

Study on Preventing Bit Error of Digital Watermark Using Orthogonal Transform

Jun Watanabe¹, Madoka Hasegawa² and Shigeo Kato²

¹Graduate School of Engineering, Utsunomiya University

²Faculty of Engineering, Utsunomiya University

Kato lab., Yoto 7-1-2, Utsunomiya, Tochigi 321-8585, Japan

Tel&Fax: +81-28-689-6256

e-mail : wataj@mclaren.is.utsunomiya-u.ac.jp

Abstract: Many digital watermarking methods have been proposed for copyright protection. Especially, methods to add a watermark in the frequency domain are often used. In those methods, only a few components of frequencies are changed according to watermark information, so that the pixel values of the image obtained by the inverse transform sometimes exceed the dynamic range and owing to the clipping process at this time watermark information is changed. This phenomenon even occurs without attacks like image processing. In this paper, we propose two methods to prevent loss of watermark information, applicable to such cases, it is said that, one is the method called "inversion of increase and decrease" and another is "conservation of amounts of changes." We can extract the watermark information correctly under the condition of no attacks by using these proposed methods.

1. Introduction

Recently, large amounts of digital contents can be accessed by computer networks and easily duplicated by personal computers. Therefore, protection of intellectual property rights of digital media information has become an important issue. Digital watermark technology is a solution of this task and various kinds of digital watermark techniques have been proposed [1].

Methods to add a watermark in the frequency domain are often used [2][3]. In those methods, only a few components of frequencies are changed according to watermark information, so that the pixel values of the image obtained by the inverse transform sometimes exceed the dynamic range and owing to the clipping process at this time watermark information is changed [4][5]. This phenomenon even occurs without attacks like image processing. In this paper, we propose two methods to prevent bit error of watermark information, applicable to such cases, it is said that, one is the method called "inversion of increase and decrease" and another is "conservation of amounts of changes." We can extract the watermark information correctly under the condition of no attacks by using these proposed methods.

2. Problem of Watermarking Method Using Orthogonal Transform

In this section, we explain a problem in case of embedding watermarks in the frequency domain. To simplify the discussion, we use 8×8 Hadamard transform as a watermarking method in the frequency domain.

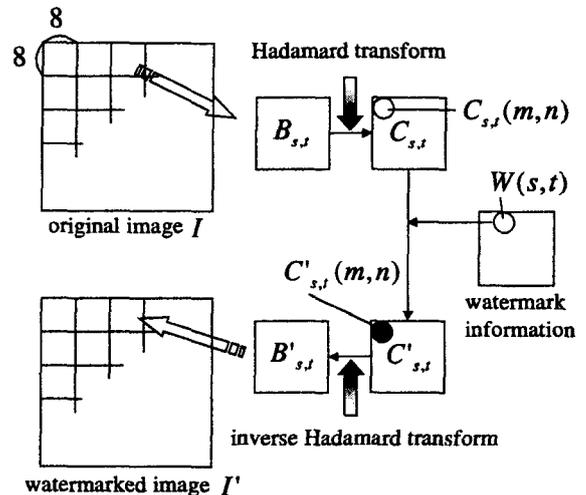


Fig.1 Watermarking method

2.1 Watermarking method using Hadamard transform

We explain watermarking method that embeds authentication information, signature or logo mark into an image. The embedding algorithm is shown in Fig.1, and the process is explained in the following steps.

1. Original image I ($X \times Y$ pixels, 8bit/pel) is divided into blocks of size 8×8 , which are denoted as $B_{s,t}$ ($s = 0, 1, \dots, S-1, t = 0, 1, \dots, T-1, S = X/8, T = Y/8$).
2. 8×8 Hadamard transform is applied to each block $B_{s,t}$.
3. Calculate index value $e = \text{int}(C_{s,t}(m,n) / Q)$, where Q is step size of quantization and $\text{int}(\cdot)$ is a function of rounding to the nearest integer.
4. Watermark information $W(s,t)$ is embedded by modifying the transform coefficient $C_{s,t}(m,n)$ as follows.
 - If $W(s,t)$ equals 0 and e is odd, or if $W(s,t)$ equals 1 and e is even, then

$$\text{if } eQ \leq C_{s,t}(m,n) \text{ then } C'_{s,t}(m,n) = (e+1)Q \quad (1.a)$$

$$\text{else } C'_{s,t}(m,n) = (e-1)Q \quad (1.b)$$
 - otherwise $C'_{s,t}(m,n) = eQ \quad (2)$
5. After above procedures, inverse Hadamard transform is applied to all blocks and a watermarked image I' is generated.

Whereas, in extracting watermarks, inverse procedure of embedding is performed. In other words, watermarked image I' is divided into blocks, and Hadamard transform is applied to each block. Then, the transform coefficients

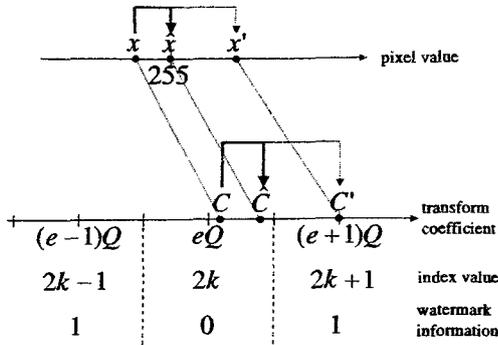


Fig.2 Bit error of watermark information

$C'_{s,t}(m, n)$ at the embedding positions are examined, and watermark information is obtained as follows.

$$\text{If } \text{int}(C'_{s,t}(m, n) / Q) \text{ is odd then } W(s, t) = 1 \quad (3)$$

$$\text{If } \text{int}(C'_{s,t}(m, n) / Q) \text{ is even then } W(s, t) = 0 \quad (4)$$

So that, embedded information can be extracted without the original image in this method.

2.2 Problem of watermarking method using orthogonal transform

In the above method, there are some cases that the embedded information can't be extracted even if they are not undergone any attacks like image processing. For example, consider that there are some pixels that values are equal or close to 255, as shown in Fig.2. In Fig.2, when the watermark information "1" is embedded by increasing the transform coefficient, the coefficient C is converted into C' . By this conversion, pixel x is equivalently converted into x' and exceeds the dynamic range. At this time the value x' is returned to 255 by clipping, and simultaneously, the coefficient C' is changed to \hat{C} . So that the embedded information is judged to "0." Thus the watermark information is changed in spite of being under no attacks. This is an example of phenomenon that occurs when watermark is embedded in the DC coefficient, but such phenomenon also occurs when watermark is embedded in the AC coefficient.

3. Embedding method preventing bit error

To solve above problem, we propose two methods called "inversion of increase and decrease" and "conservation of amounts of changes." We can extract the watermark information correctly under the condition of no attacks by using these proposed methods.

3.1 Method 1 — Inversion of increase and decrease

One of our proposed methods is "inversion of increase and decrease." This method is to inverse change of coefficient for embedding watermark to prevent exceeding dynamic range of pixel values. The concept of the method is shown in Fig.3. In Fig.3, transform coefficient is modified to $(e-1)Q$ to prevent changing of watermark information.

However, there are some cases that it is impossible to apply this method. For example, consider to embed watermark by modifying DC coefficient into a block in which there are both pixels having value of 255 or close on

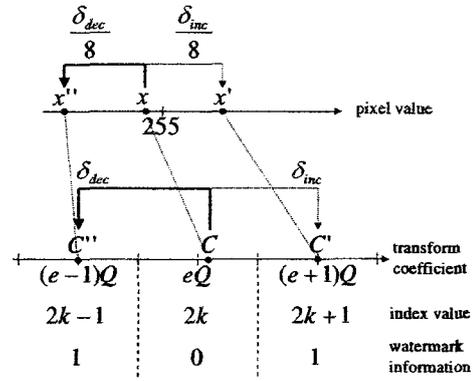


Fig.3 Inversion of increase and decrease

it and value of 0 or close on it. Change of pixel value is related to change of coefficient, so that when watermark is embedded by increasing coefficient, pixel values which are equal or close to 255 exceed the dynamic range. Similarly, when watermark is embedded by decreasing the coefficient, pixel values which are equal or close to 0 go out of the dynamic range. In such cases, it is impossible to apply "inversion of increase and decrease" method.

When transform coefficient is changed by δ_{inc} or δ_{dec} (δ_{inc} and δ_{dec} are the amount of increasing and decreasing in embedding watermark, respectively), the pixel value is changed by $\delta_{inc}/8$ or $\delta_{dec}/8$. Therefore, this method is applicable under the condition shown in equations (5.a) and (5.b).

$$\frac{\delta_{inc}}{8} \leq 255 - x_{max} \quad (5.a)$$

$$\frac{\delta_{dec}}{8} \leq x_{min} \quad (5.b)$$

where x_{max} and x_{min} are the maximum and minimum of pixel values in embedding block respectively.

When watermark is embedded in the AC component, note that relationship of changes between the pixel value and the transform coefficient depends on basis function of embedded component, i.e., pixels corresponding to the component "-1" of the basis function, which are denoted as x_- , increase (or decrease) when transform coefficient decreases (or increases). To simplify this complexity, these pixels are converted into \tilde{x}_- instead of x_- , where $\tilde{x}_- = 255 - x_-$. When \tilde{x}_- exceeds dynamic range, x_- also exceeds. Furthermore, change of \tilde{x}_- is related to change of the coefficient, i.e., when the transform coefficient increases (or decreases), \tilde{x}_- also increases (or decreases). So that, by handling \tilde{x}_- and x_+ (pixels corresponding to the component "+1" of basis function) as pixels \tilde{x} in the block, we can consider the condition applicable of this method in embedding in the AC component as if embedding in the DC component. Therefore, this method is applicable under the condition shown in equations (6.a) and (6.b).

$$\frac{\delta_{inc}}{8} \leq 255 - \tilde{x}_{max} \quad (6.a)$$

$$\frac{\delta_{dec}}{8} \leq \tilde{x}_{min} \quad (6.b)$$

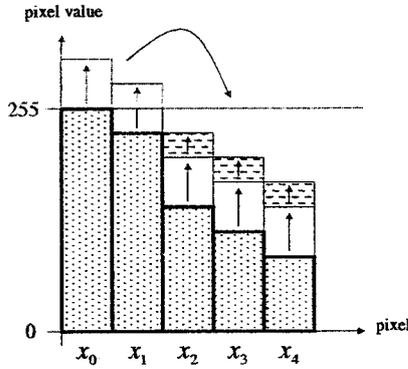


Fig.4 Conservation of amounts of changes

3.2 Method 2 – Conservation of amounts of changes –

Another proposal is “conservation of amounts of changes.” This method is to distribute pixel values lost by clipping into other pixels. The concept of this method is shown in Fig.4.

For applying “conservation of amounts of changes” method, it is necessary that *flow* (amount of pixel values exceeding or going down dynamic range) is smaller than *margin* (amounts of absorbable pixel values of *flow*). To determine applicable condition, consider to embed watermark into DC component by increasing the coefficient. Pixel values which exceed 255 are returned to 255. This causes decreasing of the coefficient value, and it is necessary to increase pixel values to preserve coefficient. *Flow*, the sum of pixel values exceeding dynamic range, is as follows.

$$flow = \sum_{i=0}^7 \sum_{j=0}^7 f \left(\left(x_{i,j} + \frac{\delta_{inc}}{8} \right) - 255 \right) \quad (7)$$

On the other hand, *margin* available is the sum of differences between 255 and pixel values which don't exceed dynamic range, so that

$$margin = \sum_{i=0}^7 \sum_{j=0}^7 f \left(255 - \left(x_{i,j} + \frac{\delta_{inc}}{8} \right) \right) \quad (8)$$

where $f(\alpha)$ is a following function.

$$f(\alpha) = \begin{cases} \alpha & \alpha \geq 0 \\ 0 & \alpha < 0 \end{cases} \quad (9)$$

We can apply “conservation of amounts of changes” method if

$$flow \leq margin \quad (10)$$

Similarly, consider to embed watermark into DC component by decreasing the coefficient. Pixel values which go down and below 0 are returned to 0 by clipping. This causes increasing of the coefficient value, and it is necessary to decrease pixel values to preserve coefficient, so that

$$flow = \sum_{i=0}^7 \sum_{j=0}^7 f \left(\frac{\delta_{dec}}{8} - x_{i,j} \right) \quad (11)$$

$$margin = \sum_{i=0}^7 \sum_{j=0}^7 f \left(x_{i,j} - \frac{\delta_{dec}}{8} \right) \quad (12)$$

When the watermark is embedded in the AC component, note that if distribution from pixels x_+ corresponding to the component “+1” of the basis function to pixels x_- corresponding to the component “-1” of the basis function

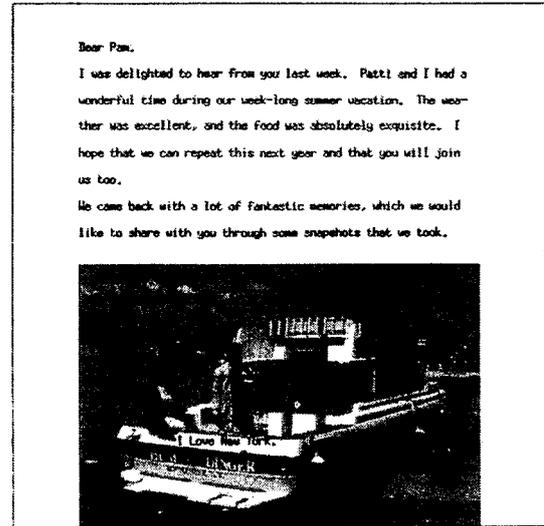


Fig.5 CMPND1 (512x512 pixels, 8bit/pel)



Fig.6 Watermark information (64x64 pixels, 1bit/pel)

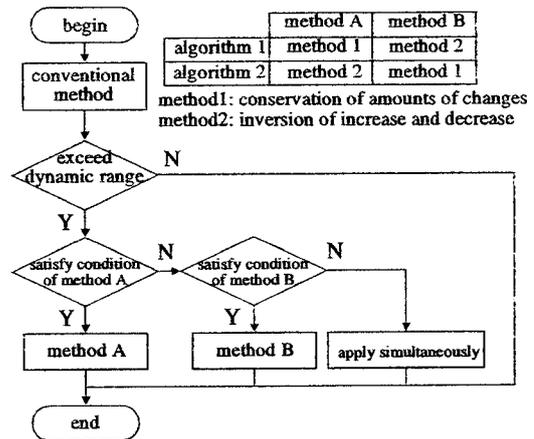


Fig.7 Watermark embedding algorithm

is performed, it is necessary to inverse the sign of changes and vice versa. To simplify this complexity, these pixels are converted into \tilde{x}_- instead of x_- , where $\tilde{x}_- = 255 - x_-$. After this conversion, we can distribute pixel values as if we distribute them when we embed watermark in the DC component.

3.3 Complementarities of “inversion of increase and decrease” method and “conservation of amounts of changes” method

There are some conditions to be applicable in both of our proposed methods. However, in most cases, we can apply either of them. There are exceptional cases of impossible to apply these methods independently, but we can prevent pixels from exceeding dynamic range by applying both of these methods simultaneously. To apply both of these methods simultaneously is possible in any situation, so that we can avoid bit error of watermark information anytime.

Table 1 Extract ratio of watermark information in non-attacking situation [%]

		step size of quantization (Q)			
		16	32	48	64
algorithm1	(0, 0) component	100.0	100.0	100.0	100.0
	(3, 3) component	100.0	100.0	100.0	100.0
	(7, 7) component	100.0	100.0	100.0	100.0
algorithm2	(0, 0) component	100.0	100.0	100.0	100.0
	(3, 3) component	100.0	100.0	100.0	100.0
	(7, 7) component	100.0	100.0	100.0	100.0
conventional method only	(0, 0) component	99.95	99.95	97.73	97.75
	(3, 3) component	100.0	99.98	99.98	100.0
	(7, 7) component	100.0	99.98	100.0	100.0

Table 2 Blocks with proposed methods

		step size of quantization (Q)			
		16	32	48	64
algorithm1	(0, 0) component	44/274/0	60/275/0	54/283/0	51/282/11
	(3, 3) component	2/1344/0	3/1384/0	3/1391/0	2/1405/0
	(7, 7) component	0/1358/0	2/1392/0	1/1396/0	1/1407/0
algorithm2	(0, 0) component	3/315/0	6/329/0	5/332/0	2/331/11
	(3, 3) component	0/1346/0	0/1387/0	0/1394/0	0/1407/0
	(7, 7) component	0/1358/0	0/1394/0	0/1397/0	0/1408/0

Table 3 SNR after watermark embedded [dB]

		step size of quantization (Q)			
		16	32	48	64
algorithm1	(0, 0) component	48.41	41.00	36.47	33.15
	(3, 3) component	43.28	37.27	33.98	31.46
	(7, 7) component	43.22	37.18	33.89	31.34
algorithm2	(0, 0) component	48.51	41.14	36.53	33.23
	(3, 3) component	43.28	37.27	33.99	31.47
	(7, 7) component	43.22	37.19	33.89	31.35
conventional method only	(0, 0) component	48.73	41.41	39.40	34.80
	(3, 3) component	47.28	41.31	37.93	35.37
	(7, 7) component	47.16	41.13	37.73	35.15

4. Experimental results and discussion

In this section, we present simulation results using the proposed two methods. Figure 5 shows the image used in the experiment, which is cropped at the point (0, 0) from CMPND1 (512×512 pixels, 8bit/pel) and Fig.6 shows the authentication image (64×64 pixels, 1bit/pel) to be embedded. We embed the watermark in positions of frequency $(m, n) = (0, 0), (3, 3),$ or $(7, 7)$.

Embedding algorithms for each block are shown in Fig.7. Two kinds of algorithms are used. In algorithm1, at first watermark is embedded by the conventional method. If pixel values exceed or go out of the dynamic range, check whether we can apply "inversion of increase and decrease" method and apply if we can. If we can't, check the applicable condition of "conservation of amounts of changes" method and apply if we can. If neither of method can be applied independently, both of the methods are applied simultaneously.

Table 1 shows the extraction ratio of embedded watermark by applying proposed methods and the

conventional method in non-attacking situation. When we embed watermark with proposed methods, we can perfectly extract correct information.

Table 2 shows the number of blocks applied proposed methods. Three numbers in table 2 mean number of blocks applied "inversion of increase and decrease" method / number of blocks applied "conservation of amounts of changes" method / number of blocks applied both of the methods simultaneously. Sometimes proposed methods are applied even if extract ratio of watermark with applying only conventional method achieves 100.0%. This means that there are some cases that pixel values exceed dynamic range in spite of correct extraction.

Table 3 shows the SNR after watermark embedded. It can be seen that the watermarked image with proposed method is degraded more than the watermarked image without proposed methods. This is because that watermark embedded with the conventional method causes clipping and change of the pixel value is lost. Therefore relative change of the pixel value by watermarking with proposed method is greater than that by the conventional one.

5. Conclusion

In this paper, a watermarking method that is possible to prevent bit error of information on non-attacking situation is proposed. We show effectiveness of our scheme by the simulation.

In our future works, we will study a watermarking method that has robustness against attacks (cropping, scaling, and so on).

Acknowledgment

This work was supported by Grants-in-Aid for Young Scientists (B) of Japan Society for the Promotion of Science (JSPS14750276).

References

- [1] Kineo Matsui, "The foundation of watermarking," The Morikita Publishing Co., 1988.
- [2] Junji Onishi, Kineo Matsui, "A Method of Watermarking with Multiresolution Analysis and PN Sequence," IEICE Trans., Vol.J80-D-II, No.11, pp.3020-3028, Nov. 1997.
- [3] Tomokazu Onuki, Takeharu Adachi, Madoka Hasegawa, Shigeo Kato, "A Study on a Digital Watermarking Method for Still Images," 2000 International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'00), pp.19-22, Pusan, Korea, July 2000.
- [4] Jun Watanabe, Madoka Hasegawa, Shigeo Kato, "Study on Preventing Data Disappearance on Non-Attacking for Digital Watermark Using Orthogonal Transform," 2001 Picture Coding Symposium of Japan (PCSJ2001), pp.99-100, Nagano, Japan, Nov. 2001. (in Japanese)
- [5] Jun Watanabe, Madoka Hasegawa, Shigeo Kato, "Study on Data Embedding Method for Digital Watermark Using Orthogonal Transform," Tech.Rep. of the Institute of Image Information and Television Engineers, Vol.26, No.24, pp.49-54, Mar. 2002. (in Japanese)