Optik 127 (2016) 5711–5721

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

ABC optimized secured image watermarking scheme to find out the rightful ownership

Irshad Ahmad Ansari [a], Millie Pant [a], Chang Wook Ahn [b,*]

[a] Department of ASE, Indian Institute of Technology Roorkee, India
[b] Department of CSE, Sungkyunkwan University, Suwon, Republic of Korea

ARTICLE INFO

Article history:
Received 10 July 2015
Received in revised form
12 December 2015
Accepted 29 March 2016

Keywords:
Singular value decomposition
False positive free
Robust image watermarking
Artificial bee colony (ABC)
Discrete wavelet transform

ABSTRACT

Digital image watermarking is one the powerful means to find out the unauthorized use of copyrighted images. It inserts secret information (watermark) into the host image that helps in finding the rightful ownership of image. SVD (singular value decomposition) based watermarking is known for providing high capacity, robustness an imperceptibility but most of the SVD based techniques suffers from false positive issue. The proposed work is carried out in order to provide a secured image watermarking with a decent capacity. DWT (discrete wavelet transform) is employed on the host image and then block wise singular components (maximum) are used for watermark's principal component insertion. The insertion of principal component makes the scheme free from false positive error. The insertion is done with the help of multiple scaling factors. The values of scaling factors are found out with the help of ABC (artificial bee colony) in order to obtain a good tradeoff between robustness and imperceptibility. The proposed scheme is also compared with earlier proposed schemes and it shows a visible improvement in performance.

1. Introduction

With the development of multimedia and information technology, sharing of images becomes very easy and fast. But the same technology makes the security and privacy of our personal/confidential images vulnerable. Many powerful image processing tools are being used to manipulate the images and claim the ownership of the same [1–3]. This practice becomes a huge threat to copyrighted material and this can even lead to big financial loss [1]. So the protection of images becomes need of the hour. One of such powerful tools, to deal with protection of images is image watermarking. In Image watermarking, a visible/invisible watermark is used to insert into the host image such that it can be used to authenticate the host afterwards [1–3]. This insertion should be done in such a way that it affect the host image minimally and reproduction of this watermark can be done easily even after intentional/unintentional attacks. Different watermarking strategies helps us to know that attacked region in host image [4,5], ownership information of host image [6] and even recovery of deleted region [7]. The watermark can be embedded either pixel wise [4,5,7] or in transform domain [8–12]. Transform domains (DCT, DWT, IWT etc.) are proven to be more robust toward the different attacks [8–12]. Middle frequencies show more robust nature toward different attack and so they become first choice for many watermarking scheme [8–12].

* Corresponding author.
E-mail addresses: 01.irshad@gmail.com (I.A. Ansari), millifpt@iitr.ac.in (M. Pant), cwan@skku.edu (C.W. Ahn).

http://dx.doi.org/10.1016/j.ijleo.2016.03.070
0030-4026/

The main objective of any robust watermarking scheme is to show robust nature toward different attacks as well as minimizing the degrading in visual quality of host due to watermark insertion. SVD based image watermarking provide a very efficient way of doing so [8–12]. The insertion of watermark in SVD, makes the scheme robust toward attack as singular values did not get much affected by the attacks. The imperceptibility of the SVD based watermarking scheme also remain very good as small change in SVD does not affect the visibility much. Even though, SVD based scheme provides very good imperceptibility and robustness but most of these schemes suffer with false positive issue [13,14].

The problem of false positive issue arises within the watermark insertion methodology itself. The SVD break down the transformed image in three vectors U, S and V. The insertion is done in the S matrix by varying singular matrix value with the help of watermark/singular value of watermark and scaling factor. The watermarked U and V matrices are generated by the help of this watermarked S matrix and it is been saved with the owner. Though, matrix S shows very robust nature toward different attacks [8–17] but it does not contain any detailed information about the image [18]. This led the authentication of wrong owner also because during the authentication process, watermarked U and V matrices are supplied by the owner itself. If attacker supplied a wrong U and V matrices then a different watermark will be generated as U and V contain the detailed information about the image [18]. The detailed theory of false positive problem can be read from Ref. [13,14].

Many researchers [19–22] implemented the SVD based watermarking in different domains like DCT, DFT, and DWT etc. Domain change help in imperceptibility and robustness but it did not help in false positive issue. So many solutions of this false positive problem are given by different researchers [23–25]. These suggestions include use of secret signature information generated by hash function, insertion of complete watermark, encryption of watermark before insertion, insertion of principal components of watermark etc. The use of signature method did not able to provide a complete security to the scheme as the signature was itself vulnerable to different attacks. The insertion of complete watermark provided complete security but the drawback was a poor capacity. The insertion of watermark's principal components in to the host provides a very efficient tradeoff between robustness, imperceptibility and capacity. Guo and Prasetyo [26] suggested a watermark approach using the insertion of watermark's principal components into the blocked host image. The use of block wise insertion makes the scheme more robust toward attack. Guo and Prasetyo [26] used fixed scaling factor for the insertion of watermark data. The use of multiple scaling factors can improve this performance even further as different blocks behave differently toward attack.

In order to improve the performance of image processing algorithms, nature inspired optimization techniques have emerged as significant tool in recent past. Few applications of these algorithms are image segmentation [27], compression [28], watermarking [29] etc. PSO [29], DE [30] and ABC [27,28] are few metaheuristics that are used quite frequently in image processing. The competitive performance of ABC is verified by the help of numerical comparisons with other metaheuristics techniques; along with an additive advantage of using less control parameters [31]. Due to its simple and fast implementation along with robust performance, it is being used widely for imaging as well as other applications [27,28,32–34]. The ABC is already been implemented into imaging optimization areas like image segmentation [27], compression [28] and enhancement [33] and it provided a very good results. ABC is known to tackle the multidimensional search quite effectively and the same is utilized in this work also.

The objective of proposed work is the development of an efficient and secure watermarking approach. To make the scheme free from false positive error, the principal components of watermark are inserted into the block wise singular values of DWT transformed host image. The security of watermark is enhanced by performing Arnold transform on it before the computation of principal components. The use of DWT domain and block wise SVD improves the imperceptibility and robustness of scheme. The imperceptibility and robustness are two inversely proportional parameters; along with that they are mainly dependent on the value of scaling factor. To get a good tradeoff between these two, ABC optimization of scaling factors is employed in present work.

2. Preliminaries

2.1. Singular value decomposition (SVD) and principal components

The singular value decomposition is used for numerical analysis of symmetric matrix by decomposing it; into three independent rectangular matrices. The decomposition (Eq. (1)) is done in such a way that original matrix can be formed back by multiplying the left singular matrix (U), singular matrix (S) and transpose of right singular matrix (V). A digital image can also be represented in the form of symmetric matrix and so SVD can also be performed on them using Eq. (1).

$$I = USV^T \tag{1}$$

In Eq. (1), matrix S is formed with the help of only diagonal elements, which is known as singular values. Suppose that ($r \leq n$) is the rank of the singular matrix S of the order $n \times n$ and d_1, d_2 ... d_n represents the diagonal elements of matrix S. Then, these singular values follow the property as shown in Eq. (2).

$$d_1 \geq d_2 \ldots d_r > d_{r+1} > d_{r+2} \ldots > d_n = 0 \tag{2}$$

$UU^T = I_n$ and $VV^T = I_n$ are two properties that are followed by left and right singular matrices U and V.

The elements of matrix S represent the participation of each layer of decomposed image in the final image formation [19]. Whereas the matrices U and V holds the detailed and decomposed information about the host image. The host image matrix

(I) can be restored with the help of lesser singular values but such restoration of host image I^* will deliver degradation in the visual quality of host.

The multiplication of left singular matrix with the singular matrix provides the principal components matrix as shown in Eq. (3).

$$PC = U \times S \tag{3}$$

These principal components contain the unique features of any matrix and so the same is utilized in this work for ownership checking.

2.2. Arnold transform

To insure the security of watermark even after the successful extraction as well as to compute the scramble SVD of watermark its pixels are randomized with the help of Arnold transform. Arnold transform is an efficient and powerful iterative method for providing randomization to the elements of any given array [35]. The randomization provided by Arnold transform can only be reversed back unique key/code. This prevents the unauthorized access of watermark even after successful extraction. This randomization is also done in order to increase the imperceptibility of recovered watermark image from the attacked watermarked image as scramble SVD provide better recovery [26].

Suppose we have an image of height 'h'. Then, each pixel position (x, y) of this image will get transformed into a new pixel position (x_i, y_i) after i^{th} iterations according to Eq. (4).

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 1 & m \\ n & mn+1 \end{bmatrix} \begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix} \bmod(h) \tag{4}$$

In Eq. (4), 'm' and 'n' can have any positive integer value. This pixel transformation is of periodic nature i.e., after a fixed number of iteration (period of transform) the pixels will again reach back to its original positions. This time period is dependent on the values of 'm', 'n' and 'h'.

Suppose that 'i' iterations are performed during the process of scrambling (Before embedding) and the period of transform is T then to get back the original image, $(T - i)$ iterations are need to be performed on this pre-scramble image (After extraction of watermark).

2.3. Artificial bee colony (ABC)

Karaboga [36] introduce a simple and robust population based optimization algorithm in the year 2005 and named it artificial bee colony (ABC). It is based on the smart foraging actions of a honey bee swarms. The possible solutions of given problem are represented by the food source in ABC algorithm and fitness of any solution is represented by nectar amount of a food source. There are three types of bees exist in ABC: employed bees, onlooker bees and scout bees. Employed bees represent the number of solutions in the given population size. ABC starts with an initial population of solutions of size N (food source locations) with each having a dimension D. i.e., initial solution can be represented as $X_i = \{x_{(i,1)}, x_{(i,2)} \ldots x_{(i,D)}\}$; where $i = 1, 2 \ldots N$.

After the initial distribution, search becomes a repetitive process of choosing best solution till the stopping criteria is reached. The onlooker bees choose the best food sources based on the fitness function value and information supplied by employed bees, whereas the scout bees leaves their current food source in order to search better food sources. In ABC algorithm, employed bees and onlookers and are responsible for exploitation process, whereas scouts bees take care of proper exploration. The ABC algorithm contain following steps:

1) N numbers of food sources are allotted randomly between the permissible lower $X_{\min} = (x_{(\min,1)}, \ldots x_{(\min,D)})$ and upper limit $X_{\max} = (x_{(\max,1)}, \ldots x_{(\max,D)})$ of allocation with each having dimension D using Eq. (5).

$$x_{(i,j)} = x_{(\min,j)} + \text{rand}(0, 1) \times (x_{(\max,j)} - x_{(\min,j)}) \tag{5}$$

2) Each employed bee generates a new solution on the basis of local information available to it and compares the fitness of generated solution with the parent solution. The better solution among these two is used for next iteration. The new solution Y_i is generated from current solution by using Eq. (6).

$$y_{(i,j)} = x_{(i,j)} + \Phi_{(i,j)} \times \left(x_{(i,j)} - x_{(k,j)}\right) \tag{6}$$

Here $\Phi_{(i,j)}$ is a randomly chosen number from [−1, 1] and indices $k \in (1, 2 \ldots N)$ and $j \in (1, 2 \ldots D)$ are randomly chosen in such a way that k are j remain different.

3) The employed bee shared the fitness information with the Onlooker bee. The Onlooker bee generates a probability (P_i) of nectar (fitness) amount using Eqs. (7) and (8).

$$P_i = \frac{\text{fit}_i}{\sum_{i=1}^{N} \text{fit}_i} \tag{7}$$

$$\text{fit}_i = \begin{cases} \dfrac{1}{f(X_i)+1} & \text{if} f(X_i) \geq 0 \\ 1 + \text{abs}(f(X_i)) & \text{otherwise} \end{cases} \tag{8}$$

Here $f(X_i)$ represents the value of objective function at the food location X_i. The objective function used in current study is defined by Eq. (20).

4) A random number is generated for each onlooker bee between zero and one; if P_i of food location have a greater value than the random number then step 2 is followed by that onlooker bee too.
5) If predetermined number of iterations is not able to change the food locations then such location are assumed to be abandoned. The value of predetermined number of iterations is important parameter and known as limit. In such situations, scout bee determines the new positions randomly in order to replace the abandoned positions.

Steps (1)–(5) kept on repeating till the predetermined stopping criteria (maximum iteration, minimum change) met.

3. Watermarking scheme

The watermarking scheme is divided into three stages: embedding process, extraction process and optimization of scaling factors.

3.1. Embedding process

The block diagram of proposed scheme is shown in Fig. 1 and it contains following steps:

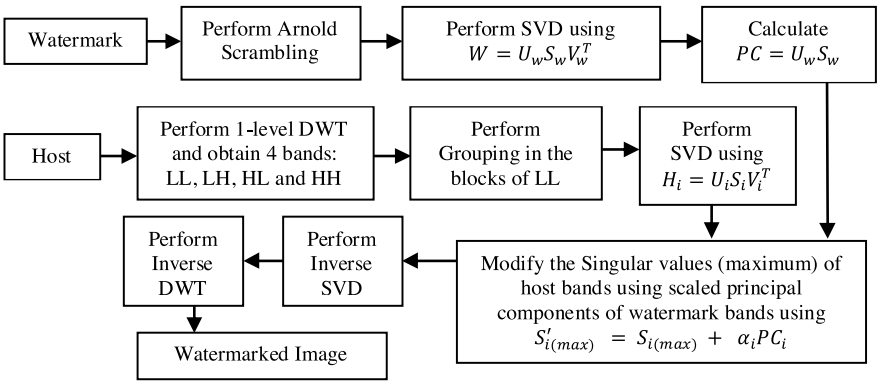

Fig. 1. Block diagram of embedding process.

1) Arnold scrambling is performed on the watermark image using a secret key (Fig. 2).
2) SVD is performed on the transformed watermark image using Eq. (9).

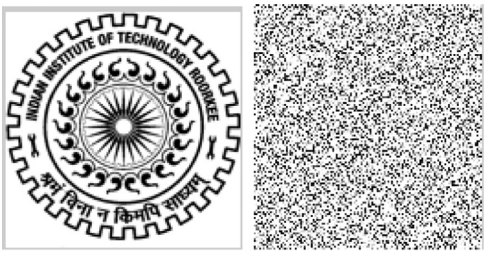

Fig. 2. (a) Watermark image and (b) Arnold transformed watermark image.

$$W = U_w S_w V_w^T \tag{9}$$

3) The principal comments are computed with the help of Eq. (10).

$$\mathrm{PC} = U_w S_w \tag{10}$$

4) DWT (1-level) is performed on the host image (H) and four bands are obtained as LL, HL, LH and HH.
5) The LL band is divided into small blocks. The block size should be in accordance with the number of principal comments. For example: if host is $M \times M$ and watermark is $N \times N$ then block size will be $n \times n$, where $n = M/2N$.
6) Block wise SVD is performed on the host image using Eq. (11).

$$H_{i(\mathrm{LL})} = U_i S_i V_i^T \tag{11}$$

7) The largest singular value of each block is modified with the help of Eq. (12).

$$S'_{i(\max)} = S_{i(\max)} + \alpha_i \mathrm{PC}_i \tag{12}$$

Here 'i' represents the number of blocks and α_i represents the scaling factor for each block.

8) Inverse SVD is performed on each block with using new singular values.
9) Inverse DWT is performed in order to get the watermarked image (H_w).

3.2. Extraction process

The attacked watermarked image is represented by H_w^*. For proper extraction of watermark, we need original host image and matrix V_w^T. Firstly, DWT is performed on the watermarked image to obtain the LL sub-band of watermarked image $H_{w(LL)}^*$. The extraction process is performed as shown Fig. 3 and describe in following steps:

1) DWT (1-level) is performed on the possibly distorted host image (H_w^*) and four bands are obtained as LL^*, LH^*, HL^* and HH^*.
2) The LL^* band is divided into small blocks. Block wise SVD is performed on the host image using Eq. (13).

$$H_{i(LL^*)} = U_i^* S_i^* V_i^{T*} \tag{13}$$

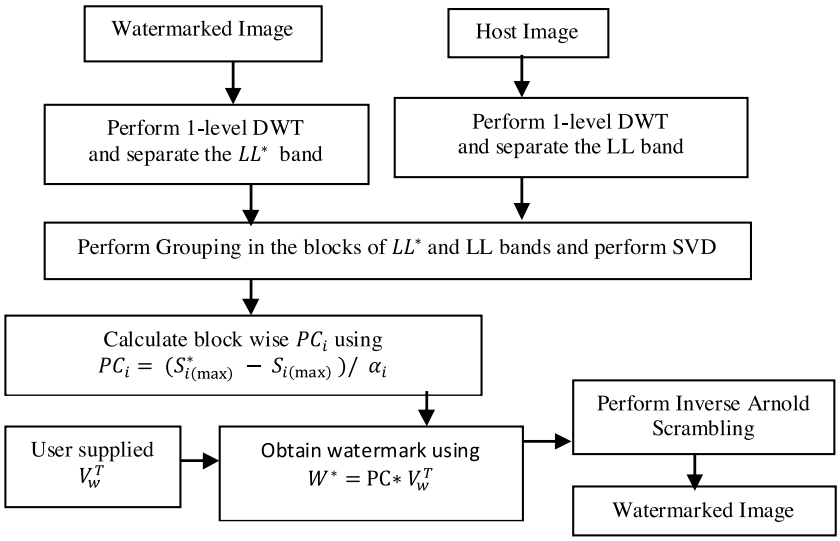

Fig. 3. Block diagram of extraction process.

3) The largest singular value of watermarked block and original host image are used to find out the principal components using Eq. (14).

Table 1
Control parameters of ABC algorithm.

Swarm size	20
Limit	25
Number of onlookers bee	50% of the swarm
Number of employed bee	50% of the swarm
Number of scout bee	Changeable

$$\mathrm{PC}_i = \frac{\left(S^*_{i(\max)} - S_{i(\max)}\right)}{\alpha_i} \tag{14}$$

4) The watermark is extracted using PC and user supplied V_w^T as shown in Eq. (15).

$$\mathrm{W}* = \mathrm{PC} \times V_w^T \tag{15}$$

5) Inverse Arnold transform is performed in order to obtained the original watermark

3.3. Optimization of scaling factors using ABC

It can easily be seen from the embedding (Section 3.1) and extraction (Section 3.2) process that the imperceptibility and robustness of proposed scheme is greatly dependent on the value of scaling factor α. A small value of α, provides a better imperceptibility at the cost of poor robustness of scheme; whereas a high value of α leads to a lower imperceptibility but provides a good robustness. So there is a need to find out an optimal value of scaling factor (α), which provides a balance between imperceptibility and robustness for each embedding in singular values. Imperceptibility and robustness are defined as below:

$$\text{Imperceptibility} = \text{correlation}\,(H, H_w) \tag{16}$$

$$\text{Robustness} = \text{correlation}\,(W, W^*) \tag{17}$$

and correlation is defined as:

$$\text{Correlation}\,(X, X^*) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\overline{X_{(i,j)} X \,\mathrm{OR}\, X^*_{(i,j)}}}{n \times n} \tag{18}$$

Here H is host image, H_w is watermarked image, W is original watermark, W^* is extracted watermark and $n \times n$ is the size of host image. Suppose that N types of attacks are performed on the watermarked image average robustness of scheme can be written as:

$$\text{Robustness}_{\text{average}} = \frac{\sum_{i=1}^{N}\text{correlation}(W, W_i^*)}{N} \tag{19}$$

The objective of proposed work is to maximize the imperceptibility and robustness. So the following fitness function (Error) is taken for minimization for fixed range of scaling factors.

$$\text{Error} = \frac{1}{\text{Robustness}_{\text{average}}} - \text{Imperceptibility} \tag{20}$$

The above shown fitness function is a multi-dimensional search; as the size of scaling factors is now considered as a vector of size 64 for search process. So the search can not be visualize graphically and needs special tool like ABC to search the optimal values of scaling factors [α].

A bounded initialization of random swarms is done in the range of 0.25–1.0. The range of initialization is chosen based on the previous research work done in the field of image watermarking. 100 generations are used in the ABC optimization. The other controlling parameters are shown in Table 1. The block diagram of optimization process is shown in Fig. 4.

4. Results and discussions

In this study, five images (gray scaled) 'Lena', 'Bird', 'Building', 'Baboon' and 'Flower' are used. The size of all the hosts images are 512×512. Two different gray scaled images 'W-1' and 'W-2' of size 64×64 are used as the watermark images. The host and watermark images are shown in Fig. 5. PSNR (Eq. (21)) and Normalized cross correlation (Eq. (18)) are used to compare the similarity of watermarked images with host images and extracted watermarks with original watermarks respectively.

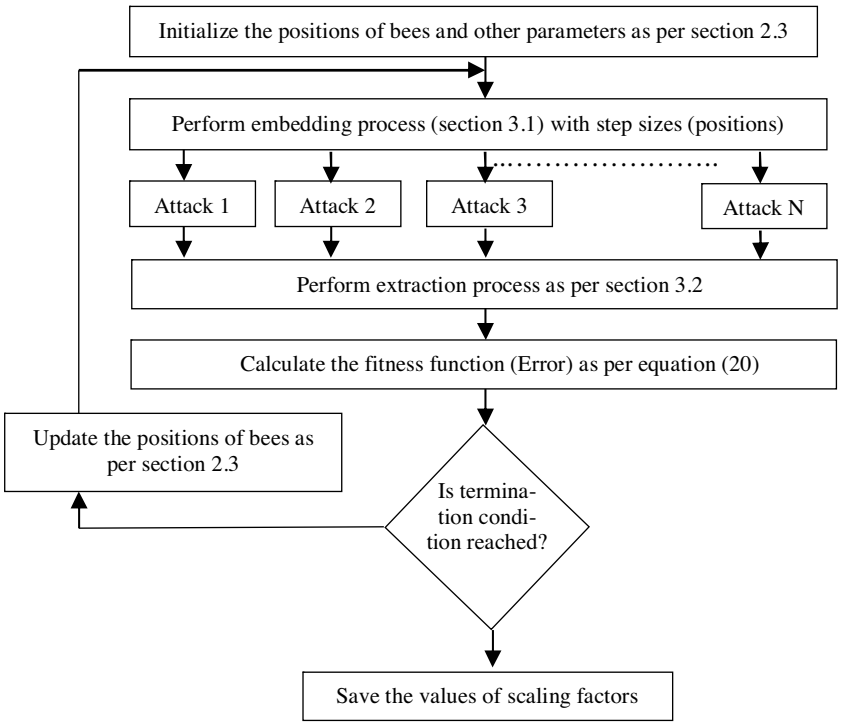

Fig. 4. Block diagram of optimization process.

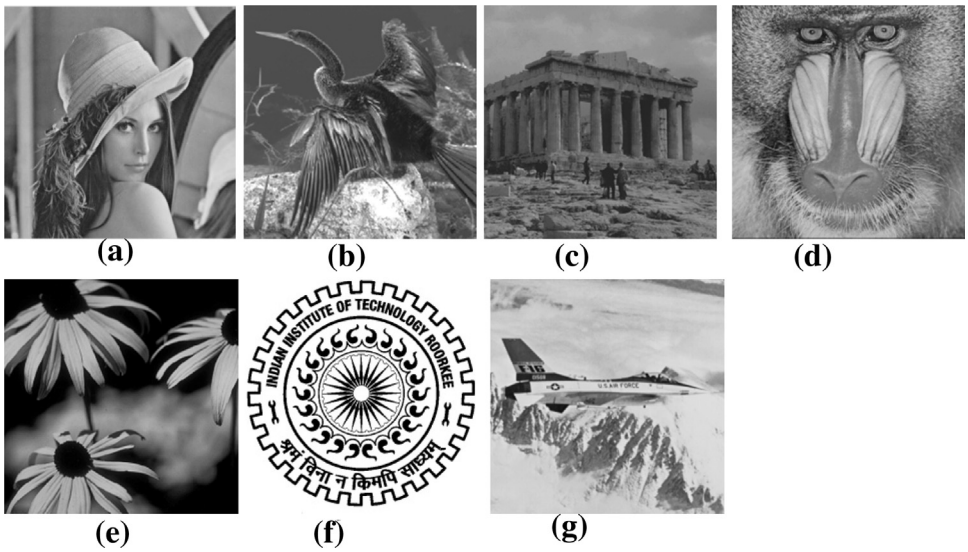

Fig. 5. Host images and watermarks: (a) Lena (b) Bird (c) Building (d) Baboon (e) Flower (f) W-1 (g) W-2.

Suppose the size of host image (X) and watermarked image (X^*) is $n \times n$ and they both can attain a maximum pixel value as X_{max}. Then PSNR can be defines as:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{n \times n \times (X_{max})^2}{\sum_{i=1}^{n}\sum_{i=1}^{n}(X(i,j) - X^*(i,j))^2}\right) \tag{21}$$

Table 2 is showing the feature comparison between proposed scheme and schemes of Guo and Prasetyo [26]. The main difference between Guo and Prasetyo [26] and proposed work is the use of multiple scaling factors instead of a single scaling factor. Guo and Prasetyo [26] did not perform the optimization of the scaling factors. The same is carried out in this work in order to improve the robustness and imperceptibility further.

Table 2
Feature comparison of proposed scheme with Guo and Prasetyo [26].

Description	Guo and Prasetyo [26]	Proposed
Scheme Type	Non-blind	Non-blind
Domain used	DWT + SVD	DWT + SVD
Sub bands used	LL	LL
Size of host	512×512	512×512
Insertion in	Pricipal component	Pricipal component
Size of watermark	Variable	Variable
False positive error	No	No
Scaling factor	Single (0.5)	Multiple

Table 3
PSNR comparison of proposed scheme with Guo and Prasetyo [26].

Host image	PSNR (dB)			
	Guo and Prasetyo [26]		Proposed	
	W1	W2	W1	W2
Lena	31.2769	31.1769	33.1242	33.0326
Bird	31.4367	31.8562	32.5326	32.4263
Building	30.8356	31.1244	33.2322	32.5468
Baboon	31.5436	30.8763	32.9243	32.8923
Flower	30.8634	31.6234	32.9321	32.7632

Fig. 6. Attacked watermarked images: (a) Gaussian noise ($M = 0$, var = 0.001), (b) multiplicative uniform noise 0.005 (c) gamma correction (0.8) (d) cropping 20 pixels each side, (e) median filter (3×3) (f) upscale re-size (g) downscale re-size (h) salt and pepper noise 0.01.

Table 3 is showing the PSNR value of watermarked images after different watermarks ('W1' and 'W2') are inserted. It is also providing a comparison of PSNR values of watermarked images obtained by different schemes.

The watermark images undergo different attacks. Fig. 6 is showing selected attacks on the watermarked image 'Lena' after embedding the watermark 'W1'. Fig. 7 is showing the extracted watermark 'W1' from LL band after those attacks. All the attacks are applied on all the host images with both the watermarks but here only one is shown because similar results are obtained from all.

Tables 4–6 are showing the robustness comparison (under different attacks) of proposed watermark scheme with existing schemes of Guo and Prasetyo [26]. Five host images and both the watermarks are used for this comparison.

Proposed method outperformed the method of Guo and Prasetyo [26] in term of imperceptibility and robustness because of the use of ABC optimization of scaling factors. The superior performance of proposed method can be verified from the results of Tables 4–6.

The proposed watermarking scheme's performance is tested with the other three hosts also. The scheme performs quite similar in terms of imperceptibility and robustness with other hosts too.

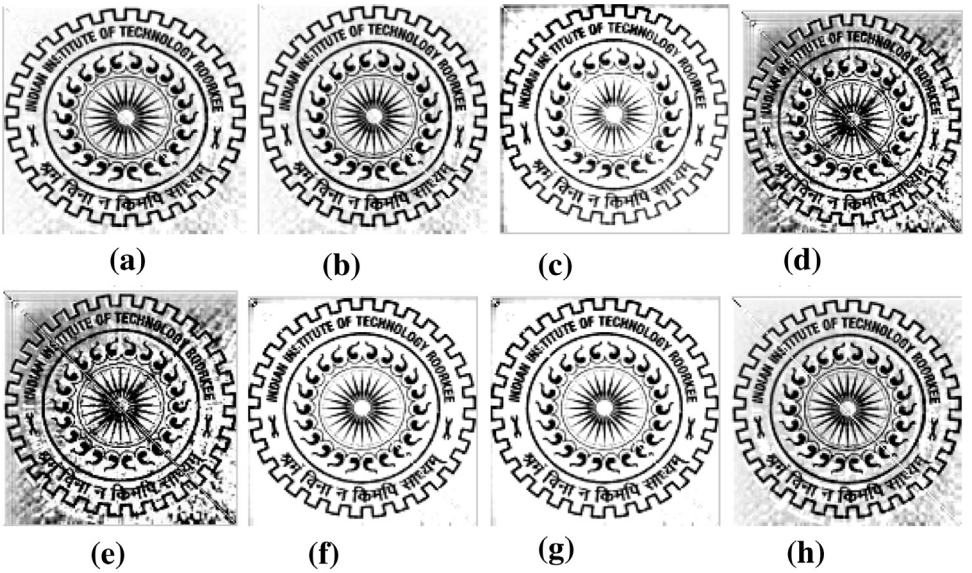

Fig. 7. Extracted watermarks W1 after: (a) Gaussian noise (M = 0, var = 0.001), (b) multiplicative uniform noise 0.005 (c) gamma correction (0.8) (d) cropping 20 pixels each side, (e) median filter (3 × 3) (f) upscale re-size (g) downscale re-size (h) salt and pepper noise 0.01.

Table 4
Robustness comparison of proposed scheme with Guo and Prasetyo [26] for host 'Lena' and watermark 'W1' and 'W2'.

Attack	Proposed (watermark W-1)	Proposed (watermark W-2)	Guo and Prasetyo [26] (watermark W-1)	Guo and Prasetyo [26] (watermark W-2)
Median filter (5 × 5)	0.8545	0.8645	0.8551	0.8446
Average filtering (3 × 3)	0.9157	0.9034	0.8989	0.8965
JPEG compression (Q = 50)	0.9880	0.9743	0.9812	0.9865
Motion blur (3, 3)	0.9678	0.9643	0.9496	0.9564
Scaling (0.5, 2)	0.9878	0.9854	0.9778	0.9754
Gaussian low pass filter (5 × 5)	0.9895	0.9843	0.9746	0.9663
Gaussian noise (M = 0 var = 0.01)	0.8272	0.8253	0.7996	0.7965
Salt & pepper (den = 0.05)	0.7256	0.7243	0.6621	0.6663
Speckle noise (0.04)	0.8132	0.8154	0.7455	0.7356
Gamma correction (0.9)	0.4634	0.4644	0.4281	0.4353
Sharpening (0.8)	0.9356	0.9354	0.8284	0.8286
Croping 20 pixels each side	0.7155	0.7143	0.6885	0.6965
Multiplicative uniform noise (0.005)	0.9247	0.9254	0.9089	0.9163
Average	0.8545	0.8523	0.8229	0.8231

Table 5
Robustness comparison of proposed scheme with Guo and Prasetyo [26] for host 'Bird' and watermark 'W1' and 'W2'.

Attack	Proposed (watermark W-1)	Proposed (watermark W-2)	Guo and Prasetyo [26] (watermark W-1)	Guo and Prasetyo [26] (watermark W-2)
Median filter (5 × 5)	0.8466	0.8632	0.8351	0.8546
Average filtering (3 × 3)	0.9245	0.9143	0.8954	0.8445
JPEG compression (Q = 50)	0.9843	0.9765	0.9864	0.9545
Motion blur (3, 3)	0.9643	0.9676	0.9344	0.9654
Scaling (0.5, 2)	0.9854	0.9854	0.9554	0.9665
Gaussian low pass filter (5 × 5)	0.9854	0.9743	0.9546	0.9654
Gaussian noise (M = 0 var = 0.01)	0.8346	0.8254	0.7456	0.7866
Salt & pepper (den = 0.05)	0.7447	0.7364	0.6435	0.6655
Speckle noise (0.04)	0.8132	0.8134	0.7656	0.7343
Gamma correction (0.9)	0.4742	0.4674	0.4546	0.4643
Sharpening (0.8)	0.9124	0.9355	0.8544	0.8245
Croping 20 pixels each side	0.7243	0.7243	0.6646	0.6843
Multiplicative uniform noise (0.005)	0.9137	0.9154	0.8546	0.8874
Average	0.8544	0.8537	0.8110	0.815

Table 6
Robustness comparison of proposed scheme with Guo and Prasetyo [26] for host 'Building' and watermark 'W1' and 'W2'.

Attack	Proposed (watermark W-1)	Proposed (watermark W-2)	Guo and Prasetyo [26] (watermark W-1)	Guo and Prasetyo [26] (watermark W-2)
Median filter (5×5)	0.8443	0.8545	0.8466	0.8346
Average filtering (3×3)	0.9334	0.9546	0.8546	0.8965
JPEG compression ($Q = 50$)	0.9435	0.9463	0.9854	0.9546
Motion blur (3, 3)	0.9433	0.9545	0.9546	0.9564
Scaling (0.5, 2)	0.9545	0.9545	0.9458	0.9754
Gaussian low pass filter (5×5)	0.9843	0.9746	0.9646	0.9546
Gaussian noise ($M = 0$ var $= 0.01$)	0.8254	0.8254	0.7964	0.7954
Salt & pepper (den = 0.05)	0.7435	0.7253	0.6421	0.6438
Speckle noise (0.04)	0.8543	0.8657	0.7645	0.7435
Gamma correction (0.9)	0.4455	0.4665	0.4264	0.4453
Sharpening (0.8)	0.9345	0.9546	0.8184	0.8246
Croping 20 pixels each side	0.7543	0.7565	0.6864	0.6545
Multiplicative uniform noise (0.005)	0.9254	0.9435	0.9189	0.9243
Average	0.8527	0.8597	0.8157	0.8156

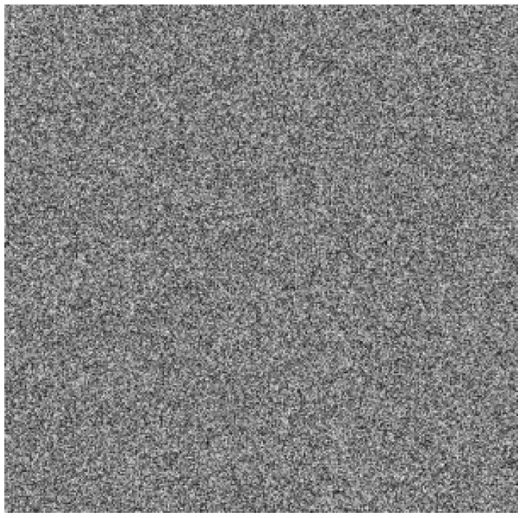

Fig. 8. Extracted watermarks from watermarked 'Lena' when matrix V_w^T is changed with wrong matrix.

4.1. False positive error checking

The proposed scheme is tested for more than thirty different false ownership claims and each time the scheme is able to detect the false claim.

False positive error take place when a wrong watermark (which is never been inserted into host) is extracted from the host image. The extraction steps are followed as describe in the Section 3.2. One can claim the false ownership by changing the user supplied matrix V_w^T. The same is checked with the proposed scheme and extracted watermark is shown in Fig. 8.

As we can see in Fig. 8, that the extracted watermark become a random noise, if you change the original V_w^T by a wrong V_w^T. So this proofs that the scheme is secured toward false positive error.

5. Conclusion

In the proposed watermarking scheme, the scaling factors were optimized with the help of artificial bee colony (ABC), which improved the robustness and imperceptibility. The principal components of watermark were inserted into the maximum singular values of each block of host image, which again helped the scheme in achieving high robustness and imperceptibility. The scheme also became free from false positive error because of the insertion of principal components. Arnold transform based encryption of watermark prevents the authentic access of it even after the successful extraction. In future, more variants of ABC along with other metaheuristic techniques will be applied to improve the performance (generation used, PSNR, cross correlation etc.) of watermarking process. Also, new insertion methodologies will to be developed in order to sustain major and deep geometrical attacks.

Acknowledgements

This work was supported by Ministry of Human Resource Development, India.

References

[1] V.M. Potdar, S. Han, E. Chang, A survey of digital image watermarking techniques, 3rd IEEE International Conference on Industrial Informatics INDIN'05 (2005) 709–716.
[2] Lin Weisi, Dacheng Tao, Janusz Kacprzyk, Zhu Li, Ebroul Izquierdo, Haohong Wang, Multimedia Analysis, Processing and Communications, vol. 346, Springer, 2011.
[3] Gary L. Friedman, The trustworthy digital camera: restoring credibility to the photographic image, IEEE Trans. Consum. Electron. 39 (4) (1993) 905–910.
[4] Shao-Hui Liu, Hong-Xun Yao, Wen Gao, Yong-Liang Liu, An image fragile watermark scheme based on chaotic image pattern and pixel-pairs, Appl. Math. Comput. 185 (2) (2007) 869–882.
[5] Adil Haouzia, Rita Noumeir, Methods for image authentication: a survey, Multimed. Tools Appl. 39 (1) (2008) 1–46.
[6] I.A. Ansari, M. Pant, SVD watermarking: particle swarm optimization of scaling factors to increase the quality of watermark, Proceedings of Fourth International Conference on Soft Computing for Problem Solving (Springer India) (2015) 205–214.
[7] I.A. Ansari, M. Pant, C.W. Ahn, SVD based fragile watermarking scheme for tamper localization and self-recovery, Int. J. Mach. Learn. Cybern. (2015) 1–15, http://dx.doi.org/10.1007/s13042-015-0455-1.
[8] Musrrat Ali, Chang Wook Ahn, Millie Pant, A robust image watermarking technique using SVD and differential evolution in DCT domain, Opt.: Int. J. Light Electron Opt. 125 (1) (2014) 428–434.
[9] Gaurav Bhatnagar, Balasubramanian Raman, A new robust reference watermarking scheme based on DWT-SVD, Comput. Stand. Interfaces 31 (5) (2009) 1002–1013.
[10] Chih-Chin Lai, Cheng-Chih Tsai, Digital image watermarking using discrete wavelet transform and singular value decomposition, IEEE Trans. Instrum. Meas. 59 (11) (2010) 3060–3063.
[11] Nasrin M. Makbol, Bee Ee Khoo, A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition, Digital Signal Process. 33 (2014) 134–147.
[12] Irshad Ahmad Ansari, Millie Pant, Chang Wook Ahn, Robust and false positive free watermarking in IWT domain using SVD and ABC, ENG. APPL. ARTIF. INTEL. 49 (2016) 114–125.
[13] Musrrat Ali, Chang Wook Ahn, Comments on "Optimized gray-scale image watermarking using DWT-SVD and Firefly Algorithm", Expert Syst. Appl. 42 (5) (2015) 2392–2394.
[14] Huo-Chong Ling, Raphael C.-W. Phan, Swee-Huay Heng, Comment on robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition, AEU: Int. J. Electron. Commun. 67 (10) (2013) 894–897.
[15] Anurag Mishra, Charu Agarwal, Arpita Sharma, Punam Bedi, Optimized gray-scale image watermarking using DWT–SVD and Firefly Algorithm, Expert Syst. Appl. 41 (17) (2014) 7858–7867.
[16] Zhen Li, Kim-Hui Yap, Bai-Ying Lei, A new blind robust image watermarking scheme in SVD-DCT composite domain, 18th IEEE International Conference on Image Processing (ICIP) (2011) 2757–2760.
[17] Ray-Shine Run, Shi-Jinn Horng, Jui-Lin Lai, Tzong-Wang Kao, Rong-Jian Chen, An improved SVD-based watermarking technique for copyright protection, Expert Syst. Appl. 39 (1) (2012) 673–689.
[18] Y. Tian, T. Tan, Y. Wang, Y. Fang, Do singular values contain adequate information for face recognition? Pattern Recognit. 36 (3) (2003) 649–655.
[19] Emir Ganic, Ahmet M. Eskicioglu, Robust embedding of visual watermarks using discrete wavelet transform and singular value decomposition, J. Electron. Imaging 14 (4) (2005) 043004.
[20] Samira Lagzian, Mohsen Soryani, Mahmood Fathy, A new robust watermarking scheme based on RDWT-SVD, Int. J. Intell. Inf. Process. 2 (1) (2011) 22–29.
[21] Saeed Rastegar, Fateme Namazi, Khashayar Yaghmaie, Amir Aliabadian, Hybrid watermarking algorithm based on singular value decomposition and radon transform, AEU: Int. J. Electron. Commun. 65 (7) (2011) 658–663.
[22] Khaled Loukhaoukha, Jean-Yves Chouinard, Mohamed Haj Taieb, Optimal image watermarking algorithm based on LWT-SVD via multi-objective ant colony optimization, J. Inf. Hiding Multimed. Signal Process. 2 (4) (2011) 303–319.
[23] U.M. Gokhale, Y.V. Joshi, A semi fragile watermarking algorithm based on SVD-IWT for image authentication, Int. J. Adv. Res. Comput. Commun. Eng. 1 (4) (2012).
[24] Akshya Kumar Gupta, Mehul S. Raval, A robust and secure watermarking scheme based on singular values replacement, Sadhana 37 (4) (2012) 425–440.
[25] Ray-Shine Run, Shi-Jinn Horng, Jui-Lin Lai, Tzong-Wang Kao, Rong-Jian Chen, An improved SVD-based watermarking technique for copyright protection, Expert Syst. Appl. 39 (1) (2012) 673–689.
[26] J.M. Guo, H. Prasetyo, False-positive-free SVD-based image watermarking, J. Vis. Commun. Image Represent. 25 (5) (2014) 1149–1163.
[27] K. Hanbay, M.F. Talu, Segmentation of SAR images using improved artificial bee colony algorithm and neutrosophic set, Appl. Soft Comput. 21 (2014) 433–443.
[28] R. Ramanathan, K. Kalaiarasi, D. Prabha, Improved wavelet based compression with adaptive lifting scheme using artificial bee colony algorithm, Int. J. Adv. Res. Comput. Eng. Technol. 2 (4) (2013) pp–1549.
[29] Irshad Ahmad Ansari, Millie Pant, Ferrante Neri, Analysis of gray scale watermark in RGB host using SVD and PSO, IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP) (2014) 1–7.
[30] Rainer Storn, Kenneth Price, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (4) (1997) 341–359.
[31] D. Karaboga, B. Akay, A comparative study of Artificial Bee Colony algorithm, Appl. Math. Comput. 214 (2009) 108–132.
[32] B. Li, Y. Li, L. Gong, Protein secondary structure optimization using an improved artificial bee colony algorithm based on AB off-lattice model, Eng. Appl. Artif. Intell. 27 (2014) 70–79.
[33] A. Draa, A. Bouaziz, An artificial bee colony algorithm for image contrast enhancement, Swarm Evol. Comput. 16 (2014) 69–84.
[34] F.J. Rodriguez, M. Lozano, C. García-Martínez, J.D. González-Barrera, An artificial bee colony algorithm for the maximally diverse grouping problem, Inf. Sci. (NY) 230 (2013) 183–196.
[35] Lingling Wu, Jianwei Zhang, Weitao Deng, Dongyan He, Arnold transformation algorithm and anti-arnold transformation algorithm, 1st IEEE International Conference on Information Science and Engineering (2009) 1164–1167.
[36] Dervis Karaboga, An idea based on honey bee swarm for numerical optimization Technical Report-tr06, vol. 200, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.