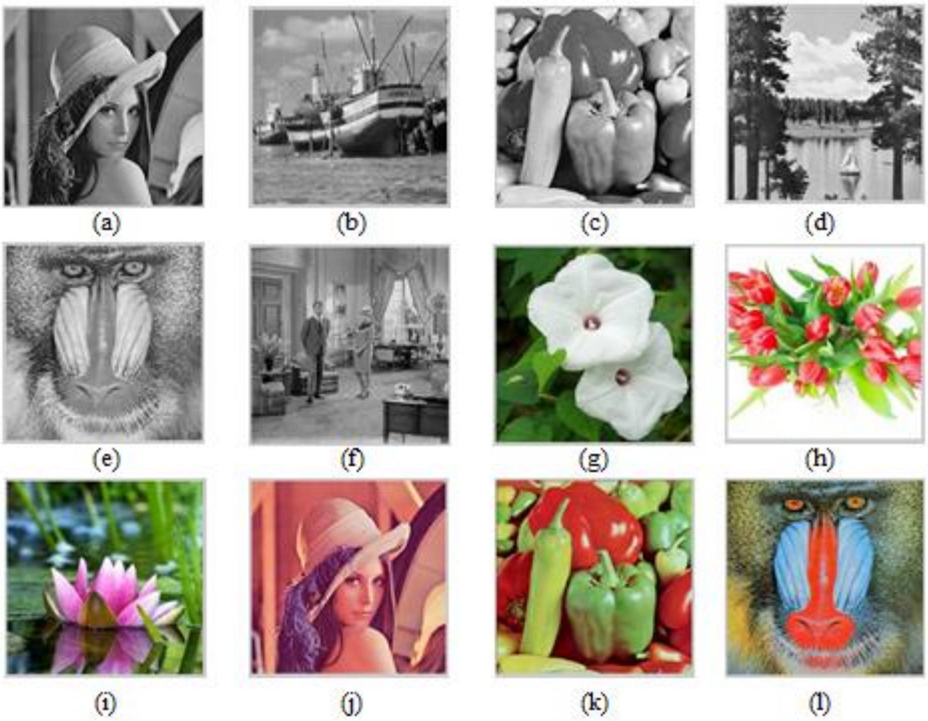


Fig.5 (a) Grey Lena, (b) Boat, (c) Grey Peppers, (d) Lake, (e) Grey Baboon, (f) Living room, (g) Flower1, (h) Flower2, (i) Flower3, (j) Color Lena, (k) Color Peppers, and (l) Color Baboon images.

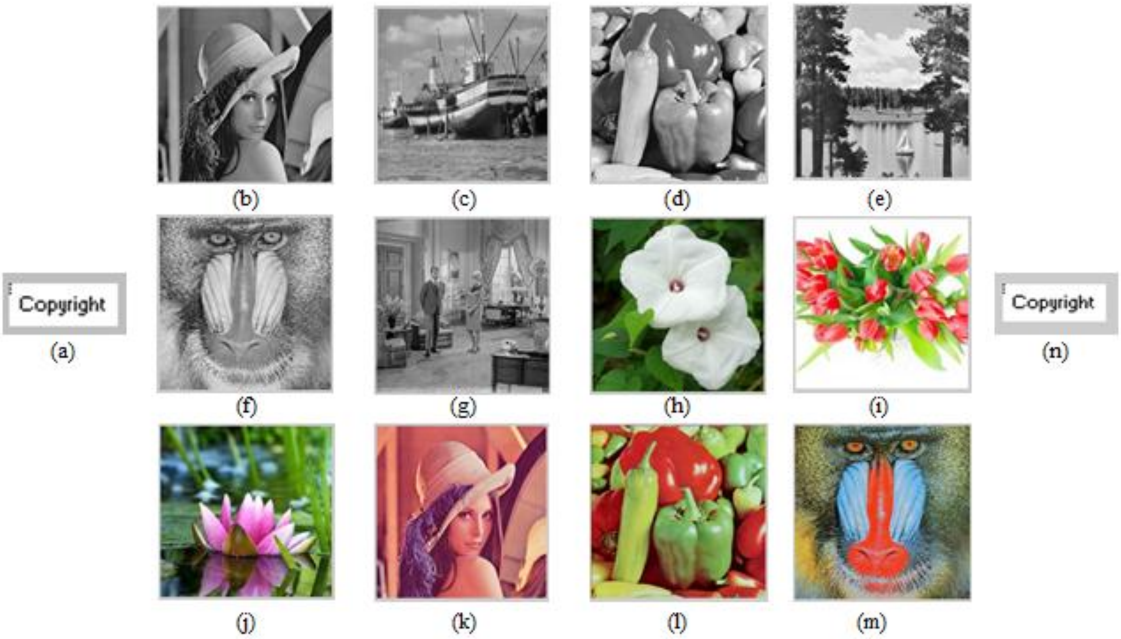


Fig.6 (a) Original Watermark, (b), (c) to (m) Watermarked images, and (n) Extracted Watermark.

3. PROPOSED CHAOS-BASED ENCRYPTION ALGORITHM

Recent researches of image encryption algorithms have been increasingly based on chaotic systems, but the drawbacks of small key space and weak security in one-dimensional chaotic cryptosystems are obvious. The proposed algorithm is based on combining four different dimensions chaotic maps (1D, 2D, 3D Logistic maps, and 2D Henon map). The encryption algorithm used an additional diffusion stage that based primarily on a new shuffling function that shuffles the bit planes of each pixel depending on a round key which is updated sequentially for every pixel. This stage was used before the confusion (permutation) stage to (i) improve the performance of the encryption algorithm against differential attacks (one pixel modification in the plain image will result in a completely different ciphered image), and to (ii) reach a specific high level of security with minimum number of iterations (rounds) in order to maximize the encryption speed. The general block diagram of the proposed encryption algorithm is shown in Fig.7.

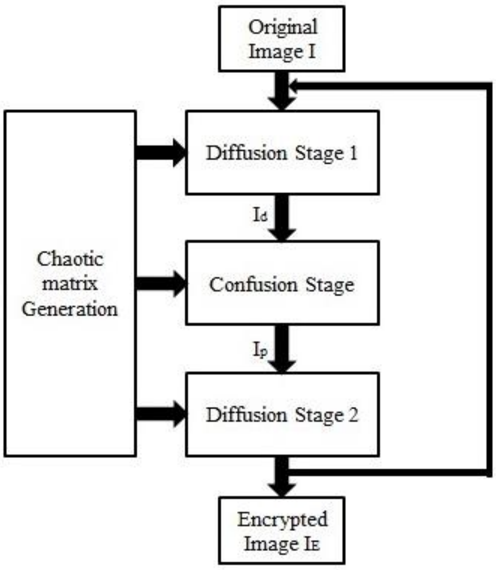


Fig.7 Block diagram of the proposed chaos-based encryption algorithm

Step-1: Chaotic Matrix Generation Block

This block is responsible for generating chaotic matrices for the whole encryption process depending on the encryption key. It contains four chaotic maps; 1D, 2D, and 3D Logistic maps, as well as 2D Henon map. The chaotic matrices can be generated using the chaotic maps equations given below which is iterated M × N times to generate the required number of elements.

- Map-1: 1D logistic map:

$$x_1(0) = \text{Key}$$

$$x_1(n+1) = 3.999x_1(n)\big(1 - x_1(n)\big) \qquad (10)$$

- Map-2: 2D logistic map:

$$x_2(0) = x_1(200), \text{and } y_2(0) = x_1(100)$$

$$x_2(n+1) = 2.95x_2(n)\big(1 - x_2(n)\big) + 0.21y_2^2(n)$$

$$y_2(n+1) =$$

$$2.79y_2(n)\big(1 - y_2(n)\big) + 0.14(x_2^2(n) + x_2(n)y_2(n)) \qquad (11)$$

- Map-3: 2D Henon map:

$$x_3(0) = y_2(100), \text{and } y_3(0) = x_2(200)$$

$$x_3(n+1) = 1 + y_3(n) - 1.4x_3^2(n)$$

$$y_3(n+1) = 0.3x_3(n) \qquad (12)$$

- Map-4: 3D Logistic map:

$$x_4(0) = y_3(100), y_4(0) = x_3(200) \text{ and, } z_4(0) = y_3(50)$$

$$x_4(n+1) = 3.8x_4(n)\big(1 - x_4(n)\big) + 0.021x_4(n)y_4^2(n) + 0.015z_4^3(n)$$

$$y_4(n+1) = 3.8y_4(n)\big(1 - y_4(n)\big) + 0.021y_4(n)z_4^2(n) + 0.015x_4^3(n)$$

$$z_4(n+1) = 3.8z_4(n)\big(1 - z_4(n)\big) + 0.021z_4(n)x_4^2(n) + 0.015y_4^3(n) \qquad (13)$$

Step-2: Diffusion Stage-1

The input of this stage is the original image 'I' and the number of rounds, while the output is the diffused image 'I_d'.

The original image I was reshaped to be a vector 'I_r'of size 1× (M×N).

Four chaotic matrices were generated using chaotic matrix generation block depending on the encryption key that was updated every round using the 1D logistic map. These matrices are Map1, Map2, Map3 and Map4.

The four matrices were discretized using the following equation:

$$\text{RMapi} = \text{Integer}_{\text{Part}((\text{Mapi}.*10^{14})\text{mod}256)}; i = 1:4. \quad (14)$$

The following equation was used to generate the diffused vector 'I_{dr}':

$$I_{dr}(1) = I_r(1) \oplus \text{RMap4}(1)$$

$$I_{dr}(i) = I_r(i) \oplus \text{RMap4}(i) \oplus f_1(I_{dr}(i-1), \text{Key});\ 2 \le i \le M * N \qquad (15)$$

where $I_{dr}(i)$and $I_{dr}(i-1)$are the ith and the (i-1)th diffused pixel value, respectively. $I_r(i)$ is the current reshaped original image pixel value, $\oplus$is a XOR function, and 'f_1' is a shuffle function that shuffles the bit planes of the pervious diffused pixel $I_{dr}(i-1)$ value depending on the round key which was updated every round.

The diffused vector 'I_{dr}' was reshaped to get the diffused image 'I_d' of size M×N.

Step-3: Confusion Stage

This stage consists of two permutation levels. The input to this stage is the diffused image 'I_d' and the output is the diffused-permuted image 'I_p'.

Level-1:

Four matrices were generated using the chaotic matrix generation block of size M×N. Two matrices were generated based on the 2D Logistic and the other two based on the 2D Henon map. The rows and columns of the two chaotic matrices of the 2D Logistic map were arranged in ascending or descending order and the element indices of the four generated matrices were stored, and repeat for the two matrices of the 2D Henon map. The indices of the four matrices were used sequentially to permute the rows and the columns of the diffused matrix.

Level-2: The output matrix of level-1 permutation is reshaped to be a vector of size 1× (M×N). Three matrices were generated using the chaotic matrix generation block of size 1× (M×N). The three matrices were generated based on the 3D Logistic map. The three matrices were arranged in ascending or descending order and the indices of the three generated matrices were stored. These indices were used sequentially to permute the vector. Reshape the permuted vector to get the diffused-permuted image matrix ‘ $\mathrm{I_p}$’ of size M×N.

Step-4: Diffusion Stage-2

The final step is the main diffusion stage that encrypts the diffused-permuted image ‘ $\mathrm{I_p}$’ to get the final encrypted image ‘ $\mathrm{I_E}$’. The Four chaotic matrices RMap1, RMap2, RMap3 and RMap4 were generated using the chaotic matrix generation block.

The diffused-permuted image ‘ $\mathrm{I_p}$’ was reshaped to be a vector ‘ $\mathrm{I_{pr}}$’. The encrypted image was obtained as follows

$$\mathrm{I_{Er}(1) = I_{pr}(1) \oplus RMap1(1) \oplus RMap2(1)} \qquad (16)$$

$$\mathrm{a = I_{pr}(i-1) \oplus RMap1(i)} \qquad (17)$$

$$\mathrm{b = [f_1(a, Key)] \oplus\ RMap2(i)} \qquad (18)$$

$$\mathrm{mod(b,4)} = \begin{cases} \mathrm{0; c = I_{Er}(i-1) \oplus RMap1(i)} \\ \mathrm{1; c = I_{Er}(i-1) \oplus RMap2(i)} \\ \mathrm{2; c = I_{Er}(i-1) \oplus RMap3(i)} \\ \mathrm{3; c = I_{Er}(i-1) \oplus RMap4(i)} \end{cases} \qquad (19)$$

$$\mathrm{I_{Er}(i)) = f_2\left(\left(a \oplus c \oplus I_{pr}(i)\right), a\right); 2 \leq i \leq M * N} \qquad (20)$$

where $\mathrm{I_{pr}(i)}$ and $\mathrm{I_{pr}(i-1)}$ are the $\mathrm{i^{th}}$ and the $\mathrm{(i-1)^{th}}$ permuted pixel value, respectively. $\mathrm{f_2(x,y)}$ is a cyclic bit-shift function that right-shifts the bit planes of an integer value x by a number of y shifts. Reshape the ‘ $\mathrm{I_{Er}}$’ vector to obtain the final encrypted image ‘ $\mathrm{I_E}$’.

3.1 Proposed Chaos-Based Encryption Algorithm Experimental Results

Different experiments are carried out in this section to show the efficiency of the proposed chaos-based encryption algorithm. Twelve Greyscale BMP plain images of size 256 × 256 are used in these experiments. These images are shown in Fig.8.

3.2 Sensitivity Analysis

High key sensitivity is required by secure image cryptosystems, which means that the cipher image cannot be decrypted correctly even if there is only a small difference between the encryption and decryption keys.

The key sensitivity of the proposed encryption algorithm is tested using five plain images of Fig.8. The five plain images are shown in Fig.9(a). These images are encrypted using the key (key = 0.010100000000001) and shown in Fig.9(b). The encrypted images are decrypted using the same encryption key and shown in Fig.9(c). Then, the encrypted images are decrypted using a wrong key (wrong key = 0.010100000000002) and shown in Fig.9(d). It can be concluded that the proposed encryption algorithm is very sensitive to a tiny change in the secret keys.

3.3 Statistical Analysis

It is well known that passing the statistical analysis on cipher image is of crucial importance for a cryptosystem actually, an ideal cipher should be strong against any statistical analysis. In order to prove the security of the proposed image encryption scheme, the following statistical tests are performed.

3.3.1 Histogram Analysis

To prevent the access of information to attackers, it is important to ensure that encrypted and original images do not have any statistical similarities. The histogram analysis clarifies that, how the pixel values of image are distributed. We test the histogram analysis of the proposed encryption algorithm using five plain images as shown in Fig.10. The histograms of the plain images contain great sharp rises followed by sharp declines as shown in Fig.10(b). The histograms of the encrypted images are shown in Fig.10(c). It shows that Greyscale values are uniformly distributed in the encrypted images. Hence the proposed encryption algorithm is safe from histogram analysis attacks and satisfies the diffusion property.

3.3.2 Correlation analysis

There is a very good correlation among adjacent pixels in the digital image [18]. The correlation between two adjacent pixels in vertical, horizontal, and diagonal orientations can be defined as

$$\mathrm{R} = \frac{\mathrm{N\sum_{j=1}^{N}(x_j \times y_j) - \sum_{j=1}^{N} x_j \times \sum_{j=1}^{N} y_j}}{\sqrt{\mathrm{[(N\sum_{j=1}^{N} x_j^2) - \left(\sum_{j=1}^{N} x_j\right)^2] \times [(N\sum_{j=1}^{N} y_j^2) - (\sum_{j=1}^{N} y_j)^2]}}} \qquad (21)$$

where x and y are intensity values of two neighboring pixels in the image and N is the number of adjacent pixels selected from the image to calculate the correlation.

Five plain images are selected from Fig.8 to be used in this experiment. We randomly select 2500 pairs of two adjacent pixels from the five plain images and the corresponding cipher images in the three directions (vertical, horizontal, and diagonal) and calculate the correlation coefficients using Eq.21. Table.5 shows the comparison of the proposed algorithm with AES, Blowfish and IDEA algorithms in terms of correlation coefficients in the three directions. It can be observed that the correlation coefficients of the proposed encryption algorithm in vertical, horizontal, and diagonal orientations are better than that of the other algorithms. The adjacent pixel correlation distribution of the proposed encryption algorithm for the Jetplane image and the corresponding cipher image in the three directions are shown in Fig.11. The results show that, the correlation between adjacent pixels in the encrypted image is very low and the proposed approach satisfies the confusion property significantly.

3.3.3 Information Entropy Analysis

The entropy is the most outstanding feature of randomness. The entropy of a message source can be defined as:

$$\mathrm{H(m) = -\sum_{i=0}^{2^N-1} p(m_i) \log_2(p(m_i))} \qquad (22)$$

where p (mi) is the probability of symbol mi, N is the number of bits for each symbol. Ideally, for a random source emitting 256 symbols, its entropy is 8 bits. Table.6 shows the comparison of the proposed encryption algorithm with AES, Blowfish and IDEA algorithms in terms of information entropy for five plain images and the corresponding cipher images. The results show that the information entropy values of the proposed encryption algorithm are slightly better than that of other algorithms. The entropy values of the proposed algorithm are very close to the theoretical value 8. This means that the information leakage may be negligible in the encryption process and the encryption algorithm is secure against the entropy attack.

3.4 Differential analysis

In general, desirable characteristic for an encrypted image is being sensitive to the little changes in plain-image (e.g. modifying only one pixel). Adversary can create a small change in the input image to observe changes in the result. By this method, the meaningful relationship between original image and cipher image can be found. If one little change in the plain image can cause a significant change in the cipher image, with respect to diffusion and confusion, then the differential attack actually loses its efficiency and becomes almost useless. To test the influence of one-pixel change on the plain image, encrypted by the proposed encryption algorithm, two common measures may be used: Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI). Consider two cipher images of size M×N×f, whose corresponding plain images have only one pixel difference, be denoted by IE1and IE2.The NPCR measures the percentage of different pixel numbers between the two images and is defined as

$$\mathrm{NPCR} = \left[\frac{1}{\mathrm{M\times N\times f}}\sum_{\mathrm{k=1}}^{\mathrm{f}}\sum_{\mathrm{i=1}}^{\mathrm{M}}\sum_{\mathrm{j=1}}^{\mathrm{N}}\mathrm{D(i,j,k)}\right] \times 100\ \% \qquad (23)$$

$$\mathrm{D(i,j)} = \begin{cases} 0, \text{if } \mathrm{I_{E1}(i,j,k)} = \mathrm{I_{E2}(i,j,k)} \\ 1, \text{if } \mathrm{I_{E1}(i,j,k)} \neq \mathrm{I_{E2}(i,j,k)} \end{cases} \qquad (24)$$

The UACI measures the average intensity of differences between the two images and is defined as

$$\mathrm{UACI} = \frac{1}{\mathrm{M\times N\times f}}\left[\sum_{\mathrm{k=1}}^{\mathrm{f}}\sum_{\mathrm{i=1}}^{\mathrm{M}}\sum_{\mathrm{j=1}}^{\mathrm{N}}\frac{\mathrm{I_{E1}(i,j,k)} - \mathrm{I_{E2}(i,j,k)}}{2^{\mathrm{l}}-1}\right] \times 100\ \% \qquad (25)$$

where ‘l’ is the number of bits per pixel of the plain image. Table.7 shows the comparison of the proposed encryption algorithm with AES, Blowfish, and IDEA algorithms in terms of NPCR and UACI values for five plain images. The typical values of NPCR and UACI for any two random images are NPCR = 99.69 % and UACI = 33.3 %. The results demonstrate that the cipher image is highly sensitive to any change in the plain image and our proposed algorithm shows better results than the other algorithms.

3.5 Encryption Speed

Apart from the security consideration, some other issues on image encryption are also important. This includes the encryption speed for real-time processing. The encryption time of 256 Greyscale Baboon BMP image of size 256 × 256 is measured using the proposed algorithm and compared to AES, Blowfish, and IDEA algorithms as shown in Fig.12. The results show the superiority of the proposed encryption algorithm over other algorithms in terms of the processing time.

3.6 Comparative Study

The proposed encryption algorithm is compared with some of the most recent encryption algorithms: O. M. Al-hazaimeh [32], T. Sivakumar et al. [33], P. V. Saraswathi et al. [34], M. Ghebleh et al. [35], R. E. BORIGA et al. [36] algorithms. The comparison is in terms of: (i) the resistance against differential attacks (NPCR, UACI), and (ii) correlation coefficients of adjacent pixels (in vertical, horizontal, and diagonal orientations). Table.8 shows that the proposed encryption algorithm has similar or better results than other algorithms.

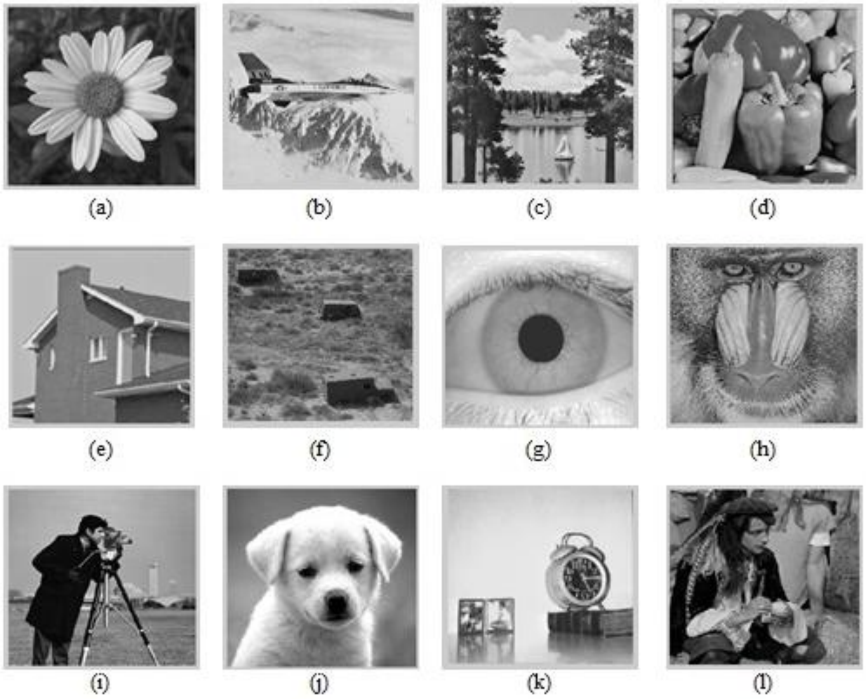


Fig.8 (a) Flower, (b) Jetplane, (c) Lake, (d) Peppers, (e) House, (f) Trucks, (g) Iris, (h) Baboon, (i) Cameraman, (j) Dog, (k) Clock, and (l) Man images.

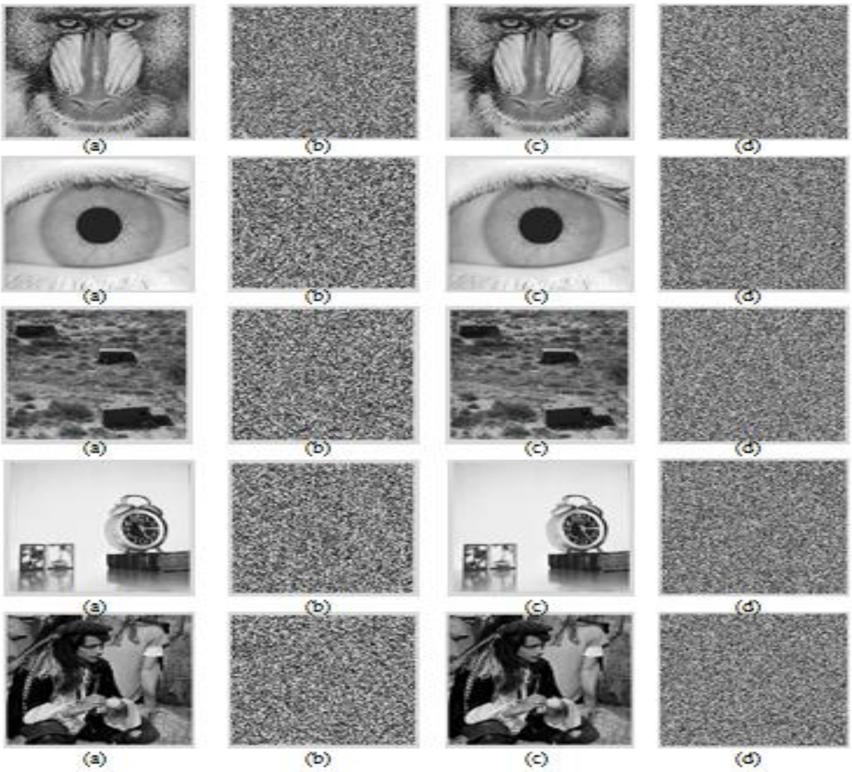


Fig.9 (a) The plain images, (b) The encrypted images, (c) The decrypted images using the correct key, and (d) The decrypted images using a wrong key.

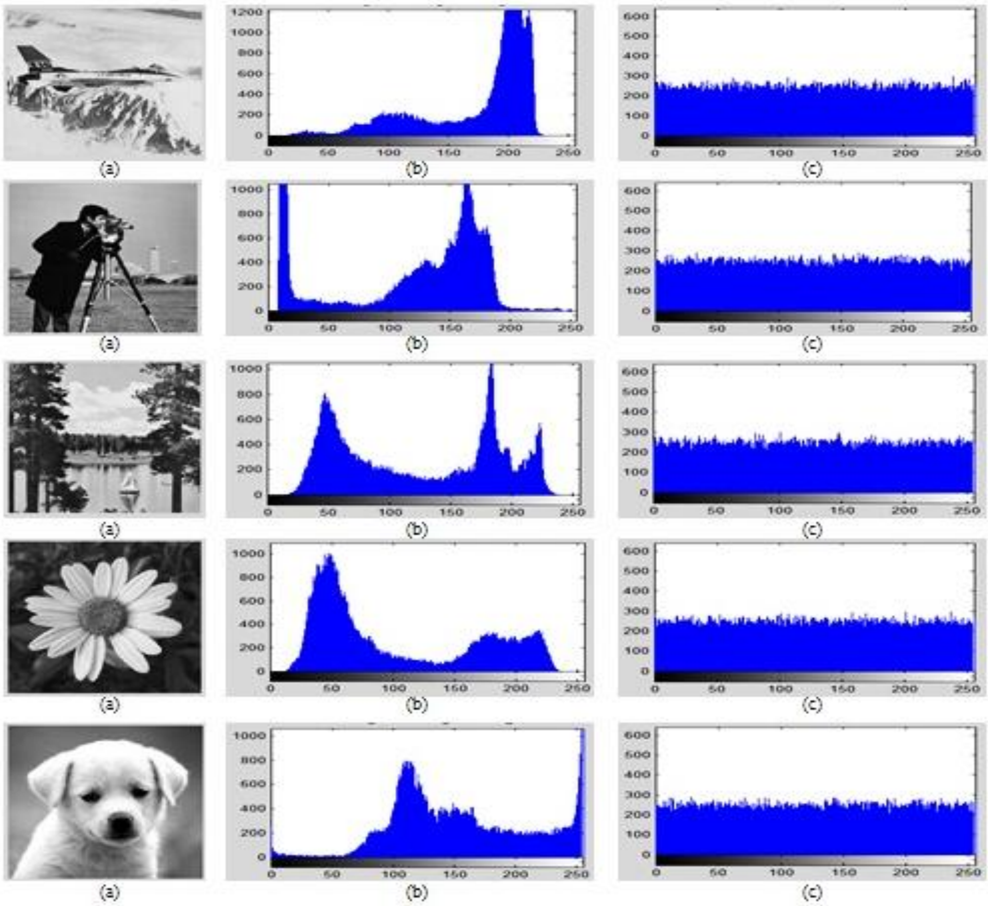


Fig.10 (a) The plain images, (b) The histograms of the plain images, and (c) The histogram of the encrypted images

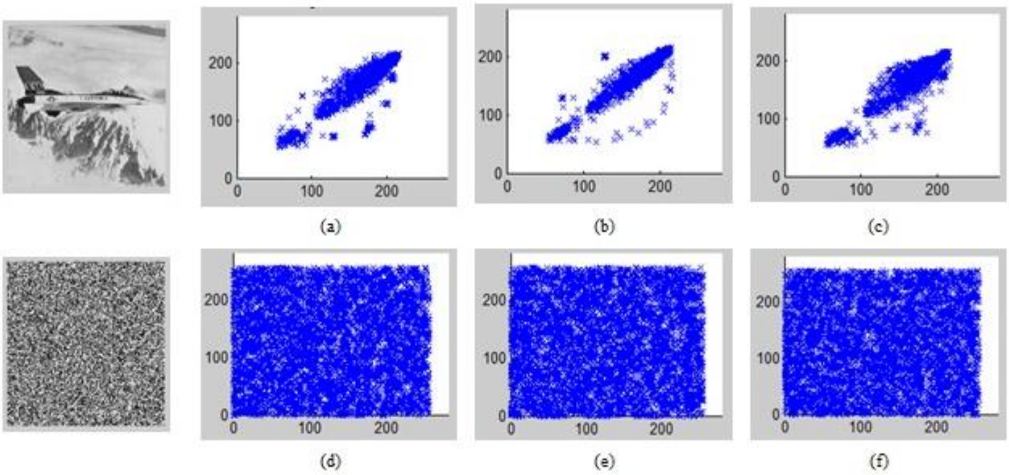


Fig.11 The correlation coefficient distribution of: (a), (b), and (c) plain image in the vertical, horizontal, and diagonal orientations and of (d), (e), and (f) encrypted image in the vertical, horizontal, and diagonal orientations, respectively.

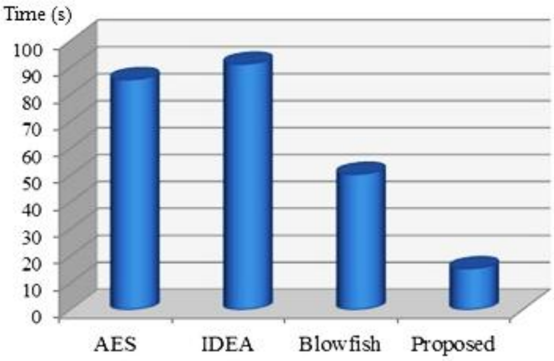


Fig.12 The encryption time of Baboon image using AES, Blowfish, IDEA and the proposed encryption algorithms

Table.5 Correlation coefficients values of AES, Blowfish, IDEA, and the proposed encryption algorithm

Images		**Plain Image**	**AES**	**Blowfish**	**IDEA**	**Proposed Encryption Algorithm**
Flower	**V**	0.9863	0.0245	0.0144	0.0220	0.0031
	H	0.9832	0.0251	0.0093	0.0480	0.0022
	D	0.9688	0.0273	0.0115	0.0270	0.0005
Jet plane	**V**	0.9152	0.0351	0.0378	0.0060	0.0007
	H	0.9619	0.0113	0.0240	0.0149	0.0030
	D	0.9049	0.0275	0.0135	0.0076	0.0023
Lake	**V**	0.9510	0.0491	0.0095	0.0084	0.0021
	H	0.9588	0.0256	0.0130	0.0132	0.0018
	D	0.9272	0.0125	0.0251	0.0211	0.0001
Peppers	**V**	0.9632	0.0370	0.0172	0.0218	0.0013
	H	0.9513	0.0145	0.0301	0.0336	0.0023
	D	0.9146	0.0099	0.0102	0.0111	0.00003
House	**V**	0.9359	0.0234	0.01379	0.0099	0.0023
	H	0.9845	0.0215	0.02081	0.0254	0.0003
	D	0.9340	0.0228	0.01420	0.0185	0.004

Table.6 The entropy values of AES, Blowfish, IDEA, and the proposed encryption algorithm.

Parameters	Plain Image	AES	Blowfish	IDEA	Proposed encryption Algorithm
Flower	7.3707	7.9975	7.9972	7.9975	7.9988
Jetplane	6.7260	7.9970	7.9970	7.9969	7.9986
Lake	7.4561	7.9976	7.9971	7.9968	7.9985
Peppers	7.5807	7.9971	7.9971	7.9970	7.9986
House	6.4961	7.9924	7.9973	7.9971	7.9979

Table.7 The NPCR and UACI of AES, Blowfish, IDEA, and the proposed encryption algorithm

Images		AES	Blowfish	IDEA	Proposed Encryption Algorithm
Flower	**NPCR**	99.6063	99.6201	99.5804	99.688
	UACI	30.5731	30.8254	29.8547	33.2405
Jet plane	**NPCR**	99.5804	99.6246	99.6053	99.6867
	UACI	32.1727	31.0062	31.2027	33.1039
Lake	**NPCR**	99.5292	99.6029	99.5895	99.6865
	UACI	31.4384	29.4853	29.3810	32.8156
Peppers	**NPCR**	99.6155	99.5819	99.6109	99.6899
	UACI	29.2488	30.2884	29.4374	33.5725
House	**NPCR**	99.5209	99.5582	99.5140	99.6938
	UACI	30.5594	29.0594	28.6070	32.9244

Table.8 Performance Comparison of the proposed encryption algorithm with other algorithms

Algorithms	NPCR	UACI	Correlation Coefficients		
			V	H	D
O. M. Al-hazaimeh [32]	99.6104	13.0755	0.053	0.051983	0.0505
T. Sivakumar et al. [33]	99.4800	30.8700	0.352	0.342	0.298
P. V. Saraswathi et al.[34]	99.8500	33.5800	0.04912	0.01776	0.00348
M. Ghebleh et al. [35]	99.6100	33.7200	0.0049	-0.0043	0.0057
R. E. BORIGA et al. [36]	99.2400	33.1300	0.0059	0.0039	0.0004
Proposed encryption Algorithm	99.6889	33.1313	0.00190	0.00192	0.001386

4. PROPOSED HYBRID ALGORITHM

In this section, we provide some experimental results to illustrate the performance of the proposed hybrid encryption-watermarking algorithm. A simple block diagram of the proposed hybrid encryption-watermarking algorithm is shown in Fig.13. The watermark image is first encrypted using the proposed chaos-based encryption algorithm and then it is embedded into the cover image using the proposed watermarking scheme.

The twelve cover images that are shown in Fig.5 will be used here to test the proposed hybrid algorithm. Fig.14 indicates that the visual quality of the watermarked images is good with PSNR (about 62 dB for greyscale images and 75 dB for color images) showing no significant artifacts or distortion due to the watermarking process. It also indicates that the original and the extracted-decrypted watermark images are identical since the NC between them is ‘1’.

In the following experiments, the imperceptibility and the robustness of the proposed hybrid encryption-watermarking scheme is estimated based on a 256 greyscale Lena BMP plain-image of size 512×512 by performing several image processing attacks, including JPEG compression, filtering, blurring, noise addition, Intensity adjustment, gamma correction, histogram equalization, resizing, rotation and distortion on the watermarked Lena image. To demonstrate the effectiveness of the proposed hybrid scheme, a comparison of the proposed hybrid encryption-watermarking scheme with hybrid AES, IDEA, Blowfish cryptography based DWT-DCT-SVD watermarking schemes is summarized in Table 8 and Table 9.

Table.8 indicates that the imperceptibility performance of the proposed hybrid algorithm in terms of PSNR is better than other hybrid algorithms. It also indicates that the proposed hybrid algorithm is faster than other algorithms since it has the least elapsed time value. Table.9 indicates that the proposed algorithm is extremely robust against various image processing attacks and the NC values of proposed algorithm are better than that of other hybrid algorithms.

5. CONCLUSION

This paper proposes a secure hybrid encryption-watermarking algorithm for copyright protection. The watermarking phase is based on combining the DWT, DCT, and SVD transformations, while the encryption phase is based on using four chaotic maps of different dimensions. Experimental tests were performed individually for the proposed DWT-DCT-SVD watermarking scheme and for the proposed chaos-based encryption algorithm. The experimental results validate that the proposed watermarking scheme is highly robust against wide set of watermarking attacks and it achieves high degree of imperceptibility. The results also showed the improvement of the proposed watermarking scheme over four existing schemes that are also based on combining between DWT, DCT, and SVD. The security of the proposed encryption algorithm is evaluated by the key sensitivity analysis, the statistical analysis, the resistance against differential attacks, and the processing speed. The proposed encryption algorithm showed better performance than traditional ciphers with respect to the security level and the encryption speed. It also showed better or similar performance as compared with some of the most recent existing algorithms. Finally, experimental results demonstrate that the proposed hybrid encryption-watermarking algorithm achieved high security performance against cryptographic attacks thanks to the encryption algorithm and it also achieved high degree of imperceptibility and robustness against different types of watermarking attacks thanks to the watermarking algorithm.

6. REFERENCES

[1] F. Hartung and M. Kutter, “Multimedia Watermarking Techniques," Proceedings of the IEEE, vol. 87, no. 7, pp. 1079-1107, July 1999.

[2] R. Ni, Q. Ruan, Y. Zhao, “Pinpoint authentication watermarking based on a chaotic system,” Forensic Science International vol. 179, no. 1, pp. 54-62, 2008.

[3] C. Deng, X. Gao, X. Li, D. Tao, “Local histogram based geometric invariant image watermarking”, Signal Processing vol. 90, no. 12, pp.3256–3264, 2010.

[4] O. Findik, I. Babaoglu, E. Ulker, “A color image watermarking scheme based on hybrid classification method: particle swarm optimization and k-nearest neighbor algorithm,” Optics Communications, vol. 283, no. 24, pp. 4916–4922, 2010.

[5] J.C. Patra, J.E. Phua, C. Bornand, “A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression,” Digital Signal Processing, vol. 20, no. 6, pp. 1597–1611, 2010.

[6] S.D. Lin, S.C. Shie, J.Y. Guo, “Improving the robustness of DCT-based image watermarking against JPEG compression,” Computer Standards & Interfaces, vol. 32, no. 1-2, pp.54-60, 2010.

[7] D. Kundur, D. Hatzinakos, “Toward robust logo watermarking using multiresolution image fusion,” IEEE Transactions on Multimedia, vol. 6, no.1, pp. 185–197, 2004.

[8] W.H. Lin, Y.R. Wang, S.J. Horng, “A wavelet-tree-based watermarking method using distance vector of binary cluster,” Expert Systems with Applications vol. 36, no. 6, pp. 9869–9878, 2009.

[9] H.M. Al-Otum, N.A. Samara, “A robust blind color image watermarking based on wavelet-tree bit host difference selection,” Signal Processing, vol. 90, no. 8, pp. 2498–2512, 2010.

[10] A.A. Mohammad, A. Alhaj, S. Shaltaf, “An improved SVD-based watermarking scheme for protecting rightful ownership,” Signal Processing vol. 88, no. 9, pp. 2158–2180, 2008.

[11] C.C. Lai, “An improved SVD-based watermarking scheme using human visual characteristics,” Optics Communications, vol. 284, no. 4, pp. 938–944, 2011.

[12] K. R. Rao, and P. Yip, “Discrete Cosine Transform: algorithms, advantages, applications,” Academic Press Professional, USA, 1990.

[13] H. M. Yang, Y. Q. Liang, X. D. Wang, S. J. Ji, “ A DWT-Based Evaluation Method of Imperceptibility of watermark in watermarked color image,” Proceeding of the 2007 International Conference on Wavelet Analysis and Pattern Recognition, vol. 1, Nov, 2007.

[14] R. Liu and T. Tan, “An SVD-based watermarking scheme for protecting rightful ownership,” IEEE Transactions on Multimedia, vol. 4, pp. 121–128, 2002.

[15] F. Huang, Z.H. Guan, “A hybrid SVD-DCT watermarking method based on LPSNR,” Pattern Recognition Letters, vol. 25, no. 15, pp. 1769–1775, 2004.

[16] P. Singh, S. Agarwal, “A hybrid DCT-SVD based robust watermarking scheme for copyright protection,” AFRICON, pp. 1-5, 2013.

[17] E. Ganic, A.M. Eskicioglu, “Robust embedding of visual watermarks using DWT-SVD,” Journal of Electronic Imaging, vol. 14, no.4, 2005.

[18] V. Aslantas, L. A. Dog¢an, and S. Ozturk, “DWT-SVD based image watermarking using particle swarm optimizer,” IEEE International Conference on Multimedia Expo, pp. 241–244, 2008.

[19] S.K. Amirgholipour, A. R. Naghsh-Nilchi, “Robust Digital Image Watermarking Based on Joint DWT-DCT, ”International Journal of Digital Content Technology and its Applications, vol. 3, no. 2, pp.42-54, 2009.

[20] A. Al-Haj, “Combined DWT-DCT Digital Image Watermarking,” Journal of Computer Science vol. 3, no. 9, pp. 740-746, 2007.

[21] P.S. Murty, M. U. Bhaskar, P. R. Kumar, “A Semi-Blind Reference Watermarking SchemeUsing DWT-DCT-SVD for Copyright Protection,” International Journal of Computer Science & Information Technology (IJCSIT), vol. 4, no. 2, 2012.

[22] R. Matthews, “On the derivation of a chaotic encryption algorithm” Cryptologia, vol. 13, no. 1, pp. 29–42, 1984.

[23] L. Zhang, X. Liao, X. Wang, “An image encryption approach based on chaotic maps,” Chaos Solitons& Fractals, vol. 24, no. 3, pp. 759-765, 2005.

[24] X. Wu, H. Hu, B. Zhang, “Analyzing and improving a chaotic encryption method,” Chaos Solitons &Fractals, vol. 22, no. 2, pp. 367-373, 2004.

[25] S. Li, X.Mou, Y. Cai, "Improving security of a chaotic encryption approach," Physics Letters A vol. 290, no. 3-4, pp. 127-133,2001.

[26] S. Katzenbeisser,et al., "First summary report on hybrid systems," European ProjectIST-2002-507932, ECRYPT-Network of Excellence in Cryptology, Jan2005.

[27] N. Merhav, "On joint coding of watermarking and encryption," IEEE Transactions on Information Theory vol. 52, pp. 190–205, Jan. 2006.

[28] M. Kutter, F. Jordan, and F. Bossen, "Digital signature of color images using amplitude modulation," in Proc. SPZE Int. Conf: Storage and Retrieval for Zmage and Video Database, vol. 3022, pp. 518-526, 1997.

[29] M.M. Rahman, "A DWT, DCT AND SVD Based Watermarking Technique to Project the Image Piracy," International Journal of Managing Public Sector Information and Communication Technologies (IJMPICT), vol. 4, no. 2, June 2013.

[30] J. Kaur, R. Khanna, and D. Sandhu, "New Watermarking Scheme for Gray Image Based on DWT and SVD-DCT," International Journal of Electronics and Communication Engineering, vol. 5, no. 4, pp. 389-397, 2012.

[31] K. Chaitanya, E. S. Reddy, and K. G. Rao, "Digital Color Image Watermarking In RGB Planes Using DWT-DCT-SVD Coefficients," International Journal of Computer Science and Information Technologies (IJCSIT), vol. 5, no. 2, pp. 2413-2417, 2014.

[32] O. M. Al-hazaimeh, "A Novel Encryption Scheme for Digital Image Based on One Dimensional Logistic Map," Computer and Information Science, vol. 7, no. 4, pp.65-73, 2014.

[33] T. Sivakumar and R. Venkatesan, "A novel approach for image encryption using dynamic SCAN pattern", IAENG International Journal of Computer Science, vol. 41, no. 2, pp. 91-101, 2014.

[34] P.V. Saraswathi and M. Venkatesulu, "A block cipher algorithm for multimedia content protection with random substitution using binary tree traversal", Journal of Computer Science, vol. 8, no. 9, pp. 1541-1546, 2012.

[35] M. Ghebleh, A. Kanso and H. Noura, "An image encryption scheme based on irregularly decimated chaotic maps", Signal Processing: Image Communication, vol. 29, no. 5, pp. 618-627, 2014.

[36] R.E. Boriga, A.C. Dascalescu, and A.V. Diaconua, "New Fast Image Encryption Scheme Based on 2D Chaotic Maps," IAENG International Journal of Computer Science, vol.41, no.4, 2014

7. APPENDIX

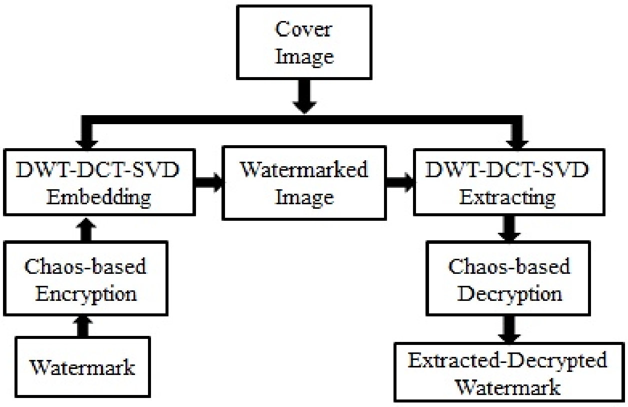


Fig.13 Simple Block Diagram of the proposed hybrid encryption-watermarking algorithm

Table.8 PSNR, MSE, and Elapsed-time of the proposed, AES, IDEA, Blowfish Encryption hybrid DWT-DCT-SVD watermarking algorithms

Parameter	**Hybrid Encryption – (DWT-DCT-SVD) Watermarking**			
	AES	**IDEA**	**Blowfish**	**Proposed Chaos-based**
PSNR (dB)	56.187	55.584	56.7059	62.2438
MSE	0.1467	0.1686	0.1302	0.0363
Elapsed-Time (s)	13.802	16.931	6.9312	1.8096

Table 9 - The NC values of the proposed, AES, IDEA, Blowfish Encryption hybrid DWT-DCT-SVD watermarking algorithms

Noise Type	Hybrid Encryption – (DWT-DCT-SVD) Watermarking			
	AES	IDEA	Blowfish	Proposed Chaos-based
No attack	0.9878	0.99383	0.9935	1.0000
Gaussian Noise	0.91387	0.947289	0.96151	0.99996
Salt & Pepper Noise	0.91951	0.958621	0.96815	0.9575
Speckle Noise	0.95675	0.952474	0.97699	0.99996
Blurring	0.851299	0.855138	0.84304	0.88529
Cropping	0.975772	0.977152	0.9843	0.99996
Intensity adjustment	0.966503	0.981822	0.98612	0.99996
Gamma Correction	0.935349	0.962034	0.97089	0.99996
Histogram Equalization	0.951743	0.973633	0.98757	0.99996
Median Filtering	0.84499	0.845175	0.840731	0.85291
JPEG Compression	0.85163	0.840007	0.84648	0.84752
Rotation	0.837515	0.854102	0.85316	0.89418

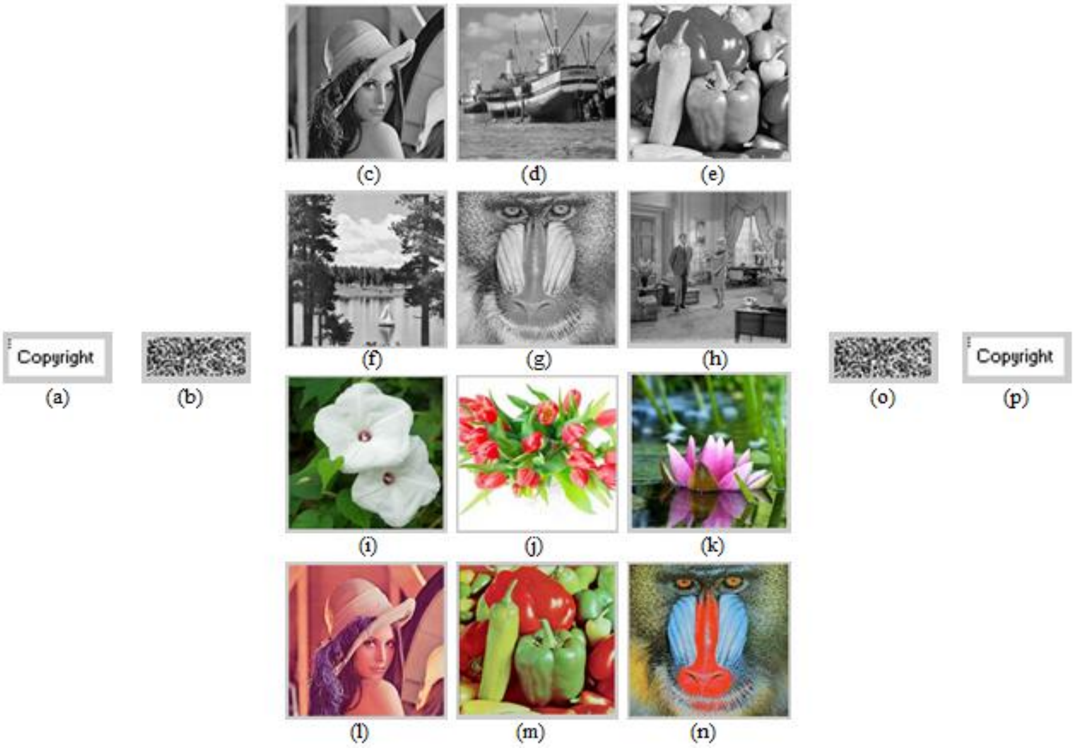


Fig.14 (a) Original watermark, (b) Encrypted watermark, (c) to (n) Watermarked images, (o) Extracted-encrypted watermark, and (p) Extracted-decrypted watermark images