

# An Optimized Image Water marking Scheme in Hadamard Transform Domain for Color images

Elham Moeinaddini  
Engineering Department  
University of Jiroft  
Jiroft, Iran  
[moeinelham96@gmail.com](mailto:moeinelham96@gmail.com)



**ABSTRACT:** *In this paper, a robust watermarking method using hadamard transform is proposed for color images. In order to resolve correlation of color channels, the Principal Component(PCA) Analysis is used and its principle component is used for watermarking. Principle component and watermark image are both divided into blocks and fast hadamard transform is applied to them. Then, hadamard coefficients of watermark blocks are embedded in AC components of blocks of cover image using different robustness factors. Since, cover image is not required for detecting watermark; this method is a blind watermarking method. Using hadamard transform compared to other transforms increases speed and reduces complexity of the algorithm and preserves transparency of the watermarked image. In order to optimize robustness factors, Artificial Bee Colony (ABC) is used. Results show that the proposed method improves results of similar methods in terms of robustness and transparency.*

**Keywords:** Watermarking, PCA, Hadamard Transform, Robustness, Transparency

**Received:** 11 January 2018, Revised 28 February 2018, Accepted 10 March 2018

© 2018 DLINE. All Rights Reserved

## 1. Introduction

Development of Internet and computer networks has made multimedia data accessible which can be copied easily. Copied digital content have quality of the main content. This fact creates bad critical condition like misusing data. Watermarking is a method proposed to resolve such problems which is a proper tool for preserving ownership right of digital contents. In primary watermarking methods, watermark is embedded in bits of the image, directly. Another method is to use frequency domain. These techniques employ different transforms like Discrete Fourier Transform, Discrete Cosine Transform, Discrete Wavelet Transform, Discrete Hadamard Transform and their combinations [1-5]. Among these transforms, hadamard transform is one of the fastest and strongest ones in terms of transparency. Researchers have proposed different watermarking methods using hadamard transform. S. P. Maity et.al. have proposed a watermarking method based on entropy in hadamard domain for grayscale images in which watermark is embedded in blocks of the image with low to medium entropy[6]. V. Santhi et.al. proposed an adaptive method for calculating scaling factors using sigmoid function in hadamard domain[7]. Robustness is controlled with a parameter which can be used to make the watermark visible or invisible. Falkowski and Lim proposed a method for making the watermark robust against JPEG compression in hadamard domain for grayscale images[8]. This method is also robust against scaling, fading and cropping.

In the method proposed by Mansoori and Soltani, AC coefficients of watermarking blocks are embedded in AC coefficients of image after applying hadamard transform[9]. DC coefficients of watermark blocks are kept as a key for detection. Detection employed in this method is semi-blind. Most watermarking methods proposed for color images, embed watermark in RGB channels of the image[10-13]. Due to high correlation of R, G and B channels, transparency of watermark is reduced. In order to resolve this correlation, some researchers have employed YCbCR, UCS and PCA for watermarking[14-16]. Changes made in content for embedding watermark should be determined such that there is balance between transparency and robustness of the watermark. These changes are usually adjusted with parameters which are known as scaling factors. Therefore, an efficient algorithm should be proposed for finding optimum scaling factors such that transparency and robustness are balanced and this parameter is determined dynamically for each content. Since, purpose is to find scaling factors for balancing transparency and robustness, this issue can be considered as an optimization problem where meta-heuristic methods can be used to solve it. Researchers have employed methods based on population and swarm intelligence (Particle Swarm Optimization, Ant Colony Optimization, Artificial Bee Colony and ...) to find optimal scaling factors in watermarking techniques and have improved results significantly [17-19]. Compared to other evolutionary algorithms, artificial bee colony has more interesting characteristics including relative simplicity, effectiveness in complex and large search spaces which make it a suitable option for multiple objective optimization [20].

According to studies, a blind algorithm for watermarking color images on hadamard transform which detecting scaling factors using ABC algorithm, has not been proposed yet. In this paper, a color images watermarking scheme in hadamard domain using PCA and multiple scaling factors is proposed. Optimizing scaling factors is performed using ABC algorithm and objective function used in this algorithm is a linear combination of normal correlation (NC) coefficient and PSNR. Rest of this paper is organized as follows. Section 2 describes concepts and employed algorithms briefly. The method proposed for embedding and extracting watermark and how ABC algorithm is used for optimizing threshold is described in section 3. Section 4 presents experiments and obtained results. Finally, the paper is concluded in section 5.

## 2. Concepts and Employed Algorithms

In this section concepts and employed algorithms are described briefly.

### 2.1 Artificial Bee Colony Algorithm

Swarm intelligence algorithms are a new context of computation algorithms which are inspired by social behavior of insects. Karaboga [20] proposed a simple optimization algorithm based on population and called it bee colony. In ABC algorithm, there are three types of bees: employed bees, onlooker bees and scout bees. This algorithm has 4 stages: First stage is initializing bees. In this stage, initial population is created and other parameters of the algorithm are also initialized. If total number of bees is  $N$ , number of food sources(solutions) is  $SN$ , maximum iteration numbers is  $MAX_{cyc}$  and objective functions and fitness function are  $f(x)$  and  $g(x)$ , respectively, initial solutions ( $X^i$ 's) are initialized on D-dimensional space of the problem as follows:

$$X_j^i = X_j^{min} + rand(0, 1) (X_j^{max} - X_j^{min}) \quad (1)$$

in which  $j = 1, 2, \dots, D$  is dimension of the solution and  $X_j^{max}$  and  $X_j^{min}$  are maximum and minimum bounds on  $j^{th}$  dimension. Then value of fitness function for each initial food source is calculated and  $NB$  bees with best value of fitness function as employee bees start working. In addition, counter  $T_i$  where  $i = 1, 2, \dots, NB$  are initialized with zero. Second stage is employee bee's turn. Each employee bee looks to replace new food source  $X_{ij}$  with its old food source using the following equation:

$$\bar{X}_{ij} = X_{ij} + \Phi_{ij} (X_{ij} - X_{kj}) \quad k \in \{1, 2, \dots, \}, i \neq k \quad (2)$$

where  $i$  shows an employee bee,  $k$  is index of random bee which should differ with index  $i$ .  $\Phi$  is a random real number between -1 and 1. Using the following equation, it can be guaranteed  $X_{ij}$  that does not exceed the defined bounds.

$$\bar{X}_{ij} = \begin{cases} X_j^{min} & \text{if } \bar{X}_{ij} < X_j^{min} \\ X_j^{max} & \text{if } \bar{X}_{ij} > X_j^{max} \end{cases} \quad (3)$$

if  $f_i(\cdot)$  is the objective function, fitness function for  $\bar{X}_i$  is calculated as follows:

$$g(\bar{X}_i) = \begin{cases} 1 + |f_i(\bar{X}_i)| & \text{if } f_i(\bar{X}_i) < 0 \\ \frac{1}{1 + |f_i(\bar{X}_i)|} & \text{if } f_i(\bar{X}_i) \geq 0 \end{cases} \quad (4)$$

if  $g(\bar{X}_i) > g(X_i)$ , employee bee stops searching source  $X_i$  and selects  $\bar{X}_i$  as the new source and counter of that source  $T_i$  becomes zero, otherwise it continues searching around  $X_i$  and one unit is added to its counter. At onlooker bees' stage, employee bees transfer information obtained about food sources to onlooker bees. Onlooker bee investigates information obtained by employee bees about nectar and select food sources with maximum nectar using the following equation.

$$P_i = \frac{g(X_i)}{\sum_{n=1}^{NB} g(X_i)} \quad (5)$$

in this equation  $g(X_i)$  is the suitable amount of nectar in food sources at location  $i$  and  $NB$  is the number of food sources equal to number of employee bees. At scout bees stage, after completion of employee and onlooker bees at each round, if food source with counter  $T_i$  is greater than predefine value of Limit, that food source is left and a new source is replaced randomly. This initialization is performed using the following equation:

$$X_{\text{new}}^j = X_j^{\min} + \text{rand}(0,1)(X_j^{\max} - X_j^{\min}) \quad (6)$$

Employee, onlooker and scout bee stages are continued until number of algorithm iterations do not exceed  $MAX_{cyc}$ .

## 2.2 Principle Component Analysis (PCA)

PCA is one of the methods used for analysis of multi-variable data which its main purpose is to reduce dimensions of the problem. In image processing problems, PCA is used to reduce dimensions of the Image. In RGB images, R, G and B channels are strongly correlated[21], which reduces transparency of watermark in these channels. Uncorrelation of PCA components increases transparency of watermark in color images. Using this method, a RGB method can be converted to three uncorrelated components. Since principal component includes most information of the main image and represents most of its characteristics, by embedding information in this component, robustness of the watermark increases[15]. For a RGB image with dimension of  $M \times N$ , PCA is calculated through following steps:

- 1) RGB image is decomposed to three monochrome components of R, G and B.
- 2) Average of components R, G and B are calculated separately.
- 3) If value of pixel  $(k, l)$  at band  $i$  is shown with  $x_i(k, l)$  and value of pixel  $(k, l)$  at band  $j$  is shown with  $x_j(k, l)$  and  $x_i$  and  $x_j$  are mean value of  $b$  and  $i$  and  $j$  respectively, then covariance of R, G and B is calculated using the following equation:

$$C_{ij} = \frac{1}{N_1 N_2} \sum_{k=0}^{N_1-1} \sum_{l=0}^{N_2-1} (x_i(k, l) - x_i) \times (x_j(k, l) - x_j) \quad (7)$$

$i$  and  $j$  are 1, 2 and 3 corresponding to bands R, G and B, therefore, covariance matrix  $C$  is a  $3 \times 3$  diagonal matrix.

- 4) If eigenvectors of covariance matrix  $C$  are represented with  $\mu_1, \mu_2$  and  $\mu_3$  and  $T$  shows transpose of the matrix, then K-L transform matrix is calculated as follows:

$$A = \begin{pmatrix} \mu_1^T \\ \mu_2^T \\ \mu_3^T \end{pmatrix} \quad (8)$$

2. Three components,  $P_1, P_2$  and  $P_3$  for each pixel  $(k, l)$  are calculated using the following equation:

$$\begin{pmatrix} P_1(k, l) \\ P_2(k, l) \\ P_3(k, l) \end{pmatrix} = A \begin{pmatrix} R(k, l) \\ G(k, l) \\ B(k, l) \end{pmatrix} \quad (9)$$

in this equation,  $P_1$  is the principal component of the initial  $RGB$  image. On the other hand, if components  $P_1, P_2$  and  $P_3$  and matrix  $A$  are known, bands  $R, G$  and  $B$  are calculated as follows:

$$\begin{pmatrix} R(i, j) \\ G(i, j) \\ B(i, j) \end{pmatrix} = A^{-1} \begin{pmatrix} P_1(i, j) \\ P_2(i, j) \\ P_3(i, j) \end{pmatrix} \quad (10)$$

and  $RGB$  image is restored with bands  $R, G$  and  $B$  [15].

### 2.3 2D Hadamard Transform

2D-Hadamard transform is based on Hadamard transform matrix and has been widely used in image processing and watermarking applications. To apply Hadamard transform on an image of size  $2^n$ , we should use a Hadamard transform matrix  $H_n$ . This matrix is a  $2^n \times 2^n$  square matrix which is consisted of 1 and -1. Due to that, the forward and inverse Hadamard transforms can be calculated using simple rules [22]. Owing to orthogonality of its rows and columns,  $H_n$  matrix has some special features. One of its significant features is shown in (11):

$$H_n = H_n^T = H_{n-1} \quad (11)$$

In this equation,  $H_{n-1}$  represents the inverse matrix of  $H_n$  and  $H_n^T$  is the transposed of  $H_n$ . Due to previous rule, the transform matrix  $H_n$  could be obtained according to following equations:

$$H_n = H_{n-1} \otimes H_1 \quad (12)$$

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix} \quad (13)$$

Finally, Hadamard coefficients of image are obtained using (14):

$$V = \frac{H_n \cup H_n}{N} \quad (14)$$

In this equation,  $H_n$  is Hadamard transform matrix  $N \times N$ , in which  $N = 2^n$ , and  $V$  is transformed image. Using orthogonality feature and considering as an identity matrix, we have:

$$H_n \times H_n = N \times 1 \quad (15)$$

Hence the inverse Hadamard transform can be calculated using (16):

$$U = \frac{H_n \vee H_n}{N} \quad (16)$$

In each rows of transformed matrix  $H_n$ , the number of sign changes represents the sequence. These sequences are equivalent to frequency in Fourier transform. Since in Hadamard transform matrix the AC components are distributed randomly, in watermarking procedure, data can be embedded into low frequency AC components. This feature increases the reliability of watermark and also it makes difficulties in attacks and removing watermark [23].

### 3. The Proposed Method

In order to embed a binary image  $W$  with  $m \times n$  dimension in a RGB image with  $M \times N$  dimension, the following steps are performed:

1. Applying PCA on cover image  $I$  and selecting principle component  $P_1$ .
2. Dividing  $P_1$  to equal  $8 \times 8$  non-overlapping blocks.
3. Applying 2D hadamard on blocks of  $P_1$ . Blocks after applying hadamard transform are shown with  $B$ .
4.  $w$  is divided to  $4 \times 4$  non-overlapping blocks.
5. Applying 2D hadamard transform to blocks of  $w$  and watermark blocks after applying hadamard transform are shown with  $b$ .
6.  $b$  blocks are embedded at upper right corner of  $B$  blocks except first row, using scaling factors  $\alpha$ . Embedding algorithm is as follows:

```
for  $i = 2: Bsize - bsize + 1$ 
    for  $j = 1: Bsize - bsize$ 
 $B_w(i, j) = b(i, j) * a_k;$ 
    end
end
```

In this code,  $Bsize$  is 8 and  $bsize$  is 4 and  $k$  shows number of the current block. Figure 1 shows embedding process for each block. In this algorithm, is scaling factor for  $k^{th}$  block which is generated using optimal ABC algorithm and saved in a file and this file used as key for extraction process.

7. Inverse hadamard transform is applied to  $B_w$  blocks and  $P'_1$  is obtained.
8. Inverse PCA is performed and watermarked RGB image,  $I_w$  is rebuild.

Watermark embedding procedure is shown in Figure 2.

In this method, watermark detection is blind and cover image is not required for detection; only the watermarked image (which might be processed or attacked) and secret key used in embedding which determine scaling factors of each block are required. Watermark detection steps are as follows:

1. PCA is performed on watermarked image  $I''$  and its principle component  $P''_1$  is selected.
2.  $P''_1$  is divided into equal  $8 \times 8$  non-overlapping blocks.
3. 2D hadamard transform is applied to each blocks.  $P''_1$  blocks after applying hadamard transform are shown with  $B'_w$ .
4. In order to detect the watermark, following pseudo-code is used:

```
for  $i = 2: Bsize - bsize + 1$ 
    for  $j = 1: Bsize - bsize$ 
 $b'(i, j) = B'_w(i, j) / a_k$ 
    end
end
```

$a_k$  is available as secret key.

5. When  $b'$ 's are located besides each other, watermark  $W'$  is restored. Extraction process of watermark is shown in Figure 3.

In many of the existing methods, scaling factors of all blocks is the same. Since texture of blocks is not the same, quality of the image is reduced. In order to solve this problem, for each block, different scaling factors are considered. So number of scaling factors is the same as number of watermark blocks. In order to optimize these scaling factors, ABC algorithm is employed and its objective function is a linear combination of quality of the watermarked image and quality of the extracted watermark after applying attacks which are evaluated using PSNR (peak signal to noise ratio) and NC (normalized correlation), respectively.

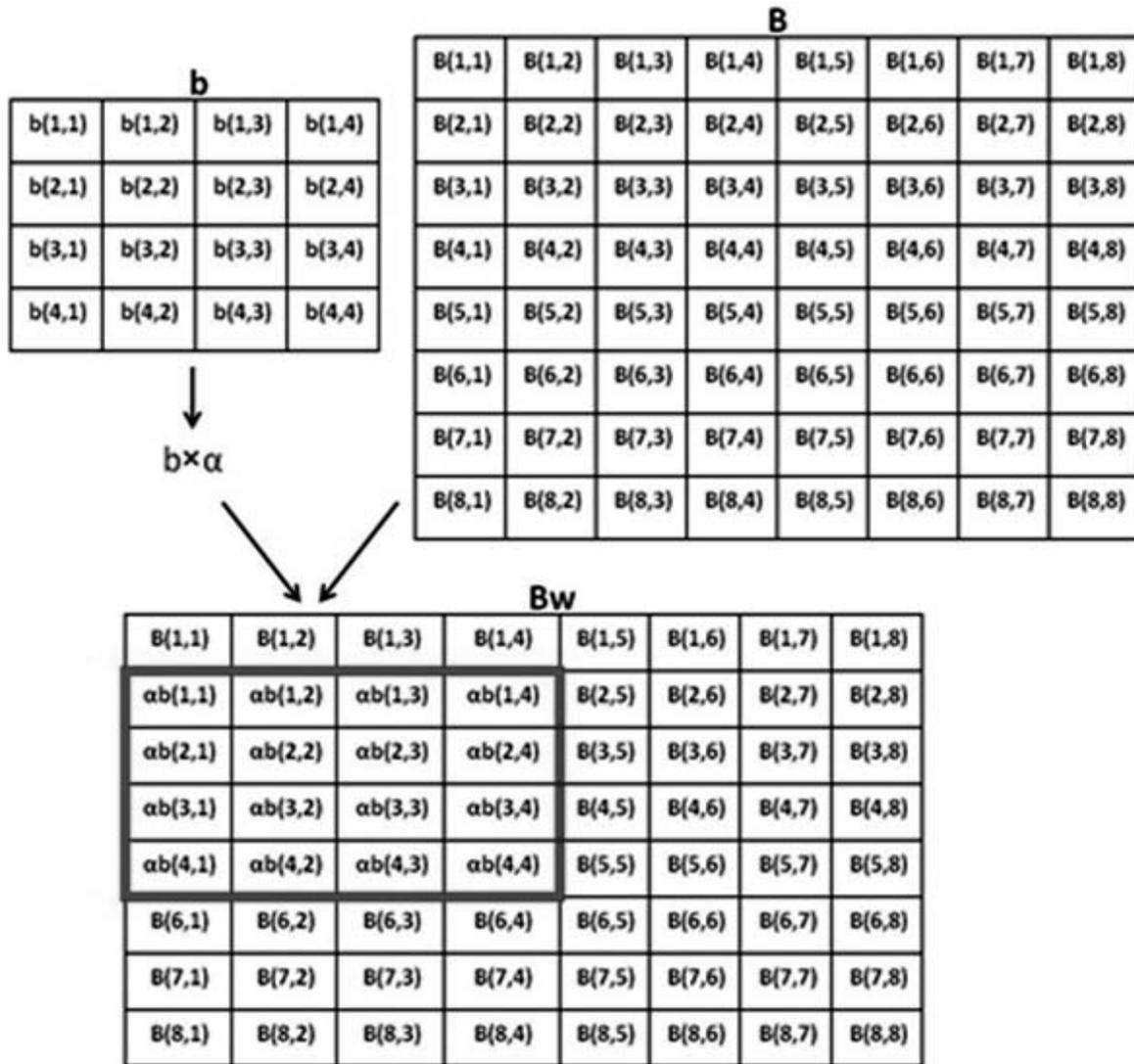


Figure 1. Structure of embedding watermark block (b) in host block (B)

PSNR is a statistical measure of similarity between cover image and the manipulated image which is measured in  $dB$ . Calculation of MSE and PSNR for cover image and watermarked image  $I_w$  is represented in the following equations.

$$MSE = \frac{1}{MN} \sum_{M,N} [I(i,j) - I_w(i,j)]^2 \quad (17)$$

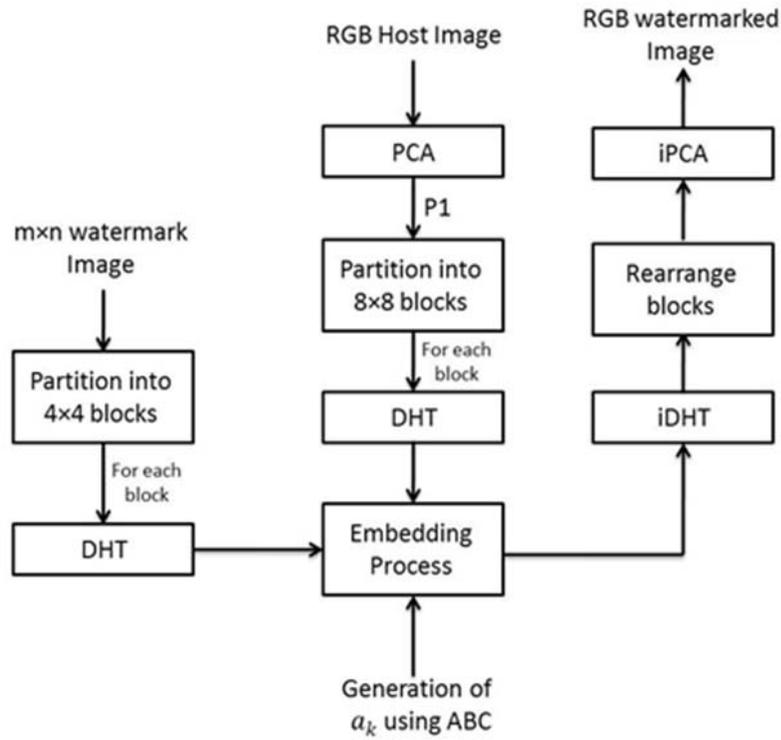


Figure 2. Proposed embedding process

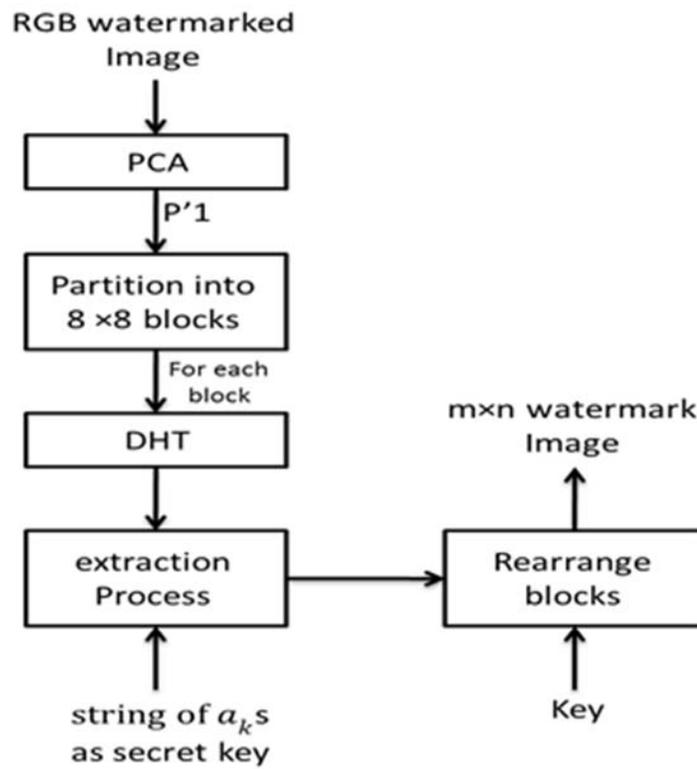


Figure 3. Proposed extraction process

$$PSNR = 10 \log_{10} \frac{R^2}{MSE} \quad (18)$$

in these equation,  $M$  and  $N$  are dimensions of the image.

$NC$  is a measure of similarity between main watermark image  $W$  and extracted watermark image  $W'$ , which can be calculated as follows:

$$NC = \frac{\sum_{i=1}^m \sum_{j=1}^n W(i,j) \cdot W'(i,j)}{\sum_{i=1}^m \sum_{j=1}^n (W(i,j))^2} \quad (19)$$

$m$  and  $n$  are dimensions of the watermark.

Value of  $NC$  is between 0 and 1. If it is 1 for two images, it means that images are completely similar. Objective function is calculated as follows:

$$f = \frac{1}{PSNR} + \sum_{i=1}^{NA} (1 - NC_i) \quad (20)$$

where  $NA$  is the number of attacks applied to images which is equal to 12. In order to optimize solutions, this function should be minimized.  $ABC$  algorithm finds the solution with minimum  $f$  as the optimal solution and its values are used as threshold values for final watermarking of the image.

#### 4. Experimental Results

In this section, performance of the proposed watermarking method using  $ABC$  algorithm is evaluated. For this purpose, seven standard  $512 \times 512$  RGB images are used as host image and two  $64 \times 64$  images are used as watermark which are shown in Figure 4. In order to evaluate robustness of the proposed method common processes are applied to the watermarked images which are represented in Table 1. In all executions, number of food sources is 20, Limit is 30, maximum number of rounds is 100, number of attacks is 8,  $X_j^{min}$  is 10 and  $X_j^{max}$  is 80. Number of dimensions of each solution should be equal to number of watermarked blocks, that is  $(m/4) \times (n/4)$  for a watermark of dimension  $m \times n$ .

Attack Number	Distortion attacks.
1	Median filter (3×3)
2	Gaussian filter (5×5)
3	Average filter (3×3)
4	Sharpening filter
5	Histogram equalization
6	Gamma correction (0.2)
7	Gaussian noise (0.01)
8	Salt & pepper noise (0.01)
9	Cropping 1/4 (center)
10	Scaling 512_256_512
11	Rotation (anticlockwise 5°)
12	JPEG compression (70)

Table 1. Attacks description



Figure 4. (a- g) Standard color images , (h) and (i) binary watermark images

In order to find the proper interval for selecting scaling factors  $[X_j^{min}, X_j^{max}]$ , watermarking is performed on Lena image with scaling factors in the range of 1 to 100 with 0.01 steps and PSNR is obtained for each step. Then attacks represented in Table 1 are applied to the watermarked image and after extracting the watermark, average NC is calculated. Figure 5 shows PSNR and average NC changes in this interval. As can be seen in Figure 5(a), by increasing scaling factors, PSNR is reduced. If a PSNR greater than 52 is desired, robustness factor should be less than 80. On the other hand, as threshold increases, NC is also increased. For average NC greater than 0.8 scaling factor should be greater than 10. Therefore  $[10, 80]$  is selected for initializing scaling. ABC algorithm determines best scaling factor after 100 rounds and uses these values to perform watermarking. Figure 6 shows best execution of the algorithm on Lena image after 30 times running program.

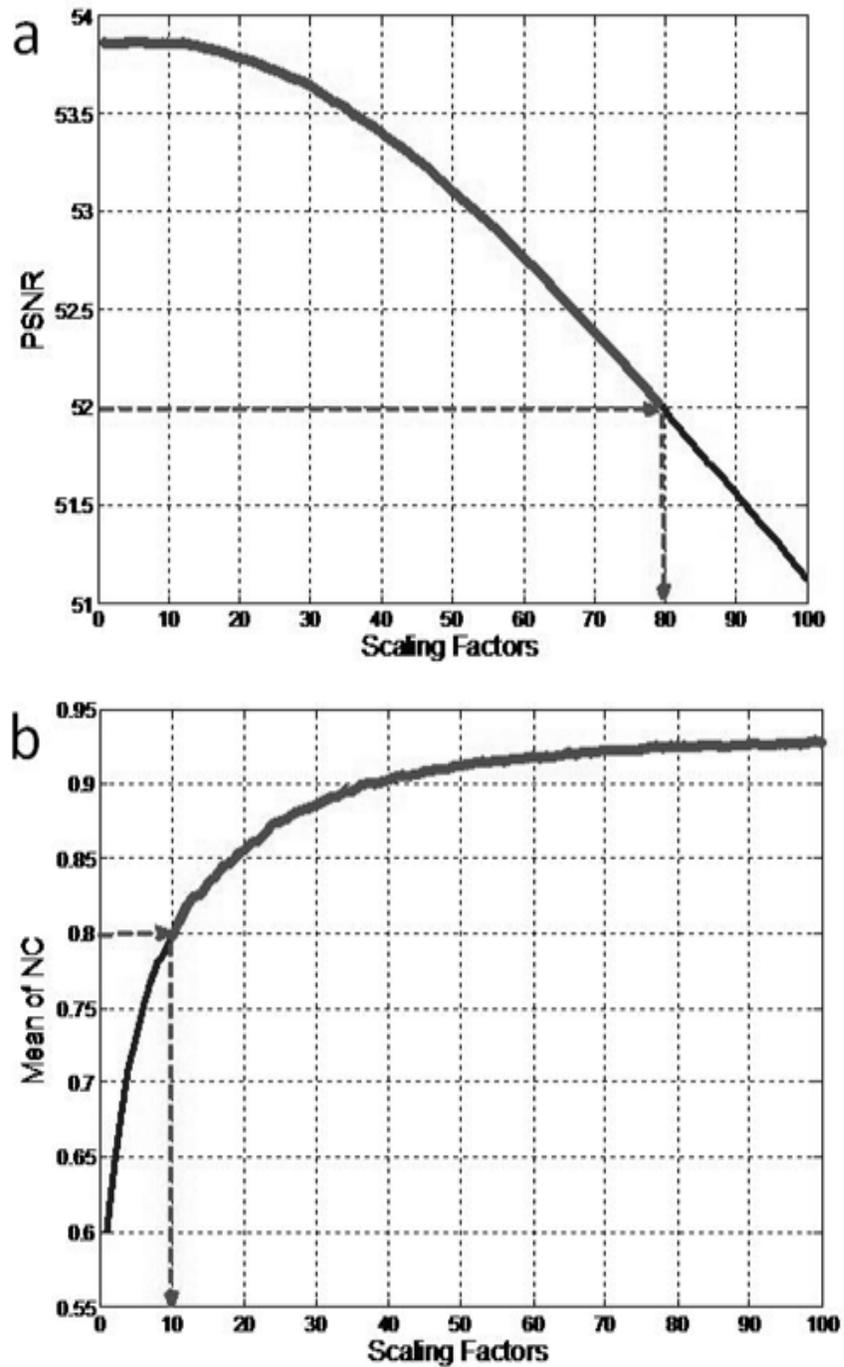


Figure 5. Effect of scaling factor on (a) PSNR and (B) mean of NC for lena test image

In order to evaluate transparency of the watermarked image, the structural similarity index (SSIM) is used besides to PSNR and NC. The dynamic range of NC is [0,1] while the best value 1 is achieved when two images are identical. Similarly, the best value of SSIM is also 1. Table 2 shows the obtained results. High values of PSNR and SSIM indicate high quality of the watermarked image. In addition, values of NC which are 1 in the absence of any attacks validate accuracy of the proposed method in extracting the watermark.

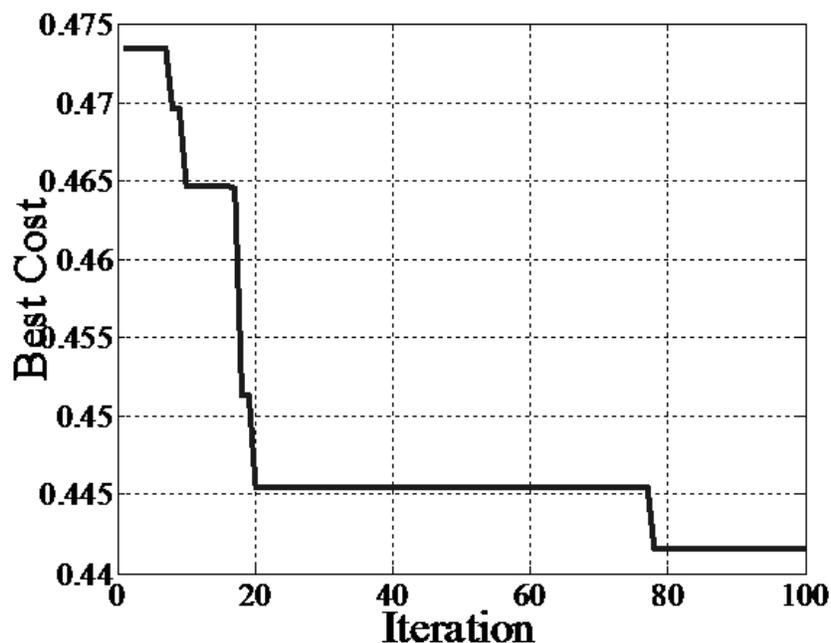


Figure 6. Objective function of ABC after 100 iterations for lena test image

Watermark image	M			A		
	PSNR	SSIM	NC	PSNR	SSIM	NC
House	53.65	0.997	1	53.8777	0.997	1
Lena	53.60	0.998	1	53.9744	0.998	1
Opera	51.25	0.997	1	52.3448	0.997	1
Baboon	52.71	0.997	1	53.4106	0.997	1
Sailboat	52.83	0.996	1	51.5736	0.996	1
F16	52.39	0.996	1	52.6089	0.996	1
Peppers	51.25	0.995	1	50.2613	0.995	1
Average	52.52	0.996	1	52.57	0.996	1

Table 2. Experimental values of PSNR, SSIM and NC for watermarked images

Quality of the watermark extracted is calculated using NC. In order to evaluate robustness of the proposed method, different processes and attacks are applied to the watermarked image and after extracting the watermark, value of NC is calculated. These attacks include filtering, adding different noises, cropping, rotation, scaling, compression and ... which are described in Tables 3 and 4. These tables show NC values for considered images using both watermark images.

Transparency and robustness of the proposed method is compared with methods proposed by V. Santhi et.al [7] and Soltani and Mansoori [9] in the following. These methods were described briefly in literature review. In both methods, hadamard transform is used to embed the watermark and no optimization is used. For empirical analyzes, all cover images are 512x512 RGB images and

Image	House		Lena		Opera		Baboon	
Attack description		<b>A</b>		<b>A</b>		<b>A</b>		<b>A</b>
No Attack	1	1	1	1	1	1	1	1
Median filter (3×3)	0.9950	0.9967	0.9963	0.9919	0.9938	0.9906	0.9938	0.9948
Gaussian filter (5×5)	0.9583	0.9520	0.9685	0.9532	0.9766	0.9583	0.98668	0.9634
Average filter (3×3)	0.9958	0.9919	0.9959	0.9900	0.9910	0.9952	0.9997	0.9904
Sharpening filter 0.9993	0.9980	0.9961	0.9934	0.9932	0.9910	0.9890	0.9817	
Histogram equalization	0.9971	1	0.9995	1	0.9890	0.9939	0.9995	0.9995
Gamma correction (0.2)	1	1	1	1	1	1	1	1
Gaussian noise (0.01)	0.9715	0.9737	0.9847	0.9854	0.9839	0.9839	0.9708	0.9825
Salt & pepper noise (0.01)	0.9865	0.9836	0.9807	0.9853	0.9895	0.9707	0.9717	0.9863
Cropping 1/8 (center)	0.9830	0.9894	0.9858	0.9802	0.9966	0.9895	0.9812	0.9855
Scaling 512_1024_512	1	1	1	1	1	1	1	0.9990
Scaling 512_256_512	0.9392	0.9328	0.9287	0.9221	0.9360	0.9363	0.9444	0.9431
Rotation (anticlockwise 5°)	0.8518	0.8097	0.8128	0.8085	0.8180	0.8018	0.8035	0.8007
Rotation (anticlockwise 45°)	1	1	1	1	1	1	1	1
Rotation (anticlockwise 90°)	0.9629	0.9702	0.9720	0.9745	0.9360	0.9508	0.9791	0.9810
JPEG compression (70)	0.9539	0.9641	0.958	0.9502	0.9609	0.9551	0.9570	0.9553
JPEG compression (80)	0.9919	0.9856	0.981	0.9766	0.9795	0.9768	0.9810	0.9741
JPEG compression (90)	1	1	1	1	1	1	1	1
Average	0.9770	0.974872	0.9755	0.9728	0.9746	0.9718	0.9754	0.9736

Table 3. Experimental values of NC obtained by taking over the House, Lena, Opera and Baboon host images corresponding to the 17 distortion attacks

watermark image is a binary 64×64 image. All methods are simulated using MATLAB 2015R on a PC with Core 2 Duo, 4GB RAM and Windows 7. In simulations, parameters used for each method are the ones mentioned in the corresponding papers. In the following, results are investigated in terms of robustness and transparency. Better results are shown in bold.

Table 5 shows PSNR and SSIM compared to methods proposed by V. Santhi [7] and Soltani and Mansoori [9] for the considered images while no attack is applied. Results indicate high quality of the water marked image using our proposed method. In order to compare robustness of the proposed method with cited methods against attacks describe in table 1, mean of NC is calculated for each image and results for each method are shown in Table 6. According to the results, the robustness of our watermarking scheme is better than the cited methods, in average. Employing hadamard transform and using ABC for optimizing multiple scaling factors increases robustness of our method. Since a secret key is used for extracting watermark, security of this method is high and false positive error may not occur.

Image	Sailboat		F16		Peppers	
Attack description		A		A		A
No Attack	1	1	1	1	1	1
Median filter (3×3)	0.9960	0.99902	0.99203	0.9900	0.9980	0.9904
Gaussian filter (5×5)	0.9544	0.9563	0.9615	0.8195	0.9575	0.9515
Average filter (3×3)	0.9915	0.9998	0.9908	0.9959	0.9915	0.9919
Sharpening filter	0.9956	0.9944	1	1	0.9875	0.9846
Histogram equalization	1	1	1	1	0.9968	0.9985
Gamma correction (0.2)	1	1	1	1	1	1
Gaussian noise (0.01)	0.9871	0.9868	0.9817	0.9805	0.9846	0.9829
Salt & pepper noise (0.01)	0.9846	0.9812	0.9795	0.9782	0.9848	0.9846
Cropping 1/4 (center)	0.9876	0.9821	0.9999	0.9928	0.9843	0.9809
Scaling 512_1024_512	0.9998	1	1	1	0.9998	0.9998
Scaling 512_256_512	0.9016	0.8804	0.9465	0.9219	0.8501	0.8193
Rotation (anticlockwise 5°)	0.8094	0.8051	0.8146	0.8114	0.8084	0.8031
Rotation (anticlockwise 45°)	1	1	1	1	1	1
Rotation (anticlockwise 90°)	0.9812	0.9852	0.9822	0.9811	0.9595	0.9736
JPEG compression (70)	0.9679	0.9667	0.9565	0.9533	0.9604	0.9686
JPEG compression (80)	0.9998	0.9921	0.9922	0.9929	0.9998	0.9998
JPEG compression (90)	1	1	1	1	1	1
Average	0.9753	0.9738	0.9776	0.9676	0.9701	0.9683

Table 4. Experimental values of NC obtained by taking over the Sailboat, F16 and Peppers host images corresponding to the 17 distortion attacks

## 5. Conclusion

In this paper, a robust watermarking method using hadamard transform for color images is proposed. In order to resolve correlation between RGB channels, watermark is embedded in principle component of PCA. Using PCA as an uncorrelated color space increases efficiency of RGB channels and transparency of the watermark. To improve robustness and transparency of watermark, multiple scaling factors are used in embedding process and these factors are optimized by ABC algorithm. Robustness of this method is evaluated by applying different attacks to test images and the results indicate that transparency and robustness of the proposed method have increased compared to other existence methods. Watermark extraction is performed blindly and FP error may not occur. Generalizing this method to robustness against other attacks and processes might be investigated in future works.

Images/ Methods	Proposed Method		[7] V. Santhi. et. al		Mansoori and Soltani [9]	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
House	53.65	0.997	48.59	0.994	52.36	0.996
Lena	53.60	0.997	47.06	0.993	50.22	0.994
Opera	51.15	0.996	49.75	0.994	51.23	0.996
Baboon	52.71	0.996	48.16	0.994	50.70	0.994
Sailboat	52.83	0.997	51.26	0.996	52.28	0.997
F16	52.39	0.997	47.06	0.992	51.06	0.995
Peppers	51.25	0.996	50.03	0.995	52.51	0.996
Average	52.52	0.996	48.84	0.994	51.48	0.995

Table 5. The PSNR and SSIM results of the proposed scheme comparing with existing schemes[7, 9]

images	Proposed Method	[7] V. Santhi. et. al	Mansoori and Soltani[9]
House	0.9770	0.9119	0.9106
Lena	0.9755	0.9088	0.9093
Opera	0.9746	0.9328	0.9202
Baboon	0.9754	0.8908	0.8914
Sailboat	0.9753	0.9203	0.9011
F16	0.9776	0.9047	0.9158
Peppers	0.9701	0.9109	0.9030
Average	0.9750	0.9114	0.9073

Table 6. The mean of NC results of the proposed scheme comparing with existing schemes [7, 9] against various attacks

## References

- [1] Maity, S. P., Maity, S. (2009). Multistage Spread Spectrum Watermark Detection Technique Using Fuzzy Logic, *IEEE Signal Processing Letters*, 16, p. 245-248.
- [2] Patra, J. C., Phua, J. E., Bornand, C. (2010). A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression, *Digital Signal Processing*, 20, p. 1597-1611, 12.
- [3] Elshazly, E. H., Faragallah, O. S., Abbas, A. M., Ashour, M. A., El-Rabaie, E.-S. M., Kazemian, H. (2015). Robust and secure fractional wavelet image watermarking, *Signal, Image and Video Processing*, 9, p. 89-98.
- [4] Lai, C.-C. (2011). An improved SVD-based watermarking scheme using human visual characteristics, *Optics Communications*, 284, p. 938-944, 2011.
- [5] Rawat, S., Raman, B. (2012). A publicly verifiable lossless watermarking scheme for copyright protection and ownership assertion, *AEU-International Journal of Electronics And Communications*, 66, p. 955-962, 11.

- [6] Maity, S. P., Kundu, M. K. (2010). DHT domain digital watermarking with low loss in image informations, *Aeu-International Journal of Electronics And Communications*, 64, p. 243-257.
- [7] Santhi, V., Arulmozhivarman, P. (2013). Hadamard transform based adaptive visible/invisible watermarking scheme for digital images, *Journal of Information Security and Applications*, 18, p. 167-179.
- [8] Falkowski, B., Lim, L.-S. (2000). Image watermarking using Hadamard transforms, *Electronics Letters*, 36, p. 211-213.
- [9] Mansoori, E., Soltani, S. (2016). A new semi-blind watermarking algorithm using ordered Hadamard transform, *The Imaging Science Journal*, 64, p. 204-214.
- [10] Al-Otum, H. M., Samara, N. A. (2010). A robust blind color image watermarking based on wavelet-tree bit host difference selection, *Signal Processing*, 90, p. 2498-2512, 8.
- [11] Rastegar, S., Namazi, F., Yaghmaie, K., Aliabadian, A. (2011). Hybrid watermarking algorithm based on Singular Value Decomposition and Radon transform, *AEU - International Journal of Electronics and Communications*, 65, p. 658-663, 7// 2011.
- [12] Su, Q., Niu, Y., Wang, G., Jia, S., Yue, J. (2014). Color image blind watermarking scheme based on QR decomposition, *Signal Processing*, 94, p. 219-235.
- [13] Ghafoor, Abdul., Imran, Muhammad. (2012). A non-blind color image watermarking scheme resistant against geometric attacks. *Radioengineering*. 21.p. 1246-1251.
- [14] Ali, M., Ahn, C. W., Pant, M., Siarry, P. (2015). An image watermarking scheme in wavelet domain with optimized compensation of singular value decomposition via artificial bee colony, *Information Sciences*, 301, p. 44-60, 4/20.
- [15] Gupta, M., Parmar, G., Gupta, R., Saraswat, M. (2015). Discrete wavelet transform-based color image watermarking using uncorrelated color space and artificial bee colony, *International Journal of Computational Intelligence Systems*, 8, p. 364-380.
- [16] Ansari, A., Pant, M., Ahn, C. W. (2016). ABC optimized secured image watermarking scheme to find out the rightful ownership, *Optik-International Journal for Light and Electron Optics*, 127, p. 5711-5721.
- [17] Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization, Technical report-tr06, Erciyes university, engineering faculty, *Computer Engineering Department*.
- [18] Santhi, V., Thangavelu, A. (2009). DWT-SVD combined full band robust watermarking technique for color images in YUV color space, *International Journal of Computer Theory and Engineering*, 1, p. 424.
- [19] Yin, C. Q., Li, L., Lv, A. Q., Li, Q. (2007). Color Image Watermarking Algorithm Based on DWT-SVD, *In: IEEE International Conference on Automation and Logistics*, p. 2607-2611.
- [20] Sun, X., Bo, S. (2010). A blind digital watermarking for color medical images based on PCA, *In: IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS)*, p. 421-427.
- [21] Rawat, S., Raman, B. (2010). A new robust watermarking scheme for color images, *In: IEEE 2nd International Advance Computing Conference (IACC)*, p. 206-209.
- [22] Ho, T., Shen, J., Tan, S. H. (2003). Robust digital image-in-image watermarking algorithm using the fast Hadamard transform, *In: International Symposium on Optical Science and Technology*, p. 76-85.
- [23] Fami, E. S., Samavi, S., Kaviani, H. R., Radani, Z. M. (2012). Adaptive watermarking in Hadamard transform coefficients of textured image blocks, *In: Artificial Intelligence and Signal Processing (AISP), 2012 16th CSI International Symposium on*, p. 503-507.