Digital Watermarking Scheme Enhancing the Robustness Against Cropping Attack

Ching-Sheng Hsu[1] and Shu-Fen Tu[2(✉)]

[1] Department of Information Management, Ming Chuan University,
Taoyuan City, Taiwan
cshsu@mail.mcu.edu.tw

[2] Department of Information Management, Chinese Culture University,
Taipei City, Taiwan
dsf3@ulive.pccu.edu.tw

Abstract. Most digital watermarking schemes using QR factorization suffer from being unable to fully utilize the elements of the R matrix. Thus, these schemes are neither secure nor robust to resist the cropping attack. Besides, these schemes do not deal with the allowable modification ranges of the R elements, thereby causing the damage to the hidden watermark. In this paper, we designed an algorithm to redundantly embed the four copies of the watermark bits to enhance the ability to against the cropping attack. During the embedding process, the property of sign wave is employed to ease the modification of real number coefficients. After the four copies of a watermark bit are extracted, they may be different due to possible attacks. Therefore, we designed a weighted strategy to resolve the watermark bit. The experimental results show that our scheme satisfy the requirements of imperceptibility and robustness. Particularly, our scheme has prominent robustness against cropping attacks.

Keywords: Digital watermarking · QR decomposition · Cropping attack

1 Introduction

Digital watermarking is a common way to protect digital images. The digital image is watermarked with a watermark, which may be a binary image or just a binary stream. Depending on the purpose of the watermarking scheme, the watermark is either robust or fragile. The purpose of a robust watermarking scheme is to protect the copyright of the digital image. The copyright of the digital image is proved by the extracted watermark, so the extracted watermark should be robust enough even if the watermarked image undergoes slight modification. On the other hand, the purpose of a fragile watermarking scheme is to protect the integrity of the digital image. The extracted watermark should be fragile when the watermarked image has been tampered with. The working domain of the digital images may be spatial or frequent [5]. Recently, some researches tried to work on another domain which is derived from matrix factorization, such as SVD [1, 2, 8], QR decomposition, or LU decomposition. The main idea is to utilize matrix factorization to decompose the image and embed the watermark into one of the decomposed matrices. Among these matrix factorization methods, QR

J. C. Hung et al. (Eds.): FC 2017, LNEE 464, pp. 84–93, 2018.
https://doi.org/10.1007/978-981-10-7398-4_9

decomposition is a good choice for digital watermarking and image steganography. The reasons are as follows. Firstly, in comparison with SVD, QR decomposition has lower computational complexity and can avoid false positive problems [12, 14]. Secondly, in comparison with LU factorization, QR decomposition is more accurate for least square problems. Thirdly, LU factorization can only applied to square matrices while QR decomposition can be applied to rectangular and square matrices. Fourthly, the first row elements of the R matrix obtained by QR decomposition are able to resist to several image processing operations, such as lossy compression, noise addition, and filtering. Finally, the first row elements of the R matrix are likely to be greater than those elements in other rows, thereby allowing a greater modification range [12, 14]. Because of the above mentioned properties of QR decomposition, some researches use the first row elements for watermark embedding and image steganography [4, 7, 11–14].

Most QR-based digital watermarking schemes suffer from being unable to fully utilize the coefficients of the R matrix. Therefore, such methods may have low data hiding capacity and watermark security. For example. Su et al. [12] adopted QR decomposition to hide a robust color watermark into a color image. They use the last element of the first row in the R matrix to hide a watermark bit, but the allowable modification range of this coefficient was not discussed. Thus, the embedded watermark may be damaged by the reverse QR decomposition without any image processing operation or any malicious attack. This problem also occurs in Subhedar and Mankar's image steganography scheme [14]. The main concern of a QR-based digital watermarking scheme is that the modification to the decomposed matrix may cause the pixel values out of range. In order to fully utilize all the coefficients in the R matrix, we have to know the allowable modification range of each coefficients.

To cope with this issue, we have built formulas to compute the allowable modification ranges, thereby increasing the capability of information hiding and watermark security [6]. Our watermark embedding scheme employ the sine wave function in trigonometry and achieve the possible minimal modification within the allowable bounds. The sign wave simplifies the modification to real number coefficients by its nature. Like Su et al.'s scheme [12], the host image is decomposed by QR decomposition, and the watermark is embedded in the R matrix. But unlike their scheme, we use all elements of the first row of matrix R and adopt Householder reflections [15] for QR decomposition because it is numerical stable relative to Gram-Schmidt process [3, 10], which is adopted by Su et al.'s scheme [12]. In addition, most QR-based watermarking scheme embeds one watermark bit in an image block; therefore, their watermarks are prone to be fragile to cropping attacks. Take the example of Su et al.'s scheme. When the watermarked image is cut 25% off, the correct rate of the extracted watermark is about 87.75%. However, when the watermarked image is cut 50%, the correct rate of the extracted watermark decreases to 62.64% [12]. Therefore, we design an embedding strategy to enhance the robustness against cropping attacks. The rest of this paper is organized as follows. In Sect. 2, we will analyze the allowable modified ranges of the elements and explain the watermark embedding and extraction scheme in detail. In Sect. 3, we will demonstrate the experimental results and give some discussions. Finally, we will give conclusions in Sect. 4.

2 The Proposed Scheme

2.1 The Allowable Modification Range

Suppose the host image is a gray-level image. Thus, the integer pixel values of the host image range from 0 to 255. In this research, 4 × 4 image blocks are used to embed watermark. Thus, the matrix A is denoted as a 4 × 4 matrix. Each matrix A is factorized to matrix Q and R via QR decomposition, which are illustrated as follows:

$$A = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}, Q = \begin{bmatrix} q_{00} & q_{01} & q_{02} & q_{03} \\ q_{10} & q_{11} & q_{12} & q_{13} \\ q_{20} & q_{21} & q_{22} & q_{23} \\ q_{30} & q_{31} & q_{32} & q_{33} \end{bmatrix}, \text{and } R = \begin{bmatrix} r_{00} & r_{01} & r_{02} & r_{03} \\ 0 & r_{11} & r_{12} & r_{13} \\ 0 & 0 & r_{22} & r_{23} \\ 0 & 0 & 0 & r_{33} \end{bmatrix}$$

If the modification applies to the first row elements of the R matrix, then the reconstructed values in matrix A should also range from 0 to 255. Thus, the allowable modification range of the R matrix should be identified to accommodate the valid pixel values. For $x = R[0, j]$, where $j = 0..3$, we define the allowable upper bound ub and lower bound lb of the modified x by Eqs. (1) and (2) [6].

$$lb = \max_{i \in \{0..3\}} (LB[i]) \tag{1}$$

And

$$ub = \min_{i \in \{0..3\}} (UB[i]) \tag{2}$$

where

$$LB[i] = \begin{cases} -\infty & \text{if } Q[i, 0] = 0, \\ \min(a, b) & \text{otherwise.} \end{cases} \tag{3}$$

$$UB[i] = \begin{cases} \infty & \text{if } Q[i, 0] = 0, \\ \max(a, b) & \text{otherwise.} \end{cases} \tag{4}$$

$$a = R[0, j] - A[i, j]/Q[i, 0] \tag{5}$$

and

$$b = R[0, j] + (255 - A[i, j]/Q[i, 0]). \tag{6}$$

2.2 Watermark Embedding

Suppose that the host image is an $M \times N$ gray-level image and the watermark is an $M/4 \times N/4$ binary image, where both M and N are multiples of four. At first, the host image is divided into n non-overlapping 4 × 4 blocks, where $n = M/4 \times N/4$. Then, each block A, which can be seen as a 4 × 4 matrix, is factorized to Q and R matrices by means of Householder reflections, and all R matrices are used to embed the watermark. Let $W = (w_0, w_1, \ldots, w_{n-1})$ denote the binary watermark, where $w_i \in \{0, 1\}$ for $i = 0, 1, \ldots n - 1$. Let $(R_0, R_1, \ldots, R_{n-1})$ denote the array of n R matrices constructed from the n image blocks. For the sake of increasing the survival of the watermark, we adopt a two-pronged strategy. Firstly, the original array $(R_0, R_1, \ldots, R_{n-1})$ is shuffled according to a pseudo-random number generator seeded by the secret key SK. Secondly, four copies of each watermark bit are redundantly inserted into the first row elements of the R matrix. Let $(R'_0, R'_1, \ldots, R'_{n-1})$ denote the shuffled array, and $R_i[0, 0]$, $R_i[0, 1]$, $R_i[0, 2]$, and $R_i[0, 3]$ respectively denote the first row elements of R_i, for $i = 0..(n-1)$. To embed a watermark bit w_i, the four copies of w_i are redundantly inserted into the four R elements: $R'_{i \bmod n}[0, 0], R'_{i+1 \bmod n}[0, 1], R'_{i+2 \bmod n}[0, 2]$, and $R'_{i+3 \bmod n}[0, 3]$. Finally, the reverse operation of the QR decomposition is used to generate the watermarked image.

The rule for embedding a watermark bit $w \in \{0, 1\}$ to an element x of a matrix R depends on a sine wave function:

$$f(x) = sin(k \cdot x), \tag{7}$$

where $k > 0$ and is a real number. Accordingly, the wavelength λ of the function f is $(360/k)$. If $w = 1$, then x is modified to x' such that $f(x') \geq 0$; otherwise, if $w = 0$, then x is modified to x' such that $f(x') < 0$. Note that the modification of x to x' is restricted to the following three constraints:

1. The range of x' needs to be within [0, 255].
2. The modification of x should be as less as possible so that x' is near to x to ensure the imperceptibility of our scheme.
3. The value of $|f(x')|$ should be as large as possible, hence our scheme can tolerate an acceptable alteration on the watermarked image.

Therefore, the proposed scheme follows the three steps below to modify x.

Step 1: For $x = R[0, j]$, where $j = 0..3$, define the allowable upper bound ub and lower bound lb of x' by Eqs. (1) and (2).

Step 2: Complying with the following rules, modify x to x' such that $|f(x')| = 1$ and the difference between x and x' is as small as possible.

Rules for w = 0:

If $(x \geq 0 \wedge r \leq 0.25\lambda)$ **then** $x' = (x - r) - 0.25\lambda$.
If $(x \geq 0 \wedge r > 0.25\lambda)$ **then** $x' = (x - r) - 0.75\lambda$.
If $(x < 0 \wedge r \geq -0.5\lambda)$ **then** $x' = (x - r) - 0.25\lambda$.
If $(x < 0 \wedge r < -0.5\lambda)$ **then** $x' = (x - r) - 1.25\lambda$.

Rules for w = 1:

If $(x \geq 0 \wedge r \leq 0.75\lambda)$ **then** $x' = (x - r) + 0.25\lambda$.
If $(x \geq 0 \wedge r > 0.75\lambda)$ **then** $x' = (x - r) + 1.25\lambda$.
If $(x < 0 \wedge r \geq -0.25\lambda)$ **then** $x' = (x - r) + 0.25\lambda$.
If $(x < 0 \wedge r < -0.25\lambda)$ **then** $x' = (x - r) - 0.75\lambda$.

The remainder r of x divided by λ is defined by

$$r = s\left(|x| - \lambda\left\lfloor\frac{|x|}{\lambda}\right\rfloor\right), \tag{8}$$

where $s \in \{1, -1\}$ represents the sign of x.

Step 3: If x' is out of allowable range, perform the following procedure to adjust x' so that x' is able to be within $[lb, ub]$, the difference between x and x' is as close as possible, and $|f(x')|$ is as large as possible.

```
Procedure ValueAdjust($x'$, $lb$, $ub$, $\lambda$):
$low \leftarrow x' - 0.25\lambda$
$high \leftarrow x' + 0.25\lambda$
If $x' < lb$ then
  If $lb \geq high$ and $ub \leq high + 0.5\lambda$ then
    Indicate that "No feasible solution for $x'$."
  If $ub \geq x' + \lambda$ then $x' \leftarrow x' + \lambda$
  Else if $lb <$ high then $x' \leftarrow lb$
  Else if $ub < x' + \lambda$ then $x' \leftarrow ub$
Else if $x' > ub$ then
  If $ub \leq$ low and $lb \geq$ low $+ 0.5\lambda$ then
    Indicate that "No feasible solution for $x'$."
  If $lb \leq x' - \lambda$ then $x' \leftarrow x' - \lambda$
  Else if $ub >$ low then $x' \leftarrow ub$
  Else if $lb > x' - \lambda$ then $x' \leftarrow lb$
Else
  Indicate that $x'$ is valid and do nothing
```

2.3 Watermark Extraction

To extract the robust watermark W, the secret key SK is used to generate the shuffled array $(R'_0, R'_1, \ldots, R'_{n-1})$. In principle, the extracted bit w' of an element of the R matrix is determined according to Eq. (9).

$$w' = \begin{cases} 1 & \text{if } f(x) \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

where $x = R[0, j]$ for $j = 0..3$.

Remember that each watermark bit is redundantly embedded in four elements of the R matrix. Therefore, the four copies of the watermark bit w_i are extracted from coefficients $R'_{i \bmod n}[0, 0], R'_{i+1 \bmod n}[0, 1], R'_{i+2 \bmod n}[0, 2],$ and $R'_{i+3 \bmod n}[0, 3]$, and each copy is determined by Eq. (9). Note that the extracted four copies may be different due to the possible alteration to the watermarked image, it is necessary to have a rule to reach a consensus about the extracted watermark bit w'_i. Intuitively, the majority voting principle is practical. That is, if two or more copies are '1', then the extracted watermark bit w'_i is determined as '1'; otherwise, the extracted watermark bit w'_i is determined as '0'. Because different elements of the R matrix may have different ability to resist image processing operations or malicious attacks [12], more sophisticated way can be used, which puts different weights on the four extracted copies. Assume that the ability to resist image processing operations of the R element r_{0i} is weighted as α_j, for j = 0, 1, 2, 3 and

$$\sum_{j=0}^{3} \alpha_j = 1$$

Also assume that $c_i(0)$, $c_i(1)$, $c_i(2)$, and $c_i(3)$ are the extracted four copies of the watermark bit w_i. Then, the extracted watermark bit w'_i is determined according to the following rule.

$$w'_i = \begin{cases} 1 & \text{if } \sum_{j=0}^{3} \alpha_j \cdot c_i(j) \geq 0.5, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

In fact, the majority voting principle is a special case of the weighted scheme. Generally, the weight α_j can be determined according experimental analysis.

3 Experiment Results and Discussions

In general, a robust watermarking scheme is evaluated by its imperceptibility and robustness. The imperceptibility means that the difference between the host image and the watermarked image should not be perceived by human eyes, and the robustness means that the extracted watermark should be similar to the original watermark. In this paper, we use *PSNR* to evaluate the imperceptibility of our scheme [9].

$$PSNR = 10 \times \log \frac{255^2}{MSE}, \tag{11}$$

where

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (p_{i,j} - p'_{i,j})^2. \tag{12}$$

The notation M and N in Eq. (12) denote the width and height of an image, respectively, and $p_{i,j}$ and $p'_{i,j}$ denote the pixel of the original and watermarked image, respectively. Generally, the difference between the watermarked image and the host image is visually imperceptible if *PSNR* is greater than 30. The robustness is evaluated by the indicator *NC* as follows.

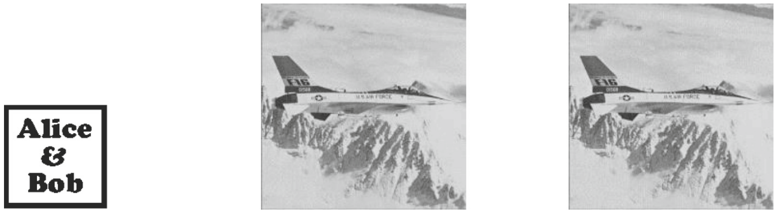
$$NC = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left(\overline{w_{i,j} \oplus w'_{i,j}} \right) \tag{13}$$

The notation *W* and *H* in Eq. (13) denote the width and height of the watermark image, respectively, $w_{i,j}$ and $w'_{i,j}$ denote the bit of the original and extracted watermark, respectively, and '⊕' represents the logical XOR operation. The larger the *NC* is, the more robust the extracted watermark is.

Figure 1(a) is our watermark, which is embedded into the host image (Fig. 1(b)). The embedding rule (Eq. (7)) needs a parameter *k*, but it is not necessary to use the same value of *k* for each element of the R matrix to embed a bit. In this experiment, we use four values (10, 90, 90, 90, 90) respectively corresponding to the four elements (R[0,0], R[0,1], R[0,2], R[0,3]). Figure 1(c) is the watermarked image, and the PSNR is 38.0081. Obviously, our scheme is qualified for the imperceptibility. With regard to the robustness, we simulate two kinds of attacks, namely cropping and JPEG lossy compression, on the watermarked image and inspect the survival of the extracted watermark. As mentioned in the above section, we can use a set of weights (α_0, α_1, α_2, α_3) for the extracted four copies to determine the extracted watermark bit. In this experiment, we test two different sets (1.0, 0.0, 0.0, 0.0) and (0.25, 0.25, 0.25, 0.25) on the robustness of the watermark.

We performed experiments on the watermarked image under cropping attack with different cropping ratios (CR) from 0.05 to 1.00. Figure 2(a)–(t) are the extracted watermarks and their respective NC values, and the corresponding cropping ratios are listed as well. Figure 2 shows that the similarity between the extracted watermark and the original watermark is greater than 80% even though the cropping ratio reaches as high as 0.6. Compared to the NC value against the attack with 25% cropping ratio of Su et al.'s scheme [12], our scheme gets 97.81%, which is higher than Su et al.'s 87.7%. When the watermarked image is cut 50% off, the NC value of our scheme is 88.59% while that of Su et al.'s is 62.64%. Thus it can be seen that our embedding strategy largely enhances the robustness against cropping attacks. The other experiments were performed on the watermarked image under JPEG lossy compression with different compression qualities, *i.e.* compressed image quality, from 0.05 to 1.00. The compression is implemented with Java Image I/O API provided in Java SE 8. Figure 3(a)–(t) are the extracted watermarks and their respective NC values, and the corresponding JPEG qualities are listed as well. Figure 3 shows that the similarity between he extracted watermark and the original watermark is greater than 80% even though the compression quality decreases as low as 0.45.

We have mentioned earlier that two sets of weights are used to test the robustness of the watermark. Actually, different weights have different effects on the robustness of the watermark. Figure 4(a) demonstrates the correlations between the NC values and cropping ratios, and Fig. 4(b) demonstrates the correlations between the NC values and JPEG compression qualities. It is observed that the two weights perform vary and each has its own strengths. The set with equal weights outperforms the set (1.0, 0.0, 0.0, 0.0) on extracting watermarks from cropped watermarked images; on the contrary, the set (1.0, 0.0, 0.0, 0.0) outperforms the extracting scheme with equal weights on extracting watermarks from compressed watermarked image. Accordingly, the set with equal weights is suitable for being against the alterations that concentrates in a region. The set (1.0, 0.0, 0.0, 0.0) means that the decision of the extracted watermark bit depends on the first element only and is suitable for being against the alterations that spread over the watermarked image. This distinct fact gives our scheme flexibility. We can juxtapose the watermarks extracted by our scheme with different weights and display the best one. Correspondingly, Fig. 2 shows the extracted watermarks with the weights (0.25, 0.25, 0.25, 0.25), and Fig. 3 shows the extracted watermarks with weights (1.0, 0.0, 0.0, 0.0).

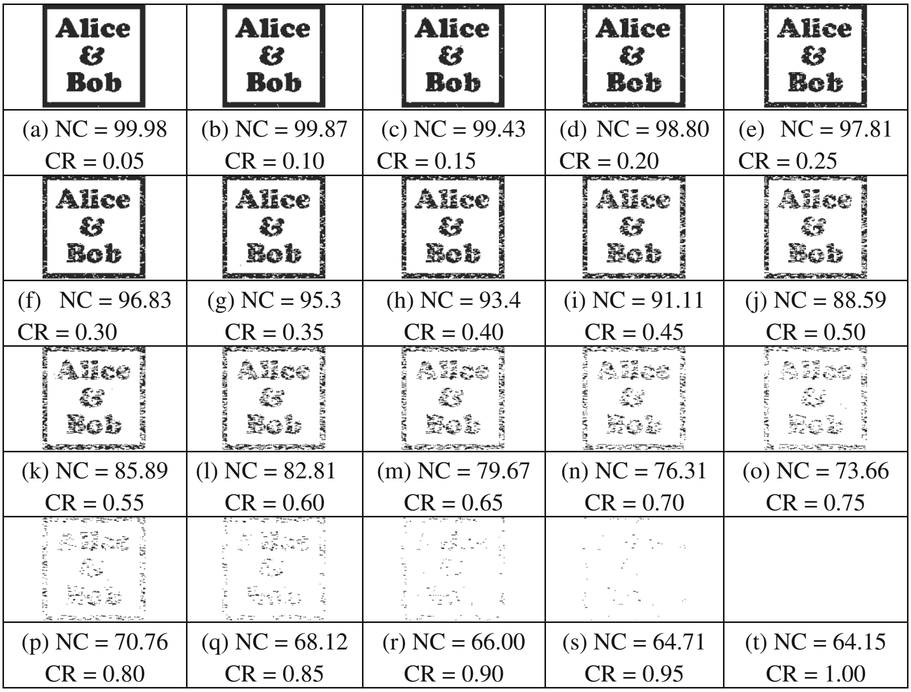


(a) Watermark

(b) Host image

(c) Watermarked image PSNR = 38.0081

Fig. 1. The watermark, the host and watermarked images



(a) NC = 99.98 CR = 0.05

(b) NC = 99.87 CR = 0.10

(c) NC = 99.43 CR = 0.15

(d) NC = 98.80 CR = 0.20

(e) NC = 97.81 CR = 0.25

(f) NC = 96.83 CR = 0.30

(g) NC = 95.3 CR = 0.35

(h) NC = 93.4 CR = 0.40

(i) NC = 91.11 CR = 0.45

(j) NC = 88.59 CR = 0.50

(k) NC = 85.89 CR = 0.55

(l) NC = 82.81 CR = 0.60

(m) NC = 79.67 CR = 0.65

(n) NC = 76.31 CR = 0.70

(o) NC = 73.66 CR = 0.75

(p) NC = 70.76 CR = 0.80

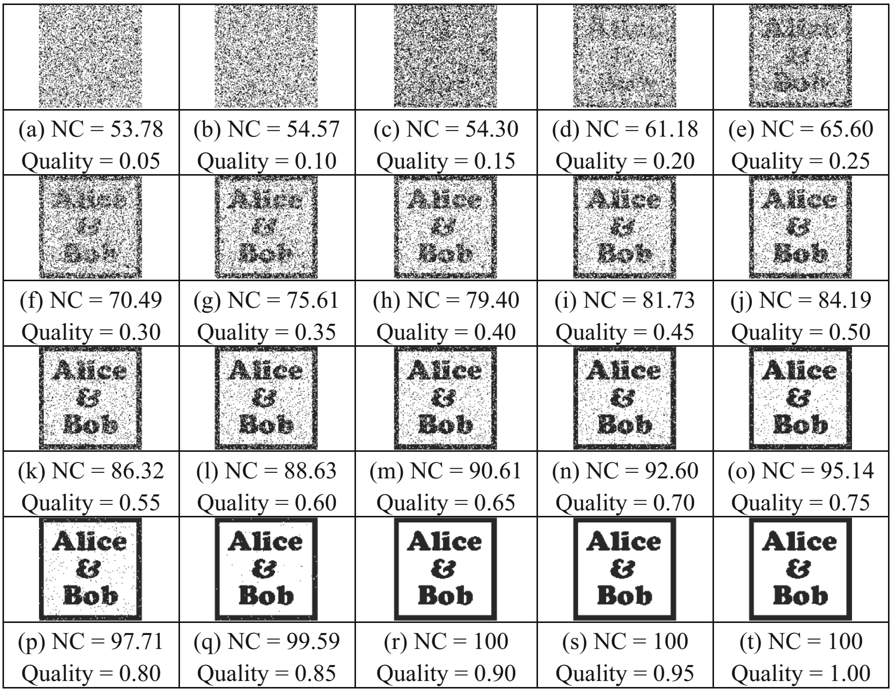
(q) NC = 68.12 CR = 0.85

(r) NC = 66.00 CR = 0.90

(s) NC = 64.71 CR = 0.95

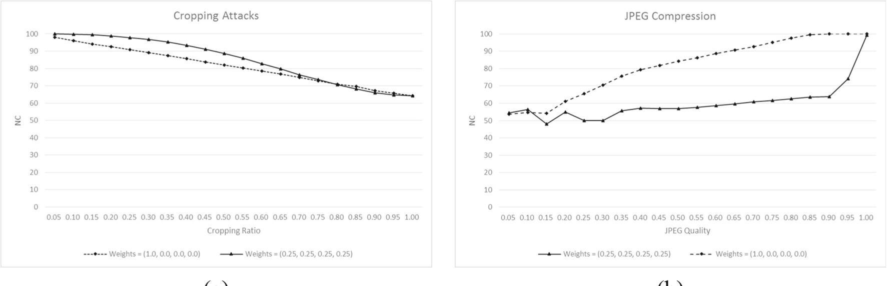
(t) NC = 64.15 CR = 1.00

Fig. 2. The NC values of extracted watermarks for cropped images with different cropping ratio values.



(a) NC = 53.78 Quality = 0.05
(b) NC = 54.57 Quality = 0.10
(c) NC = 54.30 Quality = 0.15
(d) NC = 61.18 Quality = 0.20
(e) NC = 65.60 Quality = 0.25
(f) NC = 70.49 Quality = 0.30
(g) NC = 75.61 Quality = 0.35
(h) NC = 79.40 Quality = 0.40
(i) NC = 81.73 Quality = 0.45
(j) NC = 84.19 Quality = 0.50
(k) NC = 86.32 Quality = 0.55
(l) NC = 88.63 Quality = 0.60
(m) NC = 90.61 Quality = 0.65
(n) NC = 92.60 Quality = 0.70
(o) NC = 95.14 Quality = 0.75
(p) NC = 97.71 Quality = 0.80
(q) NC = 99.59 Quality = 0.85
(r) NC = 100 Quality = 0.90
(s) NC = 100 Quality = 0.95
(t) NC = 100 Quality = 1.00

Fig. 3. The NC values of extracted watermarks for JPEG compressed images with different quality values



(a) (b)

Fig. 4. The NC values relative to different parameters

4 Conclusions

Most QR-based digital watermarking schemes suffer from being unable to fully utilize the coefficients of the R matrix. Therefore, such methods may have low data hiding capacity and watermark security. To cope with this issue, we have built formulas to compute the allowable modification ranges and employ the sine wave function in trigonometry and achieve the possible minimal modification within the allowable bounds. In the embedding scheme, the host image is decomposed by QR decomposition, and each watermark bit is redundantly embedded in the R matrix. We adopt Householder reflections for QR decomposition because it is numerical stable relative to Gram-Schmidt process. Redundantly embedding a watermark bit can increase the robustness and security of our scheme. Observing from the experiment results, our scheme indeed succeeds in enhancing the robustness against cropping attacks However, after the four copies of a watermark bit are extracted, they may be different due to possible attacks. Therefore, we design a weighted strategy to resolve the watermark bit. The experimental results show that our scheme satisfy the requirements of imperceptibility and robustness. In the future, we will test more sets of weights on extracting the watermark and give analysis of the relationship between the set of weights and the type of attacks.

References

1. Ansari, I.A., Pant, M., Ahn, C.W.: SVD based fragile watermarking scheme for tamper localization and self-recovery. Int. J. Mach. Learn. Cybern. **7**(6), 1225–1239 (2015). https://doi.org/10.1007/s13042-015-0455-1
2. Byun, S.C., Lee, S.K., Tewfik, A.H., Ahn, B.H.: A SVD-based fragile watermarking scheme for image authentication. LNCS, vol. 2613, pp. 170–178. Springer, Heidelberg (2003)
3. Cheney, W., Kincaid, D.: Linear Algebra: Theory and Applications. Jones and Bartlett, Sudbury (2009)
4. Gao, J., Fan, L., Xu, L.: Solving the face recognition problem using QR factorization. WSEAS Transl. Math. **11**(8), 712–721 (2012)
5. Hsu, C.S., Tu, S.F.: Probability-based tampering detection scheme for digital images. Opt. Commun. **283**(9), 1737–1743 (2010)
6. Hsu, C.S., Tu, S.F.: Image authentication based on QR decomposition and sinusoid. In: 11th International Conference on Computer Science and Education. IEEE Press, pp. 479–482 (2016). https://doi.org/10.1109/iccse.2016.7581627
7. Huang, H.F.: A fragile watermarking algorithm based on Chaos and QR decomposition. Intl. J. Adv. Comput. Tech. **5**(4), 117–124 (2013)
8. Kang, Q., Li, K., Chen, H.: An SVD-based fragile watermarking scheme with grouped blocks. In: 2nd International Conference on Information Technology and Electronic Commerce. IEEE Press (2014). https://doi.org/10.1109/icitec.2014.7105595
9. Katzenbeisser, S., Petitcolas, F.A.P.: Information Hiding: Techniques for Steganography and Digital Watermarking. Artech House Inc., MA, USA (2000)
10. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis, 3rd edn. Springer, New York (2002)
11. Su, O., Niu, Y., Zou, H., Zhao, Y., Yao, T.: A blind double color image watermarking algorithm based on QR decomposition. Multimed. Tools Appl. **72**(1), 987–1009 (2014)
12. Su, Q., Niu, Y., Wang, G., Jia, S., Yue, J.: Color image blind watermarking scheme based on QR decomposition. Signal Proc. **94**, 219–235 (2014)
13. Su, O., Wang, G., Zhang, X., Lv, G., Chen, B.: An improved color image watermarking algorithm based on QR decomposition. Multimed. Tools Appl. **76**(1), 707–729 (2017). https://doi.org/10.1007/s11042-015-3071-x
14. Subhedar, M.S., Manbkar, V.H.: Image steganography using redundant discrete wavelet transform and QR factorization. Comp. Elec. Eng. **54**, 406–422 (2016)
15. QR decomposition. https://en.wikipedia.org/wiki/QR_decomposition