Digital Watermarking of Images Using Neural Networks *

Min-Shiang Hwang[†§] Chin-Chen Chang[‡] Kuo-Feng Hwang[‡]

Department of Information Management [†]
Chaoyang University of Technology
Wufeng, Taiwan, R.O.C.
Email: mshwang@cyut.edu.tw

Department of Computer Science and [‡]
Information Engineering,
National Chung Cheng University,
Chaiyi, Taiwan, R. O. C.
Email: ccc@cs.ccu.edu.tw

June 12, 2000

*This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC89-2213-E-324-025.

[§]Responsible for correspondence.

Digital Watermarking of Images Using Neural Networks

Abstract

Watermarking techniques are primarily used for copyright protection. In this paper, we propose a digital watermarking scheme, which is based on neural network, cryptography, and image processing techniques. Our scheme can achieve the following two goals. The first is that illegal users do not know the location of an embedded watermark in the image. The second is that a legal user can retrieve the embedded watermark from an altered (filtering, lossy compression, and scaling) image. The security and robustness of the proposed scheme are improved using neural network technology.

KeyWords: Digital watermarking, neural networks, one-way hash function

1 INTRODUCTION

Along with the proliferation of the World-Wide Web, a huge amount of multimedia content is available for browsing and downloading by any user over the network. Today, many traditional transactions are conducted over the Internet. Many new business applications are employed on electronic networks, pay-per-view video on demand, on-line consulting, component-based software, and virtual shopping etc. Therefore, security and copyright issues have become increasingly important. Cryptography can overcome many security problems. Until now, copyright protection has been very weak in the digital world. Digital watermarking is applicable to copyright protection and the integrity of checks, etc.

There are two types of watermarking from the visual viewpoint. The first is an embedded watermark that is visible. The primary advantage of visible watermarks is

the ease of identification by the owner. However, the embedded watermark is easier to remove using image processing techniques. The second type is the embedded watermark that is invisible. This paper will address the second type of watermark. In order to achieve the target of copyright protection, digital watermarking must satisfy the following primary requirements.

1. The quality of a watermarked image must be very high.

2. There must be no requirement to retrieve the watermark from the original watermarked image.

3. As in cryptography, the security can not be based upon the assumption that possible intruders do not know how digital watermarking is applied.

4. Except by the copyright owner, there must be no method to detect or remove the watermark from the watermarked image.

5. It must be possible to retrieve the watermark after multiple and varied image processes, such as low-pass filtering, high-pass filtering, lossy compression, scaling, etc.

For more information, please consult [12, 14]. The previous watermarking techniques that have been proposed are classified into two categories. The first method embeds a watermark into the spatial domain. In general, the main advantage of this method is that it has a good computing performance, but the disadvantages are lower security and robustness [13]. The second method embeds a watermark into the frequency domain. The method transforms the original data into the frequency domain. The watermark is then embedded after using a Fourier, Discrete Cosine, or Wavelet transform.

In this paper, we propose a digital watermarking scheme which embeds the watermark into the frequency domain. In our scheme, we use a back-propagation

neural network (BPN) to improve both security and robustness of the watermarked image. A one-way hash function is also used in our scheme to decide the locations of the embedded watermark for enhancing security.

The rest of this paper is organized as follows. Section 2 surveys some related digital watermarking schemes. In Section 3, we briefly discuss the discrete cosine transform (DCT), back-propagation neural network (BPN), and one-way hash function, which are used in our scheme. Next, we propose our digital watermark scheme in Section 4. The experimental results are presented in Section 5. Finally, Section 6 and Section 7 are discussions and conclusions of this paper.

2 RELATED WORKS

Cox et al. proposed a watermarking technique, which is based on DCT [2]. They advocated that a watermark should be embedded in the most perceptually significant parts of an image. That scheme embedded a set of independent and identically distributed samples from a Gaussian distribution into the most perceptually significant frequency components. The results achieved by Cox et al. produced a technique that is remarkably robust against not only various image processing operations, but also printing and rescanning. However, it has been demonstrated that this scheme is susceptible to collusion attacks [6]. They conjectured in the literature that 8-10 copies would be sufficient for a collusion attack. Additionally, we argue that Cox's scheme may not be adaptable to a larger watermark, such as a mark larger than or equal to a 64×64 binary image.

Kutter et al. used amplitude modulation to embed a signature into a color image [5]. Since the blue channel is relatively less sensitive in the color domain, the blue channel was used to embed the signature. In this method, a single bit s is multiply embedded into randomly selected pixels $p(x, y)$. Each selected pixel's blue channel

B is modified by a fraction of luminance L, i.e.,

$$B'_{x,y} \leftarrow B_{x,y} + q(2s-1)L_{x,y}. \tag{1}$$

Here, q is a constant that represents the strength of the signature. The original value of $B_{x,y}$ is predicted as $B''_{x,y}$, and the retrieved signature is obtained by $B'_{x,y}$ and $B''_{x,y}$. The prediction $B''_{x,y}$ is computed as follows.

$$B''_{x,y} = \frac{1}{4c}\left(\sum_{k=-c}^{c} B_{x+k,y} + \sum_{k=-c}^{c} B_{x,y+k} - 2B_{x,y}\right), \tag{2}$$

where c is the size of the crossed-shaped neighborhood. The embedded bit is retrieved according to the difference δ between the predicted and coded value as

$$\delta_{x,y} = B''_{x,y} - B'_{x,y}. \tag{3}$$

The inventors claim that this algorithm is robust against translation, rotations, slight blurring, JPEG attack, and composition with other images. As mentioned in Section 1, the details of the algorithm must be open. Therefore, an intruder may modify whole pixels using the proposed prediction function. After modification, in the retrieval stage of Kutter's scheme, whole values of $\delta_{x,y}$ will be closed to zero. Hence, the embedded watermark may be destroyed. Even if c is kept secret, the range of c is limited. Consequently, Kutter's scheme has a weakness in this aspect.

3 RELATED THEORIES

In this section, we briefly discuss the discrete cosine transform (DCT), the concept of one-way hash functions and the back-propagation neural network (BPN). All of these theories are used in our scheme. The details of our proposed scheme will be described in next section.

Discrete cosine transform [1, 4] uses the cosine waveform to represent original data. DCT can concentrate an image's energy into the left-upper corner. According to this characteristic, DCT has been used for image data compression, such as JPEG.

Generally, the DCT blocks (8×8) are not overlapped in its applications. We must also emphasize that the DCT blocks are allowed to overlap partially in our scheme. In this way, the security of the digital watermarking can be improved. In other words, a pirate can not exactly obtain the DCT block in our scheme.

The concept of the one-way hash function involves providing an arbitrary parameter x to a function h, which can easily produce an output value $z = h(x)$. However, it is difficult to produce x from the output value z. The other characteristic of the one-way hash function is given x, it is very difficult to find other x' such that $h(x) = h(x')$. To demonstrate the concept of one-way hash function, the following example is provided. Let $z = g^x \bmod p$, where g, x, and p are 512 bits in length. It is simple to produce z using exponential operations, but it is very difficult to produce x from z (this is a discrete logarithm problem [8]). There are many famous one-way hash functions that have been applied, such as MD5 [10] and SHA [7], etc. In our scheme, a one-way hash function is used, inspired from Rabin's public-key cryptosystem [8], to decide the locations of DCT blocks in which the watermark is to be hidden.

The Back-Propagation Network (BPN) is one type of supervised learning neural network [9, 11, 15]. It is a very popular model in neural networks. The principle behind BPN involves using the steepest gradient descent method to reach a small approximation error.

The general model has an architecture like that depicted in Figure 1. There are three layers, input layer, hidden layer, and output layer. Each layer has one or more neurons or units. Each unit is fully connected to its adjacent layers. There may be more than one hidden layer, according to practical necessity. Two units of each adjacent layer are directly connected to one another, called a link. Each link has a weighted value, representing the relational degree between the two units. Each input vector has its own desired output vector in the BPN model. A typical

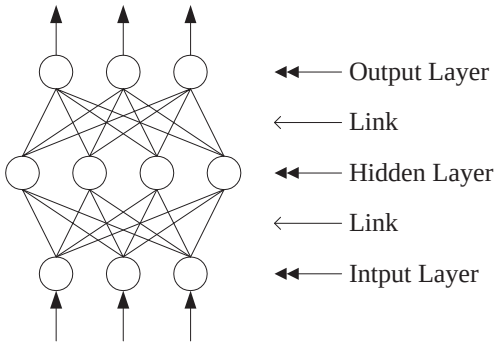


Figure 1: Architecture of BPN

unit is shown in Figure 2. The details of the training algorithm are describe by the following equations:

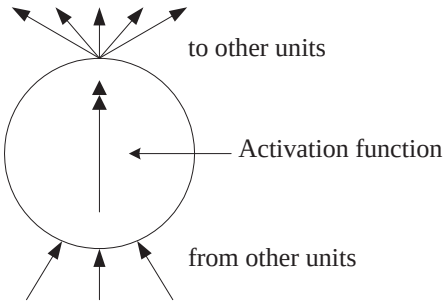


Figure 2: A typical computational unit in BPN

$$net_j(t) = \sum_i w_{ij} o_i(t) - \theta_j, \tag{4}$$

$$o_j(t+1) = f_{act}(net_j(t)). \tag{5}$$

Here

j	index for some current unit
i	index of a predecessor of current unit j
$f_{act}()$	activation function
$net_j(t)$	the activation of unit j
θ_j	threshold or bias of unit j
w_{ij}	weight of the link from unit i to unit j
$o_j(t)$	output of unit i in iteration t

The activation function used in our scheme are defined as follows.

$$f_{act}(x) = \frac{1}{1+e^{-x}}. \tag{6}$$

The activation function $f_{act}(x)$, also called a sigmoid function, is the most popular function used in BPN. Its derivative $f'_{act}(x)$, which is equal to $f_{act}(x)(1-f_{act}(x))$, is required in the training process to modify the weight w_{ij}. In each iteration, the generalized delta rule, which is described below, is used to compute the required change $\Delta\, w_{ij}$ for w_{ij}.

$$\Delta\, w_{ij} = \eta \times \delta_j \times o_i, \tag{7}$$

$$\delta_j = \begin{cases} f'_{act}(net_j)(t_j - o_j) & \text{if unit } j \text{ is an output unit} \\ f'_{act}(net_j)\sum_k \delta_k w_{jk} & \text{if unit } j \text{ is an hidden unit} \end{cases} \tag{8}$$

where

η	learning factor (a constant)
t_j	target output value of unit j
k	index of a successor unit
o_j	output value of the predecessor unit j

Generally, all initial weights w_{ij} are assigned using random values. After training, we can use novel input vectors with w_{ij} to predict the corresponding output vectors.

In our scheme, BPN is used to learn the relationship between the DCT coefficients. The watermark is hidden by changing one of the DCT coefficients. That

is one bit of the watermark is hidden in one DCT block. Since the relationship is trained before the coefficient is changed, that relationship is recorded using the weights and threshold values. We can retrieve the approximate original coefficients using these weights and thresholds. The watermark can be retrieved by the relationship between the changed coefficients and the approximate original coefficients.

4 OUR SCHEME

In this section, we describe the proposed digital watermarking scheme. The first subsection describes the algorithm of our scheme. In the second subsection, we present the linear translation function, which is used to translate the DCT coefficients into variable values in the BPN model.

4.1 Proposed Algorithm

Let O denotes an original image with 8 bits per pixel which was used to embed a watermark. The original image is usually represented as a two-dimensional (2D) arrays. We define O as follows.

$$O = \begin{bmatrix} o(0,0) & o(0,1) & \cdots & o(0,O_w-1) \\ o(1,0) & o(1,1) & \cdots & o(1,O_w-1) \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & o(i,j) & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ o(O_h-1,0) & o(O_h-1,1) & \cdots & o(O_h-1,O_w-1) \end{bmatrix} \tag{9}$$

where $o(i,j)$ is an integer, $0 \leq o(i,j) \leq 255, 0 \leq i < O_h$, and $0 \leq j < O_w$. O_h and O_w are the original image's height and width, respectively.

A binary digit image watermark, W, was used in our scheme. The watermark image must be meaningful proof for someone who owns the copyright of the original image. In general, a "trademark" of the enterprise can be selected as a watermark. A watermark is defined as follows.

$$W = \begin{bmatrix} w(0,0) & w(0,1) & \cdots & w(0,W_w-1) \\ w(1,0) & w(1,1) & \cdots & w(1,W_w-1) \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & w(i,j) & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ w(W_h-1,0) & w(W_h-1,1) & \cdots & w(W_h-1,W_w-1) \end{bmatrix} \quad (10)$$

where $w(i,j)$ is a binary, $w(i,j) \in \{0,1\}$, $0 \leq i < W_h$, and $0 \leq j < W_w$. W_h and W_w are the watermark's height and width, respectively. The watermark is represented as 2D arrays. In order to hide the watermark in the original image, we converted the stored 2D data into a one-dimensional array as follows.

$$W = (W_0, W_1, \cdots, W_k, \cdots, W_h \times W_w - 1), \quad (11)$$

where $W_k = w(i,j)$, $k = j + (i \times W_w)$, $0 \leq i < W_h$, and $0 \leq j < W_w$. Inspired from Rabin's public-key cryptosystem [8], we chose two primes p and q, and let n be equal to the product of p and q. Two secret keys k_1 and k_2 are chosen by the image's owner, which are used to decide the locations (x_i, y_i) where the watermark will be hidden. (x_i, y_i) is computed using the following location decision procedure.

1. Compute the initial location (x_0, y_0) as follows.

$$X_0 = k_1^2 \bmod n, \quad (12)$$

$$Y_0 = k_2^2 \bmod n, \quad (13)$$

$$x_i = X_0 \bmod O_w, \quad (14)$$

$$y_i = Y_0 \bmod O_h. \quad (15)$$

2. Compute the other $W_h \times W_w - 1$ locations as follows.

$$X_i = X_{i-1}^2 \bmod n, \quad (16)$$

$$Y_i = Y_{i-1}^2 \bmod n, \quad (17)$$

$$x_i = X_i \bmod O_w, \quad (18)$$

$$y_i = Y_i \bmod O_h, \quad (19)$$

where $i = 1, 2, \cdots$. Using the same locations must be avoided. In other words, if (x_i, y_i) have been chosen in this procedure, ignore the (x_i, y_i) location. After this step, we get $W_h \times W_w - 1$ locations (x_i, y_i). These locations are different from (x_0, y_0) and each other.

Therefore, using the location decision procedure, we can produce $W_h \times W_w$ available locations in total. Note that if (x_i, y_i) have the same coordinates, it seems that an infinite loop will occur in Equations (18) and (19). However, this situation can be avoided using the larger modulus n (a product of two large primes) in Equations (16) and (17). Each location (x_i, y_i) corresponds to a subimage M_i. Here M_i is produced using:

$$M_i = Submatrix(O, x_i, x_i + 7, y_i, y_i + 7). \tag{20}$$

Here the function $Submatrix(\cdot)$ denotes acquiring a DCT block from the image O's coordinate (x_i, y_i) to coordinate $(x_i + 7, y_i + 7)$. In practice, we have to avoid $(x_i + 7) > O_w$ and $(y_i + 7) > O_h$, and the simplest way is to change the moduli in Equations (18) and (19) into $(O_w - 8)$ and $(O_h - 8)$, respectively. However, there are many different methods that can be applied here [4]. These blocks may be partially overlapped with one another. An illustration of the partially overlapped DCT blocks (M_i) is shown in Figure 3. The DCT transformation on these M_i is performed next. Figure 4 shows that the first nine AC coefficients $(AC1_i, AC2_i, \ldots, AC9_i)$ are the input vectors and the twelfth AC coefficient $AC12_i$ is the output vector in the BPN model. Note that $AC10, AC11, \ldots, AC14$ are also candidates for the output vector. According to the DCT characteristics, if the AC coefficients with lower indices are used, the robustness of the watermark increases. At the same time, the quality of the watermarked image is decreased. The BPN model used in the proposed scheme is shown in Figure 5. There is only one hidden layer, which contains four units. All coefficients used in the BPN model must be translated by a linear function. We will

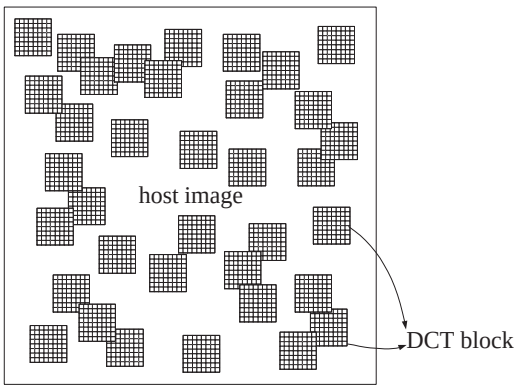


Figure 3: Selected DCT blocks which are allowed to partially overlap.

describe it in the next subsection. The initial weights w_{ij} and the unit threshold of BPN model are random values in our scheme. After training, the weights w_{ij} can be used for retrieving the watermark. The corresponding output vector $AC12_i'$ is acquired from $AC12_i$ at the end. The $AC12_i'$ is calculated below.

$$AC12_i' = N(AC1_i, AC2_i, \ldots, AC9_i). \tag{21}$$

Here $AC12_i'$ is the output of the BPN. The watermark W_i is embedded by replacing the original $AC12_i$ with $AC12_i''$, where $AC12_i''$ is computed according to $AC12_i'$ and W_i as follows.

$$AC12_i'' = \begin{cases} AC12_i' - \delta, & \text{if } W_i = 0, \\ AC12_i' + \delta, & \text{if } W_i = 1. \end{cases} \tag{22}$$

Here δ is a system parameter. A larger δ will result in a greater robustness in the watermarked image. But, the distortion will be increased too. The value δ can

DC	1	5	6	14			
2	4	7	13				
3	8	12					
9	11						
10							

Figure 4: Index of AC Components.

be determined by the user's requirements. After the twelfth coefficients are replaced and the inverse DCT is transformed, the embedding process is completed.

The retrieval procedure for the watermark from the watermarked image is similar to the embedding procedure. Except for the training procedure, inverse DCT transformation is not required in the retrieval process. When the correct secret keys (k_1, k_2) and weights are introduced, the corresponding $AC12$ and $AC12'$ can be obtained. The retrieved watermark is produced using the relationship between $AC12$ and $AC12'$ as

$$W_i = \begin{cases} 0, & \text{if } AC12_i < AC12'_i, \\ 1, & \text{if } AC12_i > AC12'_i. \end{cases} \tag{23}$$

4.2 Linear Translation Function

The activation function used in BPN always outputs the real number range from 0 to 1. Therefore, we must introduce a linear translation function to translate the DCT coefficients, which are real number between -600 and 600, into real numbers that range from 0 to 1. We use the linear translation function to translate all AC components, from number 1 to number 9 and number 12 ($AC1, AC2, \ldots, AC9$, and $AC12$). We have randomly chosen 4096 patterns (the same size as the watermark) of these AC components as our experiment. According to the frequency distribution

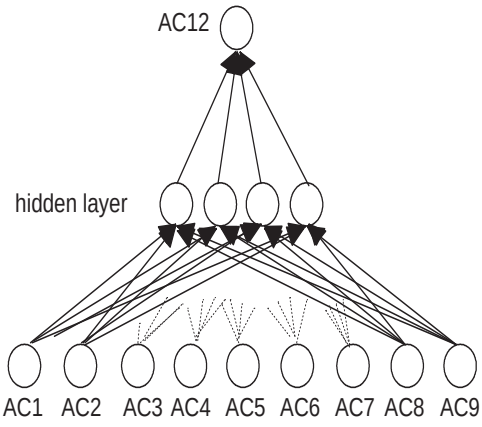


Figure 5: The BPN model used in our scheme.

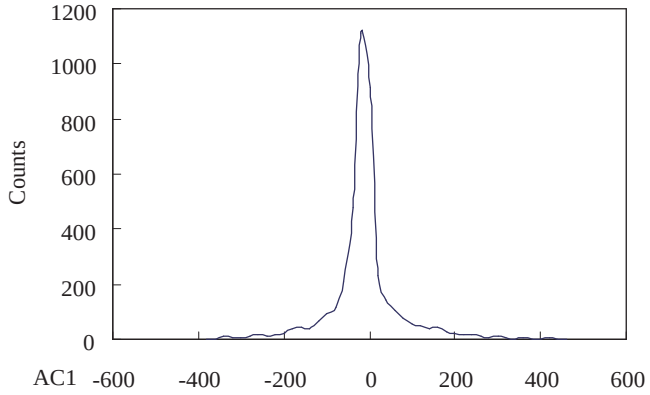


Figure 6: The frequency distribution of AC1.

of $AC1$, as shown in Figure 6, we determined that the mean is equal to -2 and the standard derivation is equal to 84. It can be seen that the frequency distribution appears as a normal distribution. The other AC components' distributions are similar to the $AC1$. In our scheme, the linear translation function $f_{trn}(x)$ $(= y)$ and its inverse function $f_{trn}^{-1}(y)$ from this distribution are defined as follows.

$$f_{trn}(x) = \frac{x + 1000}{2000}, \tag{24}$$

$$f_{trn}^{-1}(y) = 2000y - 1000. \tag{25}$$

5 EXPERIMENTAL RESULTS

In image processing, PSNR (peak signal to noise ratio) is usually used to assess the differences in image quality, from pre-processing to post processing. A larger value for PSNR means there is little difference between the original image and the processed image. A PSNR value greater than or equal to 30 dB, means that the processed image's quality is acceptable.

In our experiment, we let δ equal to 20. Figure 7 shows the SSE (Summation of Square Error) variations in the training process. It also shows that good values for η range from 0.2 to 0.3. After about 80 epochs, the training process is converged.

In our experiment, the SNNS (Stuttgart Neural Network Simulator) [15] was used for the BPN simulation.

To estimate the correctness of the retrieved watermark, the bit correct ratio (BCR) is defined as follows.

$$BCR = \frac{\sum_{i=1}^{W_h}\sum_{j=1}^{W_w}\overline{w(i,j)\oplus w'(i,j)}}{W_h \times W_w} \times 100\%. \quad (26)$$

Here $w(i,j)$ is the original watermark element, $w'(i,j)$ is the element from the retrieved watermark, and $\oplus$ denotes the exclusive-OR operator.

The original image of "Lena" ($512 \times 512, 8$ bits/pixel) is shown in Figure 8(a). Figure 8(c) is the watermark of "ChaoYang University of Technology" represented by Chinese characters (64×64 binary image). We used the proposed scheme with the above parameters to embed Figure 8(c) into Figure 8(a), which produced the watermarked image in Figure 8(b) (PSNR=37.78 dB). Figure 8(d) is the retrieved watermark from Figure 8(b). The BCR is 96.68%.

Next, we modified the watermarked image using blurring, sharpening, lossy compression, and scale processing. The retrieved watermarks from these altered images are recognizable. Figure 9(a) is a modified image of Figure 8(b) using a blurring algorithm [4]. The PSNR of the processed image is 29.46 dB. The sharpening algorithm was used to modify the watermarked image as Figure 9(b), with associated

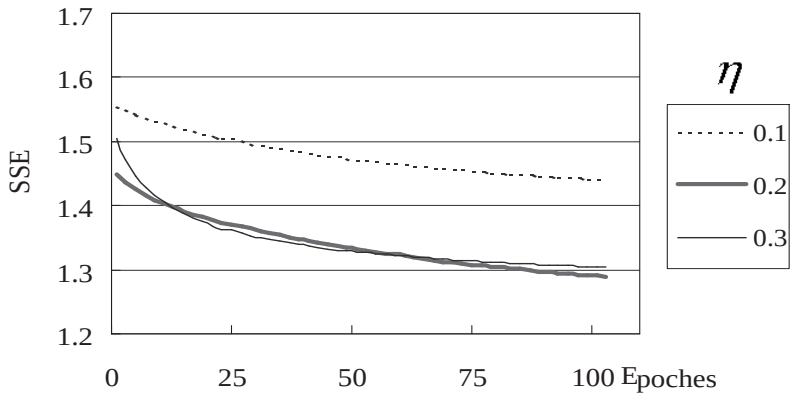


Figure 7: Variations of SSE.

(a)

(b) PSNR=37.78 dB

(c)

(d) BCR=96.68%

Figure 8: (a)Original image of "Lena", (b)Watermarked image of "Lena", (c)Watermark of "ChaoYang University of Technology", (d)Retrieved watermark from watermarked image.

(a) PSNR=29.46 dB

(b) PSNR=33.35 dB

(c) BCR=86.99%

(d) BCR=96.73%

Figure 9: (a)Blurred image with watermark, (b)Sharpened image with watermark, (c)Retrieved watermark from Figure 9(a), (d)Retrieved watermark from Figure 9(b).

PSNR=33.35 dB. The retrieved watermark from Figure 9(a) is shown in Figure 9(c). The BCR is 86.99%. The retrieved watermark from Figure 9(b) is shown in Figure 9(d). The BCR is 96.73%.

A lossy compression algorithm (JPEG) is applied to the watermarked image (Figure 8(b)). Figure 10(a) shows the reconstructed image from the compressed image, with associated PSNR=33.35 dB. Figure 10(b) is the shrunken watermarked image from from 512×512 to 256×256. Figure 10(c) is the retrieved watermark from Figure 10(a). The BCR is 86.64%. Before the watermark is retrieved from the shrunken image, the image is resized and interpolated to 512×512 using the nearest neighbor algorithm. The retrieved watermark is shown in Figure 10(d). The BCR is 84.50%.

Finally, the retrieved watermarks under different host images and several attacks,

(a) PSNR=33.35 dB

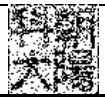
(b) PSNR=28.26 dB

(c) BCR=86.64%

(d) BCR=84.50%

Figure 10: (a)Reconstructed image from JPEG compressed image, (b)Scaled to 256×256 from watermarked image, (c)Retrieved watermark from Figure 10(a), (d)Retrieved watermark from Figure 10(b).

Table 1: Retrieved watermarks from different host images and various attacks.

Host image		Embedded	JPEG	Blurring	Sharpening	Scaling
Barbara	PSNR(dB)	36.36	30.90	25.31	17.06	22.23
	retrieved watermark					
	BCR(%)	97.51	82.03	89.75	94.02	79.86
Plane	PSNR(dB)	36.27	31.33	30.88	20.84	26.08
	retrieved watermark					
	BCR(%)	97.66	80.64	89.89	96.75	79.83

are shown in Table 1. The experimental results show that the retrieved watermarks are recognizable after being exposed to various attacks.

In the above experimental results, a legal user can retrieve the embedded watermark from an altered (lossy compression (JPEG), blurring, sharpening, and scaled) image.

6 DISCUSSIONS

The secret keys used in our scheme ranged from 0 to n. The probability of directly destroying the twelfth AC component of the watermarked image is very low. Suppose that the probability of successfully destroying one bit of the watermark is one-second and the embedded watermark cannot be recognized if more than fifty percent of watermark elements have been destroyed. The probability is

$$\left(\frac{1}{2 \times O_h \times O_w}\right)^{\frac{W_h \times W_w}{2}}, \tag{27}$$

where O_h and O_w are the image's height and width, respectively, and W_h and W_w are the watermark's height and width, respectively. Therefore, the security of the proposed scheme is very high.

The quality of the watermarked image is high (see Figure 8(b)) in our scheme. In addition, the original image is not needed to be used in the procedure of retrieving

the watermark. Consequently, as mentioned in Section 1, our scheme has achieved the primary requirement for a reliable, high quality watermarking technique.

According to the DCT characteristics, $AC12$ is chosen as the BPN output vector. However, this is not the only choice for an output vector. As the index of the AC coefficient increases, the robustness of the watermark decreases, and vice versa. For reliable robustness, the input vector using $AC1$ to $AC9$ is recommended. The DC component is not used because its value domain differs from the AC components.

Because we allow partially overlapped DCT blocks, the retrieved watermark from a non-altered watermarked image has little noise. We can also use non-overlapping DCT blocks, but, it will reduce the security at the same time and be vulnerable to attack by collusion. The advantage of allowing partially overlapped DCT blocks is that there is no way to determine the exact position of the DCT blocks except by the owner of the property. Therefore, our scheme can hold against the collusion attack.

According to the experimental results, the watermark can be retrieved from a document larger than 30 dB regardless which kind of process is used on the watermarked image. The main reason is that the DCT coefficients on the left-upper corner suffered little change when the PSNR is larger than 30 dB. Therefore, when the watermarked images were modified using a geometrical process, such as rotation, scaling, etc, we could retrieve the watermarks successfully using inverse modification. Note that if the high frequency variation is larger than the low frequency, e.g. sharpening attack, the BCR is better even though its' PSNR is lower. That is why the experimental results show that some PSNR values are lower but the BCR values are higher.

In BPN, the memory requirement for the weights w_{ij} is equal to the number of links, i.e. $U_I \times U_H \times U_O$. Here U_I, U_H, and U_O denote the number of units in the input layer, hidden layer, and output layer, respectively. Note that the memory

training set requirement size does not matter in the BPN. In our proposed scheme, the memory requirement equals $9 \times 4 \times 1 \times 2$ (bytes)=72 bytes in total. Here each weight requires 2 bytes. In other words, our scheme is practical in terms of memory requirement.

We did not address the problem of ownership deadlock [3]. This problem is common with digital signatures in cryptography. A trusted third party is required to authenticate each user's identity. An extra protocol is required to solve this problem. It is not necessary to modify the primary approach greatly to acquire this solution. Therefore, the solution to the ownership deadlock problem in digital watermarking can employ another protocol based on our scheme.

7 CONCLUSIONS

A new approach to digital watermarking was proposed in this paper. The neural network and DCT were used in the proposed scheme. The BPN was used to improve the robustness of the proposed scheme. Our method can achieve the following two goals: 1. secure locations for an embedded watermark in the image, using one-way hash function to achieve this goal, and 2. only the legal user is able to retrieve the embedded watermark from an altered image. We used BPN and DCT, in which blocks are allowed to partially overlap, to achieve this goal. Since our scheme can satisfy all of the requirements discussed in Section 1, we have thus demonstrated that a neural network can be effectively applied to digital watermarking for the purpose of copyright protection. Today, there are no schemes using the strategies rather than neural networks can achieve a wholly robust and secure watermark. Using neural networks' strategy to solve the problem is a brand new research area.

Acknowledgements

The authors wish to thank many anonymous referees for their suggestions to improve this paper. Part of this research was supported by the National Science Council, Taiwan, R.O.C., under contract no. NSC89-2213-E-324-025.

REFERENCES

[1] R. J. Clarke, *Digital Compression of Still Images and Video.* Academic Press, 1995.

[2] 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, 1997.

[3] S. Craver, N. Memon, B.-L. Yeo, and M. Yeung, "Can invisible watermarks resolve rightful ownership?," *Proc. SPIE Stroage and Retrieval for Still Image and Video Databases V*, vol. SPIE 3022, pp. 310–321, 1997.

[4] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Addison Wesley, 1992.

[5] M. Kutter, F. Jordan, and F. Bossen, "Digital watermarking of color images using amplitude modulation," *Journal of Electronic Imaging*, vol. 7, no. 2, pp. 326–332, 1998.

[6] N. Memon and P. W. Wong, "Protecting digital media content," *Communications of The ACM*, vol. 41, no. 7, pp. 34–43, 1998.

[7] National Institute of Standards and Technology, "Digital signature standard," *NIST FIPS Pub 180-16*, Apr. 1995.

[8] M. O. Rabin, "Digital signatures," *Foundations of Secure Communication*, pp. 155–168, 1978.

[9] V. B. Rao and H. V. Rao, *C++ Neural Networks and Fuzzy Logic*. MIS, 2nd edition, 1995.

[10] R. L. Rivest, "The md5 message digest algorithm," *RFC 1321*, Apr. 1992.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representation by error propagation," *Parallel Distributed Processing*, vol. 1, pp. 318–362, 1986.

[12] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proceedings of IEEE*, vol. 86, no. 6, 1998.

[13] H.-J. M. Wang, P.-C. Su, and C.-C. J. Kuo, "Wavelet-based digital image watermarking," *OPTICS EXPRESS*, vol. 3, no. 12, pp. 491–496, 1998.

[14] Minerva M. Yeung, "Digital watermarking," *Communications of the ACM*, vol. 41, pp. 30–33, jul 1998.

[15] A. Zell, N. Mache, G. Mamier, M. Vogt, and S. Doering, "Stuttgart Neural Network Simulator (SNNS)," *see http://www.informatik.uni-stuttgart.de/ipvr/bv/projekte/snns/snns.html.*

BIOGRAPHIES

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China, in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng

Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field "Electronic Engineer" in 1988. He also passed the National Telecommunication Special Examination in field "Information Engineering", qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a project leader for research in computer security at TL in July 1990. He obtained the 1997, 1998, 1999, 2000 Distinguished Research Awards of the National Science Council of the Republic of China. He is currently a professor and chairman of the Department of Information Management, Chaoyang University of Technology, Taiwan, ROC. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include database and data security, cryptography, image compression, and mobile communications.

Chin-Chen Chang received his BS degree in applied mathematics in 1977 and the MS degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. During the academic years of 1980-1983, he was on the faculty of the Department of Computer Engineering at the National Chiao Tung University. From 1983-1989, he was on the faculty of the Institute of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan. From August 1989 to July 1992, he was the head of, and a professor in, the Institute of Computer Science and Information Engineering at the National Chung Cheng University, Chiayi, Taiwan. From August 1992 to July 1995, he was the dean of the college of Engineering at the same university. From August 1995 to October 1997, he was the provost at the National Chung Cheng University. From September 1996 to Octber 1997, Dr. Chang was the Acting President at the

National Chung Cheng University. Since July 1998, he has been the director of the Ministry of Education of the R.O.C..In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.

Dr. Chang was the associate editor of Computer Quarterly, the Journal of Computers, the Journal of the Chinese Institute of Engineers, the Journal of Electrical Engineering, the International Journal on Policy and Information, the Journal of Information and Management Science, the Journal of Information Science and Engineering, and the regional editor of Information Sciences Applications of U.S.A. and is editor-in-chief of the Journal of Information and Education.

He was elected an Outstanding Youth of the Republic of China in 1984. In the same year, he was elected an Outstanding Talent in Information Science of the Republic of China. He obtained the 1986-1987,1988-1989, 1990-1991, 1992-1994, 1995-1996 Distinguished Research Awards from the National Science Council of the Republic of China. He obtained from the Chung-Shan Academic Foundation of the Republic of China the 1987 and 1997 Chung-Shan Academic Publication Awards. He was the winner of the 1990, 1991, 1992, and 1997 AceR Dragon Thesis Award for Outstanding MS Thesis Supervision and the 1990, 1997, and 1998 AceR Dragon Dissertation Award for Outstanding Ph.D Dissertation Study Supervision. He was the winner of the Best Paper Award at the Second International Conference on CISNA sponsored by the British Council. He was also the winner of the 1992 Outstanding Teaching Materials Award of the Ministry of Education of the Republic of China.

Dr. Chang has published more than 190 papers in well-known journals, more than 80 papers in international conference proceedings, and 12 books in the field of database design, data structures, information security and cryptography, in Chinese.

Dr. Chang is a fellow of the IEEE, a research fellow of National Science Council of R.O.C., and a member of the Chinese Language Computer Society, the Chinese Institute of Engineers of the Republic of China, the International Association for Cryptologic Research, the Computer Society of the Republic of China, and the Phi Tau Phi Honorary Society of the Republic of China. Dr. Chang was the chair and is the honorary chair of the executive committee of the Chinese Cryptography and Information Security Association of the Republic of China.

Kuo-Feng Hwang received the B.S. in Construction Engineering from National Lien-Ho College of Technology and Commerce, Taiwan, Republic of China, in 1991; the M.S. in Information Management from Chaoyang University of Technology, Taiwan, in 1999; He is currently pursuing his Ph.D. degree in Department of Computer Science and Information Engineering, at Nation Chung Cheng University, Taiwan. His research interests include cryptography, image processing, and information management.