A Hierarchical Approach to SVD-based Fragile Watermarking for Image Authentication

Sima Taheri*, Shahrokh Ghaemmaghami†
Department of Electrical Engineering
Sharif University Technology, Tehran, Iran
*Email: sima.taheri@mehr.sharif.edu
†Email: ghaemmag@sharif.edu

Abstract—One of the main properties of fragile watermarking schemes is to indicate the location of changes made to the image, i.e. tamper localization, in addition to detection of watermarked image modifications. In this paper, the localization property is added to the SVD(Singular Value Decomposition) based fragile watermarking scheme proposed by Byun et al. To do this, a hierarchical watermarking structure is used, where the image is partitioned into blocks in a multilevel hierarchy, and then the SVD-based algorithm is employed at each level to insert authenticating data in sub-blocks. In the verification process, blocks at the lowest hierarchical level can detect modifications made to the watermarked image, while the higher level blocks locate these modifications. It is also shown that the computational complexity of the proposed method can be reduced to that of the Byun’s method, through image partitioning. Experimental results are given to demonstrate effectiveness of our algorithm.

Keywords—Fragile watermarking, singular value decomposition, block-based authentication, tamper localization.

I. INTRODUCTION

A typical characteristic of digital products is that they are easy to manipulate, i.e. to store, duplicate, transmit, or modify. This is a critical issue in return, as unauthorized use, copying, or modification of the multimedia products could be quite easy as well. Such kinds of operations are referred to as tampering and techniques are needed to make the multimedia contents tamper resistant and guarantee integrity and originality of the image. The most common means to defeat tampering is to embed a fragile watermark into the image to identify and localize any possible image alterations.

Many schemes are proposed for fragile watermarking that are able to detect modifications made to watermarked images, but most of them lack the localization property. Tampering localization would help to find out: i) the motive for tampering; ii) possible candidate adversaries; and iii) whether the alteration is legitimate [1]. However, attempts have been made to improve watermarking algorithms to cope with both identification and localization problems.

Wong and Memon [5] introduced a secret and public key fragile image watermarking method for grayscale image authentication, which embeds a digital signature of the image block MSBs into the LSBs of the same block. Despite the elegance of the algorithm and cryptographic security of the digital signatures, its blockwise independence was exploited by Holliman and Memon for a counterfeiting attack [3]. The attacker constructs a vector quantization codebook, using blocks from a set of watermarked images, to approximate the image to be counterfeited. Since each block is authenticated by itself, the counterfeit image appears authentic to the watermarking algorithm [2].

Since the introduction of VQ codebook attack, a number of modifications to the existing algorithms have been proposed [3,6]. Most of these modifications sacrifice localization accuracy to improve security. Celik et al. [2] proposed a method that thwarts the VQ attack, while sustaining the superior localization properties of the Wong’s method. They split the image into blocks in a multilevel hierarchy and then calculate block signatures in this hierarchy. While signatures of small blocks at the lowest level of the hierarchy ensure high accuracy of tamper localization, higher level block signatures improve resistance to VQ attacks. At the highest level, a signature calculated using the whole image completely thwart the counterfeiting attack [2].

Byun et al. [1] proposed a SVD-based fragile watermarking scheme for image authentication. The basic idea is to extract authenticating data from the original image. To build the authenticating data, the singular values are converted to the binary bits using modular arithmetic. Only a few bits of authenticating data are embedded into the image, so the quality of watermarked image remains very high. This scheme can detect watermarked image modifications, but cannot localize the alterations. They also made an extension to their algorithm to acquire localization capability, which locates modifications only if the LSB of watermarked image is changed.

In this paper, we add localization property to Byun algorithm [1], based on the idea of hierarchical watermarking given by Celik [2]. Image is partitioned into blocks in a multilevel hierarchy. At the highest level of hierarchy, the image is divided into a set of blocks that are created through grouping of image pixels. The authenticating data is then constructed by computing the SVD of each level’s sub-block.

These singular values are converted into binary bits, as instructed by Byun algorithm. The resulting binary vector is incorporated into LSBs of the selected pixels within the block. The selection of pixels for embedding the vectors is made in a manner to prevent collision. In watermark extraction process,
large blocks of the lowest hierarchical level can detect any alterations to the watermarked image, where the higher-level blocks show locations of these alterations.

The rest of the paper is organized as follows. In section 2, we discuss Byun’s original scheme and its extension algorithm with localization property. Our method is proposed in section 3 and the experimental results and an analysis of computational complexity of the proposed algorithm are presented in section 4. The paper is concluded in section 5.

II. SVD BASED FRAGILE WATERMARKING

A SVD based fragile image watermarking scheme is introduced in [1] for image authentication. In this scheme, SVD of an $M \times N$ image is computed. Then, authenticating data is built by changing the singular values to the binary bits using modular arithmetic. These binary bits are inserted into the LSBs of randomly selected pixels of the original image.

A. SVD

The singular values of a matrix are computed as:

$$A = USV^T$$

where $A$ is a real $M \times N$ matrix, $U$ and $V$ are orthogonal matrices of dimensions $M \times M$ and $N \times N$, respectively, $S$ is a diagonal $M \times N$ matrix of singular values, and $T$ is for transposition. The matrix $S$ can be presented as:

$$S = \text{diag}(\sigma_1, \ldots, \sigma_k), k = \min(M, N)$$

$$\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \ldots \geq \sigma_{k-1} \geq \sigma_k$$

B. Watermark Insertion

LSBs of the N randomly selected pixels in $M \times N$ image are set to zero. The singular values, $S$, for this image are multiplied by a multiplying factor, $\alpha$, and then are set to the floor integer values. The modular arithmetic is used to generate binary bits, $B$, and the results are inserted into the LSBs of randomly selected pixels to get the watermarked image (see Fig. 1).

C. Watermark Extraction and Verification

The key used for selecting pixels in embedding process is exploited at this level to extract LSBs from the pixels. Feature information is computed using the same method employed in the insertion process. Then, the LSB string is compared to the authenticating data computed (see Fig. 2). The advantages of this scheme are that modifications to watermarked images can be detected, where the quality of watermarked images remains very high, because only a few bits of authenticating data are embedded into the image.

D. Extension algorithm for localization

The above-mentioned scheme does not have the property of tamper localization. So, an extension algorithm for localization was proposed. In this extension, a binary logo image $W$ of size $J \times K$ is used for watermark. Authenticating data is generated as described in section 2.2. The authenticating data and watermark are tiled to get the same size as that of the original image, and then are combined with each other using a bit-by-bit XOR operation. The result is embedded into the LSB of randomly selected pixels by a key (see Fig. 3).

In the extraction process, a bit-wise XOR operation is performed between tiled authenticating data and the extracted bits to form the binary watermark. This scheme can locate the alterations made to the images, only if LSBs of watermarked images are changed.

III. PROPOSED METHOD

We propose a hierarchical approach to the localization problem to obviate the LSB restriction described earlier. As well as the typical property of the alteration detection of the watermarked image, our algorithm provides a better localization feature, as compared to earlier methods. This is achieved by
dividing the image into blocks in a multilevel hierarchy, where each block is then watermarked using Byun’s method. We first introduce our method, then watermark insertion procedure is described, and finally extraction process is presented in this section.

A. Blockwise hierarchical watermarking

We partition the image hierarchically where, at the highest level of hierarchy, the image is divided into blocks in a non-overlapping manner. In each level, every block is composed of a number of the next level sub-blocks (see Fig. 4).

In an area of $4 \times 4$ blocks at a given level of hierarchy, the blocks are combined together to create a block at the level next to the current level. We perform this method for $M \times M$ images, B, in which $M = 2^h$. Any arbitrary image should be resized to be a power of 2 in each dimension, by adding zeros.

$B_{ij}^h$ denotes a block in this hierarchy, where the indices $i,j$ represent the spatial position of the block and $h$ is the level of the hierarchy, to which the block belongs. So, we have the following notation:

$$ \begin{bmatrix} B_{ij}^{h+1} & B_{ij}^{h+1} & \cdots & B_{ij}^{h+1} \\ B_{ij+1}^{h+1} & B_{ij+1}^{h+1} & \cdots & B_{ij+1}^{h+1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{ij+2^{h-1}}^{h+1} & B_{ij+2^{h-1}}^{h+1} & \cdots & B_{ij+2^{h-1}}^{h+1} \end{bmatrix} = B_{ij}^h $$

At the lowest level, we partition the image B into $P \times P$ non-overlapping blocks, $P$ should be chosen based on the desirable accuracy required for tamper localization. To obtain the highest accuracy, $P$ must be 2 or 4, depending on M. If complexity is of more concern, $P$ may be chosen larger instead. Finally, the highest level of hierarchy consists of only one block, $B$. The total number of levels in the hierarchy is given as:

$$ H = \frac{1}{2} \log_2 \frac{M}{2} + 1 $$

B. Watermark insertion

In this part of algorithm, the authenticating data, i.e. watermark, is extracted from each hierarchy block on the basis of Byun’s method. The procedure is as follows:

1) Form the block $\tilde{B}_{ij}^h$ by setting the LSBs of each pixel in $B_{ij}^h$ to zero.
2) Compute singular values (SVs) of the block $\tilde{B}_{ij}^h$ and then generate binary bits, as described in section 2, to construct the authenticating data. The length of output vector of each block is the same as the block’s dimension. We call this vector $W_{ij}^h$.
3) These binary bits, $W_{ij}^h$, are inserted into the certain pixel LSBs of the same block.

The procedure is shown in Fig 5. The selected pixels for insertion are chosen based on collision avoidance. In other words, since each level of hierarchy shares several pixels at other levels, the pixels are to be chosen deliberately. Overall, the method can be described as follows:

In the $h^{th}$ level of hierarchy, the size of $B_{ij}^h$ is $N \times N$ where $N = 2^h$ and its output vector, $W_{ij}^h$, is of size N. The size of original image is $M \times M$ and $M = 2^h$. If k is even, the locations of certain pixels are given as:

$$ (2^\frac{k}{2} m + 2^\frac{k}{2} c - 1, 2^\frac{k}{2} n + 2^\frac{k}{2} + 1 - 1) \quad m, n = 0, \ldots, 2^\frac{k}{2} - 1 $$

Otherwise, when k is odd, we have:

$$ (2^\frac{k-1}{2} m + 2^\frac{k-1}{2} c - 1, 2^\frac{k-1}{2} n + 2^\frac{k-1}{2} + 1 - 1) \quad m = 0, \ldots, 2^\frac{k-1}{2} - 1, n = 0, \ldots, 2^\frac{k-1}{2} - 1 $$

It can be shown that there is no collision with these locations. Fig. 6 shows a piece of the original image with the selected locations and illustrates why no pixel has collision. Based on this structure, we can keep the vectors of the block at each level localized inside the corresponding block. So, pixel manipulations outside a block do not affect the vector and, therefore, the verification of the particular block.

Finally we obtain the watermarked image $B_{ij}^w$. We should remove the additional parts of the watermarked image, if the image is zero added, in order to get the proper size.
C. Watermark Verification

This part of our algorithm is designed to detect modifications made to the watermarked image. Fig. 7 shows the watermark extraction and verification procedure for one hierarchical level.

Hierarchical block structure is formed based on the procedure described in section 3.1. We begin the watermark verification process from the lowest hierarchical level, i.e., $h = 1$. At each level, if the extracted vector from block $B^H_0$ does not conform with the compared vector, then we check the next level of hierarchy. In other words, we first extract the vector $W^h$, regarding $B^h_m$, from the certain pixel LSBs, and then $W^h_1$ is obtained according to the steps given in section 3.2. Comparison between $W^h_1$ and $W^h_2$ indicates whether the watermarked image is authentic. If $W^h_1$ does not equal $W^h_2$, blocks at the next hierarchical level are checked. We repeat this procedure until highest level, i.e., $h = H$, that the smallest blocks locate the manipulations.

In the case of watermarked image of an arbitrary size, we add some zeros to get the proper size, just for computing the $W^H_{i,j,m}$ vectors. This means that we still verify only the blocks belonging to the watermarked image.

IV. RESULTS AND ANALYSIS

In this section, we demonstrate the effectiveness of our method based on experiments conducted and discuss the computational complexity of the proposed algorithm.

A. Experimental Results

We used gray images of size $512 \times 512$ as inputs in our simulations. We demonstrate both the location and tamper detection abilities of our algorithm for various amounts of multiplying factor $\alpha$. During the mark embedding process, the $512 \times 512$ gray-scale image has been modified into a 4-level hierarchy with $8 \times 8$ blocks at the highest level.

$\text{Fig. 5. Watermark insertion process in the proposed method}$

$\text{Fig. 6. A} 16 \times 16 \text{ block of watermarked image with locations of certain pixels.} \text{ As shown, no pixel has collision.}$

$\text{Fig. 7. Watermark extraction and verification procedure for one hierarchical level}$

$\text{Fig. 8. Original and watermarked images that indicates no perceptible difference between the two images using subjective criteria.}$

We can see that higher efficiency is achieved with a larger $\alpha$.

B. Computational Complexity

The computational complexity of this algorithm depends on the number of SVD operations required. We compare the complexity of our proposed algorithm to the Byun's method. In the Byun's method, we compute the SVD of an $M \times M$ matrix.

$\text{This is calculated as} \frac{H^2}{2} \sum_{i=0}^{H-1} \left( \frac{M}{2^i} \right)^2 \left( \frac{M}{2^{i+1}} \right)$
matrix, where $M$ is the dimension of the original image. So, the computational complexity of this method is equal to the complexity of the SVD of an $M \times M$ matrix. QR-based method is a well-known tool for computing the singular values of a matrix. For a full matrix, $A \in \mathbb{R}^{n \times n}$, the QR iteration requires $O(n^3)$ flops [4]. Since, QR decomposition is the main part of SVD computation, we assume that the complexity of SVD is in order of $O(n^3)$.

$$C_{SVD} = f(M) \rightarrow O(M^3) \quad (7)$$

In the proposed method, the computational complexity varies based on the number of hierarchical levels and blocks at each level. Assuming that the original image is presented by an $M \times M$ matrix and the highest hierarchical blocks are of $P \times P$, since we have to calculate the SVD of each block at each level, the computational complexity is obtained as (see table 1).

$$C_h = \sum_{i=0}^{H-1} \left[ 16^{-1} \left( \frac{M}{P} \right)^2 f \left( \frac{M}{2^i(H-1-i)} \right) \right]$$

$$= f(M) + \sum_{i=0}^{H-2} \left[ 16^{-1} \left( \frac{M}{P} \right)^2 f \left( \frac{M}{2^i(H-1-i)} \right) \right] \rightarrow O \left( \sum_{i=0}^{H-1} 16^{-1} \left( \frac{M}{P} \right)^2 f \left( \frac{M}{2^i(H-1-i)} \right) \right)$$

This shows a higher complexity of our algorithm, as compared to Byun’s. To reduce the complexity, which makes sense in case of large images, we can partition the image into a number of blocks in the lowest, $h = 1$, hierarchical level. So, we have:

$$C_{h,x} = \sum_{i=0}^{H-1-x} \left[ 16^{-1} \left( \frac{M}{P} \right)^2 f \left( \frac{M}{2^i(H-1-i)} \right) \right] \quad (9)$$

Fig. 7. Watermark verification process for the proposed method.

<table>
<thead>
<tr>
<th>Variable description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
</tr>
<tr>
<td>$p$</td>
</tr>
<tr>
<td>$f(M)$</td>
</tr>
<tr>
<td>$H$</td>
</tr>
<tr>
<td>$l$</td>
</tr>
<tr>
<td>$C_h$</td>
</tr>
<tr>
<td>$16^{-1}(M/P)^2$</td>
</tr>
<tr>
<td>$M/2^6(H-1-l)$</td>
</tr>
<tr>
<td>$x$</td>
</tr>
</tbody>
</table>

A comparison between complexity of various methods for an $512 \times 512$ image is shown in table 2.

**V. Conclusion**

In this paper, we have presented a hierarchical fragile image watermarking scheme based on method introduced by Byun et al. [1]. We provide the Byun’s method with localization properties at the expense of higher computational complexity. To reduce the complexity of our method, we partition the image into a number of blocks at the lowest hierarchical level. The advantages of this scheme is that it can indicate the location of the changes made to the image. It has been shown that a higher accuracy in tamper localization can be achieved by allowing for higher computational complexity.

**REFERENCES**


Fig. 8. (a) Original image and (b) watermarked image with $\alpha = 10$ (PSNR=32.13 dB).

Fig. 9. (a) Tampered image and (b) the image underwent the detection process.
Fig. 10. (a) Original image, (b) watermarked image, (c) tampered image, and the image after the detection process with (d) $\alpha = 0.0001, 0.001$, (e) $\alpha = 0.01$, and (f) $\alpha = 10, 100$.