Engineering Applications of Artificial Intelligence 49 (2016) 114–125

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Robust and false positive free watermarking in IWT domain using SVD and ABC

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
2 December 2015
Accepted 8 December 2015

Keywords:
Secure watermarking
False positive free
SVD
Robust watermarking
Artificial bee colony (ABC)
Integer wavelet transform

ABSTRACT

Watermarking is used to protect the copyrighted materials from being misused and help us to know the lawful ownership. The security of any watermarking scheme is always a prime concern for the developer. In this work, the robustness and security issue of IWT (integer wavelet transform) and SVD (singular value decomposition) based watermarking is explored. Generally, SVD based watermarking techniques suffer with an issue of false positive problem. This leads to even authenticating the wrong owner. We are proposing a novel solution to this false positive problem; that arises in SVD based approach. Firstly, IWT is employed on the host image and then SVD is performed on this transformed host. The properties of IWT and SVD help in achieving high value of robustness. Singular values are used for the watermark embedding. In order to further improve the quality of watermarking, the optimization of scaling factor (mixing ratio) is performed with the help of artificial bee colony (ABC) algorithm. A comparison with other schemes is performed to show the superiority of proposed scheme.

1. Introduction

Development of computer based communication makes the data and image sharing very easy among people but this also increases the security threats towards the privacy of individuals. Now a days very easy to modify the image with powerful image processing tools and claim the ownership (Potdar et al., 2005) as well as produce harmful content for others (Lin et al., 2011). So there is a need of a technique that can protect/find out manipulations in images. These days, image watermarking becomes quite popular for image protection. It inserts the visible/invisible watermark in the host image, which can be used to claim the ownership afterwards as well as it provides methods to find out the manipulations in the image (Friedman, 1993). There are many watermarking schemes (Liu et al., 2007; Ansari and Pant, 2015; Haouzia and Noumeir, 2008; Rawat and Raman, 2011; Ansari et al., 2015; Ali et al., 2014; Bhatnagar and Raman, 2009), which are proposed by different researches but the sole purpose of all the schemes was to authenticate the host image correctly.

Watermark insertion can be classified into two classes: pixel wise insertion (Liu et al., 2007) and transformed domain insertion (Ansari and Pant, 2015). Transformed domains are found to be more robust as compared to the pixel wise insertion (Haouzia and Noumeir, 2008). Pixels wise insertion of watermark leads to a fragile watermarking. It has its own advantage, like finding out the distorted area (Rawat and Raman, 2011) and reproduction of original host image from the distorted one (Ansari et al., 2015). The robust watermarking is mainly used to find out the ownership of watermarked image so the watermark is inserted in such a way that it did not get destroyed even after multiple attacks. DCT (Discrete cosine transform) (Ali et al., 2014) and DWT (Discrete wavelet transform) (Bhatnagar and Raman, 2009) are the most common transform domains used for image watermarking. In DWT based watermarking, the host image is transformed into wavelet domain and then watermark is inserted into the coefficients of few (Lai and Tsai, 2010)/all (Makbol and Khoo, 2014) of the four frequency sub-bands i.e. LL (low–low), LH (low–high), HL (high–low), HH (high–high).

The improvement of robustness towards attacks without degrading the visual quality of the watermarked image is one of the basic ideas of all robust watermarking schemes. One other important parameter in the robust image watermarking is the watermark payload/capacity. The maximum amount of data (watermark) that can be embedded to the host image is known as the capacity of scheme. In many cases, the watermarking scheme needs to have a higher capacity. So that sufficient amount of ownership/other relevant information can be embedded. A low capacity design makes the scheme more robust and transparent (Potdar et al., 2005; Haouzia and Noumeir, 2008). SVD based

* Corresponding author at: Department of Computer Engineering, Sungkyunkwan University, Suwon-440746, Republic of Korea. Tel.: +82 31 299 4588.
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.engappai.2015.12.004
0952-1976/© 2015 Elsevier Ltd. All rights reserved.

watermarking are known for its high capacity. In SVD based watermarking, changes in smaller singular values have almost no effect on the imperceptibility/robustness. So, a good amount of scheme's capacity needs to be sacrificed for a slight improvement in imperceptibility/robustness.

Many SVD and DWT based hybrid schemes (Bhatnagar and Raman, 2009; Lai and Tsai, 2010; Makbol and Khoo, 2014; Ali and Ahn, 2014) are proposed in the past but most of the SVD schemes suffer with a false positive problem (Ali and Ahn, 2015; Ling et al., 2013). The SVD break down the transformed image in three vectors U, S and V as shown in Eq. (1).

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

The watermark can be inserted into any one of the three matrices i.e. U, S and V but singular values are the most common used matrix because of its robust nature towards attacks (Mishra et al., 2014; Li et al., 2011; Run et al., 2012). Moreover, a small change in the singular values does not affect the visual quality of image much as most of the information of image is carried by matrices U and V (Tian et al., 2003). So, the insertion in S matrix provides best visual quality. On contrary to this, false positive error also comes into picture; when watermark is inserted into the singular values (Ali and Ahn, 2015; Ling et al., 2013). Attacker can change the matrix U and V with their desired matrix and can extract new watermark (which is not even inserted) to claim the false ownership regardless of singular matrix (S) (Ali and Ahn, 2015; Ling et al., 2013). Researches (Lai and Tsai, 2010; Makbol and Khoo, 2014; Ali and Ahn, 2014) proposed the change in the singular values (S) by the help of scaling factor (α) and watermark (W) as per Eq. (2).

$$S_{new} = S_{old} + \alpha W \tag{2}$$

Following schemes (Ganic and Eskicioglu, 2005; Lagzian et al., 2011; Rastegar et al., 2011) involved same embedding step except the different wavelet transformations. Bhatnagar and Raman (2009) used DWT, Lagzian et al. (2011) used RDWT (redundant discrete wavelet transform) and FRAT (Finite Random Transform) is used by Rastegar et al. (2011). Different transforms are used to improve the watermarking scheme performance. Even though they used different transforms, all the schemes suffered an issue of false positive error because the main reason of false positive error is the dependence of extracted watermark on the user supplied information.

In order to solve the problem of false detection, Loukhaoukha et al. (2011) suggests applying one way hash function on U and V matrices. This provided two hashing values HU and HV that can be stored privately, and then these values can be used to authenticate the U and V matrices during extraction. Loukhaoukha et al. (2011) proposed one more solution and that was encrypting the watermark prior to the insertion into host image. The same solution is also used by Gokhale and Joshi (2012). Signature based authentication for both U and V matrices prior to the extraction is proposed by Gupta et al. (2012). Gupta et al. (2012) proposed the embedding of principal component of watermark into the singular values and use PSO (Particle swarm optimization) to get optimal scaling factors. Ali and Ahn (2014) also proposed principal component based watermarking scheme in the domain of DWT. Watermark insertion in principal components definitely make the scheme free from false positive free error but this also leads to poor imperceptibility and robustness. Ali and Ahn (2014) tried to improve the imperceptibility and robustness of their scheme by choosing DE (Differential Evolution) based optimal scaling factors for watermark insertion. A robust and secured scheme is proposed by Makbol and Khoo (2014) for image watermarking. They (Ali and Ahn, 2014) used the domain of integer wavelet transform (IWT) to get rid of rounding off errors during inverse transform. They insert the watermark in the singular values of all the bands after performing SVD on them. A digital signature was inserted into the watermarked image in order to verify the authenticity of user supplied singular matrices. Makbol and Khoo (2014) claimed their scheme false positive free, but their scheme generate errors in the signature matching step. Moreover, they did not use optimal value of scaling factors and this provides a scope for improvement in imperceptibility as well as robustness

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 (Hanbay and Talu, 2014), compression (Ramanathan et al., 2013), watermarking (Ansari et al., 2014) etc. PSO (Ansari et al., 2014), DE (Storn and Price, 1997) and ABC (Hanbay and Talu, 2014; Ramanathan et al., 2013) 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 (Karaboga and Akay, 2009). Due to its simple and fast implementation along with robust performance, it is being used widely for imaging as well as other applications (Hanbay and Talu, 2014; Ramanathan et al., 2013; Li et al., 2014; Draa and Bouaziz, 2014; Rodriguez et al., 2013). The ABC is already been implemented into imaging optimization areas like image segmentation (Hanbay and Talu, 2014), compression (Ramanathan et al., 2013) and enhancement (Draa and Bouaziz, 2014) and it provided very good results. ABC is known to tackle the multidimensional search quite effectively and the same is utilized in this work also.

In this paper, we are proposing a solution to false positive problem in IWT domain to overcome the problem of rounding off errors during inverse transform. At the same time, we are also improving the robustness and imperceptibility of proposed scheme by optimizing the scaling parameter 'α' (as shown in Eq. (2)) with the help of Artificial Bee Colony (ABC). Rest of the paper is organized as follows: in Section 2, basics of IWT, ABC and false positive solution are discussed. Section 3 talks about the proposed watermarking scheme. Result and discussions are provided in section four and finally section five provides a brief conclusion of proposed work.

2. Preliminaries

2.1. Singular value decomposition (SVD)

Singular value decomposition is a technique of linear algebra to diagonalize the symmetric matrix. A digital image is also a matrix of integer numbers so SVD can be performed on digital images straight away. SVD decomposes the given matrix into three parts i.e. left singular matrix U, right singular matrix V and singular matrix S (Eq. (1)).

The matrix S contains only diagonal element, which is known as singular values. The matrix S contains the singular values in descending order. The matrices U and V carry the decomposed and detailed information about the image. Suppose A is the rectangular matrix of the order n × n then matrix S can have maximum n diagonal elements. These elements (S) basically represent the participation of each layer of decomposed image in the final image formation (Tian et al., 2003). The parent matrix (A) can be regenerated with the lesser elements of matrix S but such regeneration of matrix A^* will lead to degradation in its quality.

The matrices U and V follow the properties $UU^T = I_n$ and $VV^T = I_n$. The diagonal values of diagonal matrix S follow the property shown in Eq. (3).

$$d_1 \geq d_2 \ldots d_r > d_{r+1} > d_{r+2} \ldots > d_n = 0 \tag{3}$$

Here, $(r \leq n)$ represents the rank of the matrix S and $d_1, d_2 \ldots d_n$ are diagonal elements of matrix S.

2.2. Integer wavelet transform

The lifting wavelet transform was proposed by Sweldens (1998) and it provides the basic structure for the adaptive wavelets. Daubechies and Sweldens (1998) proved that all the classical wavelet could be realized using lifting scheme. Lifting scheme has three basic steps; splitting, prediction and updating (Jia et al., 2010). Classical wavelet transforms perform the transformation with the assumption that inputs are floating point. Though in reality, many applications (image compression and image watermarking) involve only integer inputs. This conversion from integer to floating point and floating point to integer make them loose their perfect reconstruction property and lead to rounding off error. The lifting scheme can be used to implement the integer to integer wavelet transform (IWT) in order to remove these rounding off errors (Su et al., 2012). IWT conations reversibility property and so, it can be used for perfect reconstruction. In order to provide a speedy and efficient implementation, IWT can be implemented by the help of three basics steps of lifting operation – split, predict and update. Fig. 1 is showing the block diagram representation of lifting operation and used steps are explained as follows:

Split: in this step, original signal (C) is divided into even (C_e) and odd (C_o) parts (Jia et al., 2010; Su et al., 2012).

Predict: in this step, the odd sequence is predicted by the help of even sequence based on the predictor, which work on the correlation between the data. The difference between the predicted and actual value of odd sequence is used as new odd sequence for next level of IWT implementation (Jia et al., 2010; Su et al., 2012).

Update: in this step, a new even sample are generated by combining the predicted odd sample and original even sample based on a updater, which ensure same feature on new sample (Jia et al., 2010; Su et al., 2012).

The predicted odd sample is viewed as high frequency component and generated even sample as low frequency coefficient. The low frequency component can further be transformed using same procedure.

The Inverse lifting operation can be implemented by changing the split block by merge operation as shown in Fig. 2.

2.3. Artificial Bee Colony (ABC)

Karaboga (2005) 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.

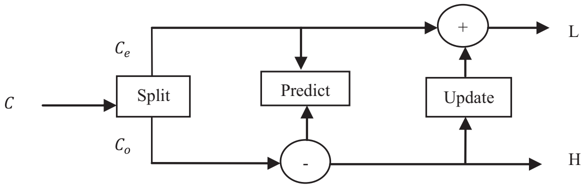

Fig. 1. Block diagram of lifting operation.

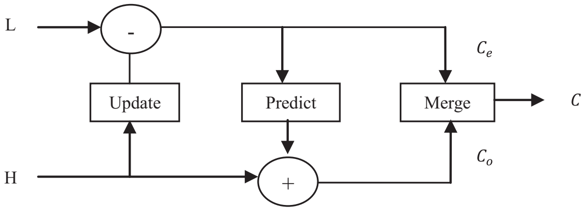

Fig. 2. Block diagram of Inverse lifting operation.

There are three types of bees existing 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 leave their current food source in order to search better food sources. In ABC algorithm, employed bees and onlookers are responsible for exploitation process, whereas scouts bees take care of proper exploration. The ABC algorithm contains 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. (4).

$$x_{(i,j)} = x_{(min,j)} + rand(0,1) \times (x_{(max,j)} - x_{(min,j)}) \tag{4}$$

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. (5).

$$y_{(i,j)} = x_{(i,j)} + \Phi_{(i,j)} \times (x_{(i,j)} - x_{(k,j)}) \tag{5}$$

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 and 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. (6) and (7).

$$P_i = \frac{\text{fit}_i}{\sum_{i=1}^{N} \text{fit}_i} \tag{6}$$

$$\text{fit}_i = \begin{cases} \frac{1}{f(X_i)+1} & \text{if} \quad f(X_i) \geq 0 \\ 1 + abs(f(X_i)) & \text{otherwise} \end{cases} \tag{7}$$

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. (21).

1) 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.

2) If predetermined number of iterations is not able to change the food locations then such locations are assumed to be abandoned. The value of predetermined number of iterations is an important parameter and is known as limit. In such situations,

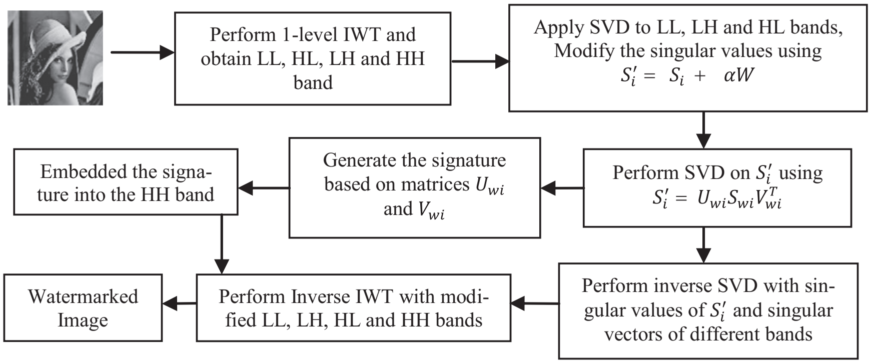
Perform 1-level IWT and obtain LL, HL, LH and HH band
Apply SVD to LL, LH and HL bands, Modify the singular values using $S_i' = S_i + \alpha W$
Embedded the signature into the HH band
Generate the signature based on matrices U_{wi} and V_{wi}
Perform SVD on S_i' using $S_i' = U_{wi} S_{wi} V_{wi}^T$
Watermarked Image
Perform Inverse IWT with modified LL, LH, HL and HH bands
Perform inverse SVD with singular values of S_i' and singular vectors of different bands

Fig. 3. Block diagram of embedding process.

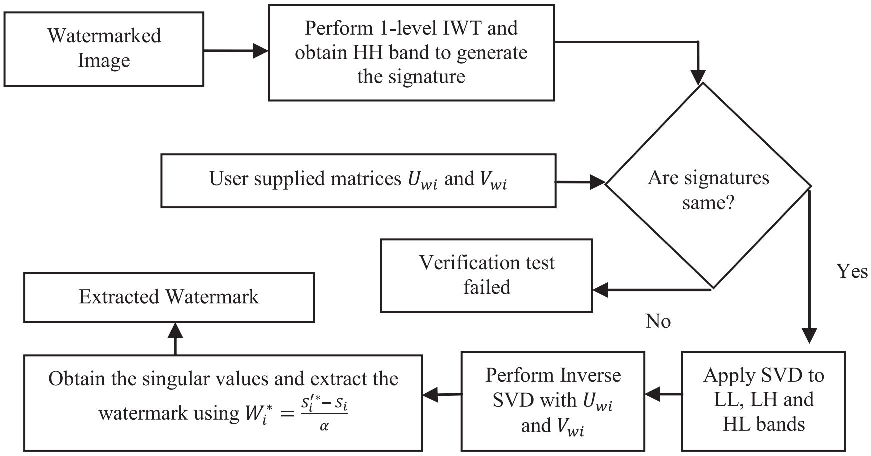
Watermarked Image
Perform 1-level IWT and obtain HH band to generate the signature
User supplied matrices U_{wi} and V_{wi}
Are signatures same?
Yes
Verification test failed
No
Extracted Watermark
Obtain the singular values and extract the watermark using $W_i^* = \frac{S_i'^* - S_i}{\alpha}$
Perform Inverse SVD with U_{wi} and V_{wi}
Apply SVD to LL, LH and HL bands

Fig. 4. Block diagram of extraction process.

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.

2.4. False positive free solution

In order to get rid of false positive error, extra information is generated based on the matrices U_{wi} and V_{wi}. So that authentication of these matrices can be done during extraction process. This signature (generated information) is embedded into the HH band of the host image. The generation and embedding of signature contains following steps:

1) Sum all the columns of U_{wi} and V_{wi} matrices so that it gets converted to 1D matrix from 2D.
2) Hash transformed U_{wi} and V_{wi} vectors using secure hash algorithm (SHA-1).

$$Hashed_U = Hashing_{SHA-1}(U_{wi}) \tag{8}$$

$$Hashed_V = Hashing_{SHA-1}(V_{wi}) \tag{9}$$

3) Change the $Hashed_U$ and $Hashed_V$ into their equivalent binary value representation and perform XOR operation between them to obtain ‘A’.
4) Select a secret binary string ‘B’ and perform XOR between A and B. The first 16 bits of result is used as signature bits.
5) Transformed the host image into wavelet domain and separated the HH band.
6) Perform DCT (discrete cosine transforms) on this band and select 16 mid-band frequency components (separate 16 components for LL, LH and HL bands).
7) Convert the coefficients to binary equivalent. Change the nth bit of each coefficient with signature bits and convert it back to decimal equivalent.
8) Apply IDCT (Inverse DCT) to obtain the modified HH band.

3. Watermarking scheme

3.1. Embedding process

The block diagram of proposed scheme is shown in Fig. 3 and it contains following steps:

1) The host image H is transformed into wavelet domain (LL, HL, LH and HH band) using IWT. LL, LH and HL bands are selected for embedding.
2) SVD (singular value decomposition) is performed on each band I_i $(i=$ LL, LH and HL$)$ using Eq. (10).

$$I_i = U_i S_i V_i^T \tag{10}$$

3) Modification of the singular values of each band is done according to Eq. (11).

$$S_i' = S_i + \alpha W \tag{11}$$

where, α is the scaling factor and W is the watermark.

4) Another SVD (singular value decomposition) is performed on the S_i' using Eq. (12).

$$S_i' = U_{wi} S_{wi} V_{wi}^T \tag{12}$$

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

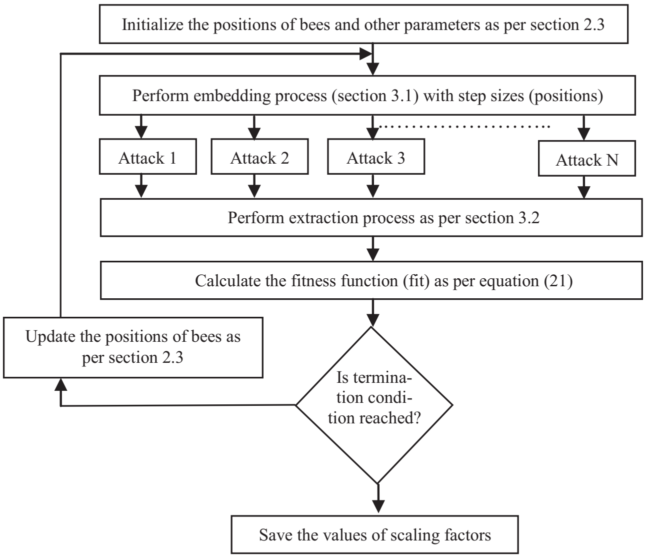
Initialize the positions of bees and other parameters as per section 2.3
Perform embedding process (section 3.1) with step sizes (positions)
Attack 1
Attack 2
Attack 3
Attack N
Perform extraction process as per section 3.2
Calculate the fitness function (fit) as per equation (21)
Update the positions of bees as per section 2.3
Is termination condition reached?
Save the values of scaling factors

Fig. 5. Block diagram of optimization process.

5) The matrix S_{wi} along with U_i and V_i used to create the watermarked sub bands I_{wi} for each block as shown in Eq. (13).

$$I_{wi} = U_i S_{wi} V_i^T \tag{13}$$

6) The signature information is generated for the LL, LH and HL bands based on the matrices U_{wi} and V_{wi} as per Section 2.4 (step (1) to step (4)) to make the scheme free from false positive error.
7) The signature bits are inserted into host image (HH band) as per Section 2.4 (step (5) to step (8)).
8) The watermarked sub bands I_{wi} along with modified HH band are used for Inverse IWT to generate watermarked image H_w.

3.2. Extraction process

For extraction of watermark (Fig. 4) from the distorted watermarked image H_w^*, we need three more matrices U_{wi}, S_i and V_{wi}. Firstly, IWT is performed on the watermarked image to obtain I_w^* then following steps performed:

1) Separate the HH band and perform the perform DCT on it. Select 16 mid-band frequency components (different for LL, LH and HL bands) that contain signature information.

Table 2
Feature comparison of proposed scheme with other schemes.

Description	Ali and Ahn (2014)	Makbol and Khoo (2014)	Proposed
Scheme type	Non-blind	Non-blind	Non-blind
Domain used	DWT+SVD	IWT+SVD	IWT+SVD
Sub bands used	All	All	LL+LH+HL
Size of host	512 × 512	512 × 512	512 × 512
Insertion in	Principal component	Singular values	Singular values
Size of watermark	256 × 256	256 × 256	256 × 256
False positive error	No	No	No
Scaling factor	Optimized with DE	0.05 (LL), 0.005 (rest)	Optimized with ABC

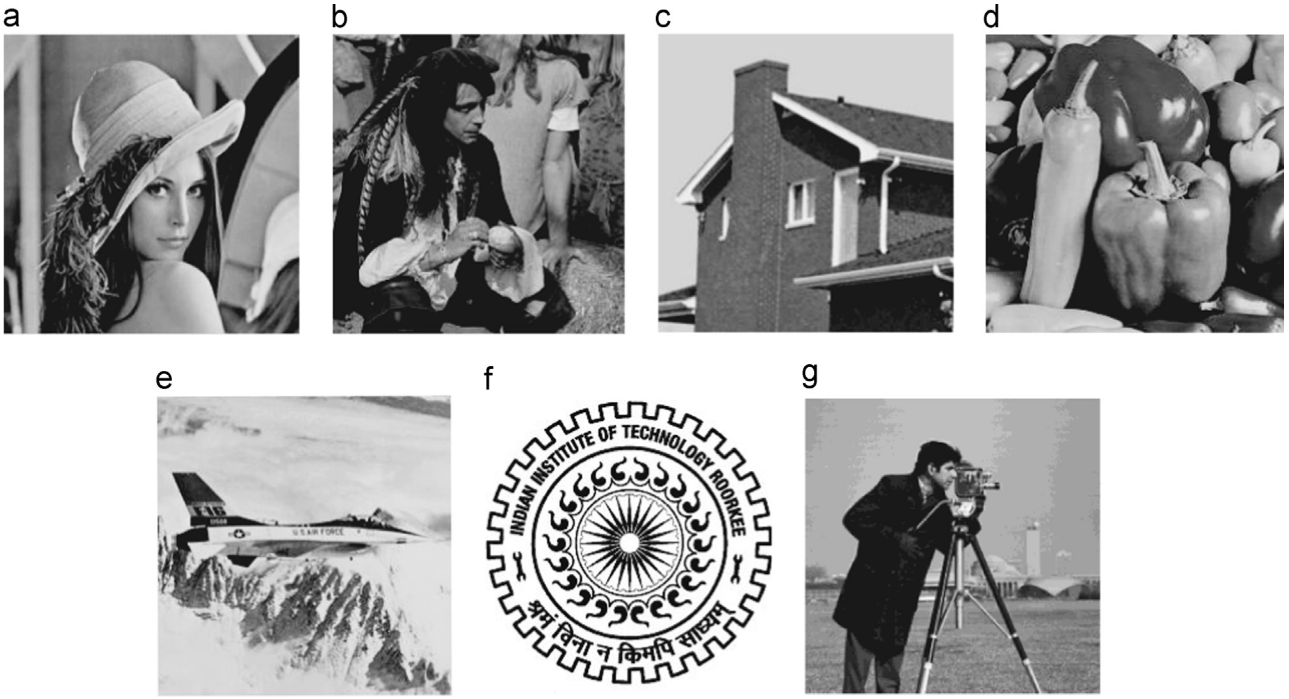
a
b
c
d
e
f
g
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
श्रमं विना न किमपि साध्यम्

Fig. 6. Host images and watermarks: (a) Lena, (b) Man, (c) House, (d) Pepper, (e) Plane, (f) $W1$ and (g) $W2$.

2) Covert the coefficients to binary equivalent. Read the nth bit of each coefficient and regenerate the signature back.
3) Generate the signature bits for the user supplied matrices U_{wi} and V_{wi} as per Section 2.4 (step (1)–step (4)).
4) Compare the signature bits generated from step (2) and step (3), if they are same then go to step (6).
5) Verification test failed. Go to step (9)
6) Perform SVD on the LL, LH and HL bands of watermarked image using Eq. (14).

$$I_i^* = U_i^* S_{wi}^* V_i^{*T} \tag{14}$$

7) An approximated version of S′for bands LL, LH and HL is retrieved back using Eq. (15).

$$S_i'^* = U_{wi} S_{wi}^* V_{wi}^T \tag{15}$$

8) An approximated version of watermark is generated for LL, LH and HL bands using Eq. (16).

$$W_i^* = \frac{S_i'^* - S_i}{\alpha} \tag{16}$$

9) Extraction process finished and result displayed.

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:

$$\text{Imperceptibility} = \text{correlation}\ (H, H_w) \tag{17}$$

$$\text{Robustness} = \text{correlation}\ (W, W^*) \tag{18}$$

and correlation is defined as:

$$\text{correlation}(X, X^*) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} \overline{X_{(i,j)} \text{XOR } X^*_{(i,j)}}}{n \times n} \tag{19}$$

here H is the host image, H_w is the watermarked image, W is the original watermark, W^* is the extracted watermark and $n \times n$ is the size of host image. Suppose that N type of attacks are

Table 3
PSNR comparison of proposed scheme with existing schemes using different watermarks.

Host image	PSNR (dB)					
	Ali and Ahn (2014)		Makbol and Khoo (2014)		Proposed	
	$W1$	$W2$	$W1$	$W2$	$W1$	$W2$
Lena	34.3274	34.2343	43.5769	43.6769	45.1242	45.0326
Man	33.1674	33.2843	42.4367	42.3562	44.5326	44.4263
House	33.7463	33.6733	42.8356	42.6244	44.7322	44.5468
Pepper	34.4356	34.3278	43.5436	43.4763	44.9243	44.8923
Plane	34.5462	34.4288	43.8634	43.6234	44.9321	44.7632

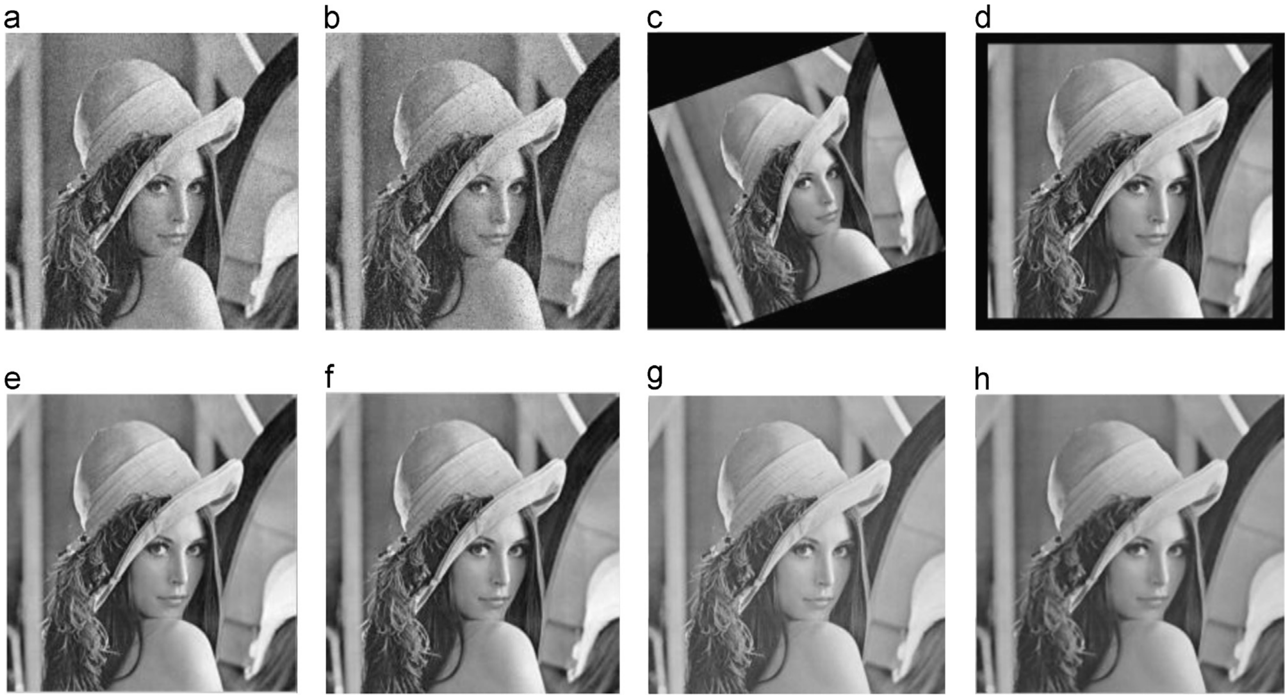

Fig. 7. Attacked watermarked images: (a) Gaussian noise (M=0, var=0.005), (b) salt and pepper noise (density=0.001), (c) rotation by 20°, (d) cropping 20 pixels each side, (e) upscale re-size, (f) downscale re-size (g) gamma correction, (0.8) and (h) median filter (5 × 5).

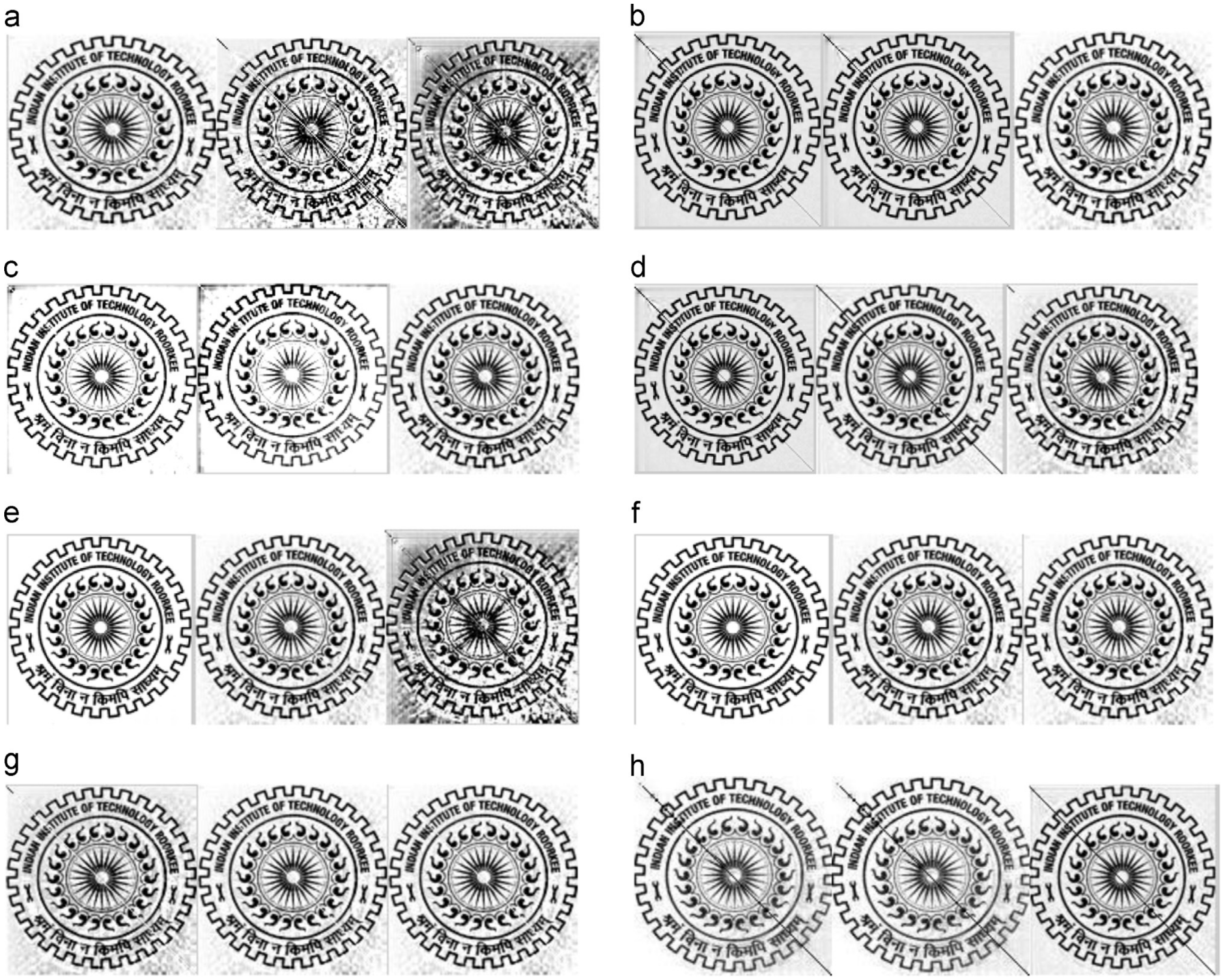

Fig. 8. Extracted watermarks (LL, LH and HL bands) $W1$: (a) Gaussian Noise ($M=0$, var$=0.005$), (b) salt and pepper noise (density$=0.001$), (c) rotation by 20°, (d) cropping 20 pixels each side, (e) scaling (0.5, 2), (f) scaling (2, 0.5), (g) gamma correction (0.8) and (h) median filter (3 × 3).

performed on the watermarked image and M bands and utilized for insertion then the average robustness of scheme can be written as:

$$\text{Robustness}_{average} = \frac{\sum_{i=1}^{N} \text{correlation}(W, W_i^*)}{M \times N} \tag{20}$$

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.

$$\text{Error} = \frac{1}{\text{Robustness}_{average}} - \text{Imperceptibility} \tag{21}$$

The above shown fitness function is a multi-dimensional search; as the size of scaling factors is now considered as a vector of size 256 for each band. So the search cannot 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 0.001–0.1. 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. 5.

4. Results and discussions

In this study, five gray scaled images 'Lena', 'Man', 'House', 'Pepper' and 'Plane' (all of size 512 × 512) are used as the host images. Two gray scaled images '$W1$' and '$W2$' of size 256 × 256 are used as the watermark images. The host and watermark images are shown in Fig. 6. PSNR (Eq. (22)) and Normalized cross correlation (Eq. (19)) are used to compare the similarity of watermarked images with host images and extracted watermarks with original watermarks respectively.

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:

$$\text{PSNR} = 10\log_{10}\left(\frac{n \times n \times (X_{max})^2}{\sum_{i=1}^{n}\sum_{i=1}^{n}(X(i,j) - X^*(i,j))^2}\right) \tag{22}$$

Table 2 shows the feature comparison between proposed scheme and schemes of Ali and Ahn (2014) and Makbol and Khoo (2014). The main difference between Ali and Ahn (2014) and Makbol and Khoo (2014) is the use of singular value at the place for principal components and use of IWT at the place of DWT. Even though, Makbol and Khoo (2014) used IWT and singular values,

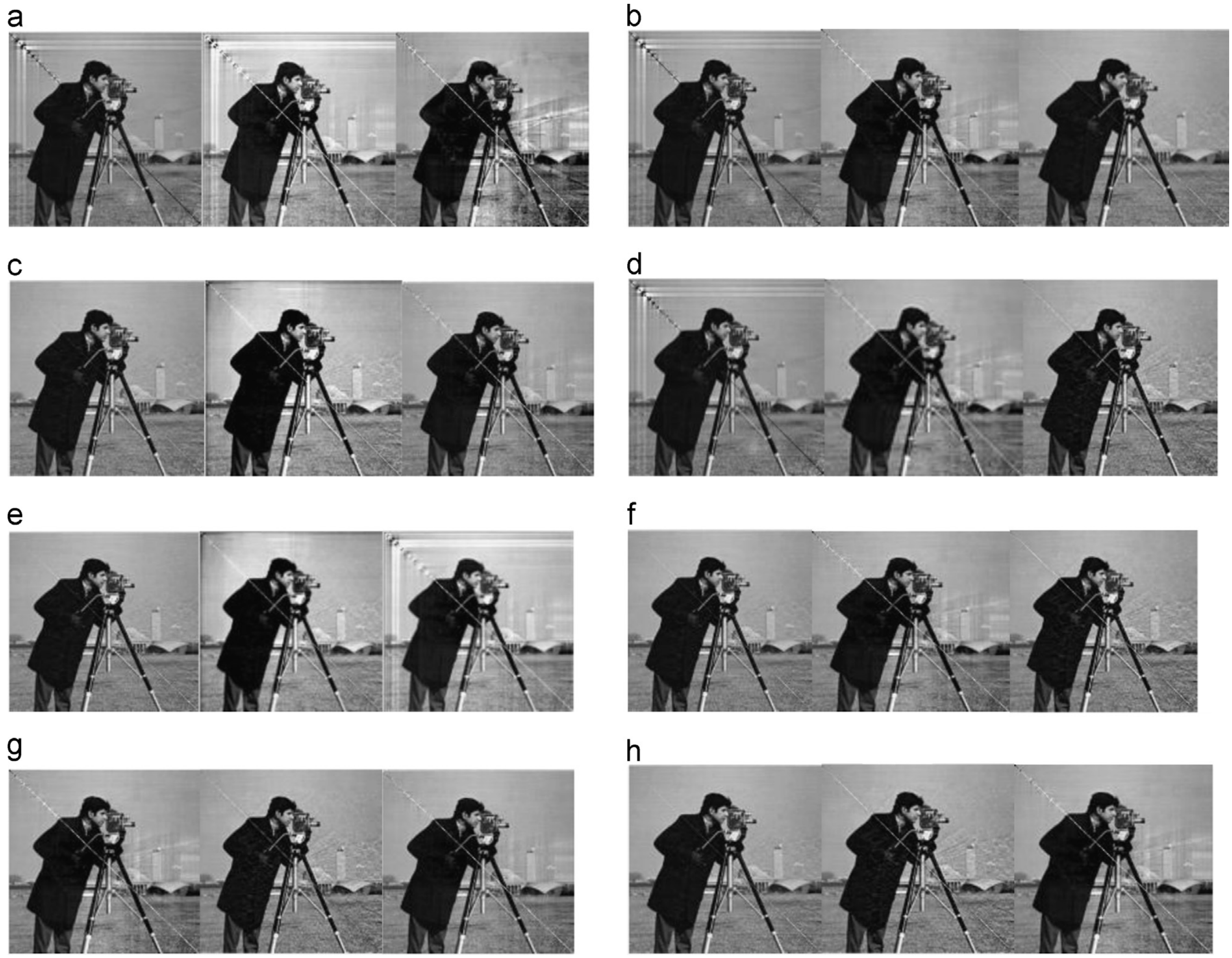

Fig. 9. Extracted watermarks (LL, LH and HL bands) *W*2: (a) Gaussian noise ($M=0$, var$=0.005$), (b) salt and pepper noise (density$=0.001$), (c) rotation by 20°, (d) cropping 20 pixels each side, (e) scaling (0.5, 2), (f) scaling (2, 0.5), (g) gamma correction (0.8) and (h) median filter (3 × 3).

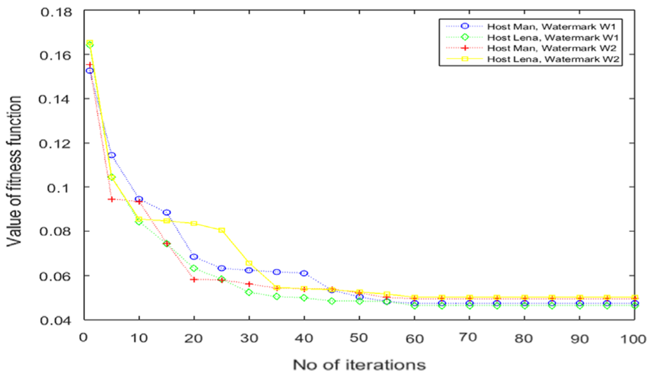

Fig. 10. Fitness value vs. iterations.

but they 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 3 is showing the PSNR value of watermarked images after different watermarks ('*W*1' and '*W*2') 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. 7 shows selected attacks on the watermarked image 'Lena' after embedding the watermark '*W*1'. Fig. 8 shows the extracted watermark '*W*1' from LL, LH and HL bands after those attacks. The same attacks are also applied on the watermarked 'Lena' with watermark '*W*2'. Fig. 9 shows the extracted watermark '*W*2' from those attacked images.

The signature information embedded in HH band of host image is used to check the authenticity of user supplied singular matrices.

In the ideal case, both the signatures (embedded and extracted) should be same i.e. the hamming distance should be zero. But in actual scenario the codes may face a slight variation; especially in case of major attacks. The scheme was able to provide zero hamming distance in the case of no attack as well as minor attacks but in case of major attacks a slight variation is noticed in extracted signature bits. Because of this observation, the hamming distance of three is considered as a valid signature in the proposed scheme.

Fig. 10 shows the convergence of fitness value with the iterations for different hosts and watermark images.

Tables 4 and 5 show the robustness comparison (under different attacks) of proposed watermark scheme with existing schemes of Ali and Ahn (2014) and Makbol and Khoo (2014). Two host images ('Lena' and 'Man') and both the watermarks are used for this comparison.

In Ali and Ahn (2014) method, the insertion of watermark is done in the principal components instead of singular values. Though, doing so makes the scheme secured towards false positive

Table 4
Robustness comparison of proposed scheme with Ali and Ahn (2014) and Makbol and Khoo (2014) for host 'Lena' and watermark '*W*1'.

Attack	Ali and Ahn (2014) for watermark '*W*2'	Makbol and Khoo (2014) (best correlation from all band) for watermark '*W*2'	Proposed (best correlation from LL, LH and HL band) for watermark '*W*1'
Average filtering (3 × 3)	0.9191	0.9716	0.9751
Scaling (0.5, 2)	0.9445	0.9838	0.9889
Gamma correction (0.8)	0.9639	0.9944	0.9949
Median filter (3 × 3)	0.9495	0.9892	0.9896
Histrogram equalization	0.9279	0.9842	0.9878
Gaussian noise (M=0 var =0.01)	0.8589	0.9383	0.9446
JPEG compression (Q=50)	0.9607	0.9992	0.9996
Gaussian filter (3 × 3)	0.9630	0.9915	0.9921
Wiener filtering (2 × 2)	0.9250	0.9948	0.9955
Sharpening (0.8)	0.8893	0.9438	0.9481
Croping 20 pixels each side	0.9313	0.9869	0.9884
Rotation (20°)	0.9467	0.9851	0.9885
Pepper and salt (den=0.001)	0.9247	0.9983	0.9989
Contrast adjustment (20%)	0.9490	0.9729	0.9797

Table 5
Robustness comparison of proposed scheme with Ali and Ahn (2014) and Makbol and Khoo (2014) for host 'Lena' and watermark '*W*2'.

Attack	Ali and Ahn (2014) for watermark '*W*2'	Makbol and Khoo (2014) (best correlation from all band) for watermark '*W*2'	Proposed (best correlation from LL, LH and HL band) for watermark '*W*2'
Average filtering (3 × 3)	0.9146	0.9724	0.9747
Scaling (0.5, 2)	0.9449	0.9840	0.9885
Gamma correction (0.8)	0.9613	0.9940	0.9952
Median filter (3 × 3)	0.9478	0.9890	0.9894
Histrogram equalization	0.9283	0.9854	0.9872
Gaussian noise (M=0 var =0.01)	0.8578	0.9360	0.9442
JPEG compression (Q=50)	0.9583	0.9990	0.9996
Gaussian filter (3 × 3)	0.9592	0.9910	0.9918
Wiener filtering (2 × 2)	0.9225	0.9950	0.9968
Sharpening (0.8)	0.8856	0.9470	0.9485
Croping 20 pixels each side	0.9316	0.9852	0.9896
Rotation (20°)	0.9478	0.9842	0.9884
Pepper and salt (den=0.001)	0.9229	0.9970	0.9976
Contrast adjustment (20%)	0.9496	0.9732	0.9792

error but it also reduces the robustness and imperceptibility of the scheme as the change in principal components change the information contained by left singular vector also. Singular vectors contains large amount of image information and so this leads to a low imperceptibility. Proposed method outperformed the Ali and Ahn (2014) method in term of cross correlation/robustness and PSNR because of the watermark insertion in singular values. Proposed method outperformed Makbol and Khoo (2014) method in terms 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 3–7.

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.

Tables 8 and 9 are showing all the correlations values (LL, LH and HL bands) of extracted watermarks '*W*1' and '*W*2' (under different attacks) for image 'Lena' and 'Man' respectively.

4.1. *False positive error checking*

The proposed scheme is tested for more than 30 different false ownership claims and each time the scheme is able to detect the false claim.

False positive error takes place when a wrong watermark (which is never been inserted into host) is extracted from the host image. The following steps are carried out to check the proposed schemes towards false positive error.

1. Watermark '*W*1' is embedded into the host 'Lena' as per the embedding steps of proposed scheme and watermarked image is named Lena$_{w1}$. The user supplied information is saved as U_{w1i} and V_{w1i} for all the bands.
2. Watermark '*W*2' is embedded into the host 'Lena' as per the embedding steps of proposed scheme and watermarked image is named Lena$_{w2}$. The user supplied information is saved as U_{w2i} and V_{w2i} for all the bands.
3. Before extraction of watermark '*W*1' from watermarked image Lena$_{w1}$, the user supplied information U_{w1i} and V_{w1i} are replaced by the U_{w2i} and V_{w2i}.
4. The extraction process of '*W*1' from watermarked image Lena$_{w1}$ is carried out as per Section 3.2.

The extraction process stopped with a result – *Verification test failed*. This happen because the signature verification step gets failed and this test proofs that the proposed scheme is secured towards false positive error.

4.2. *Performance of proposed scheme with benchmark Stirmark 4.0*

The Stirmark 4.0 is a watermarking benchmark (Petitcolas et al., 1998), which consists of a number of signal processing and

Table 6
Robustness comparison of proposed scheme with Ali and Ahn (2014) and Makbol and Khoo (2014) for host 'Man' and watermark '*W*1'.

Attack	Ali and Ahn (2014) for watermark '*W*1'	Makbol and Khoo (2014) (best correlation from all band) for watermark '*W*1'	Proposed (best correlation from LL, LH and HL band) for watermark '*W*1'
Average filtering (3 × 3)	0.9184	0.9714	0.9738
Scaling (0.5, 2)	0.9452	0.9856	0.9853
Gamma correction (0.8)	0.9636	0.9961	0.9981
Median filter (3 × 3)	0.9498	0.9836	0.9882
Histrogram equalization	0.9282	0.9872	0.9884
Gaussian noise (M=0 var =0.01)	0.8592	0.9482	0.9526
JPEG compression (Q=50)	0.9612	0.9989	0.9990
Gaussian filter (3 × 3)	0.9628	0.9921	0.9921
Wiener filtering (2 × 2)	0.9249	0.9946	0.9974
Sharpening (0.8)	0.8912	0.9432	0.9468
Croping 20 pixels each side	0.9311	0.9843	0.9891
Rotation (20°)	0.9421	0.9873	0.9884
Pepper and salt (den=0.001)	0.9232	0.9932	0.9948
Contrast adjustment (20%)	0.9482	0.9812	0.9852

Table 7
Robustness comparison of proposed scheme with Ali and Ahn (2014) and Makbol and Khoo (2014) for host 'Man' and watermark '*W*2'.

Attack	Ali and Ahn (2014) for watermark '*W*2'	Makbol and Khoo (2014) (best correlation from all band) for watermark '*W*2'	Proposed (best correlation from LL, LH and HL band) for watermark '*W*2'
Average filtering (3 × 3)	0.9182	0.9742	0.9751
Scaling (0.5, 2)	0.9447	0.9847	0.9859
Gamma correction (0.8)	0.9643	0.9942	0.9961
Median filter (3 × 3)	0.9436	0.9883	0.9898
Histrogram equalization	0.9285	0.9862	0.9879
Gaussian noise (M=0 var =0.01)	0.8613	0.9369	0.9452
JPEG compression (Q=50)	0.9624	0.9991	0.9995
Gaussian filter (3 × 3)	0.9626	0.9914	0.9914
Wiener filtering (2 × 2)	0.9242	0.9944	0.9971
Sharpening (0.8)	0.8918	0.9487	0.9483
Croping 20 pixels each side	0.9321	0.9869	0.9881
Rotation (20°)	0.9428	0.9838	0.9869
Pepper and salt (den=0.001)	0.9237	0.9965	0.9968
Contrast adjustment (20%)	0.9492	0.9768	0.9808

Table 8
NCC of proposed scheme and Makbol and Khoo (2014) for host 'Lena'.

Attack	Makbol and Khoo (2014) (watermark *W*2)			Proposed scheme (watermark *W*2)		
	LL band	LH band	HL band	LL band	LH band	HL band
Average filtering (3 × 3)	0.9724	0.9647	0.9643	0.9747	0.9652	0.9651
Scaling (0.5, 2)	0.9840	0.8995	0.8753	0.9885	0.9036	0.8932
Scaling (2, 0.5)	0.9980	0.9854	0.9834	0.9986	0.9873	0.9857
Gamma correction (0.6)	0.9840	0.9880	0.9847	0.9844	0.9896	0.9856
Gamma correction (0.8)	0.9940	0.9891	0.9870	0.9952	0.9882	0.9889
Median filter (3 × 3)	0.9890	0.9758	0.9725	0.9894	0.9762	0.9734
Histrogram equalization	0.9743	0.9854	0.9732	0.9802	0.9872	0.9739
Gaussian noise (M=0 var =0.01)	0.9360	0.8998	0.8207	0.9442	0.9026	0.8312
Gaussian noise (M=0 var =0.005)	0.9610	0.8822	0.7950	0.9694	0.8856	0.8018
JPEG compression (Q=40)	0.9990	0.9776	0.9779	0.9996	0.9768	0.9808
JPEG compression (Q=50)	0.9990	0.9784	0.9802	0.9996	0.9799	0.9852
Gaussian filter (3 × 3)	0.9910	0.9323	0.9733	0.9918	0.9356	0.9784
Wiener filtering (2 × 2)	0.9950	0.9846	0.9806	0.9968	0.9848	0.9814
Sharpening (0.8)	0.9326	0.9470	0.9243	0.9367	0.9485	0.9239
Croping 20 pixels each side	0.9827	0.9852	0.9824	0.9862	0.9896	0.9863
Rotation (20°)	0.9842	0.9040	0.9532	0.9884	0.9076	0.9612
Pepper and salt (den=0.001)	0.9950	0.9970	0.9727	0.9959	0.9976	0.9783
Contrast adjustment (20%)	0.9732	0.9711	0.9643	0.9792	0.9743	0.9699

geometric attacks. It is used to check the robustness of proposed scheme and the result of same is shown in Table 10. From Table 10 it is clear that the proposed scheme is quite robust towards different signal processing and geometric attacks.

5. Conclusion

In the proposed watermarking scheme, the scaling factors were optimized with the help of artificial bee colony (ABC), which

Table 9
NCC of proposed scheme and Makbol and Khoo (2014) for host 'Man'.

Attack	Makbol and Khoo (2014) (watermark *W*2)			Proposed scheme (watermark *W*2)		
	LL band	LH band	HL band	LL band	LH band	HL band
Average filtering (3 × 3)	0.9742	0.9652	0.9652	0.9751	0.9652	0.9668
Scaling (0.5, 2)	0.9847	0.9024	0.8882	0.9859	0.9078	0.8946
Scaling (2, 0.5)	0.9978	0.9862	0.9843	0.9989	0.9878	0.9842
Gamma correction (0.6)	0.9840	0.9884	0.9865	0.9844	0.9892	0.9884
Gamma correction (0.8)	0.9942	0.9864	0.9856	0.9961	0.9884	0.9869
Median filter (3 × 3)	0.9883	0.9785	0.9743	0.9898	0.9812	0.9784
Histrogram equalization	0.9742	0.9862	0.9743	0.9859	0.9879	0.9769
Gaussian noise (M=0 var =0.01)	0.9369	0.8954	0.8287	0.9452	0.9035	0.8353
Gaussian noise (M=0 var =0.005)	0.9621	0.8854	0.7974	0.9684	0.8859	0.8054
JPEG compression (Q=40)	0.9990	0.9733	0.9787	0.9994	0.9745	0.9811
JPEG compression (Q=50)	0.9991	0.9774	0.9846	0.9995	0.9797	0.9872
Gaussian filter (3 × 3)	0.9914	0.9354	0.9765	0.9914	0.9368	0.9792
Wiener filtering (2 × 2)	0.9944	0.9847	0.9825	0.9971	0.9859	0.9832
Sharpening (0.8)	0.9335	0.9487	0.9286	0.9396	0.9483	0.9308
Croping 20 pixels each side	0.9865	0.9869	0.9876	0.9879	0.9881	0.9889
Rotation (20°)	0.9838	0.9054	0.9584	0.9869	0.9084	0.9598
Pepper and salt (den=0.001)	0.9976	0.9965	0.9785	0.9978	0.9968	0.9794
Contrast adjustment (20%)	0.9768	0.9765	0.9687	0.9808	0.9791	0.9724

Table 10
Performance of proposed scheme with benchmark Stirmark 4.0 using different hosts.

Attack	NCC of extracted 'W2' with original 'W2'				
	Lena	Man	House	Pepper	Plane
No attack	1.0000	1.0000	1.0000	1.0000	1.0000
JPEG compression 50	0.9996	0.99994	0.9992	0.9994	0.9995
JPEG compression 70	1.0000	1.0000	1.0000	1.0000	1.0000
JPEG compression 90	1.0000	1.0000	1.0000	1.0000	1.0000
Median filter (3 × 3)	0.9894	0.9898	0.9873	0.9892	0.9884
Median filter (5 × 5)	0.9734	0.9725	0.9726	0.9735	0.9731
Random bending	0.9527	0.9521	0.9519	0.9527	0.9518
Shearing *x*-0%, *y*-5%	0.9624	0.9619	0.9624	0.9623	0.9631
Shearing *x*-5%, *y*-0%	0.9672	0.9667	0.9664	0.9669	0.9671
Shearing *x*-1%, *y*-1%	0.9824	0.9829	0.9824	0.9825	0.9824
Centered crop 5%	0.9819	0.9823	0.9816	0.9820	0.9818
Rotation 2°+crop	0.9782	0.9782	0.9772	0.9779	0.9782
Rotation 5°+crop	0.9754	0.9751	0.9742	0.9754	0.9749
Rotation 10°+crop	0.9728	0.9719	0.9725	0.9728	0.9719
Rotation 15∘+crop	0.9704	0.9708	0.9701	0.9712	0.9710
Scaling 0.75 ×	0.9921	0.9921	0.9922	0.9922	0.9919
Scaling 0.90 ×	0.9984	0.9985	0.9984	0.9985	0.9984
Scaling 1.1 ×	0.9994	0.9994	0.9995	0.9994	0.9994
Scaling 1.5 ×	0.9992	0.9992	0.9992	0.9993	0.9993
Rotation 2°+scaling	0.9897	0.9886	0.9892	0.9889	0.9893
Rotation 5°+scaling	0.9843	0.9843	0.9852	0.9855	0.9847
Rotation 10°+scaling	0.9824	0.9827	0.9828	0.9831	0.9826
Rotation 2°+scaling+crop	0.9652	0.9647	0.9641	0.9650	0.9655
Rotation 5°+scaling+crop	0.9624	0.9618	0.9636	0.9624	0.9626
Rotation 10°+scaling+ crop	0.9582	0.9584	0.9591	0.9574	0.9589
Random distortion 0.95	0.9925	0.9928	0.9921	0.9918	0.9914
Flip horizontally	1.0000	1.0000	1.0000	1.0000	1.0000
Flip vertically	1.0000	1.0000	1.0000	1.0000	1.0000
Linear geometric transform 1.008	0.9728	0.9726	0.9721	0.9727	0.9731
Linear geometric transform 1.012	0.9657	0.9649	0.9651	0.9657	0.9659

improved the robustness and imperceptibility. The watermark was embedded in the singular values of IWT bands of host image, which helped the scheme to achieve high capacity, robustness and imperceptibility. By inserting signature information into the LL band, the scheme also becomes free from false positive error as this signature information provided the authentication of user supplied singular matrices. The comparative analysis showed that the proposed scheme performs better than the DWT and IWT based schemes proposed in past. 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 under the IKRI (India Korea Research Internship) program; jointly supported by National Research Foundation of Korea (NRF-P4KR-006) and Department of Science and Technology, India.

References

Ali, Musrrat, Ahn, Chang Wook, 2014. An optimized watermarking technique based on self-adaptive DE in DWT–SVD transform domain. Signal Process. 94, 545–556.

Ali, Musrrat, Ahn, Chang Wook, 2015. Comments on optimized gray-scale image watermarking using DWT–SVD and firefly algorithm. Expert. Syst. Appl. 42 (5), 2392–2394.

Ali, Musrrat, Ahn, Chang Wook, Pant, Millie, 2014. A robust image watermarking technique using SVD and differential evolution in DCT domain. Optik – Int. J. Light Electron Opt. 125 (1), 428–434.

Ansari, I.A., Pant, M., 2015. SVD watermarking: particle swarm optimization of scaling factors to increase the quality of watermark. In: Proceedings of Fourth International Conference on Soft Computing for Problem Solving, Springer India, pp. 205–214.

Ansari, I.A., Pant, M., Ahn, C.W., 2015. SVD based fragile watermarking scheme for tamper localization and self-recovery. Int. J. Mach. Learn. Cybern. 1–15 . http://dx.doi.org/10.1007/s13042-015-0455-1.

Ansari, Irshad Ahmad, Millie, Pant, Ferrante, Neri, 2014. Analysis of gray scale watermark in RGB host using SVD and PSO. In: Proceedings of the IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing. CIMSIVP, pp. 1–7.

Bhatnagar, Gaurav, Raman, Balasubramanian, 2009. A new robust reference watermarking scheme based on DWT–SVD. Comput. Stand. Interfaces 31 (5), 1002–1013.

Daubechies, I., Sweldens, W., 1998. Factoring wavelet transforms into lifting steps. J. Fourier Anal. Appl. 4 (3), 245–267.

Draa, A., Bouaziz, A., 2014. An artificial bee colony algorithm for image contrast enhancement. Swarm Evolut. Comput. 16, 69–84.

Friedman, Gary L., 1993. The trustworthy digital camera: restoring credibility to the photographic image. IEEE Trans. Consum. Electron. 39 (4), 905–910.

Ganic, Emir, Eskicioglu, Ahmet M., 2005. Robust embedding of visual watermarks using discrete wavelet transform and singular value decomposition. J. Electron. Imaging 14 (4), 043004.

Gokhale, U.M., Joshi, Y.V., 2012. A semi fragile watermarking algorithm based on SVD-IWT for image authentication. Int. J. Adv. Res. Comput. Commun. Eng. 1, 4.

Gupta, Akshya, Kumar, Raval, Mehul S., 2012. A robust and secure watermarking scheme based on singular values replacement. Sadhana 37 (4), 425–440.

Hanbay, K., Talu, M.F., 2014. Segmentation of SAR images using improved artificial bee colony algorithm and neutrosophic set. Appl. Soft Comput. 21, 433–443.

Haouzia, Adil, Noumeir, Rita, 2008. Methods for image authentication: a survey. Multimed. Tools Appl. 39 (1), 1–46.

Jia, Z.-Z., Zhu, H.-Y., Cheng, W.-S., 2010, A blind watermarking algorithm based on lifting wavelet transform and scrambling technology. In: Proceedings of the International Conference on Electrical and Control Engineering. ICECE, pp. 4576–4579.

Karaboga, D., Akay, B., 2009. A comparative study of Artificial Bee Colony algorithm. Appl. Math. Comput. 214, 108–132.

Karaboga, Dervis, 2005. An Idea Based on Honey Bee Swarm for Numerical Optimization. vol. 200. Erciyes University, Engineering Faculty, Computer Engineering Department, Kayseri/Türkiye, Technical report-tr06.

Lagzian, Samira, Soryani, Mohsen, Fathy, Mahmood, 2011. A new robust watermarking scheme based on RDWT–SVD. Int. J. Intell. Inf. Process. 2 (1), 22–29.

Lai, Chih-Chin, Tsai, Cheng-Chih, 2010. Digital image watermarking using discrete wavelet transform and singular value decomposition. IEEE Trans. Instrum. Meas. 59 (11), 3060–3063.

Li, B., Li, Y., Gong, L., 2014. Protein secondary structure optimization using an improved artificial bee colony algorithm based on AB off-lattice model. Eng. Appl. Artif. Intell. 27, 70–79.

Li, Zhen, Yap, Kim-Hui, Lei, Bai-Ying, 2011. A new blind robust image watermarking scheme in SVD-DCT composite domain. In: Proceedings of the 18th IEEE International Conference on Image Processing. ICIP, pp. 2757–2760.

Ling, Huo-Chong, Phan, Raphael C.-W., Heng, Swee-Huay, 2013. Comment on Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition. AEU – Int. J. Electron. Commun. 67 (10), 894–897.

Liu, Shao-Hui, Yao, Hong-Xun, Gao, Wen, Liu, Yong-Liang, 2007. An image fragile watermark scheme based on chaotic image pattern and pixel-pairs. Appl. Math. Comput. 185 (2), 869–882.

Loukhaoukha, Khaled, Chouinard, Jean-Yves, Haj Taieb, Mohamed, 2011. Optimal image watermarking algorithm based on LWT-SVD via multi-objective ant colony optimization. J. Inf. Hiding Multimed. Signal Process. 2 (4), 303–319.

Makbol, Nasrin M., Khoo, Bee Ee, 2014. A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition. Digit. Signal Process. 33, 134–147.

Mishra, Anurag, Agarwal, Charu, Sharma, Arpita, Bedi, Punam, 2014. Optimized gray-scale image watermarking using DWT–SVD and firefly algorithm. Expert Syst. Appl. 41 (17), 7858–7867.

Petitcolas, F., Anderson, R.J., Kuhn, M.G., 1998. Attacks on copyright marking systems. In: Proceedings of the International Workshop on Information Hiding, LNCS 1575.

Potdar, V.M., Han, S., Chang, E., 2005. A survey of digital image watermarking techniques. In: Proceedings of the 3rd IEEE International Conference on Industrial Informatics. INDIN'05, pp. 709–716.

Ramanathan, R., Kalaiarasi, K., Prabha, D., 2013. Improved wavelet based compression with adaptive lifting scheme using artificial bee colony algorithm. Int. J. Adv. Res. Comput. Eng. Technol. 2 (4), 1549.

Rastegar, Saeed, Namazi, Fateme, Yaghmaie, Khashayar, Aliabadian, Amir, 2011. Hybrid watermarking algorithm based on singular value decomposition and radon transform. AEU – Int. J. Electron. Commun. 65 (7), 658–663.

Rawat, Sanjay, Raman, Balasubramanian, 2011. A chaotic system based fragile watermarking scheme for image tamper detection. AEU – Int. J. Electron. Commun. 65, 840–847.

Rodriguez, F.J., Lozano, M., García-Martínez, C., González-Barrera, J.D., 2013. An artificial bee colony algorithm for the maximally diverse grouping problem. Inf. Sci. 230, 183–196.

Run, Ray-Shine, Horng, Shi-Jinn, Lai, Jui-Lin, Kao, Tzong-Wang, Chen, Rong-Jian, 2012. An improved SVD-based watermarking technique for copyright protection. Expert Syst. Appl. 39 (1), 673–689.

Storn, Rainer, Price, Kenneth, 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. 11 (4), 341–359.

Su, Q., Niu, Y., Liu, X., Zhu, Y., 2012. A blind dual color images watermarking based on IWT and state coding. Opt. Commun. 285 (7), 1717–1724.

Sweldens, W., 1998. The lifting scheme: a construction of second generation wavelets. SIAM J. Math. Anal. 29 (2), 511–546.

Tian, Y., Tan, T., Wang, Y., Fang, Y., 2003. Do singular values contain adequate information for face recognition? Pattern Recognit. 36 (3), 649–655.

Lin, Weisi, Tao, Dacheng, Kacprzyk, Janusz, Li, Zhu, Izquierdo, Ebroul, Wang, Haohong, 2011. Multimedia Analysis, Processing and Communications. vol. 346. Springer.