VARIATION-FREE WATERMARKING TECHNIQUE BASED ON SCALE RELATIONSHIP

Jung-San Lee, Hsiao-Shan Wong, and Yi-Hua Wang

Department of Information Engineering and Computer Science,
Feng Chia University,
Taichung 407, Taiwan, ROC
leejs@fcu.edu.tw

ABSTRACT

Most watermark methods use pixel values or coefficients as the judgment condition to embed or extract a watermark image. The variation of these values may lead to the inaccurate condition such that an incorrect judgment has been laid out. To avoid this problem, we design a stable judgment mechanism, in which the outcome will not be seriously influenced by the variation. The principle of judgment depends on the scale relationship of two pixels. From the observation of common signal processing operations, we can find that the pixel value of processed image usually keeps stable unless an image has been manipulated by cropping attack or halftone transformation. This can greatly help reduce the modification strength from image processing operations. Experiment results show that the proposed method can resist various attacks and keep the image quality friendly.

KEYWORDS

Image watermarking, Discrete Cosine Transform (DCT), variation-free, coordinate system

1. INTRODUCTION

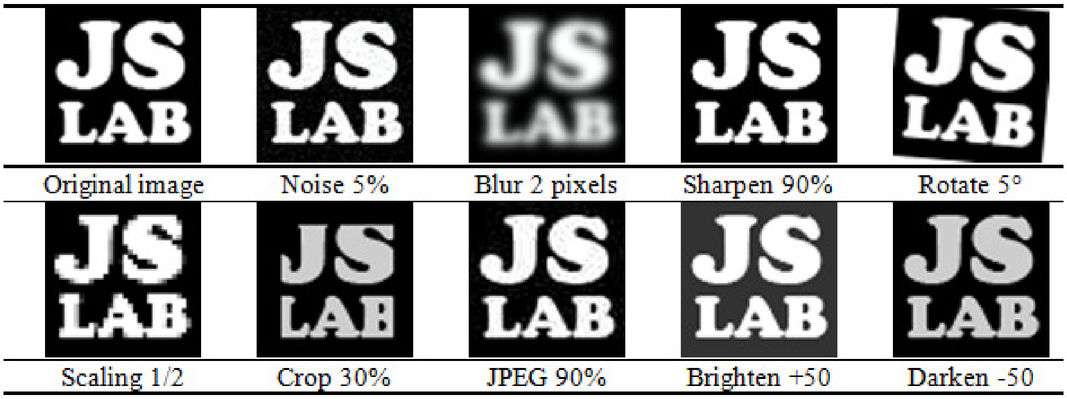
Watermarking technique is often used in anti-counterfeiting technique, and the main purpose is to solve the problem of copyright verification. It mainly marks one or more secrets and representative copyright information such as the logo of the owner in the protected digital multimedia. When this protected digital multimedia is transmitted over the insecure Internet, the secrets must be able to survive to verify the ownership[1, 9].

The digital watermarking technology can be divided into three categories: spatial domain, frequency domain, and compression domain. Spatial domain embedding technique is to modify the pixel values directly. Generally, this technique is efficient, but it is insecure once the watermark image is erased by various image processing operations. As to the frequency domain technique, it first transforms the image pixel values into coefficients via a specific conversion method such as DCT and DWT [2, 5, 6, 10]. Then the watermark bits are embedded into the coefficients. Compared with the spatial domain embedding technique, the frequency one needs more computational cost. Nevertheless, its ability to resist different image processing operations is much better. As to the compression domain watermarking [3, 4, 7, 8, 11, 12, 13], this technique is usually to compute a secret key or a codebook instead of embedding a watermark logo into the protected

David C. Wyld et al. (Eds) : CST, ITCS, JSE, SIP, ARIA, DMS - 2014
pp. 225–239, 2014. © CS & IT-CSCP 2014 DOI : 10.5121/csit.2014.4121

image. Thus, we can obtain a lossless outcome since no pixel value is modified during the embedding procedure. By this way, we can guarantee to get a satisfactory watermarked image. But, we need extra memory to record the secret key or codebook for watermark retrieval.

Based on the above mentioned literature, it is clear that the pixel values and coefficients are the commonest component used to define the judgment condition of watermark embedding and extracting. Accordingly, once a watermarked image suffers from attacks, the modified values or coefficients must lead to the incorrect logo retrieval. To avoid this misjudgment, we aim to design a more stable estimation mechanism. Thus, we introduce the XNOR operation and voting strategy to the proposed watermarking scheme. Figure. 1 illustrates an original image and the outcomes after common signal processing operations. We can find out that the pixel value in the same position usually keeps stable unless the whole image has suffered from bring seriously destroyed, such as cropping attack and halftone transformation. To enhance the stability of estimation condition, we first select two distinct pixels. Then, we apply the scale relationship of these two pixels to be the judgment condition of watermark embedding and extracting. This can make the condition more flexible even the target image has been distorted. For instance, a pair of pixel is changed from (100, 50) to (80, 60). The estimation condition will not be influenced since the relation of those two pixels still keeps the same, said $n_1 > n_2$.


Original image | Noise 5% | Blur 2 pixels | Sharpen 90% | Rotate 5°

Scaling 1/2 | Crop 30% | JPEG 90% | Brighten +50 | Darken -50

Figure.1 The common image processing operations

To prevent the estimation condition from being swayed by the modification of pixel values, the main idea of our method is to keep the target image the same. What we do to embed the watermark is to apply the XNOR operation on the watermark logo and the scale relationship of one pair of above mentioned pixels. Then we can record the outcome as a secret key. By this way, we can obtain a lossless image and get a stable estimation condition. Thus the new method can confirm the robustness and transparency.

Furthermore, how to decrease the occurrence of error extraction in a robust watermarking scheme is the challenge we are going to solve in this paper. We introduce the voting strategy to embed each watermark bit into different positions, respectively. When the watermarked image suffered from attacks, some watermarked coefficients may not be affected. Accordingly, we can determine the watermark bits through the voting strategy.

The rest of this paper is organized as follows: In Section 2, the proposed watermarking method is introduced. In Section 3, the performance is analyzed by applying various attacks to the watermarked image. Finally, the conclusion is given in Section 4.

2. The Proposed Watermarking Method

Here, we introduce the detail of how to apply the scale relationship of two distinct pixels and voting strategy to perform the watermark embedding and extracting procedure without any modification on the host image.

2.1 The pixel selection rule

For the estimation condition of watermarking embedding and extracting, we use a pair of pixels as the main component. The first pixel n_1 is chosen by PRNG (Pseudo Random Number Generator), and the other one n_2 is decided according to the selection of n_1. Actually, the difference between two neighbor pixels is usually small. That is, the scale relationship of two neighbor pixels might be the same in distinct host images. This results in the fact that we may retrieve a similar watermark logo from different host images, said a collision. Considering the uniqueness of images, two pixels at a distance might locate in distinct objects. So, the basic idea is to shift the position of n_1 for a distance s to get n_2. As shown in Figure. 2, we have shifted node A for a distance to get node B. It is clear that these two nodes have been marked in different objects; thus representing the distinguishing characteristic of images. With the help of voting strategy, the shifting can effectively prevent the occurrence of the collision.

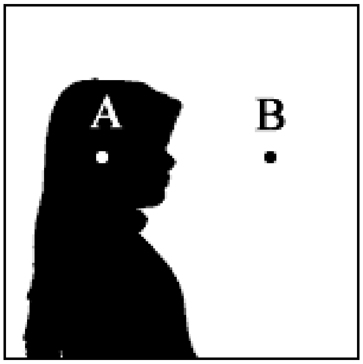


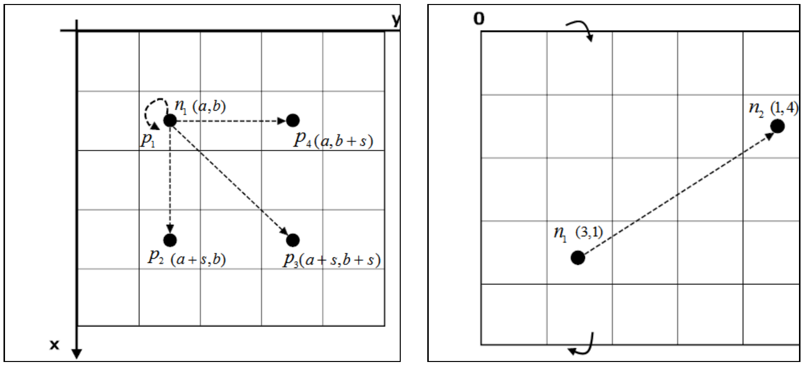
Figure. 2 The characteristic of image

We apply the coordinate system to help find the second pixel of the pair. To increase the variance and enhance the flexibility, we determine the shifting procedure according to the coordinate of the first pixel. Let $(a,\ b)$ be the coordinate of n_1 and s represent the distance. The selection of n_2 is defined as Eq. (1).

$$n_2 = \begin{cases} (a,\ b) & \text{if } a \text{ is even and } b \text{ is even} \\ (a+s,\ b) & \text{if } a \text{ is odd and } b \text{ is even} \\ (a+s,\ b+s) & \text{if } a \text{ is odd and } b \text{ is odd} \\ (a,\ b+s) & \text{if } a \text{ is even and } b \text{ is odd} \end{cases} \tag{1}$$

As illustrated in Figure. 3(a), there are four possible positions of n_2 after the shifting procedure, $p_1 = n_1 = (a,\ b)$, $p_2 = (a+s,\ b)$, $p_3 = (a+s,\ b+s)$, and $p_4 = (a,\ b+s)$. For instance, assume $s = 3$ and $n_1 = (1,\ 1)$, we have $n_2 = (1+3,\ 1+3) = (4,\ 4)$. In case that $n_1 = (2,\ 1)$, we get $n_2 = (2,\ 1+3) = (2,\ 4)$. If $n_1 = (1,\ 2)$, we obtain $n_2 = (1+3,\ 2) = (4,\ 2)$. Suppose $n_1 = (2,\ 2)$, we can infer $n_2 = n_1 = (2,\ 2)$. To guarantee that we can retrieve the exact watermark bit, we shall keep the case of $n_2 = n_1$.

Note that the shifting procedure is rotation-based. Once the shift distance runs over the bound, it continues from the opposite. Let us check the scenario in Figure. 3(b). If $n_1 = (3,\ 1)$, we have $n_2 = (3+3-5,\ 1+3) = (1,\ 4)$. As to the setting of distance s , it should be around $\left\lceil \dfrac{\text{length of side}}{2} \right\rceil$. The settings of a large s and a small s will result in the same situation that n_2 will be close to n_1 ; thus leading to a similar logo from two different host images.



(a) (b)

Figure. 3 The selection of pixel position

2.2 The embedding procedure

Assume that the size of host image O is $N \times N$ pixels, each pixel is denoted as p_{ij} for $0 \le i, j < N$, and that of the watermark image W is $M \times M$ pixels. The flowchart of the embedding phase is shown in Figure. 4, and the details are given in the following. Here, we define two secret keys K_w and K_p. K_w is used to decide the order of processed watermark bit, while K_p is adopted to determine the embedding position in the original image.

Step1. Randomly select a watermark bit w_h from the watermark image W according to K_w, for $0 \le h \le M \times M$. Set $v = 1$, where v is the number of vote.

Step2. Apply K_p to PRNG to find the first pixel n_1. Accordingly, we can obtain the second n_2 by Eq. (1).

Step3. Get parameter f by Eq. (2).

$$f = \begin{cases} 1 & n_1 \ge n_2 \\ 0 & n_1 < n_2 \end{cases} \tag{2}$$

Step4. Input f and w_h to XNOR operation (see Table 1) to obtain an r_m , for $m = 1,\ 2,$ to $h \times 3$. Record all the outcomes as a secret key.

Step5. In case $v < 3$, set $v = v + 1$ and repeat Steps 2 to 5. Repeat Steps 1 to 5 till all the watermark bits are embedded.

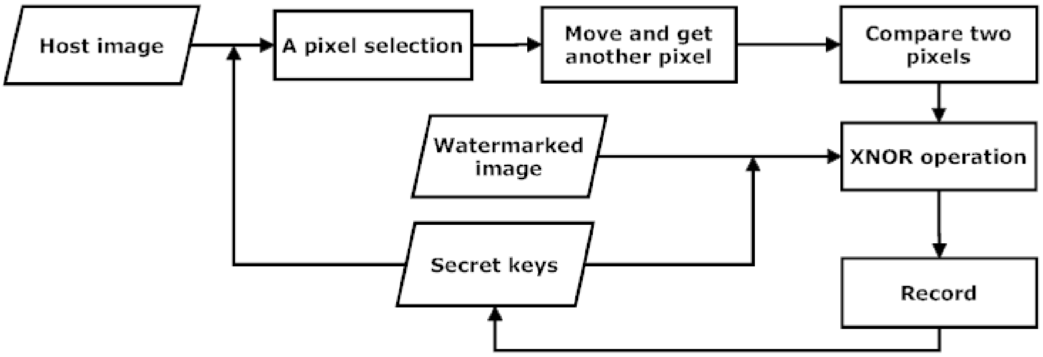


Figure. 4 The flowchart of watermark embedding

Table 1. XNOR operation

f	w_h	$r_m = \overline{f \oplus w_h}$
0	0	1
0	1	0
1	0	0
1	1	1

2.3 Watermark extracting procedure

Figure. 5 illustrates the flowchart of watermark extracting. The detail of the procedure is given as follows.

Step1. Set $v = 1$.

Step2. Decide the position of target watermark bit w_h according to K_w, where $0 \le h \le M \times M$. Employ K_p to PRNG to find the corresponding pixel n_1. Accordingly, we can shift n_1 to obtain the second pixel n_2 by Eq. (1).

Step3. Compute f value according to Eq. (2).

Step4. Extract a corresponding secret bit from r_m, where $m = 1,\ 2,$ to $h \times 3$. Apply XNOR operation to r_m and f to obtain t_x, for x = 1, 2, 3. Keep t_x in a temporary register.

Step5. If $v < 3$, perform $v = v + 1$ and repeat Steps 2 to 5. Otherwise, apply the voting strategy to t_1, t_2 and t_3 to determine w_h.

Step6. Repeat Steps 1 to 5 until all the watermark bits are retrieved.

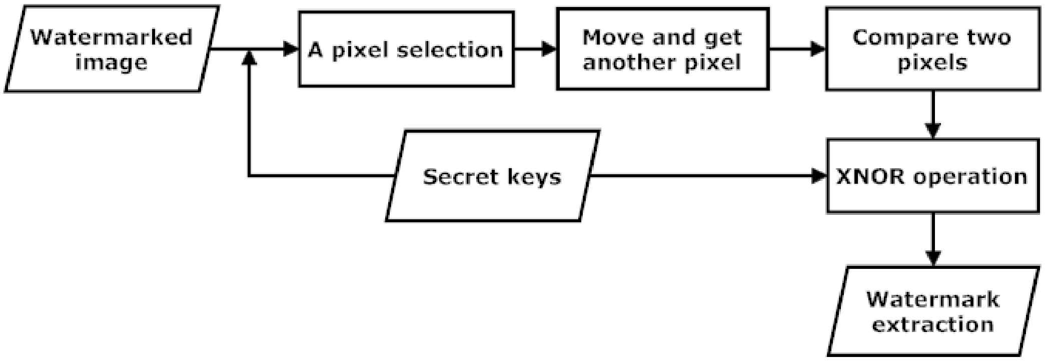


Figure. 5 The flowchart of extraction process

3. EXPERIMENTAL RESULTS

In this section, we employed JAVA to conduct all simulations to prove the practicability and robustness of the proposed scheme, including common image processing operations and attacks. Furthermore, we simulated several related works and compared the results with ours to highlight superiority. In Lin et al.'s method [8], the modulus M is set as 18 and the quality factor f = 0.3. In Lai's method [7], the threshold is tuned as 0.04. To obtain a better result in Run et al.'s method [12], the essential parameters are defined as $T = 10$, $\beta = 150$, $\gamma = 1.5$, and $\alpha = 0.95$.

Tools for simulating image processing operations and attacks include JAVA and Photoshop CS2. Simulation types and settings are introduced as follows.

1. Noise: Apply the Photoshop to add Gauss noise by 0.5% to 5%.
2. Blurring: Apply the Photoshop to perform Gauss blurring with the radius from one to five pixels.
3. Cropping: Use JAVA to simulate the cropping processing, including the inside cropping and the outside cropping. The inside cropping mainly concerns the object such as human faces, while the outside one focuses on removing the suburb of the image by 25%, which may destroy the reference information of watermark retrieval.
4. JPEG compression: Employ JAVA API to simulate the lossy compression according to the standard JPEG algorithm. The compression quality ranges from 30% to 90%.

In order to accurately evaluate image quality, aside from the human vision perception, we utilized the Peak Signal to Noise Ratio (PSNR) which is defined as Eq. (3).

$$PSNR(dB) = 10\log_{10}\left(\frac{255^2 \times H \times W}{\sum_{i=1}^{H}\sum_{j=1}^{W}(x_{ij} - \hat{x}_{ij})^2}\right), \tag{3}$$

where H and W are the height and width of the image, respectively, x_{ij} is the original image pixel value at coordinate (i, j), and $\hat{x}_{ij}$ is the camouflage image pixel value at coordinate (i, j).

Moreover, the Normalized Correlation (NC) value which is defined as Eq. (4) is introduced to measure the similarity between the original watermark image and the extracted one, and NC = [0, 1]. The similarity between two images is higher if the value gets closer to 1.

$$NC = \frac{\sum_{i=1}^{h}\sum_{j=1}^{w}(w_{ij} \times \hat{w}_{ij})}{\sum_{i=1}^{h}\sum_{j=1}^{w}(w_{ij} \times w_{ij})}, \tag{4}$$

We also utilized Tamper Assessment Function (TAF) value to evaluate the tampered level of watermark image and the formula is defined as follow.

$$TAF = \frac{\sum_{i=1}^{h}\sum_{j=1}^{w}(w_{ij}\times\hat{w}_{ij})}{h\times w}\times 100\,, \tag{5}$$

where w_{ij} and $\hat{w}_{ij}$ represent the original and extracted watermark at coordinate (i, j), respectively.

In the following experiments, all the test images are with size of 1024×1024 pixels, and the watermark logo is a binary image with size of 64×64 pixels, as displayed in Figure. 6.

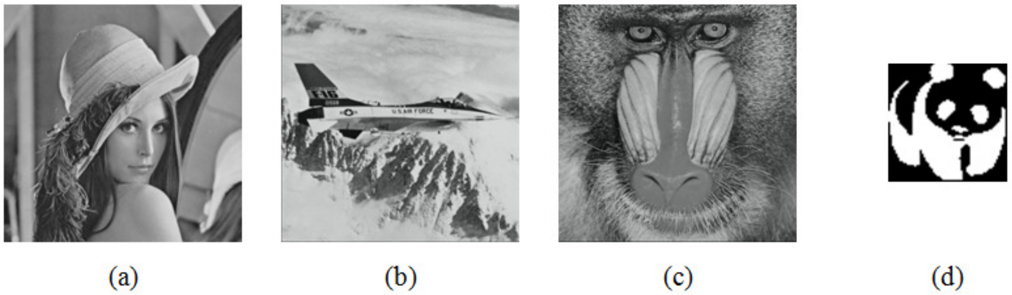
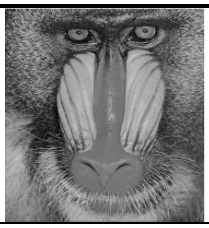
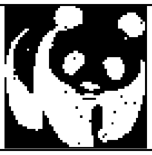
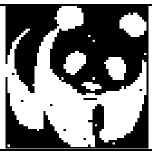
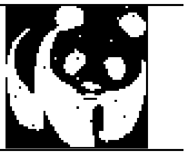
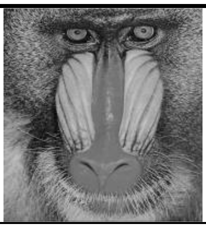
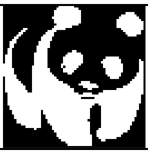
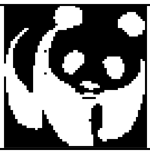
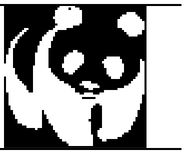
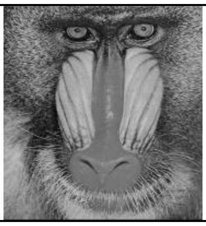
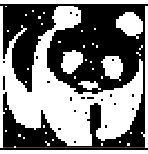
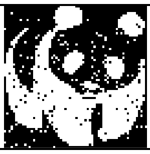
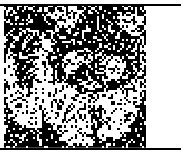
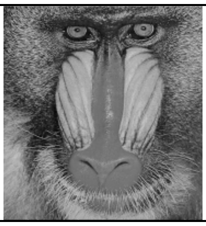
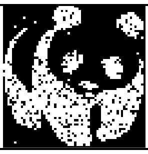
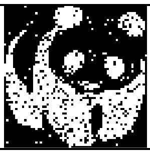
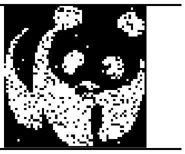
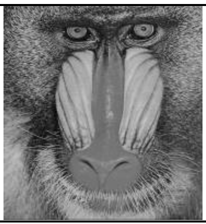


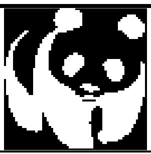
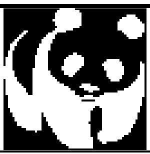
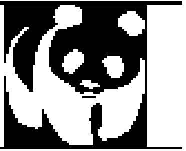
Figure. 6 Test images

Table 2 shows the simulation results of four related works and ours under three host images. Without any attacks, we can obtain the high quality watermarked images from related works. Since the host image has been modified for watermark embedding, there exist some distortions in the recovered image. Thus, we can not get lossless NC and TAF values of the extracted logo. On the contrary, the watermark logo is not really embedded in the host image in the proposed method. This results in keeping a perfect quality of watermarked image. Also, we can obtain complete NC and TAF values of the retrieved logo.

Table 2. The watermarked image and the extracted image without any attacks

		Lena	*Baboon*	*Airplane*
Patra et al.'s method [11]	Watermarked image			
	PSNR	43.81 dB	44.71 dB	45.54 dB
	Extracted image			
	NC	0.99	0.99	0.99
	TAF	0.54	0.56	0.46

		Lena	*Baboon*	*Airplane*
Lin et al.'s method [8]	Watermarked image			
	PSNR	37.55 dB	38.45 dB	38.15 dB
	Extracted image			
	NC	0.99	0.99	0.99
	TAF	0.02	0.05	0.02
Lai's method [7]	Watermarked image			
	PSNR	45.19 dB	37.3 dB	44.41 dB
	Extracted image			
	NC	0.99	0.96	0.74
	TAF	2	4.42	26
Run et al.'s method [12]	Stego image			
	PSNR	34.4 dB	35.74 dB	34.15 dB
	Extracted image			
	NC	0.85	0.86	0.84
	TAF	6.98	7.62	7.25
Proposed method	Watermarked image			
	PSNR	**Infinity**	**Infinity**	**Infinity**

	Lena	*Baboon*	*Airplane*
Extracted image			
NC	**1**	**1**	**1**
TAF	**0**	**0**	**0**

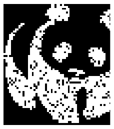
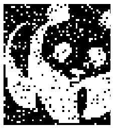
Actually, it is common that an image may suffer from malicious attacks or some signal processing operations during the network transmission or format transformation. Thus, the robustness should be taken into consideration for performance evaluation. First, we simulated the noise attack by increasing Gauss noise to the watermarked image, and the results are displayed in Table 3. It is clear that parts of pixel values may increase or decrease due to the extra noise. Considering the techniques adopted in related works, they mainly use specific transformation, including DCT, DWT, and SVD. The output of each coefficient after the transformation depends on many pixels. Once the level of noise increases, the corresponding modification and the number of affected coefficient become large. This often leads to lower down the correction ratio of extracted watermark bit. In the proposed method, we apply the scale relationship of two independent pixels for watermark embedding; thus leading to higher toleration of pixel modification in the watermarked image. With the help of voting strategy, the error rate of extracted logo can be effectively reduced, and the retrieved results can stay stable even the noise distortion becomes more serious. As shown in Table 3, the proposed method can outperform others in terms of NC value, TAF value, and human vision perception.

Table 4 illustrates the watermarked images under different levels of Gauss blurring. This operation must smooth the whole image and lower down the readability of detailed content. In particular, a small level of blurring will modify a large amount of pixels. And, this is the main reason why the frequency-based techniques can not resist the blurring attack. Nevertheless, we employ the comparison between two independent pixels to form the estimation judgment. Pixels in a pair are separated at a distance so that the difference between two corresponding pixels is usually large. Thus, the absolute difference between two pixels can stay steady under the blurring attack. This has demonstrated the robustness to blurring attack.

The cropping attack can be classified into two types: the inside cropping and the outside one. The inside cropping is mainly used to delete some objects such as face, while the outside one is often applied to cut the meaningless contour area to shorten the image. In the experiments, we used *Lena* and *Airplane* for inside cropping simulation, which are two images containing conspicuous objects. And, we cut the face of *Lena* and the body of *Airplane*. The shape of cropping could be various. For simplicity, we adopted the rectangle. The experimental results are listed in Table 5. In general, it is more difficult for a watermarking technique to withstand the outside cropping attack since the basic reference information for watermark retrieval usually locates at the suburb of the image. The main procedure of cropping is to remove some parts of the image in spatial domain. Actually, it is easy for a selected pixel to locate within the removed area in our proposed method. This must result in the inaccurate watermark extraction. However, the adoption of voting strategy has given a good solution for this weakness. As shown in the table, even the proposed method can not offer an optimal performance in this case; it still yields a recognizable watermark. This has shown that the new method has the capability of resisting the cropping attack.

To highlight the practicability of the new method, we further conducted the simulation to demonstrate its robustness to JPEG compression, which is one of the commonest compression standards during in the field of network communications. Table 6 provides the comparison results between related works and ours under different levels of compression. The procedure of sampling and quantification is the main technique used to achieve the effective compression in JPEG standard. The adoption of quantification step can help guarantee that we can obtain a high quality compressed image after JPEG algorithm. More precisely, it only slightly varies the pixel values in the spatial domain. Thus, the new method can successfully resist this signal processing operation.

Table 3. The results under different levels of Gauss noise

Noise						
		1%	2%	3%	4%	5%
Patra et al.'s method [11]	Extracted image					
	NC	0.86	0.8	0.7	0.68	0.63
	TAF	7.57	11.62	17.09	17.9	21.61
Lin et al.'s method [8]	Extracted image					
	NC	0.91	0.85	0.77	0.73	0.72
	TAF	4.13	6.67	10.3	12.08	12.23
Lai's method [7]	Extracted image					
	NC	0.93	0.89	0.82	0.78	0.76
	TAF	8.45	12.74	18.77	22.83	23.68

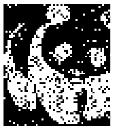
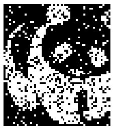
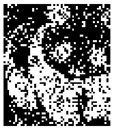
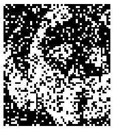
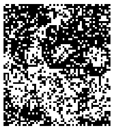
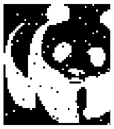
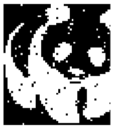
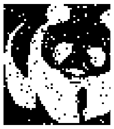
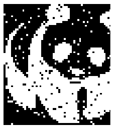
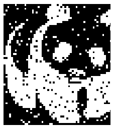
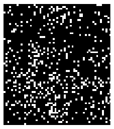
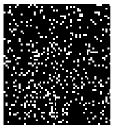
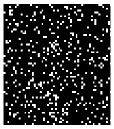
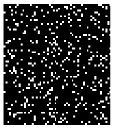
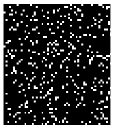
Noise						
		1%	2%	3%	4%	5%
Run et al.'s method [12]	Extracted image					
	NC	0.82	0.8	0.75	0.72	0.68
	TAF	9.64	12.28	17.97	22.79	27.39
Proposed method	Extracted image					
	NC	**0.99**	**0.98**	**0.96**	**0.96**	**0.93**
	TAF	**1.44**	**2.42**	**3.86**	**4.47**	**6.08**

Table 4. The results under different levels of Gauss blurring

Blur						
		1 pixel	2 pixels	3 pixels	4 pixels	5 pixels
Patra et al.'s method [11]	Extracted image					
	NC	0.21	0.16	0.13	0.13	0.12
	TAF	39.36	43.02	44.43	44.58	44.82

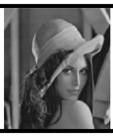
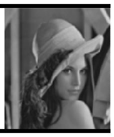
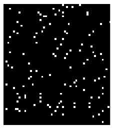
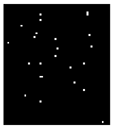
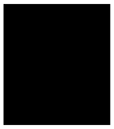
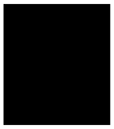
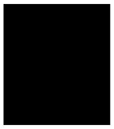
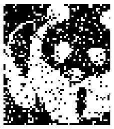
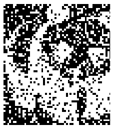
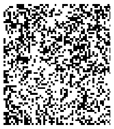
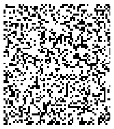
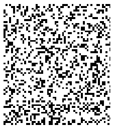
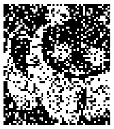
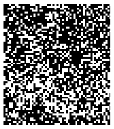
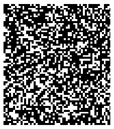
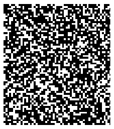
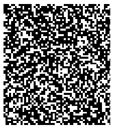
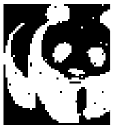
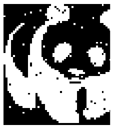
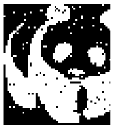
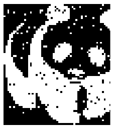
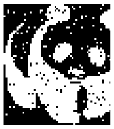
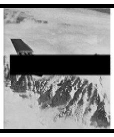
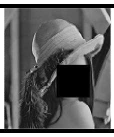
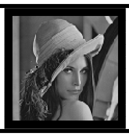
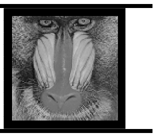
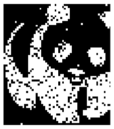
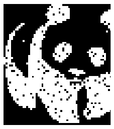
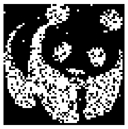
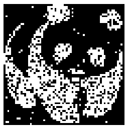
Blur						
		1 pixel	2 pixels	3 pixels	4 pixels	5 pixels
Lin et al.'s method [8]	Extracted image					
	NC	0.05	0.01	0	0	0
	TAF	43.12	44.12	44.04	44.04	44.04
Lai's method [7]	Extracted image					
	NC	0.91	0.86	0.73	0.72	0.71
	TAF	12.33	21.78	40.58	48.22	51.15
Run et al.'s method [12]	Extracted image					
	NC	0.76	0.5	0.43	0.41	0.4
	TAF	21.75	37.62	43.65	46.12	47.46
Proposed method	Extracted image					
	NC	**0.99**	**0.98**	**0.97**	**0.96**	**0.96**
	TAF	**0.73**	**1.54**	**2.78**	**3.66**	**4.3**

Table 5. The results under different sizes of cropping

Cropping					
		Inside	Inside	Outside 25%	Outside 25%
Patra et al.'s method [11]	Extracted image				

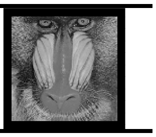
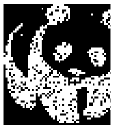
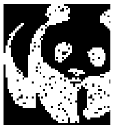
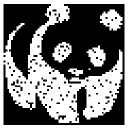
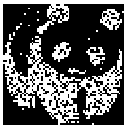
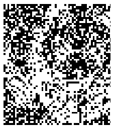
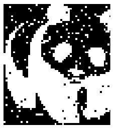
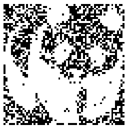
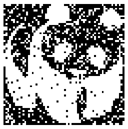
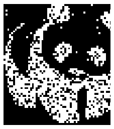
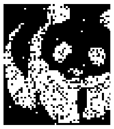
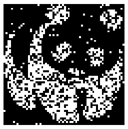
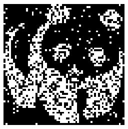
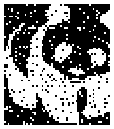
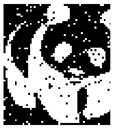
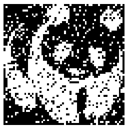
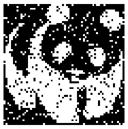
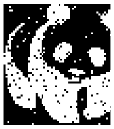
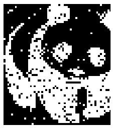
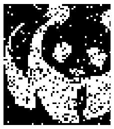
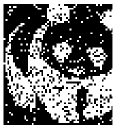
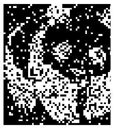
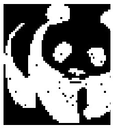
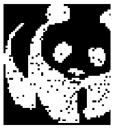
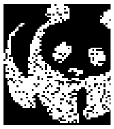
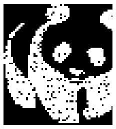
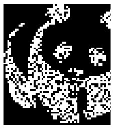
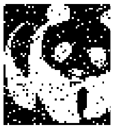
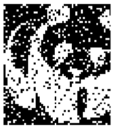
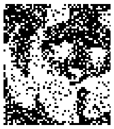
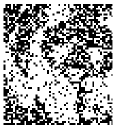
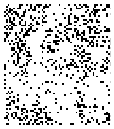
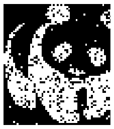
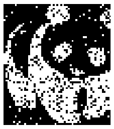
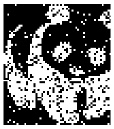
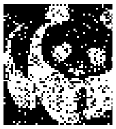
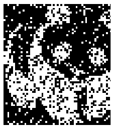
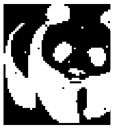
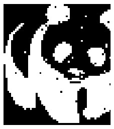
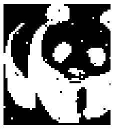
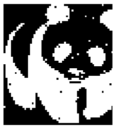
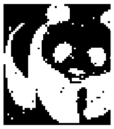
Cropping					
		Inside	Inside	Outside 25%	Outside 25%
	NC	0.84	0.91	0.73	0.73
	TAF	7.18	3.86	12.3	12.7
Lin et al.'s method [8]	Extracted image				
	NC	0.8	0.92	0.9	0.67
	TAF	8.72	3.69	4.52	15.06
Lai's method [7]	Extracted image				
	NC	0.75	0.99	0.99	0.97
	TAF	31.57	4.2	24.24	14.6
Run et al.'s method [12]	Extracted image				
	NC	0.74	0.78	0.65	0.67
	TAF	11.87	10.06	16.55	16.82
Proposed method	Extracted image				
	NC	0.92	0.96	0.88	0.87
	TAF	8.23	4.22	12.38	13.72

Table 6. The results under different ratios of JPEG compression

JPEG					
	90%	70%	60%	50%	30%

JPEG						
		90%	70%	60%	50%	30%
Patra et al.'s method [11]	Extracted image					
	NC	0.93	0.89	0.82	0.79	0.69
	TAF	4.2	6.47	9.81	12.67	18.97
Lin et al.'s method [8]	Extracted image					
	NC	0.98	0.93	0.81	0.91	0.66
	TAF	0.98	3.05	8.5	4.08	14.94
Lai's method [7]	Extracted image					
	NC	0.94	0.89	0.86	0.86	0.94
	TAF	6.27	12.23	19.68	26.88	40.31
Run et al.'s method [12]	Extracted image					
	NC	0.83	0.82	0.81	0.81	0.74
	TAF	7.96	9.64	10.6	11.67	16.72
Proposed method	Extracted image					
	NC	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
	TAF	**0.42**	**0.88**	**0.98**	**0.98**	**1.39**

4. CONCLUSIONS AND FUTURE WORKS

In this paper, we have integrated XNOR operation and voting strategy to design a watermarking scheme in the spatial domain. Based on the observation that the relativity of pixels usually keeps stable after common signal processing operations or attacks, applying the scale relationship of two pixels to be the main component of judgment condition can effectively improve the correctness of watermark retrieval. As shown in the simulation results, the new method can resist most of signal processing operations and attacks. Specifically, it can greatly outperform related works in the cases of JPEG compression and Cropping.

REFERENCES

[1] V. Aslantas, S. Ozer, and S. Ozturk, “Improving the Performance of DCT-based Fragile Watermarking using Intelligent Optimization Algorithms,” *Optics Communications*, Vol. 282, No. 14, pp. 2806-2817, 2009.

[2] M. Barni, F. Bartolini, and A. Piva, “Improved Wavelet-based Watermarking through Pixel-wise Masking,” *IEEE Transactions on Image Processing*, Vol. 10, No. 5, pp. 783-791, 2001.

[3] P. Bao and X. Ma, “Image Adaptive Watermarking using Wavelet Domain Singular Value Decomposition,” *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 15, No. 1, pp. 96-102, 2005.

[4] C.C. Chang, C.C. Lin, and C.S. Tseng, and W. L. Tai, “Reversible Hiding in DCT-based Compressed Images,” *Information Sciences*, Vol. 177, No. 13, pp. 2768-2786, 2007.

[5] J. R. Hernandez, M. Amado, and F. Perez-Gonzalez, “DCT-domain Watermarking Techniques for Still Images: Detector Performance Analysis and a New Structure,” *IEEE Transactions on Image Processing*, Vol. 9, pp. 55-68, 2000.

[6] J.R. Kim and Y.S. Moon, “A Robust Wavelet-Based Digital Watermarking Using Level-Adaptive Thresholding,” *International Conference on Image Processing*, Vol. 2, pp. 226-230, 1999.

[7] C.C. Lai, “An Improved SVD-based Watermarking Scheme using Human Visual Characteristics,” Optics Communications, Vol. 284, pp. 938-944, 2011.

[8] S.D. Lin, S.C. Shie, and J.Y. Guo, “Improving the Robustness of DCT-based Image Watermarking Against JPEG Compression,” *Computer Standards and Interfaces*, Vol. 32, pp. 54-60, 2010.

[9] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, “Watermarking Digital Image and Video Data,” IEEE Signal Processing Magazine, Vol. 17, No. 5, pp. 20-46, 2000.

[10] W. Lu, W. Sun, and H. Lu, “Robust Watermarking based on DWT and Nonnegative Matrix Factorization,” Computers and Electrical Engineering, Vol. 35, No. 1, pp. 183-188, 2008.

[11] J.C. Patra, J.E. Phua, C. Bornand, “A Novel DCT Domain CRT-based Watermarking Scheme for Image Authentication Surviving JPEG Compression,” Digital Signal Processing, Vol. 20, pp. 1597-1611, 2010.

[12] R.S. Run, S.J. Horng, W. H. Lin, T.W. Kao, P. Fan, and M. K. Khan, “An Efficient Wavelet-tree-based Watermarking Method,” Expert Systems with Applications, Vol. 38, No. 12, pp. 14357-14366, 2011.

[13] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, “Multimedia Data-embedding and Watermarking Technologies,” Proceedings of the IEEE, Vol. 86, No. 6, pp. 1064-1087, 1998.