

## Research Article

# Image Watermarking Algorithm Based on Multiobjective Ant Colony Optimization and Singular Value Decomposition in Wavelet Domain

**Khaled Loukhaoukha**

*Department of Electrical and Computer Engineering, Laval University, Québec, QC, Canada 1K 7P4*

Correspondence should be addressed to Khaled Loukhaoukha; [khaled.loukhaoukha.1@ulaval.ca](mailto:khaled.loukhaoukha.1@ulaval.ca)

Received 6 February 2013; Revised 19 April 2013; Accepted 14 May 2013

Academic Editor: Peiping Shen

Copyright © 2013 Khaled Loukhaoukha. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a new optimal watermarking scheme based on discrete wavelet transform (DWT) and singular value decomposition (SVD) using multiobjective ant colony optimization (MOACO). A binary watermark is decomposed using a singular value decomposition. Then, the singular values are embedded in a detailed subband of host image. The trade-off between watermark transparency and robustness is controlled by multiple scaling factors (MSFs) instead of a single scaling factor (SSF). Determining the optimal values of the multiple scaling factors (MSFs) is a difficult problem. However, a multiobjective ant colony optimization is used to determine these values. Experimental results show much improved performances of the proposed scheme in terms of transparency and robustness compared to other watermarking schemes. Furthermore, it does not suffer from the problem of high probability of false positive detection of the watermarks.

## 1. Introduction

With the advent of numeric era at the end of 20th century, the exchange of digital documents became a very easy task. This extraordinary technical revolution from analog to numerical technology was not achieved without generating anxiety in terms of the protection of the authors rights since multimedia documents can be quite easily duplicated, modified, and illegally attacked without deterioration. Affected by significant revenue losses multimedia documents author's are motivated more than ever to secure their documents. In this context digital watermarking was introduced: it consists of inscribing invisible (or visible) data into the multimedia documents. This is done in two stages: embedding and extracting process. Digital watermarking schemes for images can be classified into different classes according to embedding domain, embedding rule, imperceptibility, and permanency.

In terms of robustness, the watermarking algorithm can be classified into three categories: fragile, semifragile, and robust. Fragile watermarking is designed to detect any modification in such a way that slight modifications or tampering on the watermarked image will destroy the

watermark. This type is employed to ensure the integrity and image authenticity. Conversely, robust watermarking is designed to be resistant against attacks that attempt to remove or destroy the watermark without degrading the visual quality of the watermarked image significantly. Robust watermarking is typically employed for copyright protection and ownership verification. Semifragile watermarking combines the properties of fragile and robust watermarking in order to detect unauthorized manipulations while still being robust against authorized manipulations.

Generally, watermarking algorithms operate either in the spatial domain or in a transform domain such as discrete wavelet transform (DWT) [1–3], discrete Fourier transform (DFT) [4–6], and discrete cosine transform (DCT) [7–9]. Spatial domain watermarking has the advantage of low calculation complexity compared to that in transform domains. However, most watermarking schemes in the scientific literature operate in the transform domain because it provides enhanced imperceptibility and robustness compared to those in spatial domain. However, decomposition of images in a standard basis set using the transform domain does not necessarily leads to the optimal representation of an image.

Therefore, different representations were investigated for watermarking: these include nonnegative matrix factorization (NMF) [10–12] and singular value decomposition (SVD) [13–15]. Watermarking schemes based on SVD are advantageous since slight changes in the singular values do not affect significantly the image quality. Unfortunately, several SVD-based watermarking schemes suffer from typically high probability of false positive watermark detection [16–18].

According to embedding rule, watermarking algorithm can be classified into three categories: multiplicative, additive, or substitution. In multiplicative and additive embedding schemes, the trade-off between imperceptibility and robustness is controlled by a single scaling factor (SSF). Cox et al. [19] suggest to use multiple scaling factors (MSFs) instead of one. They state that single scaling factor may not be applicable for altering all the pixel values of the original image. Determining the optimal values of the multiple scaling factors can be viewed as optimization problem which is unfortunately a difficult problem. In this paper, a multiobjective ant colony optimization (MOACO) is used to solve this difficult problem and to determine these optimal values.

The rest of this paper is organized as follows. Section 2 reviews the basic concepts of the ant colony optimization (ACO). In Section 3, we presented the proposed watermarking algorithm based on discrete wavelet transform (DWT) and singular value decomposition (SVD). The proposed watermarking algorithm using multiobjective ant colony optimization is described in Section 4. The experimental results are provided in Section 5. Finally, the concluding remarks are given in Section 6.

## 2. Ant Colony Optimization Principle

The ant colony optimization (ACO) was introduced by Dorigo [20] as solution for hard optimization problems. It is inspired by observation of real ant colonies. Ants explore randomly the area surrounding their nest in order to find food. If an ant finds a food source, it evaluates and carries some food to the nest. During the return travels the ant deposits on the ground a chemical substance called pheromone trail. Other ants can smell the pheromone and follow it with some probability. This way, ants can communicate via pheromone and find the optimal path between the food source and the nest. This capability of real ant colony to find optimal paths has led to the definition of artificial ant colonies that can find the optimal solution for hard optimization problems [21] such as the traveling salesman problem. The outline of the generic ACO algorithm is presented in Figure 1.

The first point to take into account in ant colony optimization is how the colony is represented. For continuous variables, a colony of  $m$  ants is represented as  $m \times n$  matrix  $C$ , where  $C = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_m]^T$  such that  $\mathbf{x} = [x_1 \ x_2 \ \cdots \ x_n]$  is a vector of  $n$  design variables that corresponds to a single ant. The second point to consider is how to model the pheromone communication scheme. Socha and Dorigo [21] suggest to use a normal distribution for a continuous model implementation:

$$f_{\text{pheromone}}(x) = e^{-(x-x_{\min})^2/2\sigma^2}, \quad (1)$$

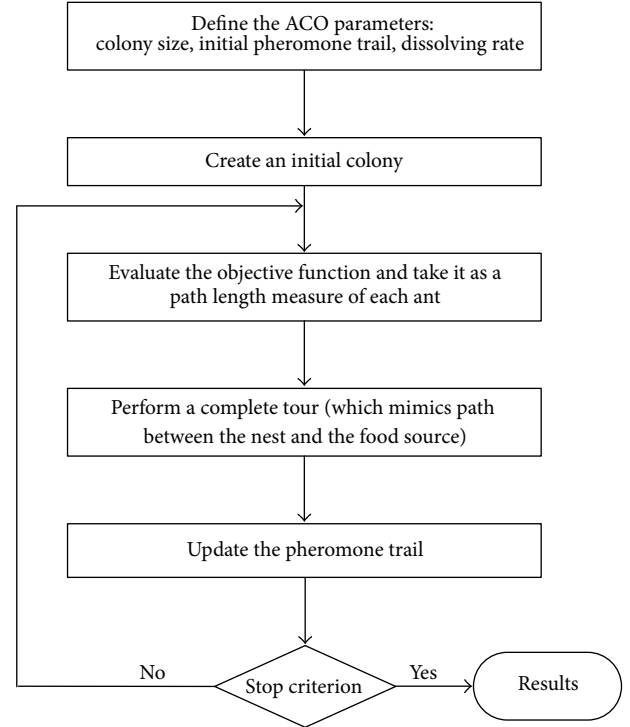


FIGURE 1: Generic ant colony optimization scheme [22].

where  $x_{\min}$  is the optimal point found within the design space and the standard deviation  $\sigma$  as an index of the ants aggregation around the current minimum. To initialize the algorithm,  $x_{\min}$  is randomly chosen in the design space, using a uniform distribution, and  $\sigma$  is taken at least three times greater than the length of the design space, to uniformly locate the ants within it. As shown in Figure 1, at each iteration, the ACO algorithm updates the values of each design variable, and this is for all the ants of the colony; that is, at each iteration each ant sets the values for the trial solution as per the distribution in (1). At the end, the pheromone distribution over the design space is updated by collecting the information acquired throughout the optimization steps. Since the pheromone is modeled by (1), it is necessary only to update  $x_{\min}$  and  $\sigma$  as

$$\sigma = \text{std}(\text{colony}), \quad (2)$$

where  $\text{std}(\text{colony})$  makes use of the colony of ants (candidate solution) to return a vector containing the standard deviation for each design variable [23]. The accumulation of pheromone increases in the vicinity of the candidate towards the optimum solution. However, to avoid premature convergence, negative update procedures are not discarded: for this, a simple method is used, which consists in dissolving the pheromone. The principle is to spread the amount of pheromone by changing the current standard deviation (for each variable) according to

$$\sigma_{\text{new}} = \gamma \cdot \sigma_{\text{old}}, \quad (3)$$

where  $\gamma > 1$  is the dissolving rate.

### 3. Watermarking Algorithm Based on SVD and DWT

The SVD-DWT watermarking algorithm presented in [14] is described by the following embedding and extraction processes.

*3.1. Embedding Process.* Consider an original image  $I$  of size  $N \times N$ , and let the watermark  $W$  be a binary image of size  $M \times M$ . The embedding process is as follows.

- (1) Decompose the original image  $I$  into  $3\ell + 1$  sub-bands by applying a  $\ell$ -level discrete wavelet transform (DWT).
- (2) Select one sub-band (SB) among the three following sub-bands:  $HH_\ell$ ,  $HL_\ell$ , and  $LH_\ell$ .
- (3) Compute the inverse DWT of the selected sub-band (SB):

$$X = \text{DWT}^{-1}(SB). \quad (4)$$

- (4) Apply a singular value decomposition (SVD) of matrix  $X$ :

$$X = U_X \cdot S_X \cdot V_X^T. \quad (5)$$

- (5) Encrypt watermark image  $W$  using image encryption algorithm to get encrypted watermark denoted by  $W_C$ .
- (6) Apply a singular value decomposition to watermark matrix  $W_C$ .

$$W = U_{W_C} \cdot S_{W_C} \cdot V_{W_C}^T. \quad (6)$$

- (7) Compute the one-way hash functions for matrices  $U_{W_C}$  and  $V_{W_C}$ :

$$H_U = \text{hash}(U_{W_C}), \quad (7)$$

$$H_V = \text{hash}(V_{W_C}).$$

- (8) Matrices  $U_{W_C}$  and  $V_{W_C}$  and their hash values  $H_U$  and  $H_V$  are stored in the private key. (Steps (5) and (7) are necessary to mitigate the problem of false positive detection of watermark. This solution was proposed in [24].)

- (9) Compute matrix  $S_Y$  according to

$$S_Y = S_X + \alpha \cdot S_{W_C}, \quad (8)$$

where  $\alpha$  is the watermark strength factor that controls the trade-off between imperceptibility and robustness of the watermarking scheme. This parameter can be used as single scaling factor (SSF) or multiple scaling factors (MSFs).

- (10) Compute matrix  $Z_W$ , according to

$$Z_W = U_X \cdot S_Y \cdot V_X^T. \quad (9)$$

- (11) Compute the discrete wavelet transform of matrix  $Z_W$ ,

$$SB_W = \text{DWT}(Z_W). \quad (10)$$

- (12) The watermarked image  $I_W$  is computed by applying the inverse  $\ell$ -level discrete wavelet transform to the modified sub-band  $SB_W$  and the  $3\ell$  unmodified sub-bands.

*3.2. Extracting Process.* The extracting process is summarized by the following steps.

- (1) A safety test is first done: hash values of matrices  $U_{W_C}$  and  $V_{W_C}$ , possibly altered by an attacker as  $\tilde{U}_W$  and  $\tilde{V}_W$ , are computed to give  $H_{\tilde{U}}$  and  $H_{\tilde{V}}$ . These hash values are compared with these stored during the embedding process:

$$\left\{ \begin{array}{l} \text{if } H_U = H_{\tilde{U}}, \quad \text{and} \quad H_V = H_{\tilde{V}} \longrightarrow \text{go to 2,} \\ \text{if } H_U \neq H_{\tilde{U}} \quad \text{or} \quad H_V \neq H_{\tilde{V}} \longrightarrow \text{stop (probable attack).} \end{array} \right. \quad (11)$$

- (2) Decompose the original and watermarked images,  $I$  and  $I_W$ , by applying the  $\ell$ -level discrete wavelet transform.
- (3) Select the same sub-band (SB) used in 2 of the watermark embedding process.  $SB_I$  and  $SB_{I_W}$  are, respectively, the subbands selected for the original and watermarked images.
- (4) Compute the inverse discrete wavelet transform of selected subbands  $SB_I$  and  $SB_{I_W}$ :

$$X_I = \text{DWT}^{-1}(SB_I), \quad (12)$$

$$X_{I_W} = \text{DWT}^{-1}(SB_{I_W}).$$

- (5) Apply the singular value decomposition on matrices  $X_I$  and  $X_{I_W}$ :

$$X_I = U_{X_I} \cdot S_{X_I} \cdot V_{X_I}^T, \quad (13)$$

$$X_{I_W} = U_{X_{I_W}} \cdot S_{X_{I_W}} \cdot V_{X_{I_W}}^T.$$

- (6) Compute matrix  $S_{\widehat{W}_C}$  as follows:

$$S_{\widehat{W}_C} = \frac{S_{X_{I_W}} - S_{X_I}}{\alpha}. \quad (14)$$

- (7) Determine the extracted encrypted watermark,  $\widehat{W}_C$ , by computing

$$\widehat{W}_C = \tilde{U}_{W_C} \cdot S_{\widehat{W}_C} \cdot \tilde{V}_{W_C}^T. \quad (15)$$

- (8) Decrypt  $\widehat{W}_C$  to get the extracted watermark  $\widehat{W}$ .

**3.3. Protection against False Positive Detection.** To protect the proposed image watermarking scheme against false positive detection, two countermeasures have been proposed in [24]. The first countermeasure consists of computing a *one-way hash function* for the matrices  $U_W$  and  $V_W$ . The hash function algorithm such as *message digest 5* (MD5) or *secure hash algorithm 1* (SHA-1) can be used for this purpose. During embedding process, the hash values of matrices  $U_W$  and  $V_W$  denoted, respectively, by  $H_U$  and  $H_V$  are stored in a private key. In the extracting process, the hash function is first computed from received (and possibly altered by an attacker) matrices  $U_{\widehat{W}}$  and  $V_{\widehat{W}}$ , denoted by  $H_{\widehat{U}}$  and  $H_{\widehat{V}}$ . Thus, if  $H_U \neq H_{\widehat{U}}$  or  $H_V \neq H_{\widehat{V}}$ , the watermark extracting process is stopped because  $U_{\widehat{W}} \neq U_W$  or  $V_{\widehat{W}} \neq V_W$ ; otherwise the watermark extracting process is performed. The authors indicate that the hash function test can be also be applied on a combined matrix from  $U_W$  and  $V_W$  such as  $U_W + V_W$  and  $U_W \times V_W$ .

The second countermeasure consists in encrypting the watermark before the embedding process. The watermark  $W$  is encrypted resulting in an encrypted watermark denoted by  $W_C$  which will then be embedded in the original image  $I$ . Suppose that for watermark extraction process, an attacker uses his own watermark  $\widehat{W}$ . Then matrices  $U_{\widehat{W}}$  and  $V_{\widehat{W}}$  will be used instead of proper matrices  $U_{W_C}$  and  $V_{W_C}$ . The first extracted watermark will be the same as  $\widehat{W}$ , but the decryption process must be performed since the embedded watermark is encrypted during the watermark embedding process. Therefore, the extracted watermark  $\widehat{W}$  will be a random-like image. Note that, if there is no attack, the first extracted watermark  $\widehat{W}_C$  will indeed be the encrypted image, but after decryption, the extracted watermark  $\widehat{W}$  will have a high correlation value with the original watermark  $W$ . Figures 2 and 3 illustrate an example of one-way hash function and encryption countermeasures, respectively, using Lena as original image  $I$  and LOGO as watermark  $W$ .

#### 4. Watermarking Algorithm Using Multiobjective Ant Colony Optimization

In general, watermarking schemes are either based on an additive or a multiplicative rule. The embedding rules themselves are usually of the following form, where  $I$  and  $I_W$  are, respectively, original and watermarked image (or their representation in other domains such as FFT, DCT, and DWT),  $\alpha$  is used to control the trade-off between imperceptibility and robustness of an image and generally is used as scaling factor:

$$\begin{aligned} I_W &= I + \alpha \cdot W \longrightarrow \text{additive rule,} \\ I_W &= I \cdot (1 + \alpha \cdot W) \longrightarrow \text{multiplicative rule.} \end{aligned} \quad (16)$$

Cox et al. [19] suggest the use of multiple scaling factors instead of one. They state that a single scaling factor may not be applicable for altering all the values of original image  $I$ . Determining the optimal values of these multiple scaling factors is a complex computational problem. To solve it, we propose to use a multi-objective ant colony optimization (MOACO).

TABLE I: Imperceptibility test results.

Image	Algorithm	PSNR ( $I, I_W$ ) (dB)	NC ( $W, \widehat{W}$ )
Cameraman	MSF-MOACO	50.118	1.000
	SSF	49.341	1.000
	Xianghong et al. [1]	49.075	1.000
	Liu and Li [3]	45.529	0.998
	Pai and Ruan [25]	55.289	1.000
Goldhill	Ouhsain and Hamza [12]	53.563	1.000
	MSF-MOACO	54.219	1.000
	Using SSF	51.124	1
	Xianghong et al. [1]	49.075	0.999
	Liu and Li [3]	45.320	1.000
Lena	Pai and Ruan [25]	54.844	1.000
	Ouhsain and Hamza [12]	51.892	1.000
	MSF-MOACO	50.942	1.000
	SSF	48.899	1
	Xianghong et al. [1]	49.075	1.000
Man	Liu and Li [3]	45.472	0.997
	Pai and Ruan [25]	55.186	1.000
	Ouhsain and Hamza [12]	53.192	1.000
	MSF-MOACO	51.026	1.000
	SSF	50.735	1.000
Man	Xianghong et al. [1]	49.075	1.000
	Liu and Li [3]	45.323	1.000
	Pai and Ruan [25]	55.186	0.998
	Ouhsain and Hamza [12]	52.569	0.999

Figure 4 illustrates block diagram of multiobjective optimization which is closed-loop control system. System input is multiple scaling factors and objective measure as system output; this measure is calculated from original image  $I$ , watermarked image  $I_W$ , watermark  $W$ , and the  $(T + 1)$  extracted watermarks ( $\widehat{W}$  and  $\widehat{W}_j$ , where  $j = \{1, 2, \dots, T\}$ ) under attacks.

The steps for applying MOACO into SVD-DWT watermarking scheme are enumerated below.

- (1) Define the colony size, the initial pheromone trail, the dissolving rate ( $\sigma$ ), the objective function, and a generation number as the algorithm stopping criterion.
- (2) Using (17), generate randomly an initial population of ants, which constitute a set of potential solutions. Each ant is denoted by  $X = \{x_1, x_2, \dots, x_n\}$ ,

$$f_{\text{pheromone}}(x_i) = e^{((x_i - x_i^*)^2 / 2\sigma_i^2)}, \quad (17)$$

where  $x_i^*$  is the  $i$ th coordinate of the best point found by optimization within the design space at the current iteration and  $\sigma_i$  is the ants aggregation index for the  $i$ th coordinate of the design space. At the first iteration,  $x_i^*$  is chosen according a uniform

TABLE 2: Correlation coefficient comparison results between MSF-MOACO and SSF algorithms and other algorithms.

Image	Algorithm	SP	GF	CR	CM	SH	SC	HE	QN	RW	CA
Cameraman	MSF-MOACO	0.956	0.972	0.947	0.902	0.989	1	0.985	0.990	1.000	1.000
	SSF	0.726	0.907	0.894	0.729	0.976	1.000	0.959	0.958	0.998	0.997
	Xianghong et al. [1]	0.495	0.847	0.983	0.621	0.580	0.993	0.621	0.750	0.850	0.914
	Liu and Li [3]	0.936	0.872	0.983	0.470	0.940	0.846	0.954	0.743	0.865	0.907
	Pai and Ruan [25]	0.407	0.898	0.975	0.537	0.697	0.989	0.474	0.504	0.867	0.924
	Ouhsain and Hamza [12]	0.710	0.840	0.881	0.775	0.653	0.982	0.920	0.532	0.860	0.902
Goldhill	MSF-MOACO	0.935	0.957	0.935	0.959	0.984	1	0.975	0.972	0.998	1.000
	SSF	0.885	0.936	0.998	0.930	0.950	1.000	0.977	0.965	0.995	0.999
	Xianghong et al. [1]	0.694	0.858	0.983	0.633	0.713	0.986	0.440	0.570	0.850	0.914
	Liu and Li [3]	0.947	0.906	0.980	0.545	0.951	0.870	0.974	0.714	0.867	0.918
	Pai and Ruan [25]	0.374	0.827	0.976	0.293	0.370	0.926	0.313	0.220	0.867	0.912
	Ouhsain and Hamza [12]	0.789	0.835	0.852	0.796	0.675	0.978	0.950	0.523	0.850	0.925
Lena	MSF-MOACO	0.933	0.974	0.918	0.948	0.976	0.999	0.974	0.967	1.000	1.000
	SSF	0.774	0.943	0.867	0.944	0.982	1.000	0.988	0.974	0.998	0.995
	Xianghong et al. [1]	0.616	0.866	0.983	0.640	0.667	0.994	0.587	0.625	0.850	0.914
	Liu and Li [3]	0.926	0.845	0.953	0.541	0.952	0.826	0.987	0.716	0.863	0.912
	Pai and Ruan [25]	0.383	0.920	0.980	0.442	0.681	0.994	0.514	0.393	0.867	0.918
	Ouhsain and Hamza [12]	0.769	0.882	0.853	0.775	0.685	0.983	0.962	0.546	0.849	0.936
Man	MSF-MOACO	0.932	0.952	0.964	0.978	0.966	1.000	0.972	0.970	1	1.000
	SSF	0.588	0.845	0.930	0.905	0.949	0.999	0.977	0.963	0.990	0.998
	Xianghong et al. [1]	0.738	0.876	0.983	0.641	0.692	0.993	0.828	0.456	0.850	0.914
	Liu and Li [3]	0.944	0.876	0.994	0.532	0.966	0.845	0.983	0.803	0.865	0.915
	Pai and Ruan [25]	0.412	0.866	0.981	0.324	0.432	0.983	0.468	0.282	0.867	0.920
	Ouhsain and Hamza [12]	0.745	0.812	0.890	0.756	0.632	0.985	0.945	0.556	0.875	0.952

TABLE 3: Correlation coefficient comparison results between the MSF-MOACO algorithms and other algorithms.

Image	Algorithm	GC	DI	RT	MB	TR
Cameraman	MSF-MOACO	0.970	0.994	0.927	0.989	0.755
	Xianghong et al. [1]	0.847	0.001	0.368	0.428	0.411
	Liu and Li [3]	0.988	0.688	0.665	0.652	0.659
	Pai and Ruan [25]	0.616	0.532	0.421	0.546	0.462
	Ouhsain and Hamza [12]	0.940	0.463	0.201	0.980	0.524
	MSF-MOACO	0.979	0.979	0.904	0.974	0.951
Goldhill	Xianghong et al. [1]	0.915	0.001	0.493	0.516	0.444
	Liu and Li [3]	0.997	0.676	0.702	0.643	0.752
	Pai and Ruan [25]	0.313	0.267	0.271	0.304	0.278
	Ouhsain and Hamza [12]	0.955	0.423	0.322	0.975	0.563
	MSF-MOACO	0.970	0.970	0.916	0.966	0.877
Lena	Xianghong et al. [1]	0.875	0.001	0.446	0.584	0.419
	Liu and Li [3]	0.986	0.686	0.641	0.666	0.683
	Pai and Ruan [25]	0.535	0.435	0.332	0.527	0.374
	Ouhsain and Hamza [12]	0.953	0.452	0.356	0.970	0.550
	MSF-MOACO	0.924	0.968	0.847	0.962	0.875
Man	Xianghong et al. [1]	0.912	0.001	0.518	0.518	0.492
	Liu and Li [3]	0.990	0.683	0.689	0.636	0.766
	Pai and Ruan [25]	0.374	0.316	0.287	0.334	0.292
	Ouhsain and Hamza [12]	0.902	0.432	0.324	0.965	0.543
	MSF-MOACO	0.924	0.968	0.847	0.962	0.875



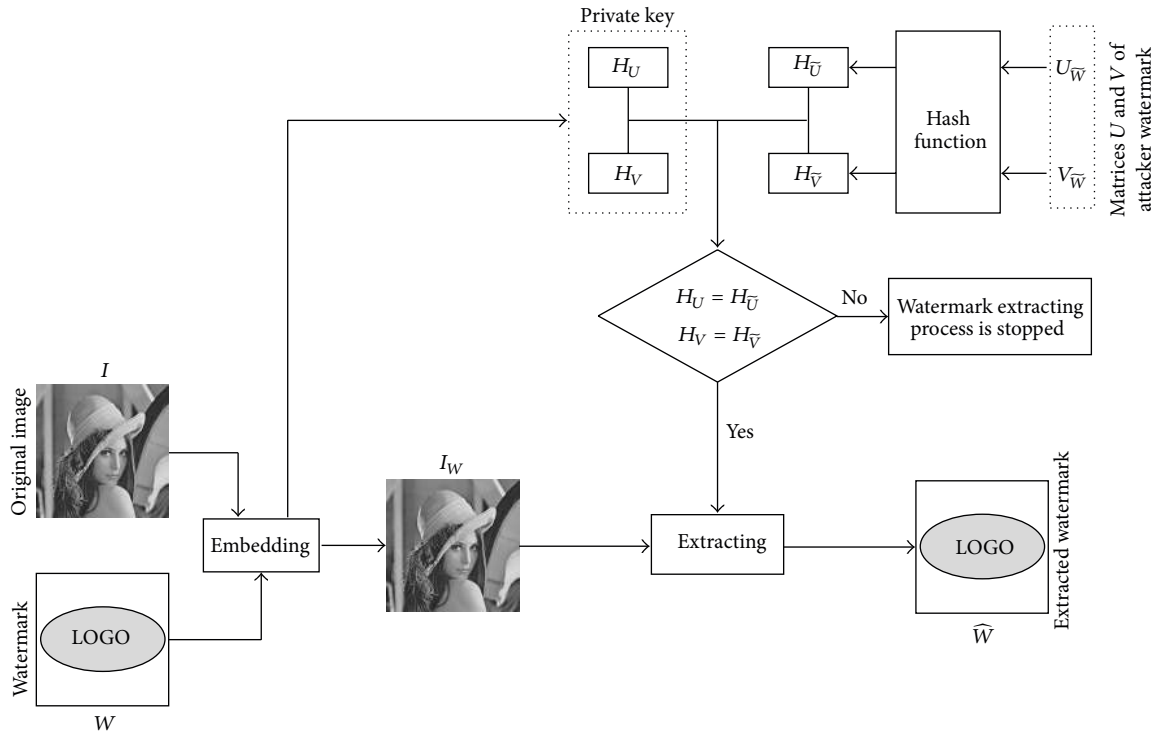


FIGURE 2: Example of the first countermeasure based on one-way hash function.

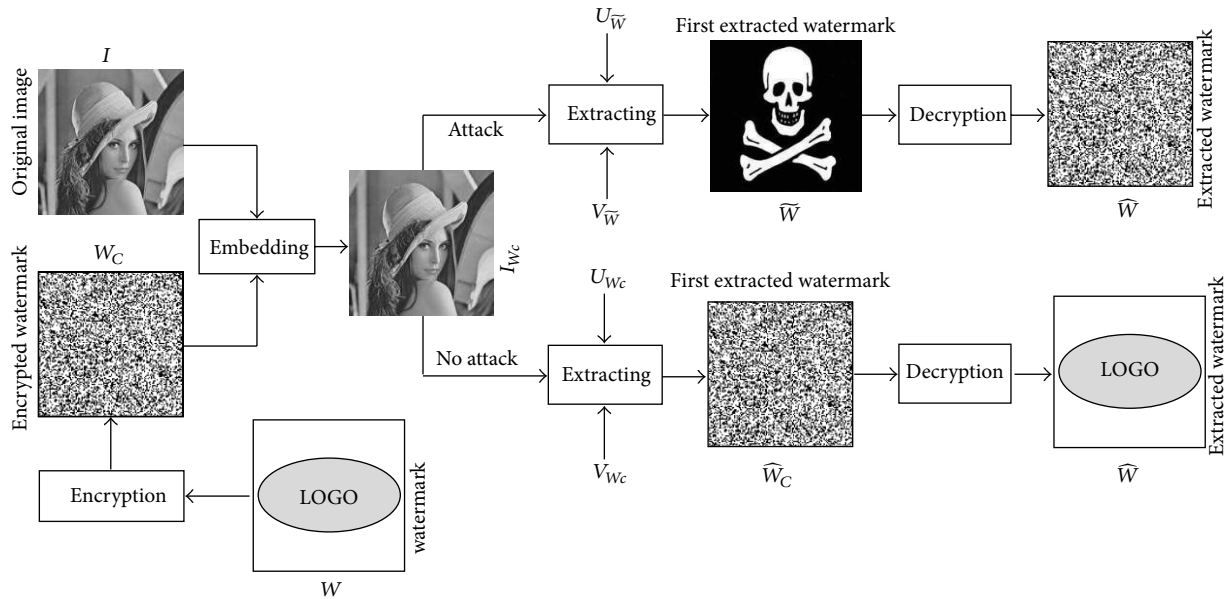


FIGURE 3: Example of the second countermeasure based on encryption.

distribution with  $\sigma_i$  taken as at least 3 times larger than the length of the search interval.

(3) For each ant  $X$  of the population, consider the following.

(i) Produce the watermarking image  $I_W$  using embedding process (Section 3.1) using the ant  $X$  as the watermark strength factor. Note that (8) in watermark embedding process is transformed into. (Note that,  $\alpha$  is changed into multiple

scaling factors instead of single scaling factor and  $\text{diag}(\alpha)$  is diagonal matrix created from the vector  $X$ , that is, the ant  $X$ )

$$S_Y = S_X + \text{diag}(\alpha) \cdot S_{W_C}. \quad (18)$$

(ii) Compute the normalized correlation  $NC(I, I_W)$  between original and watermarked images.  
 (iii) Apply a watermark attack out of a set of  $T$  selected attacks upon the watermarked image

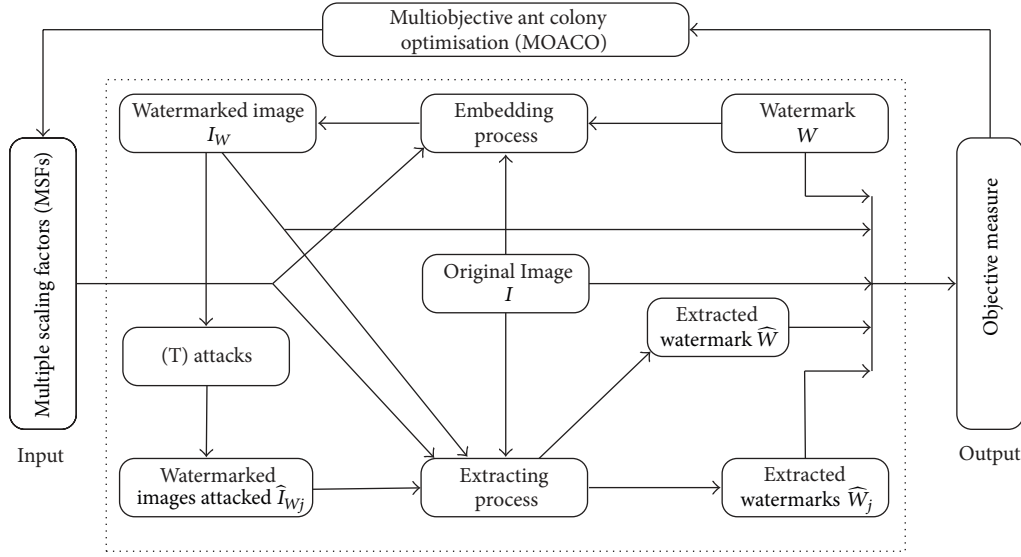


FIGURE 4: : Block diagram of multiobjective optimization.

$I_W$ . This leads to  $T$  different attacked watermarked images  $\{\hat{I}_W\}$  for each original watermarked image  $I_W$ .

- (iv) Extract the watermarks  $\hat{W}_i$  from the attacked watermarked images  $\hat{I}_W$  using the extraction process, as described in Section 3.2, where  $i = \{1, 2, \dots, T\}$ .
- (v) Compute the normalized correlation between the original watermark  $W$  and the set of extracted watermarks  $\{\hat{W}_i\}$ , that is,  $\{NC(W, \hat{W}_i)\}$ .
- (vi) Construct the vector of objective values,  $F(X)$ , defined as

$$F(p) = \left( \frac{1}{NC(I, I_{W_p})} \frac{1}{NC(W, \hat{W})} \frac{1}{NC(W, \hat{W}_{p,1})} \right. \\ \left. \times \frac{1}{NC(W, \hat{W}_{p,2})} \dots \frac{1}{NC(W, \hat{W}_{p,T})} \right)^T. \quad (19)$$

- (vii) Evaluate the vector of objective values according to the *exponential weighted method* for multiobjective optimization [26]:

$$F_{\text{obj}}(X) = \sum_{i=1}^{T+2} (e^{p \cdot w} - 1) \cdot e^{p \cdot (F(X) - F_0)}, \quad (20)$$

where  $p$ ,  $w$ , and  $F_0$  are positive constants. In experiments, we take  $p = 2$ ,  $w = 5$  et  $F_0 = 10$ .

- (4) Find the best ant  $X_{\text{best}} = \{x_1^*, x_2^*, \dots, x_n^*\}$  as the one having the smallest objective value  $F_{\text{obj}}$ .

- (5) Update the pheromone trail distribution using the formula (17); in this step the aggregation index for the  $i$ th dimension  $\sigma_i$  is given by

$$\sigma_i = \sqrt{\frac{1}{m} \sum_{i=1}^m y_i - \bar{y}}, \quad (21)$$

where  $y$  is the  $i$ th column of the colony matrix  $C$ ,  $\bar{y}$  is the mean value of the vector  $y$ , and  $m$  is the number of colony size.

- (6) Save the best ant  $X_{\text{best}}$  between this generation and the old one generation.
- (7) If the generation number is reached, the optimization process of the multiple scaling factors (MSFs) is terminated, else it goes to the next step.
- (8) Using the distribution of (17), generate a new population of ants and then go back to 3.

## 5. Experimental Results

To show the effectiveness of the proposed watermarking algorithm using multiple scaling factors optimized by multiobjective ant colony optimization instead of single scaling factor in terms of imperceptibility and robustness, a number of tests have been carried out using four  $256 \times 256$  gray-scale test images and a  $32 \times 32$  binary watermark, which are depicted in Figure 5. Note that  $\ell$  is set to be equal to 3.

We conducted several experiments to compare the imperceptibility and robustness of the proposed watermarking algorithm using multiple scaling factors optimized by multiobjective ant colony optimization (i.e., MSF-MOACO) and single scaling factor (SSF) with Xianghong et al. [1], Liu [3], Pai and Ruan [25], and Ouhsein and Hamza [12] algorithms. Table 1 lists the PSNR values of proposed and other algorithms for the same test images. It is clearly shown



FIGURE 5: Original images and binary watermark.



FIGURE 6: Watermarked images under different attacks.

that our algorithm using multiple scaling factors (MSF-MOACO) outperforms all other algorithms in terms of the visual quality of the reconstructed watermarked image except Pai and Ruan algorithm [25]. Furthermore, the normalized correlation between watermark  $W$  and extracted watermark  $\widehat{W}$  of the proposed algorithm (MSF-MOACO) is very close to unity.

For the robustness tests, ten different attacks were selected in conjunction to multiobjective optimization (i.e.,  $T = 10$ ). These attacks are salt and peppers noise (with a density of 0.05), Gaussian filtering ( $3 \times 3$ ), cropping (1/8 of the image center), JPEG compression ( $Q = 5$ ), sharpening, scaling ( $256 \rightarrow 512 \rightarrow 256$ ), histogram equalization, gray-scale

quantization (1 bit), re-watermarking using other watermarks that differ from the watermark (Letter A) and collusion attack using five watermarks, which are denoted by SP, GF, CR, CM, SH, SC, HE, QN, RW, and CA, respectively. The normalized correlation values between embedded and extracted watermarks under these attacks are depicted in Table 2, where the MSF-MOACO algorithm outperforms other algorithms [1, 3, 12, 25].

The effectiveness of the proposed algorithm (MSF-MOACO) is also tested against other attacks such as gamma correction ( $\gamma = 0.2$ ), dithering, rotation ( $25^\circ$ ), motion blur ( $45^\circ$ ), and translation (25 pixels  $\times$  25 pixels), which were denoted by GC, DI, RT, MB, and TR, respectively. Table 3



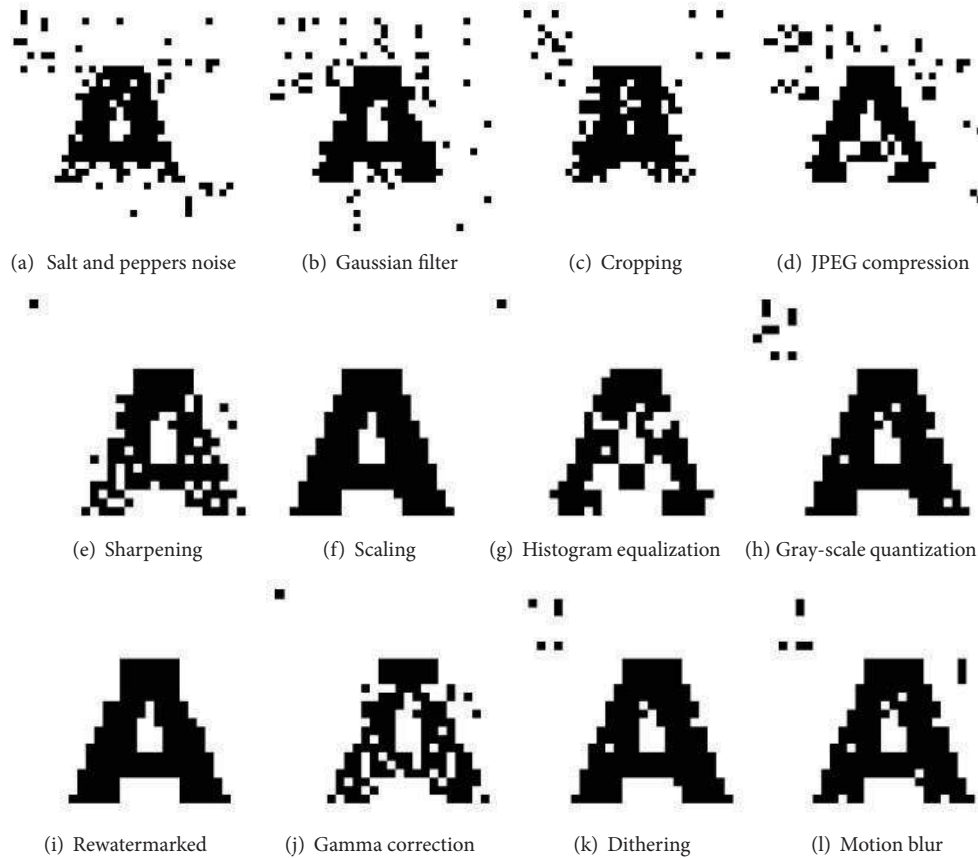


FIGURE 7: Extracted watermarks under different attacks.

depicts the normalized correlation values under these attacks, where one can observe that MSF-MOACO algorithm is more robust against these attacks compared to other algorithms. Figure 6 shows an example of watermarked images under these attacks, while Figure 7 depicts their corresponding extracted watermarks.

## 6. Conclusion

In this paper, we present a new watermarking algorithm based on lifting wavelet transform (LWT) and singular value decomposition (SVD) using multiple scaling factors (MSF) optimized by multiobjective ant colony optimization (MOACO). The MSFs are used instead of single scaling factor (SSF) to achieve a highest possible robustness without losing watermark transparency. However, determining the optimal set of multiple scaling factors is a prohibitively complex problem; to solve this problem a multiobjective ant colony optimization is used. Experimental results demonstrate that the proposed watermarking algorithm (MSF-MOACO) outperforms other watermarking algorithms in terms of imperceptibility and robustness. Moreover, MSF-MOACO algorithm showed better imperceptibility and excellent resiliency against a wide range of watermarking attacks such as additive noise, compression, filtering, and geometrical attacks. Furthermore, the problem of false positive

detection which affects most SVD watermarking algorithms is resolved using countermeasures based on one-way hash function and watermark encryption proposed in [24].

## References

- [1] T. Xianghong, L. Lu, Y. Lianjie, and N. Yamei, "A digital watermarking scheme based on DWT and vector transform," in *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP '04)*, pp. 635–638, October 2004.
- [2] L. Li, H. Xu, C. Chang, and Y. Ma, "A novel image watermarking in redistributed invariant wavelet domain," *Journal of Systems and Software*, vol. 84, no. 6, pp. 923–929, 2011.
- [3] X.-F. Liu and S.-M. Li, "Image digital watermarking algorithm based on Arnold scrambling and wavelet transform," in *Proceedings of the International Conference on Mechanical and Electronics Engineering*, vol. 130–134, pp. 2928–2931, September 2011.
- [4] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-D DFT domain," *IEEE Transactions on Image Processing*, vol. 10, no. 11, pp. 1741–1753, 2001.
- [5] W. Wang, A. Men, and X. Chen, "Robust image watermarking scheme based on phase features in DFT domain and generalized radon transformations," in *Proceedings of the International Workshop on Education Technology and Computer Science*, pp. 736–739, March 2009.

- [6] E. Ganic, S. D. Dexter, and A. M. Eskicioglu, "Embedding multiple watermarks in the DFT domain using low and high frequency bands," in *Security, Steganography, and Watermarking of Multimedia Contents VII*, vol. 5681 of *Proceedings of the SPIE*, pp. 175–184, March 2005, January 2005.
- [7] S. D. Lin, S. Shie, and J. Y. Guo, "Improving the robustness of DCT-based image watermarking against JPEG compression," *Computer Standards and Interfaces*, vol. 32, no. 1-2, pp. 51–58, 2010.
- [8] A. H. Taherinia and M. Jamzad, "A robust spread spectrum watermarking method using two levels DCT," *International Journal of Electronic Security and Digital Forensics*, vol. 2, no. 3, pp. 280–305, 2009.
- [9] J. R. Hernández, M. Amado, and F. Pérez-González, "DCT-domain watermarking techniques for still images: detector performance analysis and a new structure," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 55–68, 2000.
- [10] W. Lu, W. Sun, and H. Lu, "Robust watermarking based on DWT and nonnegative matrix factorization," *Computers and Electrical Engineering*, vol. 35, no. 1, pp. 183–188, 2009.
- [11] W. Lu and H. Lu, "Robust watermarking based on subsampling and nonnegative matrix factorization," *Informatica*, vol. 19, no. 4, pp. 555–566, 2008.
- [12] M. Ouhsain and A. B. Hamza, "Image watermarking scheme using nonnegative matrix factorization and wavelet transform," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2123–2129, 2009.
- [13] A. A. Mohammad, A. Alhaj, and S. Shaltaf, "An improved SVD-based watermarking scheme for protecting rightful ownership," *Signal Processing*, vol. 88, no. 9, pp. 2158–2180, 2008.
- [14] K. Loukhaoukha and J.-Y. Chouinard, "Hybrid watermarking algorithm based on SVD and lifting wavelet transform for ownership verification," in *Proceedings of the 11th Canadian Workshop on Information Theory (CWIT '09)*, pp. 177–182, May 2009.
- [15] R. Liu and T. Tan, "An SVD-based watermarking scheme for protecting rightful ownership," *IEEE Transactions on Multimedia*, vol. 4, no. 1, pp. 121–128, 2002.
- [16] R. Rykaczewski, "Comments on 'An SVD-based watermarking scheme for protecting rightful ownership,'" *IEEE Transactions on Multimedia*, vol. 9, no. 2, pp. 421–423, 2007.
- [17] X. Zhang and K. Li, "Comments on 'an SVD-based watermarking scheme for protecting rightful ownership,'" *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 593–594, 2005.
- [18] L. Xiao, Z. Wei, and J. Ye, "Comments on 'Robust embedding of visual watermarks using discrete wavelet transform and singular value decomposition' and theoretical analysis," *Journal of Electronic Imaging*, vol. 17, no. 4, Article ID 040501, 2008.
- [19] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.
- [20] M. Dorigo, *Optimization, learning and natural algorithms [Ph.D. thesis]*, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milan, Italy, 1992.
- [21] K. Socha and M. Dorigo, "Ant colony optimization for continuous domains," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1155–1173, 2008.
- [22] F. A. C. Viana, G. I. Kotinda, D. A. Rade, and V. Steffen Jr., "Tuning dynamic vibration absorbers by using ant colony optimization," *Computers and Structures*, vol. 86, no. 13-14, pp. 1539–1549, 2008.
- [23] S. Pourtakdoust and H. Nobahari, "An extension of ant colony system to continuous optimization problems," in *Proceedings of the International Workshop of Ant Colony Optimization and Swarm Intelligence*, pp. 294–301, September 2004.
- [24] K. Loukhaoukha and J.-Y. Chouinard, "Security of ownership watermarking of digital images based on singular value decomposition," *Journal of Electronic Imaging*, vol. 19, no. 1, Article ID 013007, 2010.
- [25] Y. Pai and S. Ruan, "A high quality robust digital watermarking by smart distribution technique and effective embedded scheme," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E90, no. 3, pp. 597–605, 2007.
- [26] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, 2004.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

