

A Method for Image Recovery in the DFT Domain

Peining Tao^a and Ahmet M. Eskicioglu^b

^a*The Graduate Center, The City University of New York
365 Fifth Avenue, New York, NY 10016*

^b*Department of Computer and Information Science, Brooklyn College
The City University of New York, 2900 Bedford Avenue, Brooklyn, NY 11210
ptao@gc.cuny.edu, eskicioglu@sci.brooklyn.cuny.edu*

Abstract

In image authentication research, a common approach is to divide a given image into a number of smaller blocks, and embed a fragile watermark into each and every block. The modifications can therefore be detected in the blocks that have been tampered with. The literature includes many authentication techniques for detecting modifications only. In this paper, we propose a method for recovering the damaged blocks using the magnitudes of DFT coefficients. If a given block is considered to be damaged, we divide it into 2×2 blocks, and replace the magnitude of the DFT coefficient $F(0,0)$ with the corresponding magnitude of the original image. As the $F(0,0)$ coefficients are always real, we quantized them and round them off. The file containing all of these coefficients are sent from the sender to the receiver using a public key scheme. As an image authentication system, we will use the scheme proposed by Wong and Memon. In their proposal, the image is partitioned into $I \times J$ blocks, and a watermark is inserted into each block.

1. Introduction

Image authentication is the process of verification of the genuineness of an image in order to establish its full or partial conformity with the original image. Such authentication systems have wide applicability in law, commerce, journalism, and national defense.

Cryptography and fragile watermarking are two approaches that can be used to guarantee integrity of multimedia data including images. Depending on the application, three types of cryptographic mechanisms are alternatives:

- *Full encryption*: Encrypting the entire message with a symmetric key cipher would provide both confidentiality and authentication [1].

- *Hashing*: As a given message can be of arbitrary length, the process called *hashing* is an essential part of most data integrity and message authentication methods. A hash function takes a message of arbitrary finite length and produces an output of fixed length [1].
- *Digital signatures*: A cryptographic primitive which is essential in authentication is the digital signature. The purpose of a digital signature is to provide a means to associate a message with some originating entity [1].

A major drawback of the above cryptographic mechanisms is that they do not permanently associate cryptographic information with the content. Digital watermarking has a number of applications. Some important examples are copyright protection, access control, broadcast monitoring, and content authentication. A digital watermark is a pattern of bits inserted into a multimedia element such as a digital image, an audio or video file. A fragile watermark is an embedded signal that can be easily altered or destroyed when the cover image goes through even the slightest modification. It can be used in applications where it is important to determine how the digital content was modified or which portion of it has been tampered with. For digital images, a common approach is to divide a given image into a number of smaller blocks, and embed a fragile watermark into each and every block.

In [2], multimedia authentication is classified into hard authentication and soft authentication. Hard authentication rejects any modifications to multimedia content. The only manipulation accepted by the hard authentication is lossless compression or format conversion that preserves visual pixel values or audio samples. Soft authentication passes certain content modifying, called incidental or admissible manipulations, and rejects all the rest, called malicious manipulations.

Most of the algorithms proposed for hard authentication are based on fragile watermarking so the authenticator is embedded into the signal to be authenticated to simplify bookkeeping and maintenance of authenticators. In fragile watermarking, the inserted watermark is so weak that any manipulation to the multimedia content disturbs its integrity. Tampered parts of the multimedia signal may be located by checking the presence and integrity of the local fragile watermark. The authors describe three major hard authentication schemes for images:

- Single pixel/sample authentication
- Block authentication
- Lossless watermarking

Two types of soft authentication algorithms are described:

- Quality-based authentication
- Content-based authentication

2. Image authentication and recovery

2.1 Image authentication scheme by Wong and Memon

In this paper, we will consider the secret key image authentication scheme proposed by Wong and Memon [3]. In their proposal, they consider a gray scale image $x_{m,n}$ of size $M_X \times N_X$ pixels. They insert an invisible watermark to create a watermarked image $x_{m,n}^w$ of the same size. $x_{m,n}$ is partitioned into blocks of $I_X J_X$ pixels, and an invisible watermark is embedded into each block. The watermark insertion procedure is shown in Figure 1.

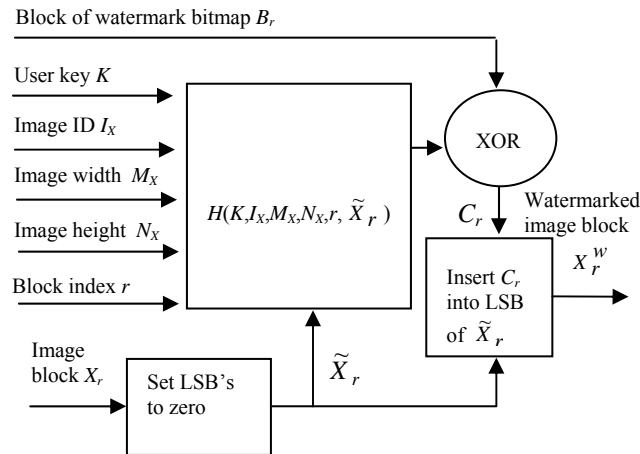


Figure 1. Block diagram of inserting a secret key authentication watermark

Suppose $a_{m,n}$ is a binary image to be used as an invisible watermark. If $a_{m,n}$ does not have the size of $x_{m,n}$, another binary image $b_{m,n}$ of size $M_X \times N_X$ is formed. Let

$$X_r = \{x_{il+k,jl+l}; 0 \leq k \leq I-1; 0 \leq l \leq J-1\}$$

be a block of size $I \times J$ taken from the image $x_{m,n}$. A single index r is used to denote the r th block in the image. The notation for the corresponding block within the binary image $b_{m,n}$ is

$$B_r = \{b_{il+k,jl+l}; 0 \leq k \leq I-1; 0 \leq l \leq J-1\}.$$

$H(S) = (d_1, d_2, \dots, d_p)$ represents a cryptographic hash function that takes a string S as input and produces a string of length p . The hash function used in [3] is MD5 [4], which is used in numerous security applications. As input, the algorithm takes a message of arbitrary length and produces a 128-bit “message digest” of the input.

Let K denote a user key consisting of a string of bits. For each block of data X_r , the corresponding block \tilde{X}_r is formed. Each element in \tilde{X}_r is equal to the corresponding element in X_r except that the least significant bit is set to zero. For each block, the hash

$$H(K, I_X, M_X, N_X, r, \tilde{X}_r) = (d_1^r, d_2^r, \dots, d_p^r)$$

is computed. The parameters I_X and r are very important for resistance to the vector quantization attack [5].

The extraction procedure for the invisible watermark is shown in Figure 2. The least significant bit of each element in the block Y_r is set to zero to obtain the block \tilde{Y}_r . For each block of data, $H(K, I_Y, M_Y, N_Y, r, \tilde{Y}_r)$ is computed, and the exclusive OR operation is performed with G_r to output a block of the binary watermark.

If the watermarked image is not modified, we have $X_r^w = Y_r$, $I_X = I_Y$, $M_X = M_Y$, and $N_X = N_Y$, implying $\tilde{Y}_r = \tilde{X}_r$, $C_r = G_r$, and $Y_r^o = B_r$. If the image is modified by changing the pixel values, changing the size of the image, or if the correct key is not used, the extracted watermark bit map will resemble random noise.

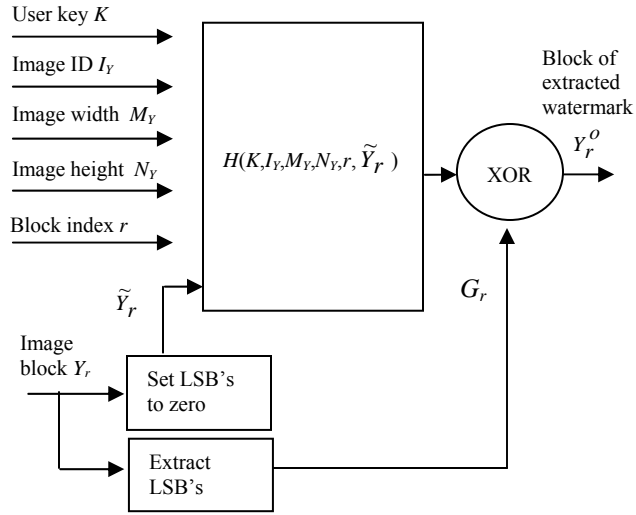


Figure 2. Block diagram of extracting a secret key authentication watermark

The properties of the invisible watermark are discussed in [3]:

- The degree of distortion introduced by the watermark is not visible.
- If the correct key is used, the extracted watermark is identical to the original.
- If an image is not watermarked, the extracted watermark resembles random noise.
- If an incorrect key is used, the extracted watermark resembles random noise.
- If a watermarked image is cropped, the extracted watermark resembles random noise.
- If certain pixels values are changed in a specific location, the watermark extraction procedure will detect the changes at this location.
- The watermark bits are embedded into the least significant bits of the image. If there is an attempt to remove the invisible watermark by changing some bit planes, the watermark extraction procedure will detect the changes.
- It is not possible to alter an image block in a way to keep the embedded watermark unchanged. Because of the properties of cryptographic hash functions, it is computationally difficult to find two inputs that hash to the same output.

2.2 A new image recovery scheme

Our proposal for recovering the damaged blocks is to use the magnitudes of DFT coefficients. If a given block is damaged because the extracted watermark resembles random noise, we divide it into 2×2 blocks, and replace the magnitude of the DFT coefficient

$F(0,0)$ with the corresponding magnitude of the original image. The other three magnitudes are made zero. As the $F(0,0)$ coefficients are always real, we quantize them and round them off. This file is then sent them from the sender to the receiver using RSA, a public key scheme [6].

Our target is to reduce the size of the file that contains the magnitudes of DFT coefficients. As the size of each test image is $n \times n$, we have $n/2 \times n/2$ magnitudes of the DFT coefficients. Consider the following algorithm to reduce the size of the file:

- Create a vector with the length $n/2 \times n/2$ containing all DFT coefficient magnitudes.
- Scan the vector, and generate different sets of values. In each set, the difference between maximum and minimum is at most 14. 15 is reserved to separate these sets.
- Obtain a string of 0 and 1 bits, containing different sets. In each set, the minimum remains unchanged while the others become the distances from the minimum.

The protocol is described as follows (where the bit string is consisted of three parts). Let us consider a 256×256 gray scale image.

Part I:

Size: 16 bits

Content: the number of groups

Part II:

Size: $128 \times 128 \times 4$ + the number of groups \times 4 bits

Contents: a long string of bits

Every 4 bits represent a number showing the distance to the minimum value in a group. The minimum distance is 0 (0000), and the maximum distance is 14 (1110). The number 15(1111) is reserved for a special purpose. Each group contains two 1111's: the first 1111 indicates the position of the minimum value in the group, and the second 1111 indicates the end of a group.

Part III

Size: the number of groups \times 7 bits

Contents: a long string of bits

Every 7 bits represent a number showing the minimum value in one group.

Example: Suppose we have three sets of data.

- $\{33 \ 1 \ 10\}$, where 33 is the minimum value, 1 and 10 are distances from the minimum.

- {3 8 14 14}, where the first 14 is the minimum value, and the other values are distances from the minimum.
- {79 5}, where 79 is the minimum value, and 5 is the distance from the minimum.

The resulting string is:

```
00000000000000111111000110101111001110001111
11101111111101011111010000100011101001111
```

It can be divided into three parts:

Part I:
0000000000000011 (16 bits, 3 groups)

Part II:
This long string can be split into three groups, each ending with {1111}:

Group 1: 1111000110101111
Group 2: 0011100011111101111
Group 3: 111101011111

Part III:
010000100011101001111

3. Experiments

The two 256x256 gray scale test images (Boat and Cameraman) are shown in Figure 3. Each 16x16 block of the binary watermark represents the tower of the library at Brooklyn College.

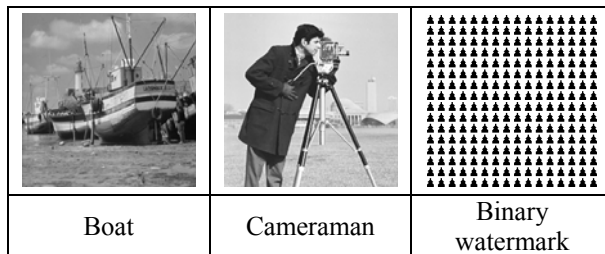


Figure 3. Test images

3.1 Watermark detection

The watermarked versions of both images and their PSNR values are shown in Figure 4.

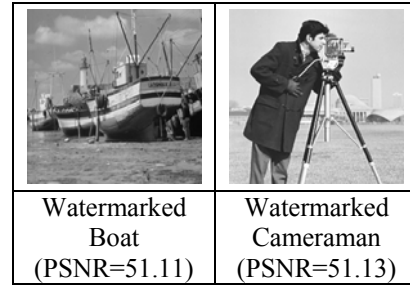


Figure 4. Watermarked images

We performed several experiments on each image. Because of its relevance, we will present the results for changing the pixel values of the watermarked test images.

Figure 5 shows the modifications in both images. A flying seagull was added to Boat, and a rabbit was added to Cameraman. As each image is modified in a specific location, the extracted watermark resembles random noise at that location.

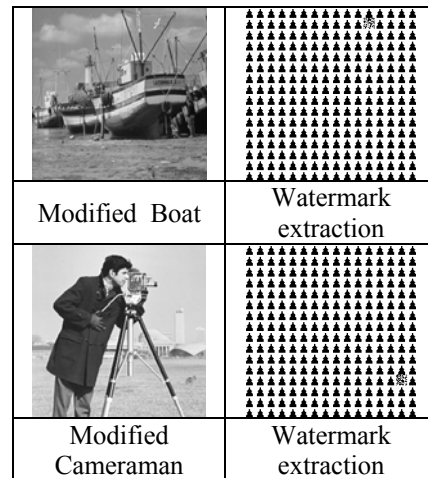


Figure 5. Using a modified image in watermark extraction

3.2 Image recovery

The size of the modified block in each image is 16x16. As it is not possible to estimate which blocks have been modified, all the DFT coefficients $F(0,0)$ are sent from the sender to the receiver in a protected way.

Because the size of each test image is 256x256, we have 128x128 magnitudes of the DFT coefficients. After quantization and rounding off, the data for both images is given in Table 1.

Table 1. Data for test images

	DFT coefficient magnitudes		# of groups	Size of file (B)	% of actual image size
	Min	Max			
Boat	4	95	2512	11684	18
Cameraman	12	102	1718	10557	16

Figure 6 shows each recovered image whose one block was damaged.

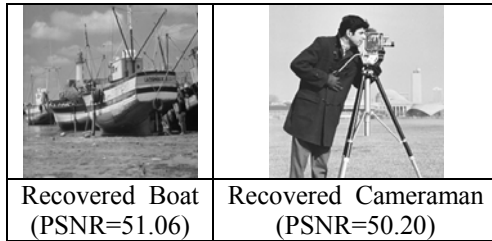


Figure 6. Recovered image

4. Conclusions

We presented a new scheme for image recovery. Our approach is to recover the damaged blocks by using the magnitudes of DFT coefficients of the original image.

As an example of image authentication scheme, we took the algorithm proposed by Wong and Memon. If a given block is damaged because the extracted watermark resembles random noise, we divide it into 2x2 blocks, and replace the magnitude of the DFT coefficient $F(0,0)$ with the corresponding magnitude of the original image. After quantization and rounding off, we send these magnitudes from the sender to the receiver using RSA.

The Wong and Memon algorithm is an image authentication algorithm, which simply detects the modifications made in an image. When our proposed scheme recovers the damaged blocks, some of the properties of the invisible watermark described in [3] will change.

Note that the actual size of each gray scale test image in our experiments is 65 KB. The additional transmission load is less than one fifth of the actual load.

The recovery of the damaged block is excellent, the PSNR value of the entire image being very high.

In future work, we will compare our algorithm with other image recovery schemes.

5. References

- [1] J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
- [2] B. B. Zhu, M. D. Swanson, and A. H. Tewfik, "When seeing isn't believing," *IEEE Signal Processing Magazine*, March 2004.
- [3] P. W. Wong and N. Memon, "Secret and public key image watermarking schemes for image authentication and ownership verification," *IEEE Transactions on Image Processing*, 10(10), October 2001.
- [4] R. L. Rivest, "Network Working Group Request for Comments: 1321," April 1992.
- [5] M. Holliman and N. Memon, "Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes," *IEEE Transactions on Image Processing*, 9(3), March 2000.
- [6] Available at <http://www.rsasecurity.com/>