An Intelligent Optimization Approach for Digital Image Watermarking in the Multi-Wavelet Domain

Prayoth Kumsawat
School of Telecommunications Engineering
Institute of Engineering
Suranaree University of Technology
Nakhon Ratchasima, Thailand

Kitti Attakitmongcol
&
Arthit Srikaew
School of Electrical Engineering
Institute of Engineering
Suranaree University of Technology
Nakhon Ratchasima, Thailand

ABSTRACT
Digital watermarking is an effective method for copyright protection and for the data security of digital contents. In this paper, the authors propose a digital image watermarking algorithm in the multi-wavelet transform domain. The watermark is embedded in the selected coefficients using the quantization index modulation technique; therefore, the original image is not needed in the watermark extraction process. The authors develop an optimization technique using genetic algorithms to search for optimal quantization steps to improve the quality of the watermarked image and the robustness of the watermark. In addition, the authors construct a prediction model based on image moments and a back-propagation neural network to correct an attacked image geometrically before the watermark extraction process begins. Experimental results demonstrate that the proposed algorithm can achieve good perceptual invisibility and security. It is also robust against various common image processing and geometrical attacks.

Keywords: Watermarking, intellectual property rights, multi-wavelet, quantization index modulation, genetic algorithms, neural networks
1. INTRODUCTION

With the rapid growth of communication networks and the rapid development of digital multimedia technologies, digital contents can be readily shared via Internet and can be easily used, processed, and transmitted. The downside is that digital contents can be copied without any loss and can be distributed without any approval from the authors. Protecting the intellectual property rights of digital multimedia contents has therefore become an important issue.

Digital watermarking is a technology that can serve this purpose. It is a technique that embeds hidden information, called a watermark, irremovably and imperceptibly into digital media, including images, audio, or video contents, called the host, by subtly modifying their perceptual data. There are three main requirements for a watermarking technique:

1. **Perceptual Transparency**: In most applications, embedding a watermark should not affect the quality of the original media.
2. **Robustness**: This refers to the ability of the watermarking system to resist the attacks.
3. **Capacity**: This refers to the amount of watermark bits that can be stored in the host media.

Many digital watermarking techniques are reported in the literature. For digital images, the embedding process can be accomplished in either the spatial domain or the transform domain. In spatial domain techniques, the values of the image pixels are directly modified based on the watermark that is to be embedded. In transform domain techniques, some invertible mathematical transform (DCT, DFT, or DWT) is first applied to the image before embedding the watermark. Previous works have shown that the transform domain scheme is typically more robust to noise, common image processing, and compression than the spatial transform scheme [Hartung and Kutter, 1999]. Since the current image compression standard JPEG 2000 is based on DWT, much attention has been focused on the wavelet transform-based watermarking algorithms.

According to the need regarding original data during the watermark detection process, watermarking algorithms are classified into two types: private algorithms, and public or blind algorithms. The private method needs the original signal during detection. In some cases, when the original data is not easy to obtain, or when it is not known which copy is the original one, it is necessary to use blind watermarking to resolve rightful ownership.
In a desired image watermarking system, the watermark should be robust to content-preserving attacks, including common image processing operations, and geometrical attacks. Common image processing not only modifies the image, but may also modify the watermark as well. Thus, the watermark may become undetectable after intentional or unintentional image processing attacks. Many methods are proposed to resist this kind of attack [Chen and Wornell, 2001; Yuan Li and Zhong, 2007].

Geometrical attack is one of the most difficult problems in watermark detection. It mostly causes a watermarking detector to fail to detect the existence of a watermark in the watermarked image. Similarly, there have also been many kinds of methods for resisting geometrical distortions, such as the transform-based scheme [Dong, Brankov, Galatsanos, Yang, and Davoine, 2005] and the feature-based scheme [Gao and Jiang, 2010; Wang, Yang, and Cui, 2008; Singh and Ranade, 2014].

In previous work, Chen and Wornell [2001] proposed a class of embedding methods called quantization index modulation (QIM) that achieves probably good rate-distortion-robustness performance. In general, the watermark must be embedded in an invisible way to avoid degrading the perceptual quality of the host image. To overcome this difficulty, many researchers exploited the characteristics of the human visual system to achieve a compromise between the invisibility and robustness of the embedding algorithm.

Yuan Li and Zhong [2007] proposed watermarking techniques for digital images. The watermark is embedded into the wavelet coefficients after the singular value decomposition of the image is computed. To satisfy the requirements on the invisibility and robustness of the embedded watermark, the human visual system is used to develop a perceptual mask for optimizing the watermark strength in the embedding process [Zhang, 2009].

Another way to achieve the main requirements of watermarking schemes is to use artificial intelligence techniques. The image watermarking problem can be viewed as an optimization problem. Therefore, it can be solved by an optimization algorithm such as neural network (NN), genetic algorithms (GA), or support vector machine (SVM). In recent years, a number of artificial intelligence watermarking techniques for digital images have been reported. The application of neural networks to digital watermarking mainly includes determining the watermark strength and improving watermark detection.

Gao and Jiang [2010] proposed a novel robust grayscale image watermarking scheme to guard against geometric attacks in the wavelet transform.
domain. The watermark robustness is improved by embedding the perceptually significant part of watermark information into the low-frequency part of the cover image coefficients. The back-propagation neural network is used to construct the forecasting model, which can be used to correct the attacked image geometrically.

Wang, Yang, and Cui [2008] proposed an image watermarking scheme in the spatial domain and applied classification techniques based on SVM to improve the performance of conventional methods.

Zhang [2009] proposed a new blind watermarking scheme based on the discrete wavelet transform and neural networks. Because of the learning and adaptive capabilities of the radial basis function in neural networks, the embedding and extracting strategies can greatly improve the robustness against various attacks.

Huang, Zhang, Feng, and Yang [2008] proposed a novel blind watermarking technique in the wavelet transform domain. The artificial neural network is applied to memorizing the relation between the watermark signal and the corresponding watermarked image. Thus, the watermark can be recovered exactly from the watermarked image without using the original image and the watermark.

Chen, Yao, Chen, and Chen [2008] proposed a new robust public watermarking algorithm. The watermark is embedded repeatedly into different coefficients of the discrete cosine transform domain using a proper embedding threshold below the perceptible threshold. Moreover, the neural network is applied to extract the watermark. Several gray-level watermarking schemes based on multi-wavelet were proposed recently.

Ghouti, Bouridane, Ibrahim, and Boussakta [2006] introduced a robust watermarking algorithm using a balanced multi-wavelet transform. The watermark embedding scheme is based on the principles of spread-spectrum communications to achieve high watermark robustness.

Kumsawat, Attakitmongcol, and Srikaew [2005] proposed the spread spectrum image watermarking algorithm using the discrete multi-wavelet transform (DMT). The GA is applied to search for optimal watermarking parameters to improve the quality of the watermarked image and the robustness of the watermark.

Chen, Zhang, and Peng [2009] proposed a novel optimal color image watermarking in the wavelet transform domain. The watermark is embedded into the host color image by selecting and modifying the wavelet coefficients. The
An Intelligent Optimization Approach for Digital Watermarking in the Multi-Wavelet Domain

A genetic algorithm is twice used to select the fit embedding wavelet coefficient location and watermark intensity to embed the watermark into the host color image. After some kinds of attacks, the extracted watermark can be identified expediently through synergetic neural networks.

Kumsawat, Attakitmongcol, and Srikaew [2007] proposed a new digital image watermarking algorithm in the discrete multi-wavelet transform (DMT) domain. The embedding technique is based on the parent-child structure of the transform coefficients called the triple tree. The watermark is a binary pseudo-random noise sequence, and this algorithm does not require the original image in the watermark extraction.

Jin and Jin [2010] proposed an adaptive digital image watermarking scheme based on fuzzy neural networks for copyright protection. The neural network is used to adaptively decide different watermark embedding positions according to different textural features of each block and luminance. The results show that this technique has no impact on the robustness of the watermark and the perceptual quality of the image.

Xu and Shujuan [2011] proposed an adaptive image watermarking algorithm based on a neural network. The watermark signal is embedded into higher frequency coefficients of DWT joined with DCT. The robustness of the algorithms can be improved through the extracted watermark on a post-processing method that uses the associative memory function of the neural network. The algorithm is shown to be effective through simulation experiments.

Ramanjaneyulu and Rajarajeswari [2012] proposed a robust and oblivious image watermarking scheme based on DWT for copyright protection. The embedding and extraction process are characterized by parameters, and a genetic algorithm is used for parameter optimization. The performance of the proposed method is tested with watermark images of different sizes.

Sridhar and Arun [2013] proposed a watermarking technique based on the wavelet domain and on the sharing of an image with the motivation to maintain the quality of the image. The original image is diagonally shared, one of the shares is horizontally merged, and the watermarking process is employed in a fusion image using a discrete wavelet transform. At the receiver end, the image is again diagonally shared and merged and the watermarked image extracted.

Hamghalam, Mirzakuchaki, and Akhaee [2014] proposed a robust image watermarking method based on geometric modeling. Eight samples of wavelet approximation coefficients on each image block are used to construct two line segments in the two-dimensional space. The authors change the angle formed
between these line segments for data embedding. Geometrical tools are used to solve the tradeoff between the transparency and robustness of the watermark data. Experimental results confirm the validity of the theoretical analyses, and the proposed scheme has high robustness against attacks.

In this paper, we propose an image watermarking method based on the discrete multi-wavelet transform for the application of copyright protection. In our algorithm, the imperceptibility and robustness of an existing image watermarking technique is enhanced through genetic algorithm optimization and neural networks. Finally, we compare our experimental results with the results of previous work.

This paper is organized as follows. In Section 2, we introduce the preliminaries of multi-wavelet transform and multi-wavelet tree. In Section 2, we describe watermarking in the DMT domain with genetic algorithm optimization and neural network training. Section 4 presents our experimental results. In Section 5, we discuss the conclusions of our study.

2. PRELIMINARIES

This section discusses the preliminaries of multi-wavelet transform and the multi-wavelet tree, genetic algorithms, and back-propagation neural networks.

2.1. Multi-Wavelet Transform

In recent years, multi-wavelet transformation has gained a lot of attention in signal- and image-processing applications. It is a relatively new concept in the framework of wavelet transform and has some important differences. In particular, whereas wavelet has one scaling function and wavelet function, multi-wavelet has two or more scaling and wavelet functions.

The main motivation for using multi-wavelet is that it is possible to construct multi-wavelets that simultaneously possess desirable properties such as orthogonality, symmetry, and compact support, with a given approximation order. A scalar wavelet cannot possess all these properties at the same time. Thus, multi-wavelets offer the possibility of superior performance for image-processing applications, compared with scalar wavelets [Attakitmongcol, Hardin and Wilkes, 2001]. Following is a brief overview of multi-wavelet transform.

Let \( \Phi \) denote a compactly supported orthogonal scaling vector 
\( \Phi = (\phi^1, \phi^2, ..., \phi^r)^T \), where \( r \) is the number of scalar scaling functions. Then \( \Phi(t) \) is to satisfy a two-scale dilation equation of the form
\[
\Phi(t) = \sqrt{2} \sum_{n} h(n) \Phi(2t-n)
\]

for some finite sequence \( h \) of \( r \times r \) matrices. Furthermore, the integer shifts of the components of \( \Phi \) form an orthonormal system, that is

\[
\langle \phi^l(-n), \phi^l'(-n') \rangle = \delta_{l,l} \delta_{n,n'}.
\]

Let \( V_0 \) denote the closed span of \( \{ \phi^l(-n) | n \in \mathbb{Z}, l = 1,2,...,r \} \) and define \( V_j = \{ f(\frac{\cdot}{2^j}) | f \in V_0 \} \). Then \( (V_j)_{j \in \mathbb{Z}} \) is a multi-resolution analysis of \( L^2(\mathbb{R}) \). Note that we choose the decreasing convention \( V_{j+1} \subset V_j \).

Let \( W_j \) denote the orthogonal complement of \( V_j \) in \( V_{j-1} \). Then, there exists an orthogonal multi-wavelet \( \Psi = (\psi^1, \psi^2,...,\psi^r)^T \) such that \( \{ \psi^l(-n) | l = 1,2,...,r \text{ and } n \in \mathbb{Z} \} \) form an orthonormal basis of \( W_0 \). Since \( W_0 \subset V_{-1} \), there exists a sequence \( g \) of \( r \times r \) matrices such that

\[
\Psi(t) = \sqrt{2} \sum_{n} g(n) \Phi(2t-n).
\]

Let \( f \in V_0 \), then \( f \) can be written as a linear combination of the basis in \( V_0 \):

\[
f(t) = \sum_{n} c_0(k) \Phi(t-k)
\]

for some sequence \( c_0 \in l_2(\mathbb{Z})^r \). Since \( V_0 = V_1 \oplus W_1 \), \( f \) can also be expressed as

\[
f(t) = \frac{1}{\sqrt{2}} \sum_{k \in \mathbb{Z}} c_1(k) \Phi(\frac{t}{2}-k) + \frac{1}{\sqrt{2}} \sum_{k \in \mathbb{Z}} d_1(k) \psi(\frac{t}{2}-k).
\]

The coefficients \( c_1 \) and \( d_1 \) are related to \( c_0 \) via the following decomposition and reconstruction algorithm:

\[
c_1(k) = \sum_{n} h(n)c_0(2k+n)
\]

\[
d_1(k) = \sum_{n} g(n)c_0(2k+n)
\]

\[
c_0(k) = \sum_{n} h(k-2n)^T c_1(n) + \sum_{n} g(k-2n)^T d_1(n).
\]

Unlike scalar wavelet, even though the multi-wavelet is designed to have approximation order \( p \), the filter bank associated with the multi-wavelet basis does not inherit this property. Thus, in applications, one must associate a given
discrete signal into a sequence of length $-r$ vectors without losing some certain properties of the underlying multi-wavelet. Such a process is referred to as pre-filtering. The block diagram of a multi-wavelet with pre-filter $Q(z)$ and post-filter $P(z)$ is shown in Figure 1. $H(z)$ and $G(z)$ are the $z$ transform of $h(n)$ and $g(n)$, respectively. Figure 2(a) illustrates a single-level multi-wavelet decomposition of the EM1 image using the DGHM multi-wavelet with optimal orthogonal pre-filter [Attakitmongcol, Hardin and Wilkes, 2001], and Figure 2(b) illustrates the sub-band arrangement. The three detail sub-bands are denoted by $LH$ (vertical orientation), $HL$ (horizontal orientation), and $HH$ (diagonal orientation), whereas the approximation sub-band is denoted by $LL$.

![Figure 1. Multi-Wavelet Filter Band](image)

![Figure 2(a). One-Level Multi-Wavelet Decomposition of EM1 Image](image)

![Figure 2(b). The Sub-Band Arrangement](image)
2.2. Multi-Wavelet Tree

Multi-wavelet transform coefficients have the property that the related coefficients in different scales are located at the same orientation and location in the multi-wavelet hierarchical decomposition. With the exception of the highest frequency sub-bands, every coefficient at a given scale can be related to a set of coefficients at the next finer scale of similar orientation. The coefficient at the coarse scale is called the parent, and all coefficients corresponding to the same spatial location at the next finer scale of similar orientation are called children.

For the four-level multi-wavelet hierarchical sub-band decomposition, the parent-child dependencies are shown in Figure 3(a). For a given parent, the set of all coefficients at all finer scales of similar orientation corresponding to the same location are called descendants. Multi-wavelet trees descending from a single coefficient in the sub-band $HH_4$, $HL_4$ and $LH_4$ are shown in Figure 3(b).

![Figure 3(a). Four-Level Multi-Wavelet Decomposition of EM1 Image](image)

![Figure 3(b). Parent-Child Dependencies of Multi-Wavelet Tree](image)

Without significant loss of generality, we shall focus on watermarking still images with 256 gray levels of size $512 \times 512$ pixels. To trade off between the invisibility and robustness of the watermark, the high-energy sub-band ($LL_4$) is not used. Furthermore, the coefficients in high-frequency sub-bands ($LH_1$, $HL_4$ and $HH_1$) are not used since they often contain low-energy coefficients.

In other sub-bands, we group the coefficients corresponding to the same spatial location together. Figure 4(a) shows an example of a group with one coefficient from $HL_4$, four coefficients from $HL_3$, and 16 coefficients from $HL_2$. 

The coefficients of the same group correspond to various frequency bands of the same spatial location and the same orientation. The total number of groups is equal to the sum of the number of coefficient in $LH_4$, $HL_4$ and $HH_4$, each of which has $32 \times 32$ coefficients. There are a total of $3 \times 32 \times 32 = 3072$ groups. We denote each group of multi-wavelet tree by $T_{g_m}$, where $m = 1, 2, ..., 3072$.

![Figure 4(a). The Group of Multi-Wavelet Coefficients in Each Tree](image)

![Figure 4(b). Example of a Triple Tree](image)

### 2.3. Genetic Algorithms

Genetic algorithms (GA) based on the laws of natural selection and genetics have been developed since 1975 [Holland, 1975] and have been applied to a variety of optimization and search problems. They have been proved to be very efficient and stable in searching for global optimum solutions. Usually, a simple GA is composed mainly of three operations: selection, genetic operation, and replacement. Figure 5 shows the elementary structure of a simple GA. Following is a brief summary on implementing GA.

Defining the solution representation of the system is the first task of applying GA. Genetic algorithms use a population that is composed of a group of chromosomes to represent the solutions of the system. The solution in the problem domain can then be encoded into the chromosome in the GA domain and vice versa. Initially, a population is randomly generated. The fitness function then uses objective values from an objective function to evaluate the fitness of each chromosome. The fitter chromosome has the greater chance to survive during the evolution process. The objective function is problem-specific; its objective value can represent the system performance index (e.g., an error). Next, a particular group of chromosomes is chosen from the population to be parents. The offspring is then generated from these parents by using genetic
operations, which normally are crossover and mutation. As is the case with their parents, the fitness of the offspring is evaluated and used in replacement processes in order to replace the chromosomes in the current population by the selected offspring. The GA cycle is then repeated until a desired termination criterion is satisfied; for example, the maximum number of generations is reached, or the objective value is below the threshold. Various techniques used in designing GA must be taken into account. These include encoding schemes, fitness evaluation, parent selection, genetic operations, and replacement strategies.

![Figure 5. Structure of a Simple Genetic Algorithm](image)

2.4. Back-Propagation Neural Networks

The back-propagation neural network (BPNN) is one type of supervised learning neural network. It is a potential tool in many signal processing applications. The principle behind BPNN involves using the steepest gradient descent method to reach small approximation [Jones, 2008]. A typical BPNN architecture usually consists of three layers: an input layer, a hidden layer, and an output layer. Each layer has one or more neurons, and each neuron is fully connected to its adjacent layers. Two neurons of each adjacent layer are directly connected to each other. This connection is called a link. Each link has a
weighting value to represent relational degree between two nodes. The illustration of BPNN architecture is shown in Figure 6.

![Figure 6. Architecture of a Typical Back-Propagation Neural Network](image)

The weighting values are determined by the training algorithm described by the following equations:

\[
net_j(t) = \sum \alpha_{i,j} o_i(t) - \theta_j
\]  

(9)

\[
o_i(t+1) = f_{act}(net_j(t))
\]  

(10)

where \(net_j(t)\) is the activation of the neuron \(j\) in the iteration \(t\), \(o_i(t+1)\) is the output of the neuron \(j\) in the iteration \(t+1\), and \(f_{act}\) is the activation function of the neuron. In general, the initial weight values \(\alpha_{i,j}\) are assigned using random values. In each iteration process, all \(\alpha_{i,j}\) are modified using the delta rule according the learning sample. Generally, when the neural network is trained with inputs, the error on the training data set decreases gradually with the epochs or goal set. The BPNN stops training when the goal is achieved or the algorithm reaches the maximum number of epochs specified, whichever condition is met first. After training, a neural network can memorize the characteristics of the learning samples, and predict a new output because of its adaptive capability.
3. PROPOSED METHOD

In this section, we first give a brief overview of the watermark embedding and watermark extracting processes in the DMT domain. We then describe the GA optimization of our proposed method.

3.1. Watermark Embedding Algorithm

Following is a description of the watermark embedding algorithm.

1. Generate a seed by mapping a signature or text through a one-way deterministic function. The seed is used as a secret key \( K \) for watermarking.

2. To increase security, perform a pseudo-random permutation in order to disperse the spatial relationship of the binary watermark pattern. Therefore, it would be difficult for a pirate to detect or remove the watermark. We use \( W \) and \( \tilde{W} \) to denote the original binary watermark image and the permuted watermark image, respectively. The relationship between \( W \) and \( \tilde{W} \) can be expressed as \( \tilde{W}(i, j)=W(i', j') \), where \( (i', j') \) is permuted to the pixel position \( (i, j) \) in a secret order using the secret key \( K \). The \( \tilde{W} \) is transformed and mapped into a binary antipodal sequence \( \hat{W} = \{\hat{w}_i\} \) for \( i = 1, 2, \ldots, N_w \), where \( N_w \) is the length of watermark and \( \hat{w}_i \in \{+1, -1\} \).

3. Transform the original image into a four-level decomposition using the DMT. Then, create multi-wavelet trees and rearrange them into 3072 groups.

4. To increase watermarking security, order the groups \( T_{g_m} \) in a pseudo-random manner. The random numbers can be generated using the secret key \( K \). Then, combine the coefficients of every three groups to form “a triple tree: \( T_{t_i} \)”, for \( i = 1, 2, \ldots, 1024 \). Each watermark bit could be embedded into one triple tree. An example of a triple tree was shown previously in Figure 4(b).

5. For watermark embedding, select the first \( N_w \) triple trees, which have the largest mean values. Then, embed the watermark sequence \( \{w_i\} \) into the selected triple trees using the quantization index modulation technique. The quantization function is given as follows:
where \([x]\) rounds to the greatest integer smaller than \(x\), and \(T_{t_i}\) and \(T'_{t_i}\) denote the triple tree of the original image and the corresponding watermarked image respectively. The variable \(S_j\), for \(j=1, 2, 3\), denotes the quantization steps corresponding to the orientation of the horizontal, vertical, and diagonal of DMT sub-bands, respectively. A large \(S_j\) makes the watermark robust, but it will destroy the original quality of the image. Thus, the value of \(S_j\) should be as large as possible under the constraint of imperceptibility.

6. In order to improve both the quality of the watermark image and the robustness of the watermark, use the genetic algorithm to search for the quantization steps. The details of the GA optimization process used in this study are described in detail in Section 3.

7. Pass the modified DMT coefficients through the inverse DMT to obtain the watermarked image.

![Figure 7. The Watermark Embedding Process](image-url)
3.2. Watermark Extracting Algorithm

Following is a description of the watermark extracting algorithm, as shown in Figure 8.

1. Transform the watermarked image into a four-level decomposition using the DMT. Then, create the multi-wavelet trees and rearrange them into 3072 groups.

2. Order the groups in a pseudo-random manner using a secret key to that which was used in the embedding process. Then, combine every three groups to form a triple tree $T_{tn}$, for $n = 1, 2, \ldots, 1024$.

3. Let $\tilde{T}_i$ denote the first $N_w$ triple trees, which have the largest mean values. The embedded watermark can be extracted from $\tilde{T}_i$ by using the following rule:

$$\tilde{w}_i = \begin{cases} +1 & \text{if } \tilde{T}_i - \left[\frac{\tilde{T}_i}{S_j}S_j \geq S_j/2 \right] \\ -1 & \text{if } \tilde{T}_i - \left[\frac{\tilde{T}_i}{S_j}S_j < S_j/2 \right] \end{cases}$$

(12)

4. After extracting the watermark, use normalized correlation coefficients $(NC)$ to quantify the correlation between the original watermark and the extracted one. A normalized correlation $(NC)$ between $W$ and $\tilde{W}$ is defined as:

---

**Figure 8. Watermark Extracting Process**

- Suspected Image
- Geometrically Correct by BPNN
- DMT
- GA parameters
- Key
- Recovered Watermark
- Inverse Permutation
- Watermark Extraction

---
\[ NC(W, \tilde{W}) = \frac{\sum_{i=1}^{N_w} w_i \tilde{w}_i}{\sqrt{\sum_{i=1}^{N_w} w_i^2 \sum_{i=1}^{N_w} \tilde{w}_i^2}} \]  

where \( W \) and \( \tilde{W} \) denote an original watermark and the extracted one, respectively, and \( \tilde{W} = \{ \tilde{w}_i \} \) for \( i = 1, 2, \ldots, N_w \).

### 3.3. Genetic Algorithm Optimization

The goals of an effective digital watermarking – such as imperceptibility, robustness, and data capacity – usually conflict [Kumsawat, Attakitmongcol, and Srikaew, 2005]. To minimize this, the current study uses genetic algorithms (GA) to search for optimal parameters, which allows the system to achieve optimum performance (Figure 9). For the optimization process, GA is applied in the watermark embedding. The parameters to be searched for are three quantization steps: \( S_1, S_2, \) and \( S_3 \). The objective function of the search process is computed using factors related to both the robustness and the imperceptibility of the watermarked image. A high-quality output image and robust watermark can then be achieved.

![Figure 9. Optimization Diagram for Digital Image Watermarking Using GA](image-url)
Chromosomes in GA represent desired parameters to be searched [Jones, 2008]. The number of chromosomes used in the current study is 30. The encoding scheme is a binary string with 32 bit resolutions for each chromosome. The parameter $S_j$ is then represented by chromosomes with a length of 96 bits. The objective function uses both a universal quality index ($UQI$) [Wang and Bovik, 2002] and a normalized correlation as performance indices. $UQI$ is used as the output image quality performance index because of its role as an imperceptibility measure. Similarly, $NC$ is used as a watermark detection performance index because of its role as a robustness measure. An objective value $W$ can be calculated from:

$$W = \delta_{UQI} \times UQI + \delta_{NC} \times NC$$

(14)

where $\delta_{UQI}$ and $\delta_{NC}$ are the weighting factors of $UQI$ and $NC$, respectively. These weighting factors represent the significance of each index used in the GA search process. If both indices are equally significant, the values of these factors will be 0.5 each, where the relationship $\delta_{UQI} + \delta_{NC} = 1.0$ must always hold. By using objective function $W$ above, one can optimally search the parameter $S_j$ in order to achieve the best of both output image quality and watermark robustness. In the current study, a ranking selection is chosen for the selection mechanism. The crossover and mutation probability is fixed at 0.7 and 0.05, respectively. The chromosomes are then partially replaced by the best chromosome for each generation. The GA process is repeated until the most-fit chromosome (i.e., parameter $S_j$) is optimally found.

3.4. Neural Network Training

In the current study, a back-propagation neural network is applied to the image watermarking scheme to improve the robustness of the watermark against geometric attacks. In the process of getting a prediction model using BPNN, the original image is geometrically transformed through a process such as rotation, scaling, or translation, to generate the training samples. Then, the eigenvectors of every training sample image, which are the Tchebichef image moments [Flusser, Suk, and Zitova, 2009], are computed and then used as the input of BPNN training. In the meantime, the corresponding geometric transformation
parameters \( R, S, Tx \) and \( Ty \), which denote the values of rotation, scaling, translation in x-axis, and translation in y-axis, respectively, are viewed as the target of BPNN training. We construct a three-layer BPNN with 5, 20, and 1 neurons in the input, hidden, and output layers, respectively. The activation function in the hidden layer is a sigmoid function, and the output neuron uses the linear activation function. The Levenberg-Marquardt algorithm is used to increase the training speed and to avoid having the training get into the local minimum. The training error is set at 0.001, and the number of maximum learning iterations is set at 4500. The training is finished either when the training error is smaller than 0.001 or when the iteration reaches the maximum iteration number. After training, the neural network can memorize the characteristics of the learning samples and predict a new output because of its adaptive capability. Therefore, the watermarked image that has undergone rotation, scaling, and translation transformations can be inverted to its original size and orientation for watermark detection.

4. EXPERIMENTAL RESULTS AND DISCUSSION

To evaluate the performance of the proposed watermarking scheme, we conducted experiments using four scanning electron microscope (SEM) images from the Center for Scientific and Technological Equipment at Suranaree University of Technology (SUT). They are all gray-level images with standard dimension 512 \( \times \) 512 pixels. These images, which have various texture characteristics, are shown in Figure 10.
An Intelligent Optimization Approach for Digital Watermarking in the Multi-Wavelet Domain

Figure 10. Original Images Used in Experiments
(a) EM1, (b) EM2, (c) EM3, and (d) EM4

We use two binary “SIP SUT” logos as visually recognizable watermarks. The size of two binary watermarks $W_1$ and $W_2$ are $16 \times 32$ pixels and $32 \times 32$ pixels, respectively. Two original and permutated watermarks are shown in Figure 11(a) and (b) and Figure 12(a) and (b).

Figure 11(a). Original Watermark $W_1$
Figure 11(b). Permuted Watermark $W_1$

Figure 12(a). Original Watermark $W_2$
Figure 12(b). Permuted Watermark $W_2$
4.1. Result of GA Optimization and Neural Network Training

Figure 13 and Figure 14 show the convergence of genetic algorithm (GA) optimization at 30 generations of the four testing images and at watermark $W_1$ and watermark $W_2$, respectively. The results of optimal parameters $S_1$, $S_2$, and $S_3$ from GA searching using the testing images are shown in Table 1 and Table 2. These parameters are optimally varied to achieve the most desirable ones for original images with different characteristics.

(a)

(b)
Figure 13. $S_1$, $S_2$, $S_3$, and PSNR from GA Optimization Process of Watermark $W_1$ and the (a) EM1, (b) EM2, (c) EM3, and (d) EM4
Figure 14. $S_1$, $S_2$, $S_3$ and $PSNR$ from GA Optimization Process of Watermark $W_2$ and the (a) EM1, (b) EM2, (c) EM3, and (d) EM4
Table 1

*PSNR, UQI, and Parameters from GA of Four Watermarked Images with Watermark W₁*

<table>
<thead>
<tr>
<th>Images</th>
<th>S₁</th>
<th>S₂</th>
<th>S₃</th>
<th>PSNR</th>
<th>UQI</th>
</tr>
</thead>
<tbody>
<tr>
<td>EM1</td>
<td>49.53</td>
<td>39.43</td>
<td>28.81</td>
<td>46.88</td>
<td>0.9730</td>
</tr>
<tr>
<td>EM2</td>
<td>49.49</td>
<td>35.80</td>
<td>25.06</td>
<td>48.19</td>
<td>0.9931</td>
</tr>
<tr>
<td>EM3</td>
<td>24.14</td>
<td>35.58</td>
<td>29.59</td>
<td>50.28</td>
<td>0.9992</td>
</tr>
<tr>
<td>EM4</td>
<td>45.88</td>
<td>39.68</td>
<td>25.28</td>
<td>47.27</td>
<td>0.9846</td>
</tr>
<tr>
<td>Average Values</td>
<td></td>
<td></td>
<td></td>
<td>48.15</td>
<td>0.9875</td>
</tr>
</tbody>
</table>

Table 2

*PSNR, UQI, and Parameters from GA of Four Watermarked Images with Watermark W₂*

<table>
<thead>
<tr>
<th>Images</th>
<th>S₁</th>
<th>S₂</th>
<th>S₃</th>
<th>PSNR</th>
<th>UQI</th>
</tr>
</thead>
<tbody>
<tr>
<td>EM1</td>
<td>48.36</td>
<td>38.81</td>
<td>28.17</td>
<td>44.37</td>
<td>0.9589</td>
</tr>
<tr>
<td>EM2</td>
<td>45.52</td>
<td>36.94</td>
<td>25.64</td>
<td>45.03</td>
<td>0.9906</td>
</tr>
<tr>
<td>EM3</td>
<td>28.56</td>
<td>39.44</td>
<td>28.63</td>
<td>46.13</td>
<td>0.9942</td>
</tr>
<tr>
<td>EM4</td>
<td>44.91</td>
<td>36.89</td>
<td>27.97</td>
<td>44.40</td>
<td>0.9766</td>
</tr>
<tr>
<td>Average Values</td>
<td></td>
<td></td>
<td></td>
<td>44.98</td>
<td>0.9801</td>
</tr>
</tbody>
</table>

Figure 15 shows the mean square error in each step when the BPNN is trained using training samples of the testing images. As can be seen from the error curves, after 3571 epochs for watermark W₁ and 2632 epochs for watermark W₂, the network’s errors reached the performance goal, and the final errors are 0.000999762 and 0.000999755, for watermark W₁ and watermark W₂, respectively.
Figure 15. Training Curve of BPNN
Using (a) Watermark $W_1$ and (b) Watermark $W_2$
4.2. Imperceptibility Test Results

We test the output image quality by watermarking four original images with the resulting parameters from GA. Then, we measure the quality of the watermarked image using the peak signal-to-noise ratio ($PSNR$) and the universal quality index ($UQI$). The $PSNR$ and $UQI$ values of the watermarked images are also shown earlier in Table 1 and Table 2, which indicate that the average values of $PSNR$ and $UQI$ for the tested images are 48.15 dB and 0.9875 for watermark $W_1$, and 44.98 dB and 0.9801 for watermark $W_2$, respectively.

4.3. Robustness Test Results

To investigate the robustness of the watermark, we attack watermarked images by various common image operations and geometric distortions. Then, we perform the watermark extraction process and compute the normalized correlation ($NC$). Since our system is a multi-bit watermarking system, the bit error rate ($BER$) is a very useful measure of performance.

We first examine the robustness against common signal processing such as filtering, JPEG compression, and JPEG2000 compression.

Smoothing operations such as median filtering are used to decrease spurious effects that may be present in images from a poor transmission channel. Table 3 shows the results of applying a median filter to a watermarked image. The extracted watermarks and the original watermarks are of high $NC$.

A desirable property of image watermarking algorithm is the robustness of the watermark against lossy compression. We first attack the watermarked image using JPEG and JPEG2000 compression. Table 3, Figure 16 and Figure 17 show the results from a JPEG-compressed version of the watermarked images with JPEG quality factors of 20%, 50%, 80%, and 90%. As indicated, the extracted logos are still visually recognizable, even for the low quality factor of 20%. This fact demonstrates that the proposed method is very robust to JPEG compression.

Table 3, Figure 18, and Figure 19 show the results from a JPEG2000-compressed version of watermarked images with compression ratios 1:10, 1:20, 1:30, and 1:40. The quality of the watermarked images is still good, even under a high compression ratio. The extracted watermarks and the original watermarks are of high $NC$, and the extracted watermarks are still visually recognizable. The bit error rates of the extracted watermarks are quite small at 7.42% and 8.59% for watermark $W_1$ and watermark $W_2$, respectively, even under the situation of compression ratio 1:40.
Table 3
The Average Values of NC and BER Values Under Common Image Operations of the EM1-EM4 Images

<table>
<thead>
<tr>
<th>Attack Method</th>
<th>Watermark W₁</th>
<th></th>
<th>Watermark W₂</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NC</td>
<td>BER (%)</td>
<td>NC</td>
<td>BER (%)</td>
</tr>
<tr>
<td>3x3 Median Filtering</td>
<td>0.9589</td>
<td>5.66</td>
<td>0.9142</td>
<td>9.76</td>
</tr>
<tr>
<td>JPEG Quality 20%</td>
<td>0.9420</td>
<td>8.01</td>
<td>0.9345</td>
<td>7.42</td>
</tr>
<tr>
<td>JPEG Quality 50%</td>
<td>0.9986</td>
<td>0.19</td>
<td>0.9983</td>
<td>0.19</td>
</tr>
<tr>
<td>JPEG Quality 80%</td>
<td>1.0000</td>
<td>0.00</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>JPEG Quality 90%</td>
<td>1.0000</td>
<td>0.00</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>JPEG2000 (1:10)</td>
<td>1.0000</td>
<td>0.00</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>JPEG2000 (1:20)</td>
<td>0.9986</td>
<td>0.19</td>
<td>0.9957</td>
<td>0.48</td>
</tr>
<tr>
<td>JPEG2000 (1:30)</td>
<td>0.9591</td>
<td>5.66</td>
<td>0.9471</td>
<td>6.05</td>
</tr>
<tr>
<td>JPEG2000 (1:40)</td>
<td>0.9462</td>
<td>7.42</td>
<td>0.9246</td>
<td>8.59</td>
</tr>
</tbody>
</table>

Figure 16. Logo W₁ Extracted from EM1 Watermarked Image After JBEG Compression Attack of Various Quality Factors
(a) 20%, (b) 50%, (c) 80% and (d) 90%
Figure 17. Logo $W_2$ Extracted from EM1 Watermarked Image
After JPEG Compression of Various Quality Factors
(a) 20%, (b) 50%, (c) 80% and (d) 90%

Figure 18. Logo $W_1$ Extracted from EM1 Watermarked Image
After JPEG2000 Compression of Various Compression Ratios
(a) 1:10, (b) 1:20, (c) 1:30 and (d) 1:40
Next, we test robustness with respect to geometrical attacks such as rotation, scaling, and translation. Figure 20 illustrates different types of geometric attacks to the watermarked image. Before watermark extraction is performed, the back-propagation neural network is used to memorize the relations between a geometric distortion and the corresponding watermarked image to correct the attacked image geometrically. The experiment results are shown in Figure 21, Figure 22, and Table 4. The results show that the watermark detection algorithm exhibits very good performance and that the extracted watermark is still visually recognizable.

The results from our proposed algorithm called GANW1 are compared with the method based on multi-wavelet-tree quantization [Kumsawat, Attakitmongcol, and Srikaew, 2007] called Method1. The comparison is made using watermark $W_1$. The original image is a 256 gray-level image with the size of 512 x 512 pixels and the watermark length $N_w = 512$. For a fair comparison, the quality of the watermarked EM1 image (PSNR around 38 dB) and embedding capacity (512 bits) for both schemes must be the same. From the comparison results in Table 4, it can be seen that the proposed method is very robust to various attacks and yields a significantly more robust watermark than Method1 does.
Figure 20. Different Types of Attacks to EM1 Watermarked Image
(a) Rotation 5°, (b) Rotation 15°, (c) Rotation 90°, (d) Scaling 200%,
(e) Translation 20 Pixels in x-axis, and (f) Translation 20 Pixels in x- and y-axis
Figure 21. Logo $W_1$ Extracted from EM1 Watermarked Image After
(a) Rotation 5°, (b) Rotation 15°, (c) Rotation 90°, (d) Scaling 200%,
(e) Translation 20 Pixels in x-axis, and (f) Translation 20 Pixels in x and y-axis
Figure 22. Logo $W_2$ Extracted from EM1 Watermarked Image After
(a) Rotation 5°, (b) Rotation 15°, (c) Rotation 90°, (d) Scaling 200%,
(e) Translation 20 Pixels in x-axis, and (f) Translation 20 Pixels in x and y-axis
An Intelligent Optimization Approach for Digital Watermarking in the Multi-Wavelet Domain

Table 4
Average Values of \( NC \) and \( BER \) Under Geometrical Attacks of the EM1-EM4 Images

<table>
<thead>
<tr>
<th>Attack Method</th>
<th>Watermark ( W_1 )</th>
<th>Watermark ( W_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( NC )</td>
<td>( BER ) (%)</td>
</tr>
<tr>
<td>Rotation 5°</td>
<td>0.9065</td>
<td>12.69</td>
</tr>
<tr>
<td>Rotation 15°</td>
<td>0.8779</td>
<td>16.21</td>
</tr>
<tr>
<td>Rotation 90°</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>Rotation -90°</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>Scaling 150%</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>Scaling 200%</td>
<td>1.0000</td>
<td>0.00</td>
</tr>
<tr>
<td>Translation 5 pixels in x-axis</td>
<td>0.9775</td>
<td>3.12</td>
</tr>
<tr>
<td>Translation 5 pixels in y-axis</td>
<td>0.9763</td>
<td>3.32</td>
</tr>
<tr>
<td>Translation 20 pixels in x-axis</td>
<td>0.9648</td>
<td>4.84</td>
</tr>
<tr>
<td>Translation 20 pixels in y-axis</td>
<td>0.9546</td>
<td>6.25</td>
</tr>
<tr>
<td>Translation 20 pixels in x and y-axis</td>
<td>0.9256</td>
<td>10.15</td>
</tr>
</tbody>
</table>

Finally, the results obtained from our proposed method using watermark \( W_2 \) (which is called GANW2) are compared with the method based on wavelet-tree quantization [Jin and Jin, 2010] called Method2. The original image is a 256 gray-level image with the size of \( 512 \times 512 \) pixels, and the watermark length \( N_w \) is 1024. For our proposed method, the results of optimal parameters \( S_1, S_2, \) and \( S_3 \) from GA searching using four test images and watermark \( W_2 \) are shown in Table 5. The \( PSNR \) of the watermarked images and watermark length of both methods are about 43 dB and 1024 bits, respectively. The comparison results in

Journal of International Business and Information
Table 6 and Table 7 show that our proposed method outperforms the Method2 in most cases.

**Table 5**

**Comparison of NC Values Under Signal processing attacks of the EM1 Images**

<table>
<thead>
<tr>
<th>Attack Method</th>
<th>Proposed Method: GANW1</th>
<th>Method1</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 × 3 Median Filtering</td>
<td>0.9585</td>
<td>0.5820</td>
</tr>
<tr>
<td>JPEG Quality 20</td>
<td>0.9420</td>
<td>0.4141</td>
</tr>
<tr>
<td>JPEG Quality 50</td>
<td>0.9986</td>
<td>0.9023</td>
</tr>
<tr>
<td>JPEG Quality 80</td>
<td>1.0000</td>
<td>0.9922</td>
</tr>
<tr>
<td>JPEG2000 (1:10)</td>
<td>1.0000</td>
<td>0.9610</td>
</tr>
<tr>
<td>JPEG2000 (1:20)</td>
<td>0.9986</td>
<td>0.7188</td>
</tr>
<tr>
<td>Rotation 5°</td>
<td>0.9065</td>
<td>0.3867</td>
</tr>
<tr>
<td>Rotation 15°</td>
<td>0.8779</td>
<td>0.4140</td>
</tr>
<tr>
<td>Scaling 150%</td>
<td>1.0000</td>
<td>0.3633</td>
</tr>
<tr>
<td>Scaling 200%</td>
<td>1.0000</td>
<td>0.3867</td>
</tr>
<tr>
<td>Translation 5 Pixels in x-axis</td>
<td>0.9775</td>
<td>0.2266</td>
</tr>
<tr>
<td>Translation 20 Pixels in x-axis</td>
<td>0.9648</td>
<td>0.2305</td>
</tr>
</tbody>
</table>
An Intelligent Optimization Approach for Digital Watermarking in the Multi-Wavelet Domain

Table 6
The Results of Parameters Using the GA Optimization Process and Comparison of PSNR of the Four Standard Test Images

<table>
<thead>
<tr>
<th>Images</th>
<th>Proposed Method: GANW2</th>
<th>Method2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$S_1$</td>
<td>$S_2$</td>
</tr>
<tr>
<td>Lena</td>
<td>47.89</td>
<td>39.12</td>
</tr>
<tr>
<td>Cameraman</td>
<td>45.52</td>
<td>35.77</td>
</tr>
<tr>
<td>Baboon</td>
<td>46.90</td>
<td>36.77</td>
</tr>
<tr>
<td>Pepper</td>
<td>44.62</td>
<td>34.64</td>
</tr>
<tr>
<td>Average of PSNR (dB)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7
Comparison of NC Values Under Signal Processing and Geometrical Attacks of the Four Standard Test Images

<table>
<thead>
<tr>
<th>Attacks</th>
<th>Lena</th>
<th>Cameraman</th>
<th>Pepper</th>
<th>Baboon</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAN W2 Method2</td>
<td>5 x 5 Mean Filtering</td>
<td>0.7879</td>
<td>0.5982</td>
<td>0.6300</td>
</tr>
<tr>
<td>GAN W2 Method2</td>
<td>7 x 7 Median Filtering</td>
<td>0.6282</td>
<td>0.5730</td>
<td>0.6384</td>
</tr>
<tr>
<td>GAN W2 Method2</td>
<td>Rotation 10°</td>
<td>0.8832</td>
<td>0.6721</td>
<td>0.8055</td>
</tr>
<tr>
<td>GAN W2 Method2</td>
<td>Scaling 85%</td>
<td>1.0000</td>
<td>0.7304</td>
<td>1.0000</td>
</tr>
</tbody>
</table>
5. CONCLUSIONS

In recent years, the digital watermarking technique has become a potential solution for copyright protection and authentication and integrity verification of digital media such as images, video, audio, and multimedia content. Many digital watermarking techniques are reported in the literature. Among the widely used watermarking techniques, transform domain-based techniques are appealing for various applications because of their inherent advantage of greater robustness. This paper has proposed a digital image watermarking algorithm for copyright protection. The watermark is embedded in selected coefficients using quantization index modulation in the multi-wavelet transform domain. This watermark can be extracted and then used to authenticate the legal copyright holder.

In this study, we have developed an optimization technique using genetic algorithms to search for optimal quantization steps to improve the quality of the watermarked image and the robustness of the watermark. In addition, a back-propagation neural network is applied to the image watermarking scheme to improve the robustness of the watermark against geometric attacks. In the process of obtaining a prediction model using a back-propagation neural network, we geometrically transformed the original image using techniques such as rotation, scaling, and translation, to generate the training samples. Then, we computed the eigenvectors of every training sample image, which are the Tchebichef image moments, and then used these as input for back-propagation neural network training. Our experimental results show that the proposed algorithm can achieve a watermarked image with good perceptual invisibility and security. The watermark is also robust against various common image processing and geometrical attacks.

ACKNOWLEDGMENT

This work was supported by Suranaree University of Technology (SUT) and by the Office of the Higher Education Commission under the NRU project of Thailand.

REFERENCES


ABOUT THE AUTHORS

Prayoth Kumsawat was born in Maehongson, Thailand, in 1969. He received his B. Eng. degree in electrical engineering from the Royal Thai Air Force Academy, Bangkok, Thailand, in 1994, his M. Eng. degree in electrical engineering from Kasetsart University, Bangkok, Thailand, in 1997, and his Ph.D. in electrical engineering from Suranaree University of Technology, Nakhon Ratchasima, Thailand, in 2006. Since 1999, he has been with the Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand, where he is currently an assistant...
An Intelligent Optimization Approach for Digital Watermarking in the Multi-Wavelet Domain

Kitti Attakitmongcol was born in Satun, Thailand, in 1972. He received his B. Eng. degree in electronics engineering from King Mongkut’s Institute of Technology, Ladkrabang, Bangkok, Thailand, in 1994, and his M.S. degree and Ph.D., both in electrical engineering, from Vanderbilt University, Nashville, Tennessee, USA, in 1996 and 1999, respectively. Since 1999, he has been with the Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand, where he is currently an associate professor in the School of Electrical Engineering. His research interests include digital signal processing, image processing, wavelet transform, and multi-wavelet transform.

Arthit Srikaew was born in Ubol Ratchathani, Thailand, in 1972. He received his B. Eng. degree from King Mongkut’s Institute of Technology Ladkrabang, Bangkok, Thailand, in 1994, and his M.S. degree and Ph.D., both in electrical engineering, from Vanderbilt University, Nashville, Tennessee, USA, in 1997 and 2000, respectively. Since 2000, he has been with the School of Electrical Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand, where he is currently an associate professor. His main research interests are in the areas of computer and robot vision, image processing, neural networks, artificial Intelligence, and intelligent systems.