A blind digital image watermarking method using interval wavelet decomposition

Teruya Minamoto and Kentaro Aoki
Department of Information Science, Saga University, JAPAN
minamoto@is.saga-u.ac.jp

Abstract

In this paper, we present a new blind digital image watermarking method. We introduce interval wavelet decomposition, which is a combination of a discrete wavelet transform and interval arithmetic, and we examine its properties. According to our experimental results, this combination is a good way to produce a kind of redundancy from the original image and to develop new watermarking methods. Thanks to this property, we can obtain specific frequency components where the watermark is embedded. We describe the procedure of our method in detail and its relations with the human visual system (HVS). We also give some experimental results demonstrating that our method gives watermarked images of better quality and is robust against attacks such as clipping, marking, and JPEG and JPEG2000 compressions.

Keywords: *interval arithmetic, digital watermarking, wavelet decomposition*

1. Introduction

Digital watermarking refers to specific information hiding techniques whose purpose is to embed secret information inside multimedia content, like images, video, or audio data. The watermark is typically added to a specific field in the original content to protect its copyright. In this paper we treat only still digital images, so we focus our discussions on images in the following.

Many digital watermarking methods have been proposed over the last decade [2]. According to whether or not the original signal is during the watermark detection process, digital watermarking methods can also be roughly categorized into two types: non-blind and blind. Non-blind methods require the original image at the detection end, whereas blind methods do not. Blind methods are more useful than non-blind ones because the original image may not be available in actual scenarios.

If we understand correctly, almost all existing watermarking methods utilize redundancy in the original image to embed the digital watermark. Therefore, it is important to find or produce the parts of the original image that are redundant for human beings based on the characteristics of the human visual system (HVS). In short, we need only obtain the redundant parts of the original image to embed the watermark.

Interval arithmetic is mainly used in fields where rigorous mathematics is associated with scientific computing, including computer graphics and computer-aided design [5,7]. Interval arithmetic has the property that the width of an initial interval that includes the original data expands in proportion to the number of arithmetic computations. This property is sometimes called *interval expansion*, and an example of it is described in the next section. Interval expansion is, so to speak, the redundancy produced by the original data and interval

arithmetic. This implies that there is a possibility of applying interval arithmetic to digital watermarking.

Based on this idea, we previously proposed a new non-blind digital image watermarking method [6], where we used discrete wavelet transforms and interval arithmetic. According to our experiences, it seems that the discrete wavelet transform is suited for interval arithmetic to obtain the redundant parts of the original image. In this paper we call such a combination of a discrete wavelet transform and interval arithmetic *interval wavelet decomposition*.

Up to now, we have not investigated the properties of the interval wavelet decomposition, nor have we considered the blind type in Ref. [6]. Therefore, our goal in this paper is to examine the properties of the interval wavelet decomposition and its relations with the HVS, and to propose a new blind digital watermarking method using interval wavelet decomposition.

This paper is organized as follows: in Section 2, we briefly describe the basics of interval arithmetic. In Section 3, we introduce the interval wavelet decomposition, and in Section 4, we propose a blind digital watermarking method. Then, in Section 5, we describe how to choose the parameters appearing in our experiments, and the relationship between our method and the HVS. We show experimental results in Section 6, and conclude this paper in Section 7.

2. Interval arithmetic

Since many readers are probably not familiar with interval arithmetic, here we summarize its basic properties, described in Ref.[7].

An interval is a set of the form

$$A = [a_1, a_2] = \{t \mid a_1 \le t \le a_2, a_1, a_2 \in R\},$$

where R denotes the set of real numbers. We denote the lower and upper bounds of an interval A by $\inf(A) = a_1$ and $\sup(A) = a_2$, and the width of any non-empty interval A is defined by $w(A) = a_2 - a_1$.

The four basic operations, namely, addition (+), subtraction ($-$), multiplication ($*$), and division (/) on two intervals $A = [a_1, a_2]$ and $B = [b_1, b_2]$ are a set of the form

$$A \circ B = \{c = a \circ b \mid a \in A, b \in B\}, \quad \circ \in \{+, -, *, /\},$$

and we can explicitly calculate them as

$$\begin{aligned} A + B &= [a_1 + b_1, a_2 + b_2] \\ A - B &= [a_1 - b_2, a_2 - b_1] \\ A * B &= [\min\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\}, \max\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\}] \\ A / B &= [a_1, a_2] * [1/b_2, 1/b_1], \quad 0 \notin B. \end{aligned} \tag{1}$$

For interval vectors and matrices whose elements consist of intervals, these operations are executed at each element.

From the basic operations (1), the width of the interval expands in proportion to the number of computations in general. For example, let $A = [-1, 2]$, $B = [1, 4]$, and $C = [-2, 1]$; then, $A + B = [0, 6]$, $A + B - C = [-1, 8]$. The widths of A, B, and C are 3, whereas the widths of $A + B$ and $A + B - C$ are 6 and 9, respectively. This phenomenon is sometimes called *interval expansion*. Many researchers who study mathematically rigorous computer-assisted proofs [5] are often annoyed by interval expansion. This is a typical demerit of interval arithmetic. On the contrary, however, we

regard interval expansion as a useful tool to produce the redundant part from the original image.

3. Interval wavelet decomposition

Let us denote the original image by S . It is well known from Refs. [3,4] that the usual Daubechies wavelet decomposition formulae for images with support width $2N-1$ are given by:

$$\begin{aligned} C(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} p_m p_n S(m+2i, n+2j) \\ D(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} p_m q_n S(m+2i, n+2j) \\ E(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} q_m p_n S(m+2i, n+2j) \\ F(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} q_m q_n S(m+2i, n+2j). \end{aligned} \tag{2}$$

Here p_n and q_n are real parameters which have the relation $q_n = (-1)^n p_{2N-1-n}$, and the indices i and j are the locations in the horizontal and vertical directions, respectively. $C(i,j)$, $D(i,j)$, $E(i,j)$ and $F(i,j)$ indicate low-frequency components and high-frequency components in the vertical, horizontal, and diagonal directions, respectively.

To accelerate the interval expansion, we define the interval wavelet decomposition based on (2) by

$$\begin{aligned} IC(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} [1-\Delta_{m,n}, 1+\Delta_{m,n}] p_m p_n S(m+2i, n+2j) \\ ID(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} [1-\Delta_{m,n}, 1+\Delta_{m,n}] p_m q_n S(m+2i, n+2j) \\ IE(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} [1-\Delta_{m,n}, 1+\Delta_{m,n}] q_m p_n S(m+2i, n+2j) \\ IF(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} [1-\Delta_{m,n}, 1+\Delta_{m,n}] q_m q_n S(m+2i, n+2j), \end{aligned} \tag{3}$$

where $\Delta_{m,n}$ are positive real numbers.

4. Watermarking Algorithm

We assume that the binary-valued watermark W consists of -1 and 1 . To develop a new blind method, we modify one of the best-known block-based blind watermark schemes described in Chapter 3 in Ref. [2] and apply it to the non-blind method in Ref. [6]. In our method, we use a sliding window instead of a block, because the sliding-window-based scheme is superior to the block-based one in many cases according to our experience. Then, the embedding procedure is as follows:

Choose one of the four interval components (IC , ID , IE , IF) and decide the position at which the watermark is to be embedded. For simplicity, in this section we choose IF and decide $F' = \sup(IF)$.

Replace other components with floating point ones. In this case, replace IC , ID , and IE with C , D , and E , respectively.

Compute the following sequence using a sliding window of $2k+1$:

$$\overline{F}(i,j) = \mathrm{sgn}(F'(i,j)) \cdot \frac{1}{(2k+1)} \sum_{l=-k}^{k} | F'(i+l,j) |, , \tag{4}$$

where k is a fixed natural number and

$$\mathrm{sgn}(x) = \begin{cases} 1 & (x > 0) \\ 0 & (x = 0) \\ -1 & (x < 0). \end{cases}$$

Embed the watermark W by computing

$$F_w(i,j) = \overline{F}(i,j)\{1 + \alpha W(i,j)\}. \tag{5}$$

Here $0 < \alpha < 1$ is a given hiding factor which adjusts the robustness.

Reconstruct the image using $C \square E$, F_w and the inverse wavelet transform; then the watermarked image S_w is obtained.

In the extraction procedure, we do not need the original image, and we just compute (6) below.

Decompose S_w into four components C_{S_w} , D_{S_w} , E_{S_w} , and F_{S_w} by the wavelet transform.

Compute

$$W_e(i,j) = \mathrm{sgn}(| F_{S_w}(i,j) | - | \overline{F_{S_w}}(i,j) |) \tag{6}$$

to extract the watermark. Here we note that we must consider the absolute values of F_{S_w} and $\overline{F_{S_w}}$ to avoid changing the sign of $\alpha W \approx (F_{S_w} - \overline{F_{S_w}}) / \overline{F_{S_w}}$ corresponding to $(F_w - \overline{F}) / \overline{F}$ in (5). For example, assuming that $W = 1$, $(F_w - \overline{F}) / \overline{F} < 0$ when $\overline{F} < 0$, whereas $(F_w - \overline{F}) / \overline{F} > 0$ when $\overline{F} > 0$.

5. Considerations in the proposed algorithm

In this section, we consider how to choose the parameters described in the previous section and the relationship between the redundancy and the HVS.

5.1. Parameter selection

Digital watermarking methods involve a trade-off between robustness and quality of the watermarked image. $\Delta_{m,n}$ in (3) represents this relation. In fact, the higher the values of $\Delta_{m,n}$, the higher the redundancy of the original image. That is, the robustness is proportional to $\Delta_{m,n}$, but the quality of the watermarked image is inversely proportional to $\Delta_{m,n}$, and vice versa. Since the hiding parameter α in (5) also determines the robustness of our watermarking method, it is important to find suitable values of $\Delta_{m,n}$ and α . Furthermore, the natural number k in (4) will also affect the robustness.

We decide the parameters so as not to considerably decrease the value of the peak signal to noise ratio (PSNR), expressed in decibels, which is computed by

$$PSNR = 20\log_{10}\left(\frac{255}{\sqrt{\frac{1}{N_x N_y}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}(S_w(i,j)-S(i,j))^2}}\right),, \quad (7)$$

where N_x and N_y are the sizes of the image in the horizontal and vertical directions, respectively.

In our experiment, we require that the PSNR exceeds 30. According to our preliminary experiments, we must set $\Delta_{m,n} \leq 0.008$ to fulfill this requirement. On the other hand, k in (4) does not affect the PSNR too much, but it affects the quality of the extracted watermark (Fig. 1).

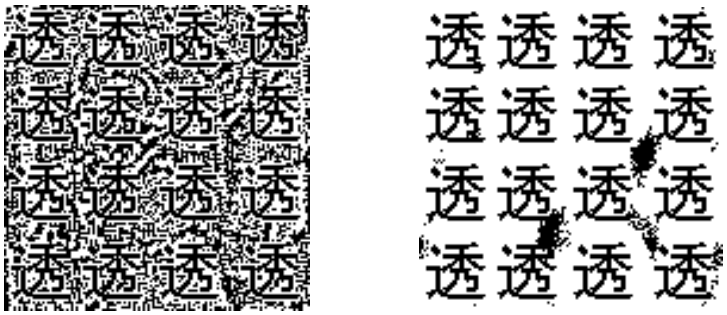


Figure 1. Extracted watermarks when $k = 1$ (left) and $k = 10$ (right).

Moreover, α should be chosen as large as possible in order to maintain the robustness. Table 1 shows some of our preliminary experimental results.

Table 1. PSNRs when α =0.7 (top) and $\alpha = 0.9$ (bottom).

	$\Delta_{m,n} = 0.006$	$\Delta_{m,n} = 0.008$	$\Delta_{m,n} = 0.01$
$k = 1$	32.8667	30.5639	28.7266
$k = 2$	32.8114	30.5450	28.7212
$k = 3$	32.7917	30.5459	28.7321
$k = 4$	32.7916	30.5535	28.7452
$k = 5$	32.7924	30.5637	28.7580
$k = 6$	32.7948	30.5731	28.7692
$k = 7$	32.7955	30.5772	28.7781
$k = 8$	32.7970	30.5819	28.7843
$k = 9$	32.7944	30.5852	28.7899
$k = 10$	32.7948	30.5867	28.7932

	$\Delta_{m,n} = 0.006$	$\Delta_{m,n} = 0.008$	$\Delta_{m,n} = 0.01$
$k = 1$	32.2870	30.0218	28.1984
$k = 2$	32.2563	30.0127	28.2037
$k = 3$	32.2488	30.0211	28.2192
$k = 4$	32.2537	30.0356	28.2372
$k = 5$	32.2601	30.0488	28.2520

$k=6$	32.2673	30.0614	28.2673
$k=7$	32.2711	30.0666	28.2760
$k=8$	32.2716	30.0726	28.2845
$k=9$	32.2729	30.0770	28.2882
$k=10$	32.2724	30.0804	28.2942

As a result, we choose $\Delta_{m,n}=0.008$ in (3), $k=10$ in (4), and $\alpha=0.9$ in (5). Of course, we can also set different values of $\Delta_{m,n}$ at each pixel, but we use the same constant $\Delta_{m,n}$ at all pixels in our numerical experiments for simplicity.

5.2. Relationship between our method and HVS

In this subsection, we investigate the relationship between the redundancy produced by the interval wavelet decomposition and the HVS. To this end, we prepare the following theorem.

Theorem 1. The interval high-frequency components ID, IE, and IF contain a low-frequency component C. More precisely, the following relations hold:

$$\begin{aligned} ID(i,j) &= D(i,j)+\theta_D\big(IC(i,j)-C(i,j)\big)+R_D(i,j),\\ IE(i,j) &= E(i,j)+\theta_E\big(IC(i,j)-C(i,j)\big)+R_E(i,j), \\ IF(i,j) &= F(i,j)+\theta_F\big(IC(i,j)-C(i,j)\big)+R_F(i,j). \end{aligned} \tag{8}$$

Here θ_D, θ_E, and θ_F are certain constants depending on p_n in (2), and $R_D(i,j)$, $R_E(i,j)$ and $R_F(i,j)$ are certain components whose values are expected to be small.

(Proof)

We only prove the first equality in (8), because the other equalities can be shown in the same manner. In this proof, we abbreviate $S(i,j)$ and $[-\Delta_{m,n},\Delta_{m,n}]$ to $S_{i,j}$ and $I(\Delta_{m,n})$, respectively, to save space.

From (3), we first note that

$$\begin{aligned} ID(i,j) &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}[1-\Delta_{m,n},1+\Delta_{m,n}]p_m q_n S_{m+2i,n+2j}\\ &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}p_m q_n S_{m+2i,n+2j}+\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}I(\Delta_{m,n})p_m q_n S_{m+2i,n+2j}\\ &= D(i,j)+\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}I(\Delta_{m,n})p_m q_n S_{m+2i,n+2j}. \end{aligned}$$

Using the relation $q_n=(-1)^n p_{2N-1-n}$ appearing in Ref. [3], the second term of the right-hand side is denoted by

$$\begin{aligned} \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}I(\Delta_{m,n})p_m q_n S_{m+2i,n+2j} &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}I(\Delta_{m,n})p_m p_{2N-1-n} S_{m+2i,n+2j}\\ &= \sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}I(\Delta_{m,n})\frac{p_{2N-1-n}}{p_n}p_m p_n S_{m+2i,n+2j}. \end{aligned}$$

Next, setting $c_n=p_{2N-1-n}/p_n$ and $\theta_D=\min_n|c_n|$, the last term above is represented by

$$\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} I(\Delta_{m,n})(c_n+\theta_D-\theta_D)p_m p_n S_{m+2i,n+2j}$$
$$=\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}\left(I(\Delta_{m,n})\theta_D p_m p_n S_{m+2i,n+2j}+I(\Delta_{m,n})(c_n-\theta_D)p_m p_n S_{m+2i,n+2j}\right)$$
$$=\theta_D\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} I(\Delta_{m,n})p_m p_n S_{m+2i,n+2j}+R_D(i,j),$$

where

$$R_D(i,j)=\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} I(\Delta_{m,n})(c_n-\theta_D)p_m p_n S_{m+2i,n+2j}.$$

Then, we obtain

$$\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} I(\Delta_{m,n})p_m p_n S_{m+2i,n+2j}$$
$$=\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}(I(\Delta_{m,n})+1-1)p_m p_n S_{m+2i,n+2j}$$
$$=\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}([1-\Delta_{m,n},1+\Delta_{m,n}]-1)p_m p_n S_{m+2i,n+2j}$$
$$=\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1}[1-\Delta_{m,n},1+\Delta_{m,n}]p_m p_n S_{m+2i,n+2j}-\sum_{m=0}^{2N-1}\sum_{n=0}^{2N-1} p_m p_n S_{m+2i,n+2j}.$$

Therefore, we get

$$ID(i,j)=D(i,j)+\theta_D(IC(i,j)-C(i,j))+R_D(i,j).$$

Barni *et al.* proposed a weighting function using the masking characteristics of the HVS [1]. To compare their method with our method, we introduce the following notation:

$$Y_l^{\theta}(i,j)=\begin{cases} D_l(i,j), & \text{if } \theta=0\\ F_l(i,j), & \text{if } \theta=1\\ E_l(i,j), & \text{if } \theta=2\\ C_l(i,j), & \text{if } \theta=3, \end{cases}$$

where the numerical subscript $l\,(l=0,1,2,3)$ stands for the resolution level, and C_0, D_0, E_0, and F_0 are identical to C, D, E, and F, respectively, in Section 3. That is, the value $l+1$ represents the number of times that the discrete wavelet decomposition is carried out.

They embed the watermark $w^{\theta}(i,j)$ according to the rule

$$\tilde{Y}_0^{\theta}(i,j)=Y_0^{\theta}(i,j)+\beta q^{\theta}(i,j)w^{\theta}(i,j) \tag{9}$$

using the weighting function

$$q^{\theta}(i,j)=\Theta(\theta)\Lambda(i,j)\Xi(i,j)^{0.2}. \tag{10}$$

We note that their rule (9) corresponds to

$$F_w(i,j)=\overline{F}(i,j)+\alpha W(i,j)\overline{F}(i,j) \tag{11}$$

in our method by the relation (5).

The first term in (10),

$$\Theta(\theta)=\begin{cases}\sqrt{2}, & \text{if } \theta=1\\ 1, & \text{otherwise}\end{cases}, \tag{12}$$

corresponds to the fact that the eye is less sensitive to noise in high-resolution bands. Since Y_l^1 (that is, F_l) is a high-resolution band in general, they set Y_l^1 larger than the others. This operation corresponds to our embedding the watermark into F.

The second term is computed in the following way:

$$\Lambda(i,j)=1+L(i,j) \tag{13}$$

with

$$L(i,j)=\frac{1}{256}C_3\left(1+\left\lfloor\frac{i}{2^3}\right\rfloor,1+\left\lfloor\frac{j}{2^3}\right\rfloor\right), \tag{14}$$

where $L(i,j)$ is redefined as

$$L'(i,j)=\begin{cases}1-L(i,j), & \text{if } L(i,j)<0.5\\ L(i,j), & \text{otherwise}\end{cases}. \tag{15}$$

Since the eye is less sensitive to noise in those areas of the image where the brightness is high or low, the second term takes into account the local brightness based on the graylevel values of the low-pass version of the image. This term corresponds to $\overline{F}$ in our method, because $\overline{F}$ depends on IF, and IF contains the low-frequency component by Theorem 1.

The third term

$$\begin{aligned}\Xi(i,j)=&\sum_{k=0}^{3}\frac{1}{16^k}\sum_{\theta=0}^{2}\sum_{x=0}^{1}\sum_{y=0}^{1}[Y_k^\theta(y+\frac{i}{2^k},x+\frac{j}{2^k})]^2\\ &\cdot\mathrm{Var}\{Y_3^3(1+y+\frac{i}{2^3},1+x+\frac{j}{2^3})\}_{x=0,1\ y=0,1}\end{aligned} \tag{16}$$

gives a measure of texture activity in the neighborhood of the pixel. This term is formed by the product of two contributions: the first is the local mean square values of Y_k^θ in all components, whereas the second is the local variance of the low-frequency component Y_3^3. Thanks to these contributions, they can take into account the fact that the eye is less sensitive to noise in highly textured areas, but, among these, is more sensitive near the edge. The third term also corresponds to $\overline{F}$ in our method, because $\overline{F}$ is produced by the sliding window which takes into account the neighborhood of the pixel, and $\overline{F}$ contains both high- and low-frequency factors.

From these discussions, we may conclude that our method takes the HVS into consideration. In particular, the high-frequency component $\overline{F}$ has information about the low-frequency component thanks to the interval wavelet decomposition, and the mathematical expressions are simpler and more sophisticated than the rule proposed by Barni *et al*. This is a newly discovered merit of interval arithmetic.

5.3. Our algorithm with a pseudorandom binary sequence

Since we embed the watermark into one of the high-frequency components according to (5), there is a possibility of seeing the watermark when the watermarked image is transformed by the same wavelet decomposition that we use to embed the watermark. To eliminate this

problem, we adopt one of the easiest known solutions in our algorithm; that is, we multiply the watermark by a pseudorandom binary sequence according to the rule

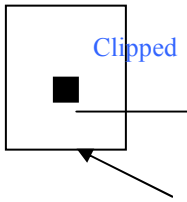
$$F_w(i,j) = \overline{F}(i,j)\{1 + \alpha W(i,j)x(i,j)\} \tag{17}$$

instead of (5) in the embedding procedure, where $x(i,j)$ is a pseudorandom binary sequence. On the other hand, we just multiply $x(i,j)$ by F_{S_w} to extract the watermark in the extraction procedure.

Nevertheless, because the pseudorandom binary sequence is embedded into the whole original image, the clipped image contains part of this sequence. Therefore, we need another technique when we extract the watermark from a clipped image. To extract the watermark from a clipped image, we decompose the original image and the clipped image by the wavelet decomposition. Let $C(i,j)$ and $CC(i,j)$ be the low-frequency components of the original image and the clipped image, respectively. At the position (ox,oy), we compute the mean square error

$$\delta(ox,oy) = \sum_{x=0}^{m-1}\sum_{y=0}^{m-1}(C(ox+x,oy+y) - CC(x,y))^2$$

and find the minimum point (mx,my). After that, we extract the pseudorandom binary sequence corresponding to the clipped image based on (mx,my) and carry out the extraction procedure using this pseudorandom binary sequence. In this case, we need the original image in the extraction procedure. Although the technique is very simple, this is one disadvantage of the method.



6. Experimental results

To evaluate the performance of the proposed method, we adopted the 256-grayscale Lenna image of size 256×256 pixels and a binary watermark of size 128×128 pixels, as shown in Fig. 2. We implemented our method using INTLAB [8], which is a MATLAB toolbox that supports interval arithmetic, and used the PSNR to evaluate the quality of the watermarked image.

In this experiment, we set $\Delta_{m,n} = 0.008$ and $N = 3$ in (3), $k = 10$ in (4), and $\alpha = 0.9$ in (5), and chose the component F in Step 1 of the embedding procedure described in Section 4.

Fig. 3 shows the watermarked image with a PSNR of 30.08 obtained by the proposed blind method and the watermark extracted from the watermarked image without any attack. Figs. 4–7 illustrate the watermarks extracted from the watermarked images under attacks such as marking, clipping, and JPEG and JPEG2000 compressions. The extracted images are degraded, but we are able to identify the existence of the watermark at a single glance in Figs. 4, 5 and 7, but barely in Fig. 6.

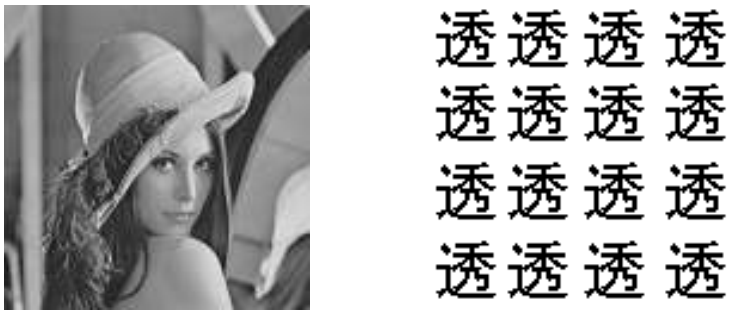


Figure 2. Original image and watermark.

Figure 3. Watermarked image and extracted watermark without any attack

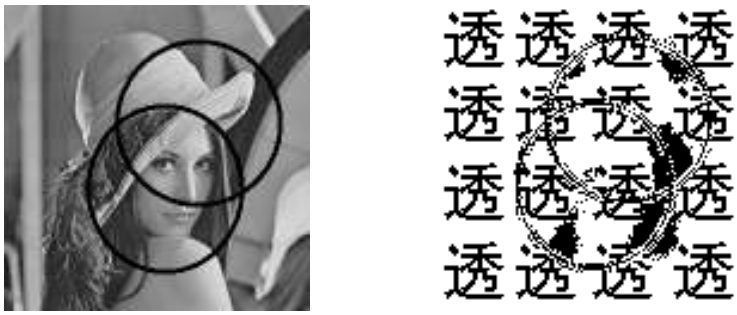


Figure 4. Watermarked image with marked areas and extracted watermark.

Figure 5. 188 × 210 fragment of watermarked image and extracted watermark.

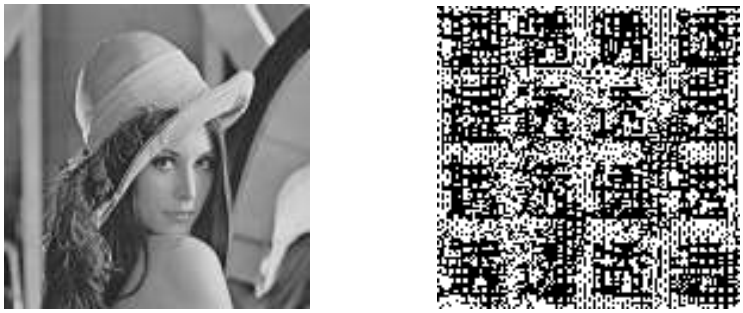


Figure 6. Watermarked JPEG image compressed with file size ratio 32% and extracted watermark.

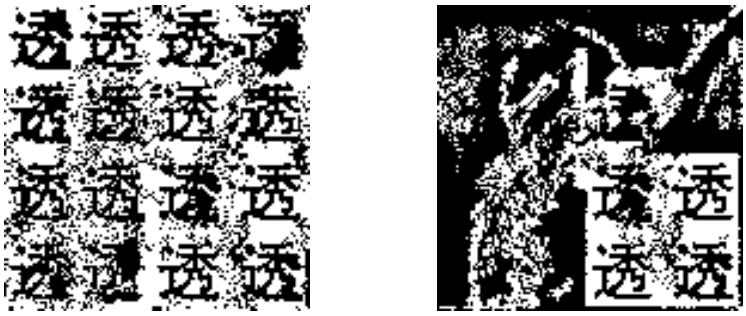


Figure 7. Watermarks extracted from watermarked JPEG2000 images compressed with file size ratios 11.8% (left) and 7.8% (right).

We demonstrate the experimental results of our method with a pseudorandom binary sequence described in Section 5.3, shown below.

Figure 8. Watermarked image and extracted watermark without any attack

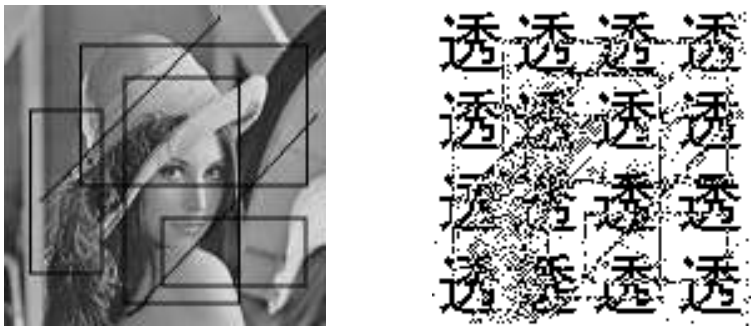


Figure 9. Watermarked image with marked areas and extracted watermark.

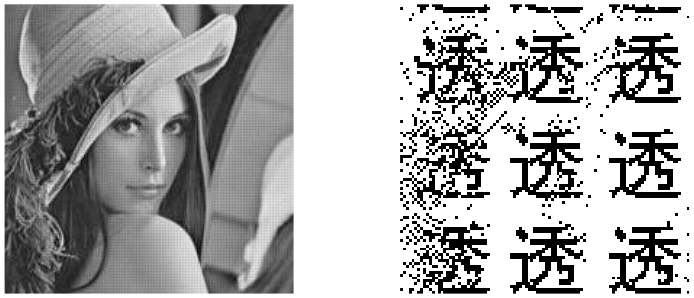


Figure 10. 200 × 200 fragment of watermarked image and extracted watermark.

Figure 11. Watermarks extracted from watermarked JPEG images compressed with file size ratios 55.8% (left) and 41.2% (right).

Figure 12. Watermarks extracted from watermarked JPEG2000 images compressed with file size ratios 32.3% (left) and 24.4% (right).

As shown in Figs. 11 and 12, the robustness of the watermarking with a pseudorandom binary sequence against image compression is inferior to the method without a pseudorandom binary sequence. Since there is a possibility that the pseudorandom binary sequence is corrupted in the image compression process, we think that these results are natural. In future, we will develop a new method that rigorously takes into account the features of image compression methods.

7. Conclusions

We proposed a blind digital image watermarking method. It seems that our method is relatively easy to implement compared with other methods based on the frequency domain [1,2]. To develop our method, we introduced an interval wavelet decomposition formula and investigated its properties. Using the interval wavelet decomposition formula, we realized that every high-frequency component contains the low-frequency component. Since the low-frequency component is the main part of the original image, these high-frequency components also contain it. This means that every high-frequency component has a certain redundancy. We also examined the relationship between this redundancy and the human visual system (HVS), and realized that our method takes the HVS into consideration automatically; that is to say, it does not require additional functions, such as function (10) and its related functions, to consider the HVS.

Experimental results demonstrate that our method gives better-quality watermarked images and has the robustness against some attacks such as clipping, marking, and JPEG and JPEG2000 compressions.

At present, our digital watermarking method may not have sufficient robustness in practice. However, this work gives a novel digital watermarking method, and besides, it may open up new possibilities for interval arithmetic. In this sense, we believe that our approach is very important in the fields of both digital watermarking and interval arithmetic.

References

[1] Barni, M., Bartolini, F. and Piva, A.: Improved wavelet-based watermarking through pixel-wise masking. IEEE Trans. Image Processing, Vol.10, No.5, 783-791 (2001)

[2] Cox, I. J., Miller, M. L. and Bloom, J. A.: Digital Watermarking. Morgan Kaufmann Publishers (2002).

[3] Daubechies, I.: Orthonormal bases of compactly supported wavelets. Comm. Pure Appl. Math. Vol.41, 909-996 (1988)

[4] Daubechies, I.: Ten Lectures on Wavelets. SIAM (1992)

[5] Minamoto, T. and Nakao, M. T.: Numerical method for verifying the existence and local uniqueness of a double turning point for a radially symmetric solution of the perturbed Gelfand equation. Journal of Computational and Applied Mathematics Vol.202/2, 177-185 (2007)

[6] Minamoto, T., Yoshihara, M. and Fujii, S.: A Digital Image Watermarking Method Using Interval Arithmetic. IEICE Trans. Fundamentals Vol.E90-A, No.12, 2949-2951 (2007)

[7] Moore, R. E., Kearfott, R. B. and Cloud, M. J.: Introduction to Interval Analysis. SIAM (2009)

[8] Rump, S. M.: INTLAB - Interval Laboratory.

Authors

Teruya Minamoto received his B.Ed. degree in mathematics education, in 1992, and his M.S. degree in mathematics, in 1994, both from Ehime University, Japan, and a Dr. Math. degree in 2000, from Kyushu University, Japan. He joined the Department of Informatics at Kyushu University as a research associate. In 2000, he joined the Department of Information Science, Saga University, where is now an associate professor. He received the Excellent Educator Award from IPSJ Japan in 2005. His current research interests are numerical verification methods, signal and image processing, and mathematics education in universities.

Kentaro Aoki was born in 1985. He received his B.S. degree in computer science in 2008 from Saga University, where he is researching image analysis as a graduate student.