Robust Digital Image-in-Image Watermarking Algorithm Using the Fast Hadamard Transform

Anthony T.S. Ho, Jun Shen, Andrew K.K. Chow[*], Jerry Woon[*]

School of Electrical and Electronic Engineering
Nanyang Technological University
Nanyang Avenue
Singapore 639798
Email: etsho@ntu.edu.sg

[*] DataMark Technologies Pte Ltd
Singapore Technologies Building
100 Jurong East Street 21
Singapore 609602
http://www.datamark-tech.com

Abstract

In this paper, a robust and efficient digital image watermarking algorithm using the fast Hadamard transform (FHT) is proposed for the copyright protection of digital images. This algorithm can embed or hide an entire image or pattern as a watermark such as a company's logo or trademark directly into the original image. The performance of the proposed algorithm is evaluated using Stirmark 3.1which consists of 90 different types of image attacks. Results show that the proposed algorithm is very robust and can survive most of the Stirmark attacks.

These attacks were tested on a number of test images of size 512×512×8 bits, embedded with a watermark image of size 64×64×8 bits. The simplicity of the fast Hadamard transform also offers a significant advantage in shorter processing time and ease of hardware implementation than other orthogonal transforms, such as the discrete cosine transform and wavelet transform.

Index Terms: Fast Hadamard Transform, Copyright Protection, Digital Watermarking

1 Introduction

With the advent of the Internet, the online purchasing and distribution of digital images can now be performed relatively easily. However, there exists one major problem associated with the distribution of any digital images is the important issue of copyright protection and the proof of rightful ownership. Over the past few years, the technology of digital watermarking has gained prominence and emerged as a leading candidate that could solve the fundamental problems of legal ownership and content authentications for digital multimedia data (e.g. audio, image, and video).

A digital watermark is a sequence of information, containing the owner's copyright information for the protected multimedia data. It is an invisible mark inserted into the digital multimedia data so that it can be detected in the later stage for evidence of rightful ownership. A great deal of research efforts has been focused on digital image watermarking in recent years. The techniques proposed so far can be divided into two main groups according to the embedding domain of the container image [1].

One group is the spatial domain approach. The earliest watermarking techniques are mainly this kind and the simplest example is to embed the watermark into least significant bits (LSBs) of the image pixels [2]. However, this technique has relatively low information hiding capacity and can be easily erased by lossy image compression. The other is the frequency domain approach. This approach can embed more information bits and is relatively robust to attacks. Cox et al. [3] used the spread spectrum communication for digital multimedia watermarking. They embedded a Gaussian distributed sequence into the perceptually most significant frequency components of container image.

Hsu and Wu embedded an image watermark into selectively modified middle frequency of discrete cosine transform (DCT) coefficients of container image [4]. Joseph et al. developed a digital image watermarking using the Fourier-Mellin transform that is invariant to image manipulations or attacks due to rotation, scaling and translation [5]. Several other methods used discrete wavelet transform (DWT) to hide the data to the frequency domain [6-8]. Wei et al. applied JND (just-noticeable distortion) feature of HVS (human visual system) in wavelet transform domain and hid 236 information bits into the 'lenna' image [6]. The major problem with many of these watermarking schemes is that they are not very robust against different types of image manipulations or attacks such as the ones found in Stirmark. Moreover, some of these techniques are quite complicated to implement in real-time.

In this paper, we propose a fast Hadamard transform (FHT) based watermarking approach that embeds a grayscale image as a watermark. The watermark grayscale image is decomposed into Hadamard coefficients for embedding. To increase the invisibility of the watermark, a visual model based on the original image characteristics, such as edges and textures are incorporated to determine the watermarking strength factor. This factor is used to scale the watermark coefficients to a similar range to the coefficients from the Hadamard coefficients of the sub-blocks of the container image. The FHT embedding algorithm was found to provide a robust and efficient approach to perform digital watermarking of digital image data for copyright protection and proof of rightful ownership. The simplicity of FHT offers a significant advantage in shorter processing time and ease of hardware implementation than most orthogonal transform techniques such as DCT and DWT.

2 2D-Hadamard transform of signal

The 2D-Hadamard transform has been used extensively in image processing and image compression [9,10]. In this section, we give a brief overview of the Hadamard transform representation of image data, which is used in the watermarking embedding and extraction process. The reason of choosing FHT domain is also discussed.

Let $[U]$ represents the original image and $[V]$ the transformed image, the 2D-Hadamard transform is given by

$$[V] = \frac{H_n [U] H_n}{N} \tag{1}$$

Where H_n represents an $N \times N$ Hadamard matrix, $N=2^n$, $n=1,2,3\ldots$, with element values either +1 or –1. The advantages of Hadamard transform are that the elements of the transform matrix H_n are simple: they are binary, real numbers and the rows or columns of H_n are orthogonal. Hence the Hadamard transform matrix has the following property:

$$H_n = H_n^{\ *} = H^T = H^{-1} \tag{2}$$

Since H_n has N orthogonal rows $H_nH_n = NI$ (I is the identity matrix) and $H_nH_n = NH_nH_n^{-1}$, thus $H_n^{-1} = H_n /N$. The inverse 2D-fast Hadamard transform (IFHT) is given as

$$[U] = H_n^{\ -1} [V] H_n^{\ *} = \frac{H_n [V] H_n}{N} \tag{3}$$

In our watermarking algorithm, the forward and reverse Hadamard transform is applied to the sub-blocks of the original or watermarked images. A two-dimensional FHT of the segmented 8×8 blocks is performed by applying a one-dimensional FHT on the rows first and then followed by a 1-D FHT on the columns.

The Hadamard matrix of the order n is generated in terms of Hadamard matrix of order n-1 using Kronecker product, $\otimes$, as

$$H_n = H_{n-1} \otimes H_1 \tag{4}$$

or

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix} \tag{5}$$

Since in our algorithm, the processing is performed based on the 8×8 sub-blocks of the whole image, the third order Hadamard transform matrix H_3 is used. By applying (4) or (5), H_3 becomes:

$$H_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \qquad (6)$$

For a H_3 matrix, the number of transitions for row 1 to row 8 is 0, 7, 3, 4, 1, 6, 2 and 5. The number of sign changes is referred to as sequency [9]. The concept of sequency is analogous to frequency for the Fourier Transform. Zero sign transition corresponds to a DC component. While a large number of sign transitions correspond to high frequency components. For a Hadamard matrix H_3, the elements are not arranged in an increasing sequency, such that the transitions are 0, 1, 2, 3, 4, 5, 6 and 7. If the order of rows is in an ascending of sequency, the transform matrix is called a Walsh transform matrix.

A Walsh transform may cause the transformed matrix to have a DC value at the upper left corner with AC coefficients arrange in a Zigzag order from low frequency components to high frequency components. The Walsh transform is not suitable for watermarking because the middle and high frequency AC components have shown to be somewhat unreliable and the performance worse than the existing DCT watermarking algorithms. On the contrary, the Hadamard transform matrix H_3 has its AC components in a random order. In the watermarking process, some of the watermark information can be embedded into the low frequency AC components. This property increases the watermark reliability and makes it more difficult to attack and remove.

3 Image Watermarking in FHT Domain

The proposed watermarking algorithm does not require the original image information at the watermark identification stage. This refers to a "blind" watermarking process. The block diagram of the proposed watermarking system is shown in Figure 1.

Copyright information in the form of a trademark or logo can be used as an image watermark. In the watermarking embedding process, the watermark image, *w(x,y)*, is first transformed into FHT coefficients. We use a grayscale image of size 64×64 as a watermark for our testing. As such, a Hadamard transform of matrix H_6 is used. After transformation, 64×64 Hadamard transform coefficients are obtained. The DC component is stored in a key file and the AC components are then selected for embedding. The original image, *f(x,y)*, is also decomposed into a set of non-overlapped blocks of $h \times h$, denoted by $f_k(x',y')$, k=0, 1, ···, K-1, where the subscript k denotes the index of blocks and K denotes the total number of blocks. In our experiment, a test image of size 512×512 and sub-block size of 8×8 is used. The algorithm pseudo-randomly selects the sub-blocks for watermark insertion using an m-sequence random number generator. The seed of this m-sequence and initial state are also stored in the key file. After that, a FHT is performed on each selected sub-blocks of the original image. Since the sub-block size is 8×8, a Hadamard transform of matrix size H_3 is used. For each 8 x 8 sub-block, 64 Hadamard transform coefficients are obtained.

Let the watermark FHT coefficients denote by m_i. The AC components of FHT coefficients of the original image sub-blocks, before and after inserting watermark are denoted by x_i and x_i^* respectively. Where $i \in (0, n]$, with n the number of the watermarked coefficients which is 16 in our experiment. The watermark strength factor is denoted by α . The embedding formula is

$$x_i^* = \alpha m_i \qquad (7)$$

The original coefficient x_i is replaced by x_i^*. After the watermark insertion, a new 8×8 matrix of FHT coefficients is obtained. The IFHT is then applied to the 8×8 matrix using equation (3) to obtain the luminance value matrix of the watermarked image sub-block, $f_k'(x',y')$. After performing the watermark insertion for all the relevant sub-blocks of the original image, the watermarked image, *f'(x,y)*, is obtained. At the same time, a key file is also generated, which is needed for the decoding process. The image-in-image watermark embedding process is shown in Figure 2: (original image lenna.bmp marked with watermark image dmt.bmp)

The watermarked image is then passed through a channel distorted by channel noise and external attacks. The watermark is extracted by using the embedding position and the watermark strength factor α. Let the received watermarked image denote by *f''(x,y)*. The information of watermark embedded sub-blocks' position is extracted from the seed of the m-sequence and initial state number stored in the key file. By transforming all the relevant sub-blocks, $f_k''(x',y')$, into the FHT domain, we obtain all the Hadamard transform coefficients embedded with the watermark. For example, in each of the sub-block FHT coefficients, the watermark bits are inserted into the bottom right sixteen middle and high frequency components. Let these components denote by $x_i^{*'}$, the retrieved watermark FHT coefficients denote by $m_i^{'}$, where $i \in (0, n]$, and the number of the watermarked coefficients $n = 16$. The watermark extraction formula is given as:

$$m_i^{'} = \frac{x_i^{*'}}{\alpha} \tag{8}$$

The watermark FHT coefficients are extracted from all the sub-blocks of the original image. Along with the DC component stored in the key file, the AC coefficients are rearranged into a 64×64 FHT coefficients matrix. The extracted watermark image, $w^{'}(x,y)$, is obtained by IFHT of the 64×64 Hadamard coefficients matrix using equation (3).

4 Watermark Strength Factor

Two important criteria of digital watermarking are to minimize data degradation and to increase the robustness of algorithm against external attacks. Minimizing data degradation is equivalent to increase the invisibility of the watermark. This is achieved through the watermarking strength factor, which is used to control the watermark embedding strength adaptive to the original image characteristics.

The determination of the watermark strength factor is based on the original image textures and edges characteristic. It is found that edge information of an image is the most important factor for our perception of the image [11]. In fact, this information is required to be transmitted if the final receiver is the HVS (human visual system) [12]. It is essential to maintain edge integrity so as to preserve the image quality. Smooth areas influence our visual perception together with the edge information. The JND perception thresholds are relatively low as compared to the coarse textured regions. For the texture areas, the distortion visibility is low. A coarse textured region has a very high noise-sensitivity level. With this knowledge, we can adaptively control the watermarking strength applied to the different areas of the image.

The classification of different areas is based on the Hadamard transformed space energy analysis and Canny edge detection algorithm [13]. The first visual mask model is determined by the Hadamard transformed space image energy distribution. The analysis is performed on the FHT coefficients of sub-blocks for watermarking. For coarse texture and outstanding edge areas, most of the signal energy is concentrated in the AC components of the Hadamard transform. For smooth areas, the energy is mainly concentrated in the low AC components and DC component. We use a squared sum of AC components to generate this visual mask, $mask_1(j,k)$ to distinguish the smooth and coarse texture areas. The second visual mask model is generated using a Canny edge detection algorithm.

The Canny edge algorithm was selected because of its ability in detecting weak edges by using two different thresholds. It is applied to each selected sub-blocks of the original image used for watermarking. Counting the number of edge points in each sub-block, we obtain another visual mask, $mask_2(j,k)$. This mask is used to determine the coarse texture or outstanding edge in the image block. Large values in this mask indicate that the corresponding block is highly textured. Smaller values indicate that the block contains outstanding edges [14]. The two mask values are multiplied and scaled to a specific range. The watermark strength factor α is obtained as follows:

$$\alpha = \beta * mask_1(j,k) * mask_2(j,k) \tag{9}$$

Where β is the scaling factor, *j* and *k* indicate the positions of the sub-blocks.

The watermark strength factor α can be adaptively controlled according to the texture areas. High textured areas are watermarked with higher strength. Outstanding edge areas and smooth areas are watermarked with less strength.

In this way, the invisibility of the watermarked image can be improved. The watermark strength factor for the sub-blocks of image, lenna.bmp, is shown in figure 3.

In figure 3, the light areas represent coarse texture areas watermarked with higher strength than the dark areas that represent smooth areas. For this algorithm, a maximum image size of 256×256×8 bits can be hidden into a container image of size 512×512×8 bits. For our investigation, a watermark grayscale image, dmt.bmp, of size 64×64×8 bits is used.

5 Experimental Results

Results showed that there were no perceptually visible degradations on the watermarked images. The extracted watermark was also highly correlated with the original watermark with correlation factor of approximately 0.989. Test results using Stirmark 3.1 on the Lenna image are shown Table 1. Some sample attacks are illustrated in figure 4.

The proposed FHT watermarking algorithm was able to survive up to 60% against all the Stirmark attacks. It was robust against jitter attacks. Cropping attacks up to 50% of the watermarked image could be resisted. By incorporating post-processing techniques, the algorithm correctly retrieved the watermark against upscaling to twice the size of the watermarked image, downscaling to approximately 75%, changing of aspect ratios either in the x-axis or y-axis and small angle rotation attacks.

The proposed algorithm was also able to resist frequency mode Laplacian removal (FMLR) and 3×3 sharpening attacks. It survived some level of JPEG compression, to a compression factor of 30. However, It failed for large cropping at 75. The algorithm did not perform well against large factor of scaling-down attacks for factor 0.5. Even with some post-processing applied against this kind of attacks, the scaling-down attacks introduced a significant amount of degradation due to averaging of the original pixel values. It also performed relatively poorly against 3×3 Gaussian filtering, 2×2 median filtering and low factor JPEG compression (factor < 30). It was also not so effective against minor random geometric transforms, such as shearing and general linear transforms.

This algorithm was very efficient in term of processing time. It took approximately 2 seconds for the embedding process and approximately 1 second for extraction using a MATLAB 6 platform running on a Pentium III 400 MHz PC system. The simplicity of FHT also offered an advantage over the commonly used DCT and DWT techniques in terms of ease of hardware implementation.

6 Conclusion

This paper has presented a robust hybrid watermarking technique for embedding characters or grayscale image watermark into a container image based on the FHT. The embedding and extracting processes have been described in detail. In the proposed method, the embedding scheme takes the spatial information into consideration, and generates the watermark strength factor according to the visual mask. This increases the invisibility of the watermark in the watermarked image. The experimental results show that the proposed method is robust against approximately 70% of Stirmark attacks.

The Hadamard transform has more useful middle and high frequency bands than several high gain transforms, such as DCT. When compared with the DCT, it found to be more robust against various attacks. It also offers a significant advantage in terms of a shorter processing time and the ease of hardware implementation than many common transform techniques.

References

1. Miller, M. Cox, I.J., Bloom, J., “Watermarking in the Real World: An Application to DVD”, Proc. Wksp. Multimedia and Security at ACM Multimedia 98, Bristol, U.K., Sept. 1998.
2. Van Schyndel, R.J., Tirkel, A.Z., Osborne, A.F., “A digital watermark,” Proc. IEEE Int. Conf. Image Processing, vol. 2, 86–90, 1994.
3. Cox, I.J., Kilian, J., Leighton, F.T., Shamoon, T., “Secure spread spectrum watermarking for multimedia,” IEEE Trans. Image Processing, vol.6, 1673–1687, Dec. 1997.
4. Hsu, C.T., Wu, J.L., “Hidden digital watermarks in images,” IEEE Trans. Image Processing, vol. 8, pp. 58–68, Jan. 1999.
5. Joseph, J.K., O’ Ruanaidh, Pun T., “Rotation, Scale and Translation Invariant

Digital Image Watermarking”, Signal Processing, Vol. 66, No. 3, 303-317, 1998

6. Wei, Z.H., Qin, P., Fu, Y.Q., “Perceptual digital watermark of image using wavelet transform”, IEEE Trans. on Consumer Electronics, Vol. 44, No. 4, 1267-1272, Nov. 1998.
7. Dugad, R., Ratakonda, K., Ahuja, N, “A new wavelet-base for watermarking image,” in Proc. Int. Conf. Image Processing, vol. 2, 1998, 419–423.
8. Hsu, C.T., Wu, J.L., “Multiresolution watermarking for digital images,” IEEE Trans. Consumer Electron., vol. 45, 1097–101, Aug. 1998.
9. Hall, E.H., “Computer Image Processing and Recognition,” New York: Academic Press, 1979.
10. Gonzalez, R.C., Wintz, P, “Digital Image Processing,” Reading, MA: Addison-Wesley, 1977.
11. Kankanhalli, M.S., Rajmohan, Ramakrishnan, K.R., “Content Based Watermarking of Images,” ACM Multimedia 98 - Electronic Proceedings, Sept 14-16, 1998
12. Torres, L., Kunt, M., “Video Coding, The Second Generation Approach,” Kluwer Academic Publishers, 1995.
13. Canny, J., “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 8, No. 6, Nov 1986
14. Taskovski, D., Bogdanova, S., Bogdanov, M., “A Low Resolution Content Based Watermarking of Image in Wavelet Domain,” Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on, 200,1 604-608

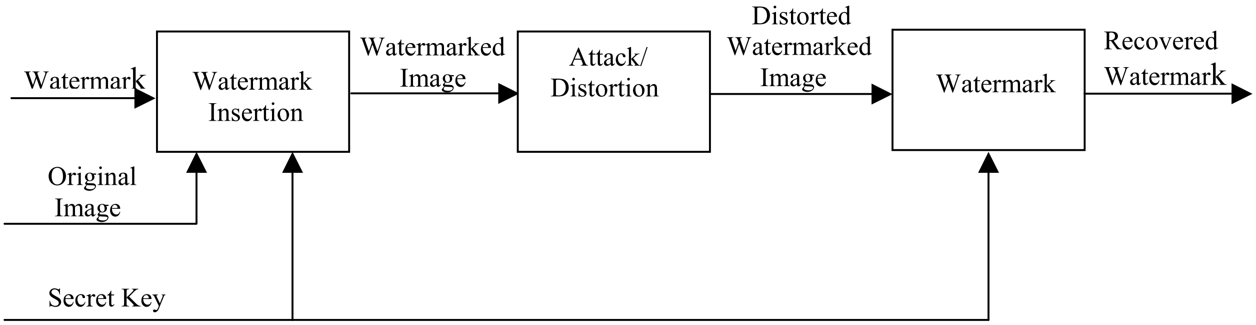
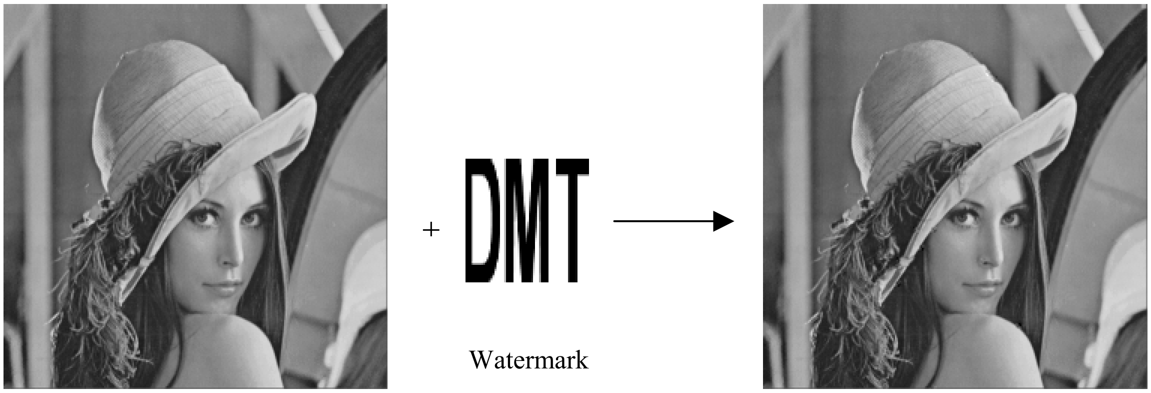


Figure 1: Block diagram of “blind” watermarking system



Watermark

Original image

Watermarked image

Figure 2: image-in-image watermarking embedding process

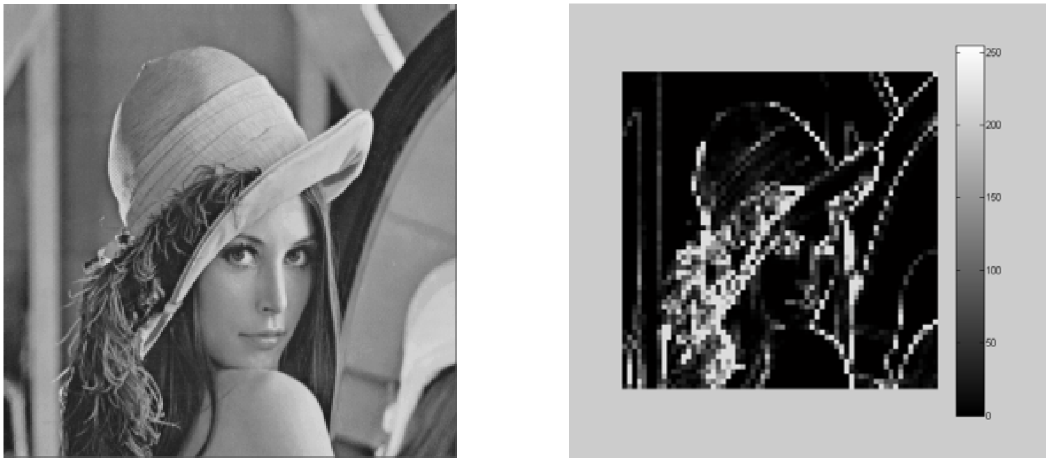
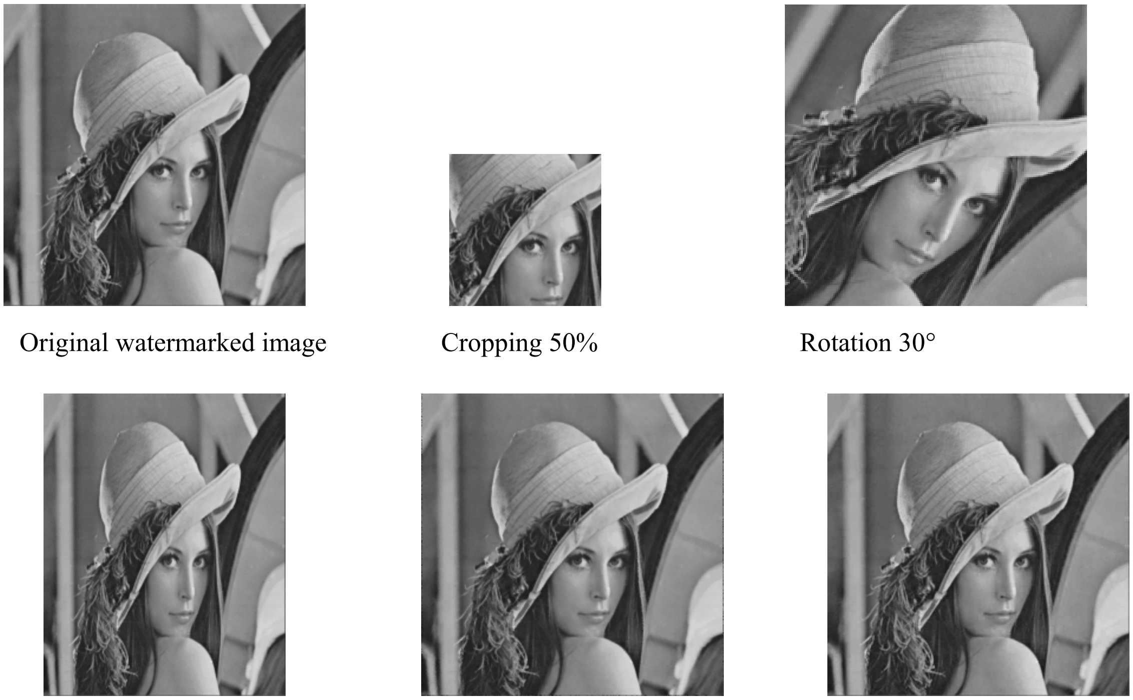


Figure 3: original lenna image and the strength level for each 8×8 sub-block



Original watermarked image

Cropping 50%

Rotation 30°

Changing aspect ratio
x-1 y-0.8

3×3 median filtering

JPEG compression
factor 35

Figure 4: Stirmark attack examples on watermarked image lenna.bmp

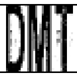
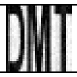
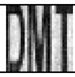
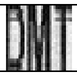
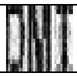
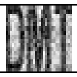
Image operations	**Extracted watermark**	**Correlation**
Sharpening 3×3		0.9573
1 rows 1 column removed		0.9866
Frequency Mode Laplacian removal		0.9580
Scaling 0.75		0.9354
JPEG Compression of factor 30		0.8688
Change aspect ratio x_1.00_y_1.20		0.8199

Table 1: Results of some Stirmark tests for image-in-image embedding algorithm