

# Coevolutionary genetic watermarking for owner identification

Yueh-Hong Chen · Hsiang-Cheh Huang

Received: 1 December 2013 / Accepted: 13 May 2014 / Published online: 27 May 2014  
© Springer-Verlag London 2014

**Abstract** In this paper, we propose a coevolutionary genetic watermarking scheme based on wavelet packet transform. Wavelet packet transform can be viewed as a generalization of the discrete wavelet transform, and a best wavelet basis in the sense of some cost metric can be found within a large library of permissible bases. Coevolutionary genetic algorithm is employed to select an appropriate basis from permissible bases of wavelet packet transform, and it determines the subbands for watermark embedding. Experimental results demonstrate that proposed method can increase the capability to resist specific image processing methods while keeping quality of the watermarked image acceptable. Moreover, the architecture of cooperative coevolutionary genetic algorithm is particularly suitable in distributed computing environment. This characteristic would make genetic watermarking schemes more applicable in real-world applications.

**Keywords** Image watermarking · Cooperative coevolutionary genetic algorithm · Owner identification · Data hiding

## 1 Introduction

Nowadays, people can easily retrieve multimedia contents from the Internet via personal computers, notebooks, and smart mobile phones. With the digital nature of unlimited copying and ease of alteration, the copyright of multimedia contents should be properly protected. Against piracy behavior, digital watermarking has been regarded as an effective solution [3, 13, 16].

Image watermarking is to embed secret information (i.e., watermarks) into images for certain purposes such as owner identification or copy control. Spatial domain watermarking algorithms are famous for their ease of implementation; however, the limited capacity for ownership protection is the major drawback with this type of schemes. Correspondingly, transform domain (or frequency domain) watermarking algorithms provide more flexibilities for algorithm design. Researchers can choose the low-, mid-, or high-frequency coefficients for watermark embedding. For natural images, low-frequency coefficients reside a large amount of energy. Hence, for embedding watermark bits into low-frequency coefficients, large amount of alteration may be expected, and watermarked image quality may get degraded drastically. Conversely, for embedding watermark bits into high-frequency coefficients, it would lead to small alteration of image; however, external signal processing (or *attack*), such as low-pass filtering, may cause the disappearance of embedded watermark. Conceptually, because watermark embedding into low- or high-frequency coefficients have their drawbacks, embedding into mid-frequency coefficients seems a proper solution [10].

By following the discussions above, an excellent image watermarking scheme would be hoped to have following properties:

---

Y.-H. Chen  
Department of Computer Science and Information Engineering,  
Far East University, Tainan, Taiwan, ROC  
e-mail: yuehhong@gmail.com

H.-C. Huang (✉)  
Department of Electrical Engineering, National University of  
Kaohsiung, Kaohsiung, Taiwan, ROC  
e-mail: huang.hc@gmail.com

1. *High fidelity*: The watermarked image quality should be perceptually similar to the original image.
2. *Good robustness*: The watermark can be successfully detected even though some attacks are applied to the watermarked image.
3. *Large data capacity*: The larger the data capacity is, the more secret information can be embedded into original images.

These properties are usually used as performance measures of watermarking schemes. As we mentioned above, since these properties may have conflict with each other, a watermarking scheme is difficult to meet the three properties above at the same time. First of all, increasing fidelity of watermarked images, e.g., embedding into low-frequency coefficients, would lower the strength of watermarks, and thus, it would decrease the robustness of watermarks. Secondly, embedding larger amount of information leads to the more alteration of image, and hence, it would reduce the fidelity of watermarked images. Finally, increased capacity may bring the flexibility for watermark generation, for instance, the inclusion of error control coding for the better protection of watermark, and thus, the robustness may get enhanced. Therefore, typical watermarking schemes embed watermarks in accordance with some heuristic rules so that all three properties mentioned above are acceptable.

A number of methods have been proposed to insert robust and invisible watermarks [2, 9]. Among all methods, the schemes based on wavelet packet transform (WPT) have attracted much attention. WPT can be viewed as a generalization of the discrete wavelet transform (DWT). In the usual dyadic wavelet decomposition, only the low-pass-filtered subband is recursively decomposed, and thus, it can be represented by a logarithmic tree structure. A wavelet packet decomposition allows the decomposition to be represented by any pruned subtree of the full tree topology. Therefore, this representation of the decomposition topology is isomorphic to all permissible subband topologies. The leaf nodes of each pruned subtree represent one permissible orthonormal basis [12]. Thus, a best wavelet basis in the sense of some cost metric can be found within a large library of permissible bases.

Due to the conflict of above three performance measures, embedding watermarks into images can be referred to as an optimization problem. If these properties are evaluated and combined into a weighted sum form appropriately, genetic algorithm (GA) could be used to solving this problem [9, 11, 15]. Different from typical watermarking schemes, conventional genetic watermarking finds out the optimal coefficients in frequency domain in the sense of a certain evaluation function to embed watermarks. Under the constraint of keeping other properties

acceptable, GA can be used to optimize the robustness of watermarks or the fidelity of watermarked images [8, 16]. However, if two or more types of parameters must be determined by GA, it is difficult to appropriately encode these parameters into chromosomes in GA.

In [14], authors indicated that the watermarking scheme employing the zerotree of WPT provided the best performance in term of PSNR compared to schemes employing DWT or discrete cosine transform (DCT). However, the robustness against various types of attack of the scheme employing WPT got slightly decreased. Therefore, we propose a coevolutionary genetic watermarking scheme based on wavelet packet transform in this paper. In the proposed method, coevolutionary genetic algorithm is employed to select an appropriate basis from permissible bases of wavelet packet transform and to determine the subbands for watermark embedding. Experimental results shows that comparing to watermarking methods based on DWT or GA, the proposed method can increase the capability to resist image processing operations.

The rest of the paper is organized as follows. Section 2 introduces the concepts of the coevolutionary genetic algorithm and then describes the proposed coevolutionary genetic watermarking scheme. Section 3 presents the experimental results of the proposed method. Finally, Sect. 4 summarizes the proposed method and draws a brief concluding remarks and future works.

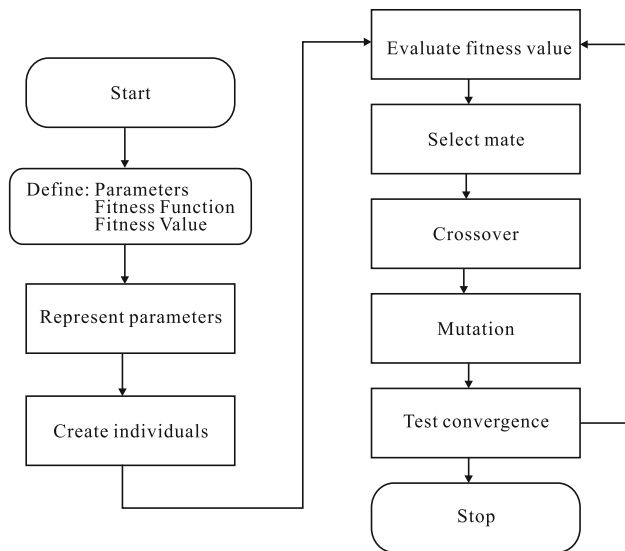
## 2 Coevolutionary genetic algorithm in wavelet packet domain

Assuming that a watermark consists of 0's and 1's, all bits of the watermark are embedded into an image with the same manner, separately. To embed the watermark, the cooperative coevolutionary genetic algorithm (CCGA) is first employed to select an appropriate basis of WPT, and subbands are used to embed watermark bits. Then, a number of coefficients are randomly chosen and modified. The random seed, the WPT decomposition tree, and subbands used in the watermark embedding process are preserved as secret key.

### 2.1 An introduction to CCGA

Genetic algorithm (GA), based on the concept of natural genetics, is a directed random search technique. The exceptional contribution of this method was developed by Holland [7] over the course of 1960s and 1970s, and finally popularized by Goldberg [6].

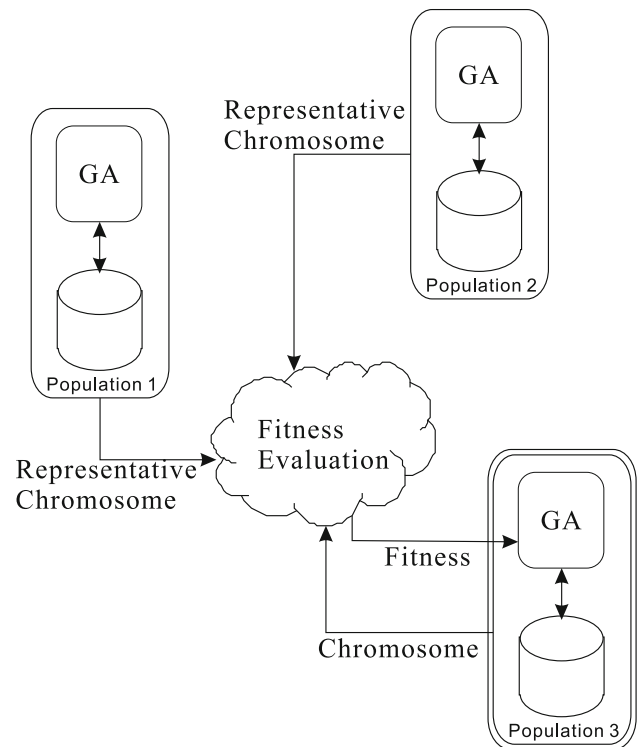
As depicted in Fig. 1, a typical binary GA begins by defining the optimization parameters, the fitness function,



**Fig. 1** The flowchart of a typical binary genetic algorithm

and the fitness value, and it ends by testing for convergence. The optimization parameters are represented by an encoded binary string, called a chromosome. The elements in the binary strings, or the genes, are adjusted by GA to minimize or maximize the fitness value. According to the applications for optimization, designers need to carefully define the necessary elements to be optimized. The fitness function, which is composed of multiple variables to be optimized, is used to evaluate the fitness value of chromosomes. In every iteration of GA, a pre-determined number of chromosomes will be evaluated to obtain fitness values. Then, GA is stopped if the terminating criteria is reached; otherwise, the natural selection, crossover, and mutation operations are applied on the chromosomes in a reasonable way to generate the next generation of chromosomes. A description of GA in more detail can be found in [5, 11].

Comparing to GA, all parameters of the fitness function are not encoded and represented by a single chromosome in CCGA [18]. Instead, the parameters are separated into several subsets. Each subset of the parameters is encoded with one kind of chromosomes, which compose an individual population. Consequently, when fitness value of a chromosome is evaluated, chromosomes in other populations are necessary so that all parameters of the fitness function can be provided. These chromosomes chosen from other populations for fitness value evaluation are referred to as *representative chromosomes*, which can be chosen randomly or according to the fitness value. The flowchart of CCGA with three populations is depicted in Fig. 2. At the beginning of CCGA, chromosomes in these three populations are randomly initialized. Chromosomes are then evaluated in accordance with fitness function and representative chromosomes. Then, mate selection,



**Fig. 2** The flowchart of coevolutionary genetic algorithm, in which chromosomes of population 3 are evaluated

crossover, and mutation operations are applied separately to chromosomes of these populations to generate next generations. In other words, GA is applied on these three populations, respectively.

### 2.2 Selection of the best basis with CCGA

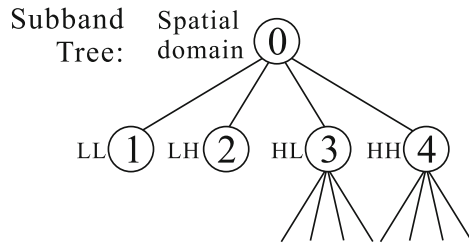
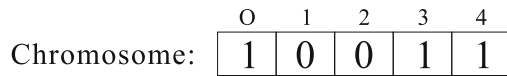
Chromosomes of the genetic algorithm used to selecting the best basis consist of a binary string. The length of the binary string is equal to  $\frac{4^D - 1}{4 - 1}$ , where  $D$  is decomposition level of WPT. A bit equal to ‘1’ in the string indicates the corresponding subband should be decomposed further, and the decomposition will stop if the bit value equals to ‘0’. If the allele at index  $i$  equals to ‘1’, the indices of the resulting four subbands can be derived by:

$$i_m = 4 \times i + m, \quad m \in \{1, 2, 3, 4\}. \tag{1}$$

An example of the chromosome and its corresponding WPT decomposition tree is illustrated in Fig. 3.

### 2.3 Selection of subbands with CCGA

The chromosome of GA used for subband selection is also composed of a binary string. The length of the chromosome equals to the chromosome used in previous subsection. The chromosome is encoded in the following manner:



**Fig. 3** An example of a chromosome and its corresponding subband tree

1. If the allele at index  $i$  equals to '1' and its corresponding subband is not decomposed further, the subband is selected for watermark embedding.
2. If the  $i$ th allele equals to '1' and subbands decomposed from  $i$ th subband are leaf nodes in the WPT decomposition tree, the decomposed LL subband is selected for watermark embedding.
3. Otherwise, the corresponding subband are not selected for watermark embedding.

Figure 4 depicts an example of the chromosome.

#### 2.4 Fitness function in CCGA watermarking

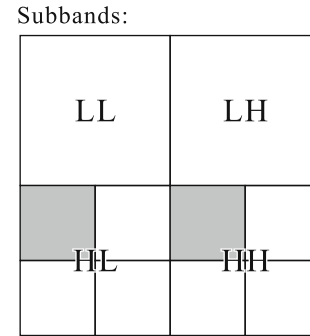
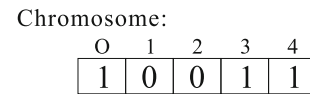
The fitness function employed in the proposed scheme is as follows:

$$\text{fitness} = P_{\text{MSE}} \times \prod_{i=1}^K P_i, \tag{2}$$

where  $P_i$  is the percentage of the watermark bits that still survive after a certain attack method was applied, and  $K$  is the number of attack methods adopted for robustness evaluation. Perceptual quality of the watermarked images was measured with mean squared error ratio (MSE), referred to as  $P_{\text{MSE}}$ . All these terms are combined in multiplication to obtain a fitness value.

#### 2.5 Binary watermarking in wavelet packet transform

The method to embed one binary value into an image was mainly based on [2]. When one bit of the watermark is embedded, a number of coefficients, which is pre-specified by the user, are chosen randomly. These coefficients are then modified such that the first one, in the order of being chosen, is the largest if an '1' is embedded. If a '0' is embedded, the coefficients should be modified such that the



**Fig. 4** An example of a chromosome and corresponding subbands it selected. Subbands filled with gray are selected to embed a watermark

first one is the smallest. Suppose  $c_i, i = 1, \dots, n$  are the chosen coefficients, and  $n$  is the number of coefficients. The modified coefficients will satisfy Eq. (3),

$$\begin{cases} c'_1 \geq \max(c'_2, c'_3, \dots, c'_n) + \delta, & \text{if } W = 1; \\ c'_1 < \min(c'_2, c'_3, \dots, c'_n) - \delta, & \text{if } W = 0. \end{cases} \tag{3}$$

Here,  $c'_i, i = 1, \dots, n$  are the modified coefficients,  $W$  is one bit of the watermark, and  $\delta, \delta \geq 0$ , is the strength parameter specifying the difference between the first coefficient and the largest (smallest) one among remaining coefficients. To maximize the PSNR value, the algorithm described in Sect. 2.6 is used to find the optimal values of the coefficients to be modified.

The extracting algorithm proposed is to compare the first coefficient to the largest and smallest ones among remaining coefficients. If the value of the first coefficient is closer to the largest one among remaining coefficients, an '1' will be extracted; otherwise, a '0' will be extracted. The extracting process is described in Eq. (4):

$$W' = \begin{cases} 1, & \text{if } c''_1 \geq \frac{c''_{\max} + c''_{\min}}{2}; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Note that

$$\begin{aligned} c''_{\max} &= \max(c''_2, c''_3, \dots, c''_n); \\ c''_{\min} &= \min(c''_2, c''_3, \dots, c''_n). \end{aligned} \tag{5}$$

Here,  $c''_i, i = 1, \dots, n$  are the coefficients obtained from an image, and  $W'$  is the extracted bit. Then, according to a threshold, we can compare the extracted binary string with the watermark to determine whether the watermark exists or not.

### 2.6 Extreme value-based watermarking (EVBW)

The main idea of EVBW is to modify more than one coefficient at the same time. To embed a ‘1’, if the first coefficient  $c_1$  is increased to  $x + \delta$ , all coefficients larger than  $x$  should be decreased to  $x$  to fit the rule shown in Eq. (3). Therefore, it is possible to find the optimal value of  $x$  such that the watermarked image have the best quality according to an appropriate quality metric.

If the mean square error (MSE) of the modified coefficients is minimized, the peak signal-to-noise ratio (PSNR) value is maximized simultaneously. Suppose a bit of ‘1’ is to be embedded into  $n$  coefficients. If  $c_1$  is increased to  $x + \delta$  and all coefficients larger than  $x$  are decreased to  $x$ , the square error (SE) value can be calculated as Eq. (6):

$$SE(x) = ((x + \delta) - c_1)^2 + \sum_{c_i > x} (c_i - x)^2. \tag{6}$$

Then, the minimum of  $SE(x)$  can be obtained by finding out the value of  $x$  where the first derivative of  $SE(x)$  is equal to 0. The first derivative of  $SE(x)$  is shown in Eq. (7), and the optimal value of  $x$  is shown in Eq. (8).

$$\frac{d}{dx} SE(x) = 2 \times (x + \delta - c_1) + 2 \times \sum_{c_i > x} (x - c_i). \tag{7}$$

$$x = \frac{\left( \sum_{c_i > x} c_i \right) + c_1 - \delta}{k + 1}, \quad i = 1, \dots, n. \tag{8}$$

Here,  $k$  is the number of coefficients larger than  $c_1$ . In Eq. (8), it is assumed that only  $k$  largest coefficient and  $c_1$  be modified. Therefore, the value of  $x$  should be larger than the  $(k + 1)$ -th largest coefficient but smaller than  $k$ th largest coefficient. The algorithm to find the optimal value  $x$  is stated as follows:

```

Obtain  $d_1, d_2, \dots, d_n$  by sorting  $c_1, c_2, \dots, c_n$ 
such that  $d_1 \geq d_2 \geq \dots \geq d_n$ 
Suppose  $c_1$  is the  $(k + 1)$ -th largest value
If  $(k + 1) = 1$ 
     $x_{opt} = d_2$ , Stop
End If
For  $i = 1$  to  $k$ 
     $x = \frac{(\sum_{j=1}^i d_j) + d_{k+1} - \delta}{i+1}$ 
    If  $d_{i+1} < x \leq d_i$ 
         $x_{opt} = x$ , Stop
    End If
End For
 $x_{opt} = c_1$ , Stop
    
```

After the algorithm finishes, the optimal value of  $x$  can be found.  $c_1$  can then be modified to  $x + \delta$ , and all coefficients larger than  $x$  be modified to  $x$  to embed a bit of ‘1’. A similar algorithm to find the optimal value to embed a ‘0’ is stated as follows:

```

Obtain  $d_1, d_2, \dots, d_n$  by sorting  $c_1, c_2, \dots, c_n$ 
such that  $d_1 \leq d_2 \leq \dots \leq d_n$ 
Suppose  $c_1$  is the  $(k + 1)$ -th smallest value
If  $(k + 1) = 1$ 
     $x_{opt} = d_2$ , Stop
For  $i = 1$  to  $k$ 
     $x = \frac{(\sum_{j=1}^i d_j) + d_{k+1} + \delta}{i+1}$ 
    If  $d_{i+1} > x \geq d_i$ 
         $x_{opt} = x$ , Stop
    End If
End For
 $x_{opt} = c_1$ , Stop
    
```

Finally,  $c_1$  is decreased to  $x - \delta$ , and all coefficients smaller than  $x$  be increased to  $x$  to embed a bit of ‘0’.

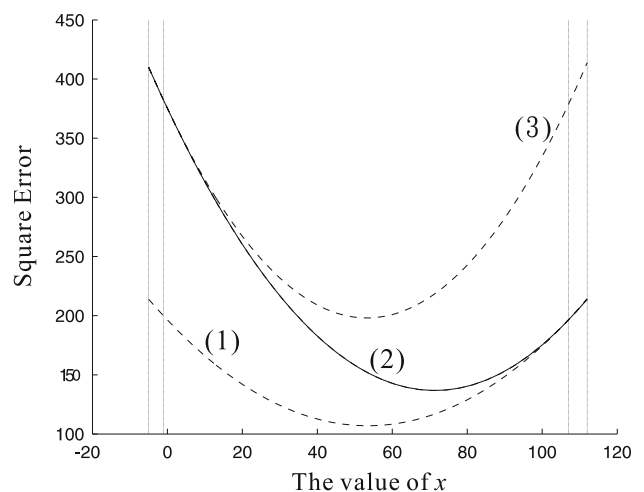
Continuing the example in previous subsection, if  $\delta = 0$ , the  $SE(x)$  value is:

$$SE(x) = \begin{cases} (x - 112)^2 + (x + 5)^2 & \text{if } 107 \leq x < 112 \\ (x - 112)^2 + (x - 107)^2 + (x + 5)^2 & \text{if } -1 \leq x < 107 \\ (x - 112)^2 + (x - 107)^2 + (x + 1)^2 + (x + 5)^2 & \text{if } -5 \leq x < -1 \end{cases}$$

By applying the proposed algorithm, the optimal value of  $x$ , about 71.3, can be obtained. The curve of  $SE(x)$  is shown in Fig. 5. It is clear that  $SE(x)$  is the minimum when  $x = 71.3$ .

### 3 Experimental results

In this section, we present some experimental results to demonstrate the performance of the proposed method. An 1,000-bit watermark was generated randomly and used throughout the experiments. The numbers of chromosomes of two populations were 200 and 200, maximal

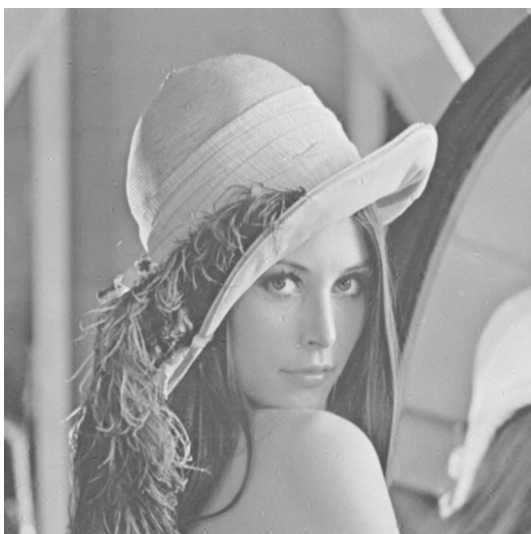


**Fig. 5** Curves of  $(x - 112)^2 + (x + 5)^2$  (curve 1),  $(x - 112)^2 + (x - 107)^2 + (x + 5)^2$  (curve 2) and  $(x - 112)^2 + (x - 107)^2 + (x + 1)^2 + (x + 5)^2$  (curve 3)

decomposition level was 7, the selection and mutation rates were set to 0.5 and 0.04, respectively, and the number of training iterations was 400.

In our experiments, performances are assessed by the output image quality, represented by PSNR value between original and watermarked images. Six  $512 \times 512$ , 8 bits/pixel gray scale test images, including Lena, Baboon, F16, Fishing Boat, Pentagon, and Peppers are employed for performance comparisons. Three image processing operations: JPEG compression with quality factor = 30, Gaussian filtering, and sharpening were applied on the watermarked images to evaluate fitness values of the chromosomes. To compare with [1] and [2], the watermark was only embedded into subbands decomposed from *LH2*, *HL2* and *HH2*. The parameters used in the watermarking scheme were the same with that in [2]. To embed one bit of the watermark, twelve coefficients were chosen from *LH2*, *HL2* or *HH2* subband. In other words,  $n = 12$  in our experiments. The strength parameter  $\delta$  was assigned to 0.

A watermarked image Lena is displayed in Fig. 6. The results of applying three image processing operations on six test images that are watermarked with [1, 2] and the proposed scheme are summarized in Table 1 for the purpose of comparison. Similar experimental results obtained by applying [1, 2] and the proposed scheme on 100 test images randomly selected from Corel Gallery are also depicted in Table 2. Figure 6, and Tables 1 and 2 reveal that the proposed watermarking scheme can generate watermarked image with higher robustness and similar perceptual quality. It is because a proper basis and subbands are selected by CCGA aiming to the three image processing methods. Moreover, due to the adaptivity of



**Fig. 6** The watermarked image with PSNR = 44.6, obtained with the proposed CCGA watermarking scheme

**Table 1** Performance of the proposed method comparing to [1] and [2]

Test image	Watermarking scheme	JPEG ( $Q = 30$ ) (%)	Gaussian filtering (%)	Sharpening (%)
Lena	The proposed scheme	85.7	90.5	99.8
	[1]	83.4	88.0	99.8
	[2]	76.6	83.1	99.7
Baboon	The proposed scheme	95.3	93.2	99.0
	[1]	95.0	92.4	98.6
	[2]	94.8	91.8	98.2
F16	The proposed scheme	83.3	88.3	99.1
	[1]	82.4	87.5	99.1
	[2]	82.1	87.2	99.0
Fishing Boat	The proposed scheme	87.6	87.2	98.6
	[1]	86.7	86.1	98.4
	[2]	86.3	85.4	98.3
Pentagon	The proposed scheme	89.6	87.8	97.6
	[1]	88.9	87.2	96.4
	[2]	88.7	86.8	95.9
Peppers	The proposed scheme	76.0	81.6	97.5
	[1]	75.4	80.7	96.9
	[2]	75.2	80.3	96.6

The results shown in the table are percentage of survival watermark bits after three image processing operations were applied on watermarked test images

**Table 2** Performance of the proposed method comparing to [1] and [2]

Watermarking scheme	Statistical indicator	JPEG ( $Q = 30$ )	Gaussian filtering	Sharpening
The proposed scheme	Mean	87.8 %	86.5 %	98.6 %
	Variance	11.6	13.3	0.71
[1]	Mean	85.1 %	83.7 %	96.9 %
	Variance	11.4	12.5	0.92
[2]	Mean	86.2 %	85.8 %	97.4 %
	Variance	12.0	12.8	0.78

The results shown in the table are mean and variance of percentage of survival watermark bits after three image processing operations were applied on 100 watermarked test images

CCGA, the robustness of the proposed scheme is better than [1] and [2] in average.

In addition to [1, 2], the performance comparison among the proposed method and other genetic watermarking

**Table 3** Performance of the proposed method comparing to [17] and [4]

Test image	Watermarking method	PSNR PSNR	JPEG compression	Low-pass filtering
Lena	Our scheme ( $n = 12, \delta = 0$ )	44.20	Survival ( $Q = 15$ )	Survival
	Our scheme ( $n = 2, \delta = 1$ )	48.12	Survival ( $Q = 40$ )	Survival
	[17]	42.25	Survival ( $Q = 75$ )	Survival
	[4]	42.88	Survival ( $Q = 20$ )	Survival
Baboon	Our scheme ( $n = 12, \delta = 0$ )	38.31	Survival ( $Q = 15$ )	Survival
	Our scheme ( $n = 2, \delta = 1$ )	41.52	Survival ( $Q = 40$ )	Survival
	[17]	36.37	Survival ( $Q = 75$ )	Survival
	[4]	41.04	Survival ( $Q = 20$ )	Survival
Fishing Boat	Our scheme ( $n = 12, \delta = 0$ )	40.45	Survival ( $Q = 15$ )	Survival
	Our scheme ( $n = 2, \delta = 1$ )	43.71	Survival ( $Q = 40$ )	Survival
	[17]	37.19	Survival ( $Q = 75$ )	Survival
	[4]	42.08	Survival ( $Q = 20$ )	Survival
Peppers	Our scheme ( $n = 12, \delta = 0$ )	42.48	Survival ( $Q = 15$ )	Survival
	Our scheme ( $n = 2, \delta = 1$ )	45.91	Survival ( $Q = 40$ )	Survival
	[17]	40.61	Survival ( $Q = 75$ )	Survival
	[4]	40.39	Survival ( $Q = 20$ )	Survival

approaches are also provided. Because the experimental results may be different while genetic algorithm are used for watermark embedding, the results used in comparison are all from the original articles. In order to demonstrate the best PSNR that can be achieved by the proposed method, the PSNR values of the watermarked images while  $n$  is assigned to 2 and  $\delta = 1$  is assigned to 1 are also provided. As shown in Table 3, watermarks embedded by the proposed scheme would survive even the JPEG compression quality factor ( $Q$ ) is assigned to 15. Thus, it can be claimed that for specific image processing methods, the proposed method can select a best WPT basis and subbands to increase watermark robustness. Furthermore, image fidelity and watermark robustness could be adjusted with parameter  $n$  and  $\delta$ , so the proposed scheme are also flexible.

## 4 Conclusion

Embedding watermarks into images can be referred to as an optimization problem. Therefore, cooperative coevolutionary genetic algorithm is used to solving this problem in this paper. The cooperative coevolutionary genetic algorithm would select an appropriate basis from permissible bases of wavelet packet transform and select proper subbands for watermark embedding. Experimental results demonstrate that proposed method increase the capability to resist specific image processing methods while keeping quality of the watermarked image acceptable. Moreover, the architecture of cooperative coevolutionary genetic algorithm is particularly suitable in distributed computing environment. This characteristic would make genetic watermarking schemes more applicable in real-world applications.

## References

- Chen YH, Huang HC (2009) Genetic watermarking based on wavelet packet transform. In: Ninth international conference on hybrid intelligent systems (HIS2009), vol 1, pp 262–265
- Chen YH, Su JM, Huang HC, Pao HT, Fu HC (2005) Adaptive watermarking using relationships between wavelet coefficients. In: 2005 IEEE international symposium on circuits and systems (ISCAS 2005), vol 5, pp 4979–4982
- Cox JJ, Miller ML, Bloom JA, Fridrich J, Kalker T (2007) Digital watermarking and steganography. Elsevier, London
- Dey N, Samanta S, Yang XS, Das A, Chaudhuri SS (2013) Optimisation of scaling factors in electrocardiogram signal watermarking using cuckoo search. *Int J Bio-Inspired Comput* 5(5):315–326
- Gen M, Cheng R (1997) Genetic algorithms and engineering design. Wiley series in engineering design and automation. Wiley, New York
- Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading
- Holland JH (1992) Adaptation in natural and artificial systems. MIT Press, Cambridge
- Huang CH, Wu JL (2000) A watermark optimization technique based on genetic algorithms. In: SPIE electronic imaging 2000. San Jose
- Huang HC, Chen YH (2009) Genetic fingerprinting for copyright protection of multicast media. *Soft Comput* 13(4):383–391
- Huang HC, Fang WC (2010) Metadata-based image watermarking for copyright protection. *Simul Model Pract Theory* 18(4):436–445
- Huang HC, Pan JS, Huang YH, Wang FH, Huang KC (2007) Progressive watermarking techniques using genetic algorithms. *Circuits Syst Signal Process* 26(5):671–687
- Li ZN, Drew MS (eds) (2003) Fundamentals of multimedia. Prentice-Hall, Englewood Cliffs
- Mathon B, Cayre F, Bas P, Macq B (2014) Optimal transport for secure spread-spectrum watermarking of still images. *IEEE Trans Image Process* 23(4):1694–1705
- Schwindt S, Amornraksa T (2002) Performance comparison of zerotrees based digital watermarking. In: IEEE international

- conference on industrial technology (IEEE ICIT 02), vol 1, pp 78–81
15. Shieh CS, Huang HC, Wang FH, Pan JS (2003) An embedding algorithm for multiple watermarks. *J Inf Sci Eng* 19(2):381–395
  16. Wang S, Zheng D, Zhao J, Tam WJ, Speranza F (2014) Adaptive watermarking and tree structure based image quality estimation. *IEEE Trans Multimed* 16(2):311–325
  17. Wei Z, Dai J, Li J (2006) Genetic watermarking based on dct domain techniques. In: Canadian conference on electrical and computer engineering, 2006. CCECE '06, pp 2365–2368
  18. Yang Y, Yu X (2007) Cooperative coevolutionary genetic algorithm for digital IIR filter design. *IEEE Trans Indust Electron* 54(3):1311–1318