Wireless Personal Communications (2022) 122:211–227
https://doi.org/10.1007/s11277-021-08895-1

Fragile Watermarking Based on QR Decomposition and Fourier Transform

Fatemeh Nejati[1] · Hedieh Sajedi[2] · Alireza Zohourian[2]

Accepted: 8 August 2021 / Published online: 22 August 2021

Abstract

In this study, a fragile watermarking technique is introduced for verifying images based on QR decomposition and Fourier Transform (FT). At first, we apply the FT to the host image to achieve the frequency domain, yielding a high-quality image. Then the resulting image is decomposed using QR factorization. In the meantime, the watermark image is decomposed only via QR factorization. Then, we add a coefficient of matrix R from the watermark image to the matrix R from the host image. This process embeds a proportion of the watermark image inside the host image. According to the experiments, our scheme is susceptible to the weakest attacks, so it is a fragile watermarking technique. So it helps to understand whether an image is manipulated or not. The validation dataset is composed of some images from the USC-SIPI image database.

Keywords Fragile watermarking · Image authentication · Fourier transformation · QR factorization · Host image · Watermark image

1 Introduction

When confidential images are transmitted through insecure environments, they might be exposed to manipulations. For some vital data like medical and military images, changes may be so risky because these manipulations might alter essential information inside images. As a result, these data should be kept safe against any changes.

The digital watermarking algorithms can be categorized into robust, fragile, and semi-fragile methods based on data during the watermark extraction trend.

✉ Hedieh Sajedi
hhsajedi@ut.ac.ir

Fatemeh Nejati
fatemehnejati25@gmail.com

Alireza Zohourian
alireza.zohourian@gmail.com

[1] Faculty of Mathematical Sciences and Computer, Kharazmi University, 50 Taleghani Avenue, 1561836314 Tehran, Iran

[2] Department of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Tehran, Iran

Robust watermarking does not react to attacks easily. Therefore, it plays a vital role in copyright protection. Despite different manipulations, the watermarked image does not change, and owners will not be concerned about illegal copyright infringement.

A fragile watermarking algorithm detects the weakest manipulations, and it is useful for recognizing any weak attacks. A semi-fragile algorithm is a method that is appropriate for both of the above goals. It is sensitive to attacks, so it helps to recognize attacks, and it is robust against a weak attack so that it can play a role in copyright protection. Nevertheless, it is not as sensitive as a fragile method. If we have valuable data, and it is crucial to keep it safe from any weak attacks, a semi-fragile method is not appropriate.

So far, several watermarking methods from the three mentioned categories have been introduced. Based on the application they developed to have higher visual quality and/or higher robustness against attacks compared to the existing watermarking algorithms.

In this article, a fragile watermarking algorithm is presented based on Fourier transform and QR decomposition. The goal of the method is to have high visual quality and high sensitivity against attacks. In the beginning, the Fourier transformation is used on the host image. After that, QR factorization is used to factorize the transformed host image along with the watermark image. Two upper triangular matrices are obtained. By choosing a proper coefficient of matrix R from the watermark image, a sign of the watermark image is embedded into the host image, resulting in an upper triangular matrix. The Q matrix of the host image and this upper triangular matrix should be multiplied together, and as a result, the matrix of the watermarked image is achieved. In the experimental part, it is understandable that although embedding and extraction parts work well, this algorithm is susceptible to the weakest attacks. So if the watermarked image is exposed to any threats, one will not be able to extract the watermark image. The current paper is an extension of the conference paper [1]. Experimental results show higher visual quality measured with two criteria compared to the existing watermarking algorithms.

The novelty of the proposed method is in the way of employing FT and QR decomposition. Although both FT and QR decomposition have been used previously in the watermarking field, the way of embedding mechanism and altering the value of coefficients affects the visual quality of the watermarked images directly. Therefore, in this paper, we proposed a novel embedding process to have higher performance.

2 Related Works

Recently, researchers have been working on fragile watermarking methods to improve the results. In this section, some fragile watermarking algorithms are summarized. Some of them employed a transform to transfer the host image from the spatial domain to the transform domain. The previous research works have been shown that embedding the bits in the transform coefficients results in watermarked images with higher visual quality.

The first fragile watermarking algorithm was presented in 1995 [2]. Based on this algorithm, a checksum is calculated from the seven most significant bits and is placed into the least important bit. Afterward, another algorithm was proposed to localize tampered image content [3] effectively. In this method, the watermark embedding procedure is comprised of two phases- i.e., authentication code generation from some selected salient bits of each pixel of the original image content.

Another fragile watermarking scheme was suggested for authenticating color-images. In this scheme, the input image is factorized into two blocks that do not overlap. For each i-th block, the watermarks are embedded into a different block according to an embedding sequence given by a permutation process [4]. A prediction-error expansion based on a reversible watermarking strategy was proposed to detect and localize malicious modifications and recovers back the original data at watermark detection [5].

A fragile watermarking was presented using the Local Binary Pattern (LBP) and Discrete Wavelet Transform (DWT) for image authentication. In this method, the LBP pattern of low-frequency wavelet coefficients is adopted as a feature watermark [6]. A pixel-based fragile watermarking algorithm for image tamper recognition was presented in [7]. By evaluating the left and right singular matrices of Singular Value Decomposition (SVD), it is found that the matrix product between the first column of the left singular matrix and the change of the first column in the right singular matrix is nearly related to the image texture attributes.

Another method with high-quality recovery capability according to an overlapping embedding approach was presented in [8]. The block-wise system is used for tampering localization, and the pixel-wise system for content reconstruction participates in this algorithm. In another fragile watermarking method for digital image tamper localization (TL) along with the self-recovery capability, at first, the host image is divided into blocks of size 4×4; then, singular value decomposition is performed on each block [9].

A fragile watermarking scheme for stereo image authentication was proposed in [10]. Using SHA-256 hash function, a block-based fragile watermark embedding and tamper detection method is proposed [11]. At first, the host image is divided into 32×32 non-overlapped blocks. Each 32×32 block is then divided into four 16×16 non-overlapped sub-blocks. The entire hash value of the first three sub-blocks is generated as a watermark using the SHA-256 hash function.

A Region of Interest (RoI) based fragile watermarking scheme for medical image tamper detection is presented in [12]. An image authentication approach is designed using block-based fragile watermarking [13]. Moreover, an effective recovery technique based on unsupervised machine learning is proposed. The authentication data is generated, for each 8×8 image block, using the Discrete Cosine Transform. Another fragile watermarking method for the authentication of digital images is proposed based on a binary rotation invariant and noise-tolerant (local texture descriptor and an Extreme Learning Machine (ELM) [14].

A self-embedding fragile watermarking scheme for medical images is presented [15]. Both self-recovery information and authentication code for each block is generated in advance and then embedded into other blocks separately with the turtle shell and embedding table's help. A frequency data-hiding scheme, which will be examined in accordance with the Linear Cellular Automata Transform using Manhattan distances, is proposed in [16]. Various invertible integer mappings are applied in order to find out the Manhattan distances from coordinates.

In a fragile watermarking technique for securing the copyrights of sensitive images, a combination of Compressive Sensing (CS) theory, Discrete Wavelet Transform (DWT), and Non-Subsampled Contourlet Transform (NSCT) are employed [17]. In a self-recovery based fragile watermarking [18], to improve the watermarked image quality, a bit-reduction technology is employed to generate a watermark with fewer bits. The watermark is then embedded into the original image using the turtle shell, based data hiding technique.

In [19], the authors present a methodology to protect multichannel images' integrity, having some highly redundant channels, using a reversible fragile watermarking algorithm. The watermark embedding phase uses a lossless compression method to compress the high redundancy channels, stores the compressed stream into their most significant bits, then embeds a secret fragile watermark by modifying the least significant bits of the high redundancy channels.

A secure fragile image watermarking scheme used to detect modification or tampering on image content is proposed in [20]. The scheme is developed into two steps, i.e., computation of secure authentication code/watermark bit from some of the most significant bits of each pixel, and subsequently hides the watermark bit in the Least Significant Bit (LSB) of each pixel by suggested watermark embedding procedure. In another fragile watermarking method, the watermark embedding procedure is comprised of two phases- i.e., authentication code generation from some selected salient bits of each pixel of the original image content, and encryption of the authentication code before realizing it for embedding into the insignificant bits of each pixel in the original cover image. The authentication code is computed from each block using Hamming Code. Subsequently, the encrypted code is concealed into the pixels of that particular block using the suggested payload embedding strategy [27]. The authors of [27] proposed two other fragile watermarking methods [28, 29], which embed the watermark in the least Significant Bits (LSBs) of pixels. Usually, LSB-based watermarking techniques result in higher visual quality measures. In some research works, different encryption methods (e.g. [30–32]) have been applied to enhance the performance of watermarking methods.

Although the research works about fragile watermarking, have been used different techniques for watermark production and watermark embedding processes, both in spatial (e.g. LSB based methods) and transform (e.g. LBP and DWT transforms) domains, nevertheless improving the visual quality of the resulted watermarked images is still an important issue.

3 Background

3.1 Fourier Transform

A grayscale image has two dimensions, so a two-dimensional Fourier transformation is applied to it. In this case, a matrix is taken as an amount of the Discrete Fourier Transformation (DFT), and it returns another matrix as output. Both matrices are of the same size.

Suppose i and j to be the indices. Then, $f(i,j)$ shows the original matrix values, and $F(u,v)$ is the output matrix values. F is f's Fourier transformation (Eq. 1).

$$F = \mathcal{F}(f) \tag{1}$$

Suppose the image yields an M×N matrix. The forward transform is calculated using Eq. (2). For simplicity, we consider i indices to range from 0 to M-1 and j indices to range from 0 to N – 1.

$$F(u,v) = \sum_{i=0}^{\mathrm{M}-1} \sum_{j=0}^{\mathrm{N}-1} f(i,j)\exp\left[-2\pi i\left(\frac{iu}{\mathrm{M}} + \frac{jv}{\mathrm{N}}\right)\right] \tag{2}$$

3.2 Matrix Factorization

Matrix factorization is a linear algebra technique with various applications in a large number of domains, such as image processing, data analysis, and machine learning. There are diverse matrix factorization methods like QR factorization, Singular Value Decomposition

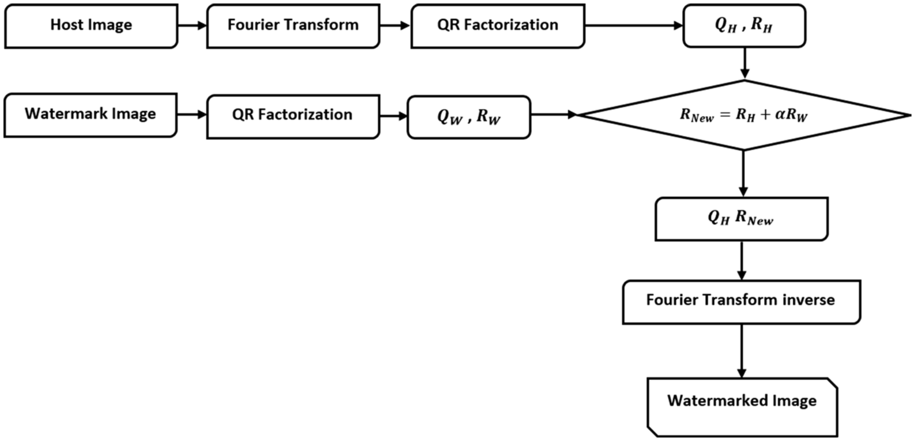

Fig. 1 Embedding algorithm

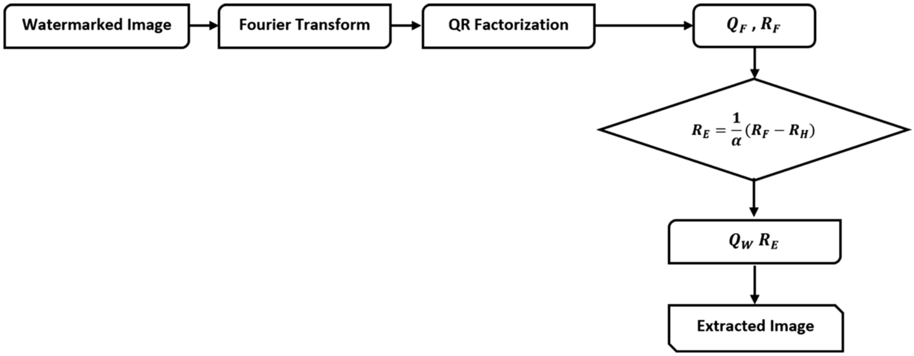

Fig. 2 Extraction algorithm

(SVD), LU decomposition, Schur decomposition, etc. In this article, to decompose both the watermark and the host images, QR factorization is used. It is worth noting that the host image needs to be transformed by FT before using QR factorization.

QR Factorization As it was said before, one of the most popular matrix factorizations is QR decomposition (also called QR factorization). Suppose A is a matrix after using QR factorization on this matrix. In that case, an equal A = QR will achieve in which matrices Q and R are orthogonal (i.e., $Q^T \cdot Q = 1$) an upper triangular, respectively. If A is nonsingular, then this factorization is unique. There are several methods for actually computing the QR decomposition. In this factorization, $A \in \mathbb{R}^{n\times k}$, $Q \in \mathbb{R}^{n\times k}$, and $R \in \mathbb{R}^{k\times k}$.

4 Proposed Algorithm

Here, the watermarking method is introduced. In Figs. 1 and 2, embedding and extraction phases are illustrated, respectively.

4.1 Embedding Algorithm

4.1.1 Preprocessing

In the beginning, before using Fourier transformation and QR decomposition to both images, grayscale conversion and resizing operations was run on them.

4.1.2 Fourier Transform

Fourier transform is mostly used on images to enhance their visual quality. The result of such a transformation is a transformed version of the initial image with the original size. It is noticeable that in this algorithm, FT is just applied to the host image. Manipulation of selected transform coefficients results in less perceptible altering in images.

4.1.3 Matrix Decomposition

After the host image is transformed as mentioned, it has to be decomposed by QR decomposition. In the meantime, the watermark image must be decomposed by QR decomposition too.

By denoting the transformed host image with H and the watermark image with W, using QR decomposition on H and W yields $\mathrm{Q_H}$,$\mathrm{R_H}$, $\mathrm{Q_W}$ and $\mathrm{R_W}$, respectively.

4.1.4 Embedding Watermark Images

After decomposing the matrices, one of the watermark matrix components should be chosen to be embedded into the host image. We choose matrix R because it is triangular and has fewer nonzero elements, and thus, it will not alter the initial image so much. It is a better choice for the embedding phase. To do so, a coefficient of matrix R_W is added to $\mathrm{R_H}$. As a result, an approximation of $\mathrm{R_H}$ is obtained, which is an upper triangular matrix. (Eq. 3).

$$\mathrm{R}_{new} = \mathrm{R_H} + \alpha \mathrm{R_W} \tag{3}$$

In this equation, α is called the embedding strength, which its lower values can significantly affect a better image incomprehensibility. The value of α is obtained experimentally. The $\mathrm{Q_H}$ should be multiplied to $\mathrm{R_{new}}$ in order to obtain the host image with a sign of the watermark image (Eq. 4).

$$H_{new} = Q_H + R_{new} \tag{4}$$

Finally, the inverse FT is applied to $\mathrm{H_{new}}$, and the watermarked image will be obtained. If α is selected correctly, the watermarked image resembles the host image and the watermark image cannot be perceptible inside it.

4.2 Extraction Algorithm

While a watermarked image is sent over an insecure social channel such as the Internet, it may face various threats, which will decrease the visual quality and will even cause to create incorrect information. As mentioned before, it is necessary for some vital data such as military and medical data to understand if images are exposed to attacks. For this application, a fragile watermarking algorithm is advantageous because it is even susceptible to the feeblest attacks. In this type of watermarking algorithm, in the extraction phase, if the watermark image can be extracted properly, it is implied that the watermarked image had not been exposed to any attacks. On the other hand, if the algorithm cannot extract the host image's watermark image, it means the watermarked image has been manipulated. The information of images may have changed and can no longer be trusted. The extraction part is expressed as follows:

4.2.1 Fourier Transform

At first, the receiver should use Fourier transformation on the watermarked image to obtain its transformation.

4.2.2 QR Factorization

After Fourier transformation, QR factorization is applied to obtain two matrices, namely, Q_F and R_F. The goal is to find an approximation of R_W, which is called, $R_{Extracted}$ and is calculated as in Eq. (5).

$$R_{Extreacted} = \frac{1}{\alpha}\left(R_F - R_H\right) \tag{5}$$

4.2.3 Generate Extracted Image

The extraction image will be achieved by multiplying Q_W into $R_{Extracted}$ (Eq. 6)

$$S = Q_W R_{Extracted} \tag{6}$$

Here, S is the extracted watermark image. If the watermarked image has not been exposed to any attacks while being transmitted, S is very close to the watermark image. However, if the extraction part cannot extract a proper image, we conclude that the watermarked image has been manipulated and cannot be trusted.

5 Experimental Results

Medical information plays an essential role in recognition of human diseases. If these images are manipulated, it causes incorrect diagnosis of diseases, creating many problems for people's health. So realizing whether an image is manipulated is a critical point for medical images. In Fig. 3, some different medical images are illustrated.

In this paper, the simulations have been done on Matlab R2018a and the images have been chosen from the USC-SIPI image database [21]. Various experiments have been done to check out imperceptibility in the watermarking algorithm. In the preprocessing part, to achieve the grayscale of images, a suitable function is applied. The performance evaluation metrics used in this paper will be expressed in the following.

To measure our schemes' invisibility, some quality metrics that include Peak Signal to Noise Ratio (PSNR), Root Mean Square Error (RMSE), Normalized Correlation (NC), and Structural Similarity (SSIM) are used. PSNR is calculated by Eq. (7).

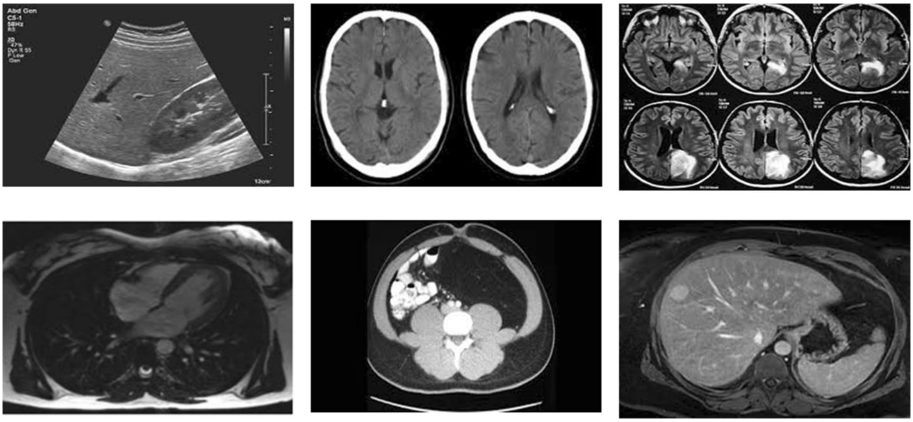

Fig. 3 Sample of medical images

$$\mathrm{PSNR} = 20\log_{10}\frac{255}{\mathrm{RMSE}} \tag{7}$$

which RMSE (Root Mean Square Error) is calculated in Eq. (8).

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{MN}}\sum_{i=1}^{N}\sum_{j=1}^{M}\left(\mathrm{H_{ij}} - \mathrm{W_{ij}}\right)^2} \tag{8}$$

In this equation, both images have sizes of M×N, $\mathrm{H_{ij}}$ is used to show the host image, and $\mathrm{W_{ij}}$ is also used for the watermark image. These metrics show the degree of visual quality. It means higher PSNR, and on the other hand, lower RMSE show that the image has more visual quality. SSIM is described in Eq. (9).

$$\mathrm{SSIM}\left(\mathrm{I},\mathrm{I}^*\right) = \mathrm{l}\left(\mathrm{I},\mathrm{I}^*\right)\mathrm{c}\left(\mathrm{I},\mathrm{I}^*\right)\mathrm{s}\left(\mathrm{I},\mathrm{I}^*\right) \tag{9}$$

where $\mathrm{l}\left(\mathrm{I},\mathrm{I}^*\right)$ is luminance comparison function, $\mathrm{c}\left(\mathrm{I},\mathrm{I}^*\right)$ is contrast comparison function and $\mathrm{s}\left(\mathrm{I},\mathrm{I}^*\right)$ is structure comparison function. The last metric which we use is NC. It is shown in Eq. (10).

$$\mathrm{NC} = \frac{\sum_{j=1}^{3}\sum_{x=1}^{p}\sum_{y=1}^{q}\left(\mathrm{C}(\mathrm{x},\mathrm{y},\mathrm{j})\right)*\left(\mathrm{W}^*(\mathrm{x},\mathrm{y},\mathrm{j})\right)}{\sqrt{\sum_{j=1}^{3}\sum_{x=1}^{p}\sum_{y=1}^{q}\left(\mathrm{C}(\mathrm{x},\mathrm{y},\mathrm{j})\right)^2}\sqrt{\sum_{j=1}^{3}\sum_{x=1}^{p}\sum_{y=1}^{q}\left(\mathrm{W}^*(\mathrm{x},\mathrm{y},\mathrm{j})\right)^2}} \tag{10}$$

where $C(x,y,j)$ and $W^*(x,y,j)$ show the value of pixel (x,y) in layer j of the original host image and watermarked image and p, q represent the row and column size of the original and watermarked images, respectively.

Our experiments are done using six images, which are shown in Fig. 4.

As it was said before, α is the embedding strength, and its different values bring various PSNRs for embedding and extraction parts. The effect of changing α on PSNR at embedding and extraction phases is shown in Fig. 5.

In the embedding phase, it is clear that an increase in α leads to a decrease in PSNR, because only a small amount of the watermark image will be added to the host image.

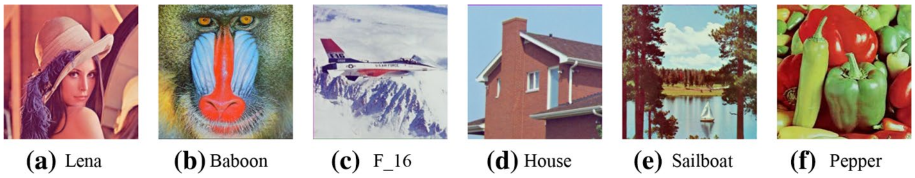
(a) Lena (b) Baboon (c) F_16 (d) House (e) Sailboat (f) Pepper

Fig. 4 **a**–**e** Host images, **f** watermark image

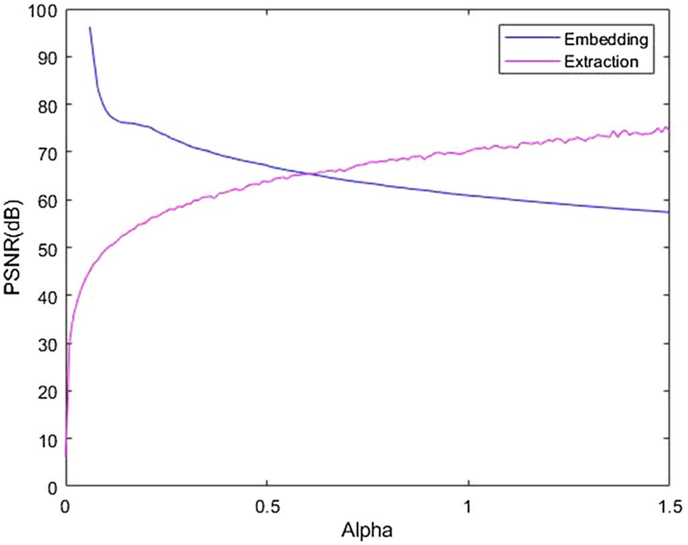

Fig. 5 The effect of changing α on PSNR in the embedding phase

Thus, if α is chosen small enough, PSNR will be larger, and as a result, our scheme will work well, and it means in the embedding phase, this method's imperceptibility is sufficient.

Choosing an appropriate α also plays an important role in the extraction part. The plot shows that the value of α and PSNR have an opposite relation. It means if α is chosen large enough, the PSNR value will be large too.

As a result, choosing α, should be considered from two sides. The best value of α occurs at the intersection of two plots. At this point, both of embedding and extraction's PSNRs are near 65, and this value shows that this scheme works well in both of these phases.

By choosing a correct α, the extraction part works correctly as well as the embedding phase, and the watermark image can be extracted entirely. In Fig. 6, different watermarked images are shown, and in Fig. 7 extracted images from the watermarked images in Fig. 6 are figured.

As mentioned before, choosing the best α can be done by considering the intersection point. By doing experiments on different host images, we found that a specific α is achieved for each host image, and as a result, for each host image, its own α obtains the PSNR value. This result is shown in Table 1. This table shows that our scheme works well in both phases and all PSNRs are high. To obtain this result, experiments have been done on images with 512×512 sizes.

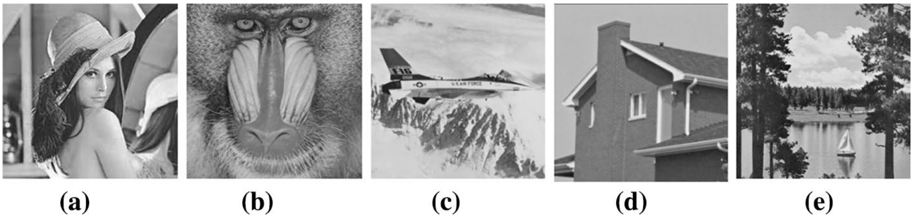

Fig. 6 **a**–**e** Watermarked Images

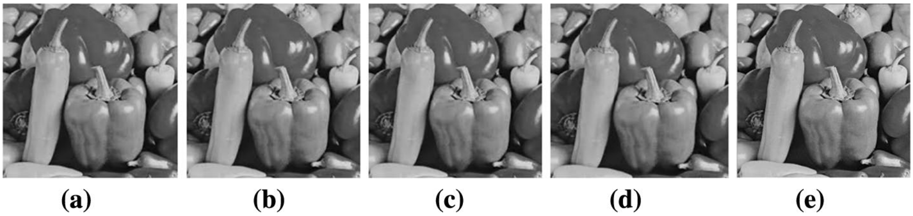

Fig. 7 **a**–**e** Extracted images

Table 1 PSNR (dB) of watermarked images from different sizes and different host images

Different cover	Lenna (α = 0.55)	Baboon (α = 0.32)	F-16 (α = 0.61)	House (α = 1.22)	Sailboat (α = 0.46)
Embedding PSNR	66.0079	70.7537	65.0743	59.0465	67.4265
Extraction PSNR	65.9253	70.3077	65.2910	59.0253	67.6165

We also evaluated our scheme in different sizes. The results are shown in Fig. 8. By considering this plot, we understand that the larger the image size in the embedding phase, the larger the PSNR value becomes. Nevertheless, in the extraction part, the situation is different, and the relation between the size of images and PSNR value is the opposite. It means that in order to have a better PNSR in the extraction phase, the size of the images should be smaller. By regarding this trade-off, the size of images should be chosen near 512 × 512. As a result, all of the next experiments have been done on 512 × 512 image size. It should be noted that the time required for embedding the watermark and the extraction is less than one second.

As mentioned before, the amounts of α affect PSNR values. Besides, it has a significant effect on other metrics, such as SSIM and NC. α's changes on SSIM and NC from embedding and extraction phases are shown in Figs. 9 and 10. The plots of Fig. 9 are from the embedding phase of these metrics, and the plots of Fig. 10 are from the extraction phase.

By considering Fig. 9 in the embedding phase, larger amounts of α, lead to lower SSIM and NC. Our method's preferences are that although α is so large, these metrics' values are close to 1.

In our scheme, to have higher PSNR in the extraction phase, α should be chosen large enough. Therefore, having a small α, leads to a decrease in PSNR, and as a result, it leads to having larger SSIM and NC. Figure 10 shows how the PSNR and SSIM values

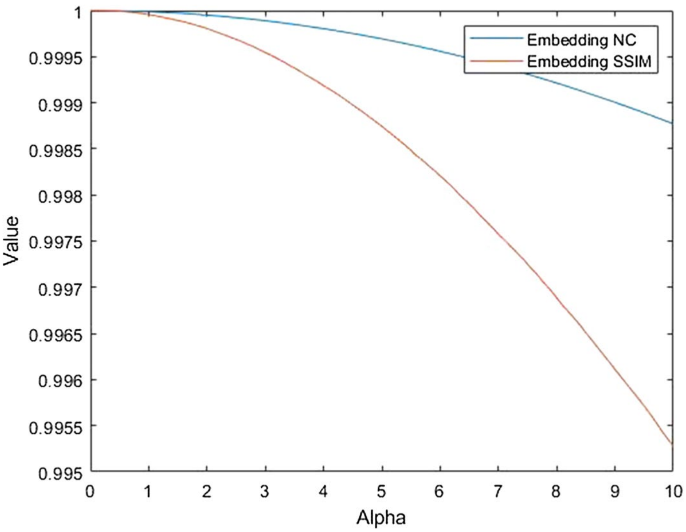

Fig. 8 The effect of image sizes on PSNR value

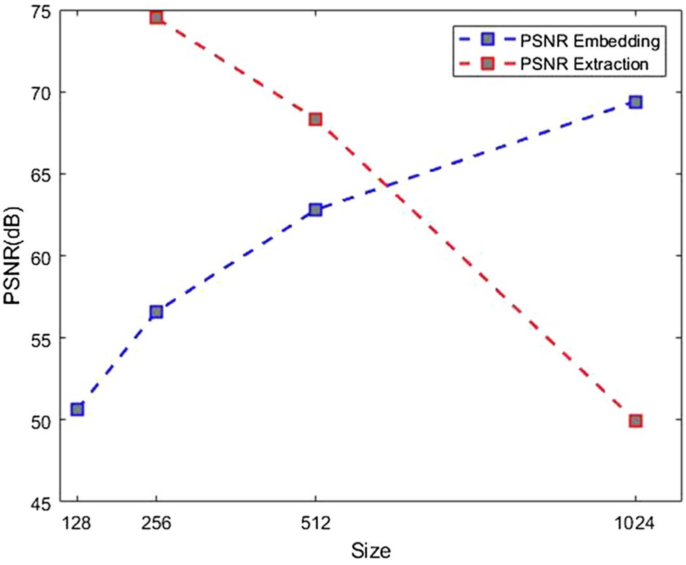

Fig. 9 SSIM and NC plots from the embedding phase

change by altering the value of α. As shown in Fig. 10, although we have chosen α very small, NC and SSIM move so fast to one, and in α = 0.01, these metrics' values are near one.

To continue our experiments, we need to have a fixed α and do the next experiments with it. To achieve this fixed α, we have done experiments on a larger dataset with 21 images. For each host image, we have found its intersection point. By calculating their

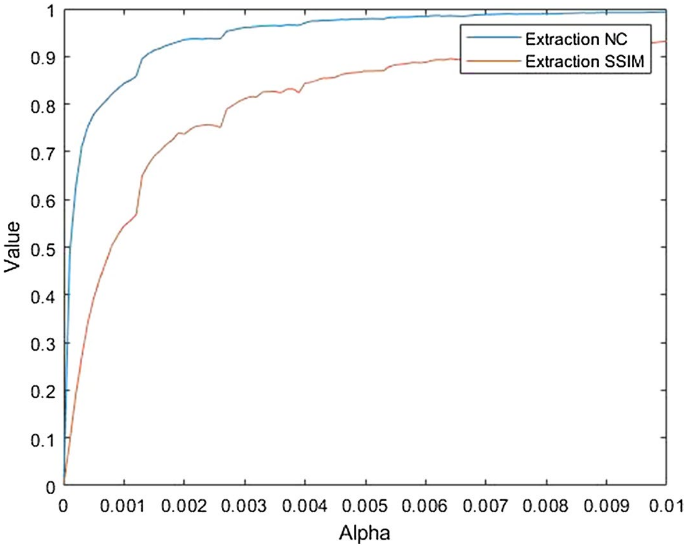

Fig. 10 SSIM and NC plots from the extraction phase

average, the best alpha is equal to 0.7833. Using this amount of α, the experimental results are shown in Table 2. The results show that our scheme's embedding phase works well.

This result from Tables 1 and 2 shows that this method is powerful in both embedding and extraction phases and it has a good visual quality. To illustrate this performance, we have compared our results with other methods in Table 3. The bold number in the last row of the table displays the superiority of the proposed method compared to the previous ones.

As it is shown in Fig. 7, in the extraction phase, if the watermarked image is not exposed to any attacks, the watermark image will be extracted correctly, and when we look at the watermark and extracted images, they seem to be the same, and no one can distinguish any differences, and it means extraction phase works perfectly.

But it will be different when a watermarked image is exposed to any weak attack because this algorithm is susceptible to every kind of attack. It means, if the watermarked image is exposed to any attacks, the extraction phase does not work. So the watermark image is not accessible.

This paper uses some weak attacks to evaluate this sensitivity by using Lena as the host image. The results are shown in Fig. 10. Besides, Table. 4. shows the PSNR from these attacks for all of the host images. All of the PSNR values are so low, and these

Table 2 PSNR, SSIM, NS, and RMSE of watermarked image from different host image

Image name	Lenna	Baboon	F-16	House	Sailboat	Average
PSNR	62.77	62.60	62.79	62.90	62.66	62.74
SSIM	0.9997	0.9999	0.9999	0.9997	0.9999	0.9998
NC	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
RMSE	0.1852	0.1889	0.1847	0.1824	0.1877	0.1858

Table 3 The comparison of PSNR and SSIM with other schemes

		Lenna	Baboon	F_16	House	Sailboat	Average
Singh [22]	PSNR	37.90	37.90	37.69	37.88	37.81	37.81
	SSIM	0.9307	0.9763	0.9194	0.9319	0.9493	0.9312
Dadkhah [23]	PSNR	44.13	44.14	44.12	44.19	44.08	44.08
	SSIM	0.9820	0.9941	0.9782	0.9815	0.9868	0.9814
Tong [24]	PSNR	37.90	37.90	37.88	37.88	37.81	37.81
	SSIM	0.9307	0.9763	0.9194	0.9319	0.9494	0.9312
Fan [25]	PSNR	44.13	44.12	44.11	44.18	44.07	44.07
	SSIM	0.9820	0.9941	0.9781	0.9815	0.9867	0.9814
Tai [26]	PSNR	44.12	44.14	44.12	44.18	44.08	44.08
	SSIM	0.9820	0.9941	0.9781	0.9815	0.9868	0.9814
Garcia [4]	PSNR	44.60	44.64	44.69	44.66	44.63	44.63
	SSIM	0.9840	0.9947	0.9812	0.9834	0.9884	0.9839
Parasad [27]	PSNR	42.01	42.29	41.85	42.32	42.11	42.11
	SSIM	0.9993	0.9994	0.9994	0.9996	0.9997	0.9995
Parasad [29]	PSNR	42.79	42.29	42.52	42.18	42.76	42.44
	SSIM	0.9932	0.9994	0.9832	0.9984	0.9951	0.9935
Our scheme	PSNR	62.77	62.60	62.79	62.90	62.66	62.7440
	SSIM	0.9997	0.9999	09,999	0.9997	0.9999	0.9998

Table 4 PSNR (dB) and SSIM value of different attacks by considering different host images

Name of attack		Lenna	Baboon	F_16	House	Sailboat
Sharpen	PSNR	1.2952	1.1045	1.4734	2.0641	1.1941
	SSIM	0.0035	0.0086	0.0018	0.0024	0.0018
Rotate (5°)	PSNR	1.2530	1.0727	1.0504	1.2179	1.0846
	SSIM	0.0057	0.0033	0.0162	0.0070	0.0051
Crop	PSNR	3.6301	1.3829	2.5642	6.4448	2.6829
	SSIM	0.0189	0.0067	0.0076	0.0590	0.0072
Blur	PSNR	1.1640	1.0742	1.0574	1.8753	1.0721
	SSIM	0.0038	0.0026	0.0313	0.0052	0.0032

results prove the sensitivity against any attack. As a result, this algorithm is fragile, and it can help protect vital images against manipulation.

The values of PSNR and SSIM in Table 4 and the visual quality of the extracted watermarks in Fig. 10, all show that the occurrence of attacks, even with minor modifications, can be recognizable (Fig. 11).

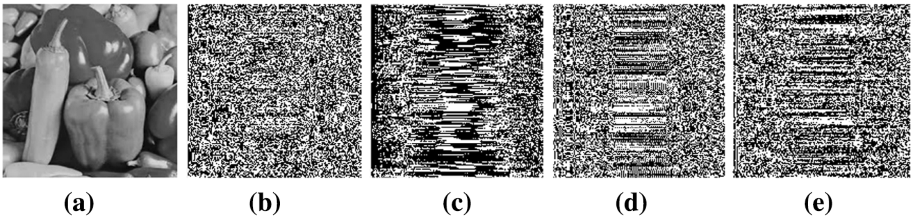

Fig. 11 **a** Extracted image without any attack. **b** Extracted image with sharpen attack. **c** Extracted image with rotation attack. **d** Extracted image with crop attack. **e** Extracted image with blur attack

6 Discussion

In the proposed method, the amounts of α affect PSNR values. In addition, it has a significant effect on other metrics, such as SSIM and NC. In the embedding phase, larger amounts of α, lead to lower SSIM and NC. In our scheme, to have higher PSNR in the extraction phase, α should be chosen large enough. Therefore, having a small α, leads to a decrease in PSNR, and as a result, it leads to having larger SSIM and NC. The experimental results show that the proposed method is influential in both embedding and extraction phases, and it has an excellent visual quality. Compared our results with other methods expresses that it is superior compared to other methods.

The proposed algorithm is vulnerable to every kind of attack. It means, if the watermarked image is exposed to any attacks, the extraction phase does not work. Consequently, the watermark image is not accessible.

Since the proposed method is fragile, it is susceptible to any changes (e.g., light compression on the Internet). If a watermarked image is manipulated, the extraction process cannot extract the watermark. Accordingly, it is applicable only when the detection of changes is significant.

7 Conclusion

We proposed a fragile watermarking method based on QR decomposition and Fourier transform, which fulfills the goal of identifying manipulated images. According to the experimental results, this method works well in the embedding and extraction phases and is susceptible to the weakest attacks, so it acts beneficially in protecting valuable data and detecting any modification. Fine-tuning the parameters especially alpha is a handwork, which is directly, affects the visual quality of the watermarked and extracted images. Setting the parameters adaptively to the feature of images and evaluation the proposed method against other attacks can be considered for the future work.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Nejati, F., Sajedi, H., & Mohammadi, M. (2019). Fragile watermarking for image authentication using QR factorization and Fourier transform. In *2019 5th international conference on web research (ICWR)* (pp. 45–49). IEEE, https://doi.org/10.1109/ICWR.2019.8765292
2. Walton, S. (1995). Image authentication for a slippery new age. *Dr. Dobb's Journal, 20*(4), 18–26.
3. Prasad, S., & Pal, A. K. (2020). A tamper detection suitable fragile watermarking scheme based on novel payload embedding strategy. *Multimedia Tools and Applications, 79*(3), 1673–1705. https://doi.org/10.1007/s11042-019-08144-5
4. Molina-Garcia, J., Garcia-Salgado, B. P., Ponomaryov, V., Reyes-Reyes, R., Sadovnychiy, S., & Cruz-Ramos, C. (2020). An effective fragile watermarking scheme for color image tampering detection and self-recovery. *Signal Processing: Image Communication, 81*, 115725. https://doi.org/10.1016/j.image.2019.115725
5. Hamadou, A., Camara, L., Issaka Hassane, A. A., & Naroua, H. (2020). Reversible fragile watermarking scheme for relational database based on prediction-error expansion. *Mathematical Problems in Engineering*. https://doi.org/10.1155/2020/1740205
6. Wang, C., Zhang, H., & Zhou, X. (2018). LBP and DWT based fragile watermarking for image authentication. *Journal of Information Processing Systems, 14*(3), 666–679. https://doi.org/10.3745/JIPS.03.0096
7. Zhang, H., Wang, C., & Zhou, X. (2017). Fragile watermarking for image authentication using the characteristic of SVD. *Algorithms, 10*(1), 27. https://doi.org/10.3390/a10010027
8. Qin, C., Ji, P., Zhang, X., Dong, J., & Wang, J. (2017). Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. *Signal Processing, 138*, 280–293. https://doi.org/10.1016/j.sigpro.2017.03.033
9. Ansari, I. A., Pant, M., & Ahn, C. W. (2016). SVD based fragile watermarking scheme for tamper localization and self-recovery. *International Journal of Machine Learning and Cybernetics, 7*(6), 1225–1239. https://doi.org/10.1007/s13042-015-0455-1
10. Yu, M., Wang, J., Jiang, G., Peng, Z., Shao, F., & Luo, T. (2015). New fragile watermarking method for stereo image authentication with localization and recovery. *AEU-International Journal of Electronics and Communications, 69*(1), 361–370. https://doi.org/10.1016/j.aeue.2014.10.006
11. Gul, E., & Ozturk, S. (2019). A novel hash function based fragile watermarking method for image integrity. *Multimedia Tools and Applications, 78*(13), 17701–17718. https://doi.org/10.1007/s11042-018-7084-0
12. Goléa, N. E. H., & Melkemi, K. E. (2019). ROI-based fragile watermarking for medical image tamper detection. *International Journal of High Performance Computing and Networking, 13*(2), 199–210. https://doi.org/10.1504/IJHPCN.2019.097508
13. Abdelhakim, A., Saleh, H. I., & Abdelhakim, M. (2019). Fragile watermarking for image tamper detection and localization with effective recovery capability using K-means clustering. *Multimedia Tools and Applications, 78*(22), 32523–32563. https://doi.org/10.1007/s11042-018-7084-0
14. AlShehri, L., Hussain, M., Aboalsamh, H., & Wadood, A. (2020). Fragile watermarking for image authentication using BRINT and ELM. *Multimedia Tools and Applications, 79*(39), 29199–29223. https://doi.org/10.1007/s11042-020-09441-0
15. Su, G. D., Chang, C. C., & Lin, C. C. (2020). Effective self-recovery and tampering localization fragile watermarking for medical images. *IEEE Access, 8*, 160840–160857. https://doi.org/10.1109/ACCESS.2020.3019832
16. Al-Ardhi, S., Thayananthan, V., & Basuhail, A. (2019). Fragile Watermarking based on linear cellular automata using manhattan distances for 2D vector map. *International Journal of Advanced Computer Science and Applications (IJACSA)*. https://doi.org/10.14569/IJACSA.2019.0100651
17. Hemida, O., Huo, Y., He, H., & Chen, F. (2019). A restorable fragile watermarking scheme with superior localization for both natural and text images. *Multimedia Tools and Applications, 78*(9), 12373–12403. https://doi.org/10.1007/s11042-018-6664-3
18. Chang, C. C., Lin, C. C., & Su, G. D. (2020). An effective image self-recovery based fragile watermarking using self-adaptive weight-based compressed AMBTC. *Multimedia Tools and Applications, 79*(33), 24795–24824. https://doi.org/10.1007/s11042-020-09132-w
19. Botta, M., Cavagnino, D., & Pomponiu, V. (2020). Reversible fragile watermarking for multichannel images with high redundancy channels. *Multimedia Tools and Applications, 79*(35), 26427–26445. https://doi.org/10.1007/s11042-020-08986-4
20. Prasad, S., & Pal, A. K. (2019). A secure fragile watermarking scheme for protecting integrity of digital images. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*. https://doi.org/10.1007/s40998-019-00275-7

21. USC-SIPI. 1997. http://sipi.usc.edu/database.
22. Singh, D., & Singh, S. K. (2016). Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability. *Journal of Visual Communication and Image Representation, 38*, 775–789. https://doi.org/10.1016/j.jvcir.2016.04.023
23. Dadkhah, S., Abd Manaf, A., Hori, Y., Hassanien, A. E., & Sadeghi, S. (2014). An effective SVD-based image tampering detection and self-recovery using active watermarking. *Signal Processing: Image Communication, 29*(10), 1197–1210. https://doi.org/10.1016/j.image.2014.09.001
24. Tong, X., Liu, Y., Zhang, M., & Chen, Y. (2013). A novel chaos-based fragile watermarking for image tampering detection and self-recovery. *Signal Processing: Image Communication, 28*(3), 301–308. https://doi.org/10.1016/j.image.2012.12.003
25. Fan, M., & Wang, H. (2018). An enhanced fragile watermarking scheme to digital image protection and self-recovery. *Signal Processing: Image Communication, 66*, 19–29. https://doi.org/10.1016/j.image.2018.04.003
26. Tai, W. L., & Liao, Z. J. (2018). Image self-recovery with watermark self-embedding. *Signal Processing: Image Communication, 65*, 11–25. https://doi.org/10.1016/j.image.2018.03.011
27. Prasad, S., & Kumar Pal, A. (2020). A tamper detection suitable fragile watermarking scheme based on novel payload embedding strategy. *Multimedia Tools and Applications, 79*, 1673–1705. https://doi.org/10.1007/s11042-019-08144-5
28. Prasad, S., & Pal, A. K. (2020). Hamming code and logistic-map based pixel-level active forgery detection scheme using fragile watermarking. *Multimedia Tools and Applications, 79*, 20897–20928. https://doi.org/10.1007/s11042-020-08715-x
29. Prasad, S., & Pal, A. K. (2020). A secure fragile watermarking scheme for protecting integrity of digital images. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering, 44*, 703–727. https://doi.org/10.1007/s40998-019-00275-7
30. Dua, M., Suthar, A., Garg, A., & Garg, V. (2021). An ILM-cosine transform-based improved approach to image encryption. *Complex & Intelligent Systems, 7*(1), 327–343.
31. Nancharla, B. K., & Dua, M. (2020). An image encryption using intertwining logistic map and enhanced logistic map. In *2020 5th international conference on communication and electronics systems (ICCES)* (pp. 1309–1314). IEEE.
32. Dua, M., Wesanekar, A., Gupta, V., Bhola, M., & Dua, S. (2019). Differential evolution optimization of intertwining logistic map-DNA based image encryption technique. *Journal of Ambient Intelligence and Humanized Computing, 11*, 3771–3786.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Fatemeh Nejati received BS and MS in Mathematics from Department of Mathematical and Computer Sciences, Kharazmi University. Her research interests include Image Processing and Machine Learning.

Hedieh Sajedi received a B.Sc. degree in Computer Engineering from AmirKabir University of Technology in 2003, and M.Sc. and Ph.D. degrees in Computer Engineering (Artificial Intelligence) from Sharif University of Technology, Tehran, Iran in 2006 and 2010, respectively. She is currently an Associate Professor at the Department of Computer Science, University of Tehran, Iran. Her research interests include Image Processing, Machine Learning, and Data Mining.

Alireza Zohourian recieved a BS in Mathematics and a MS in Computer Science. Currently he is PhD student of Computer Science at the University of New Brunswick. He is also a Research Assistant in the Canadian Institute for Cybersecurity.