Multimedia Tools and Applications (2024) 83:64663–64682
https://doi.org/10.1007/s11042-023-17875-5

A robust blind color watermarking algorithm based on the Radon-DCT transform

Bin Bao[1] · Yu Wang[2]

Received: 20 September 2022 / Revised: 16 October 2023 / Accepted: 13 December 2023 /
Published online: 17 January 2024
© The Author(s) 2024

Abstract
Digital watermarking is an effective technique for image copyright protection. Many digital image watermarking algorithms are sensitive to geometric distortions. These distortions make it difficult to detect and extract watermarks. In this study, a robust color watermarking algorithm combining the Radon transform and DCT transform is proposed. First, the color carrier image is converted to the YUV color space from RGB, and the U component of YUV is transformed to the Radon domain using the Radon transform. Then, a two-dimensional 8×8 discrete cosine transform (2D-DCT) is carried out on the selected blocks of the Radon domain, and some fixed midfrequency coefficients are selected to embed watermark information. Moreover, the Arnold transform is applied to encrypt the watermarks and the random permutation function to scramble the embedding positions. Finally, the watermark embedding and blind extraction processes are completed by modifying the midfrequency coefficients using the presented rules. A series of simulation experiments show that the watermark algorithm has high imperceptibility and good robustness to various attacks.

Keywords 2D-DCT transform · Geometric attack · Radon transform · Robustness

1 Introduction

With the rapid development of the Internet, the world has become increasingly digitalized. Digital media such as images and videos are easy to spread and duplicate; however, it will cause the rights and interests of digital media owners to be infringed. Therefore, the security and protection of digital media has become an important research topic [1–4]. Invisible digital watermarking is a potential method for protecting digital media because of its imperceptibility and robustness.

✉ Bin Bao
baob@shafc.edu.cn

[1] Department of Intelligent Agricultural Engineering, Shanghai Vocational College of Agriculture and Forestry, Shanghai 201699, China

[2] College of Software, Henan University of Engineering, Zhengzhou 451191, China

According to the difference of the watermark insertion domain, watermarking techniques can be classified into spatial domain watermarking methods [5–7] and frequency domain watermarking methods [8–11]. The spatial domain methods are techniques that directly manipulate image pixels to embed watermark bits. For example, a spatial domain color image watermarking method was proposed by Abraham et al. [5], which supports imperceptible watermarking and high resistance to attacks. The frequency domain watermarking methods use mathematical transforms such as redundant discrete wavelet transform (RDWT) [12],the discrete wavelet transform (DWT) [13–16], discrete cosine transform (DCT) [8, 9, 17], and discrete Fourier transform (DFT) [18] to transform the carrier image to the frequency domain, and the watermark information is embedded in the frequency coefficients. Compared with the spatial domain method, the frequency domain method has better imperceptibility and robustness.

Therefore, many frequency domain watermarking methods have been proposed. For instance, Jane and Elbaşi [13] presented a new nonblind watermarking method based on singular value decomposition (SVD) and DWT. This method embeds watermark information in the LL subband of DWT coefficients to increase the imperceptibility of the watermark and combines DWT and SVD to increase the watermark robustness. However, this algorithm is a nonblind watermarking algorithm that requires an original carrier image during the watermark extraction process and has a very weak ability to withstand rotation attacks. Ernawan and Kabir [8] proposed a reliable DCT-based digital watermarking scheme with the best DCT psychovisual threshold. This scheme uses the lowest modified entropy to choose the embedded image blocks and achieves high imperceptibility and robustness. However, the carrier images used in the scheme are grayscale images, and the embedded watermark image is a simple binary image. Currently, color images are more widely utilized than grayscale images because color images are more informative and detailed. Therefore, blind and color digital image watermarking algorithms have been presented [10, 17, 19]. For example, Yuan et al. [10] proposed a color image blind watermarking scheme based on DCT. The carrier and watermark images used in the scheme are RGB [20] color images, and all the pixel values of the watermark image are converted from decimal to binary information that is embedded into the DCT midfrequency coefficients of the red, green, and blue components of the color carrier image by modifying the selected DCT midfrequency coefficients. The watermark extraction process of this scheme does not require the original data, and therefore, it belongs to a blind watermarking scheme. However, it is not sufficiently robust against cropping, translation, and rotation attacks.

Furthermore, considering the imperceptibility and robustness of watermarks, many watermarking algorithms based on the Radon transform have been proposed to protect copyright information over the years. In [21], the FDCT, SVD, and Radon transform techniques were utilized to build a watermarking method to increase the robustness and security of watermark. In [22], Pranab Kumar et al. proposed a color image watermarking method based on Radon transform and Jordan decomposition, which has high robustness against geometrical attacks. In [23], a robust digital watermarking methodology based on Radon transform was introduced to embed watermark by adding a weak signal to the strong background of an original image.

Although many methods have been presented to enhance the performance of watermarking algorithms in some aspects, such as imperceptibility, security, or robustness [17, 24–26], achieving a balance of these aspects is still the focus of current research. In this study, a robust color image watermarking scheme based on the Radon transform [21, 22, 27] and DCT is proposed. The bit information of the color watermark image is embedded into the DCT middle-frequency coefficients of the Radon transform domain of the carrier image using the proposed rules. In the case of geometric distortion attacks, the synchronization error can be induced; therefore, the watermark is difficult to be extracted. To combat geometric distortions, the geometric correction method is proposed to recover the synchronization of watermarking. Numerical experiments show that the proposed scheme has high imperceptibility and robustness under different types of attacks.

The remainder of this paper proceeds as follows. Section 2 explains the relevant knowledge. Section 3 describes the scheme for watermark embedding and watermark extraction. Section 4 presents the experimental results and analysis. Finally, Section 5 presents the conclusion.

2 Preliminaries

2.1 Radon transform

The Radon transform is a projection of the image matrix along a specified direction. The Radon transform of a 2-D image $f(x, y)$ can be computed as follows:

$$P(\rho, \theta) = R(\rho, \theta)\left[f(x, y)\right] = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x, y)\delta(\rho - x\ cos\ \theta - y\ sin\ \theta)dx\ dy \tag{1}$$

where R $(\rho,$ $\theta)$ is the path integral of the function f $(x,$ $y)$ taken over Line L: $\rho =$ x cos $\theta +$ y sin θ, ρ is the distance between L and the origin, $\theta \in [0,$ π] is the angle between the normal of the line L and the x-axis, and δ(.) is the Dirac delta function. Because the digital image is discrete, the continuous Radon transform equation must be discretized. We use the radon() function, a built-in function of Matlab, to calculate the Radon transform of an image.

The linear discrete Radon transform (DRT) has several useful geometric properties:

Translation transforms:

$$R(\rho, \theta)\left[f\left(x - x_0, y - y_0\right)\right] = P\left(\rho - \rho_0, \theta\right) \tag{2}$$

where $\rho_0 = x_0$ cos $\theta + y_0$ sin θ.

Rotation transforms:

$$R(\rho, \theta)\left[f\left[\left(x\ cos\ \theta_0 + y\ sin\ \theta_0, -x\ cos\ \theta_0 + y\ sin\ \theta_0\right)\right]\right] = P\left(\rho, \theta + \theta_0\right) \tag{3}$$

where θ_0 is the angle of image rotation. If an image $f(x, y)$ is rotated by the angle θ_0, $R(\rho, \theta)$ is also translated according to same size in the θ direction.

Scale transforms:

$$R(\rho, \theta)\left[f(\lambda x, \lambda y)\right] = \lambda^{-1} P(\lambda\rho, \theta) \tag{4}$$

where λ is the scale factor of the image. If an image $f(x, y)$ is resized as λ, $R(\rho, \theta)$ is also changed with same size.

Therefore, the Radon transform has the translation, rotation and scale invariant properties. Furthermore, the Radon transform is line integrals of the image. For these reasons, embedding watermark in Radon domain will provide good robustness to various types of attacks.

2.2 The discrete cosine transform

Discrete cosine transform (DCT) is a method which can transform an image into frequency domain, and DCT has the advantages of good performance, accuracy, and good energy concentration. So DCT has been widely employed in image processing applications. To improve the DCT transform performance, the carrier image is first divided into subblocks of size 8×8, and then DCT is performed on each block. The DCT results include three frequency subbands: high-frequency, middle-frequency, low-frequency. Because low-frequency coefficients carry important perceptual information, embedding watermarks in them causes obvious image distortion. Embedding in high-frequency coefficients makes the watermarked image vulnerable to low-pass filtering because high-frequency coefficients include detailed information. Therefore, the middle-frequency DCT block coefficients are often selected to embed the watermark [5, 8, 19].

Two-dimensional discrete cosine transform (2D-DCT) can be calculated through 1D-DCT along columns and rows of images separately. The 2D-DCT of an $N \times N$ image $f(x, y)$ is defined by Eq. (5):

$$F(u, v) = \alpha(u)\alpha(v) \times \left[\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \times \cos\frac{(2x+1)u\pi}{2N} \cos\frac{(2y+1)v\pi}{2N} \right] \tag{5}$$

where,

$$\alpha(u) = \begin{cases} \sqrt{1/N}, & u = 0 \\ \sqrt{2/N}, & u = 1, 2, \dots, N-1 \end{cases} \tag{6}$$

x and y are the coordinates in the spatial domain, u and v are the coordinates in the frequency domain, $F(u, v)$ is the frequency coefficient at coordinate (u, v), and $\alpha(v)$ is similar to $\alpha(u)$. The two-dimensional inverse discrete cosine transform (2D-IDCT) is defined as follows:

$$f(x, y) = \alpha(u)\alpha(v) \times \left[\sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \times \cos\frac{(2x+1)u\pi}{2N} \cos\frac{(2y+1)v\pi}{2N} \right] \tag{7}$$

2.3 Arnold transform

To improve the watermarking scheme security, the watermark image needs to be scrambled before watermark embedding. The Arnold scrambling transform is an effective method for scrambling watermark images to ensure watermark algorithm security. The two-dimensional Arnold transform is defined as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^n \begin{bmatrix} x \\ y \end{bmatrix} mod\ N \tag{8}$$

where (x, y) is the position of a pixel in the original watermark image, (x', y') is the position of the pixel in the scrambled watermark image, N is the size of the watermark image, n is the number of iterations that is considered as the private key K_a to strengthen security, and the entries of the transformation matrix are set to $a=b=c=1$, $d=2$. An inverse

Arnold transform is applied to descramble the scrambled watermark image to retrieve the original watermark image.

3 The proposed scheme

3.1 Watermark embedding process

The detailed steps of the watermark embedding process are given in Fig. 1.

Step 1: Convert a carrier image I with a size of $M \times M$ from RGB to YUV color space. As the human eye is more sensitive to brightness than color, the watermark is embedded into the U component of the YUV color space. Employ the Radon transform on the U component to obtain the Radon domain and divide it into non-overlapping blocks of size 8×8.

Step 2: A 24-bit RGB color watermark image W with size of $N \times N$ is split into three different components (red, green, and blue). To enhance the watermark security, each component is scrambled by the Arnold transform using the private key K_{a_i}, $i=1,2,3$. Each decimal pixel value of the image' three components is converted into an eight-bit binary number and then all the binary numbers are sequentially concatenated to obtain a binary sequence S of length $24N^2$.

Step 3: In order to enhance the security level of digital watermarking algorithm, we use MATLAB's built-in function randperm(n,k) to select k unique random numbers from $\{1,2,\ldots,n\}$, where n is greater than k and less than the total number of blocks in the Radon domain, k unique random numbers are used to select the embedded blocks, and two integers n, k can be used as key K_b. Every selected block is then transformed using 2D-DCT.

Step 4: Embed the watermark.

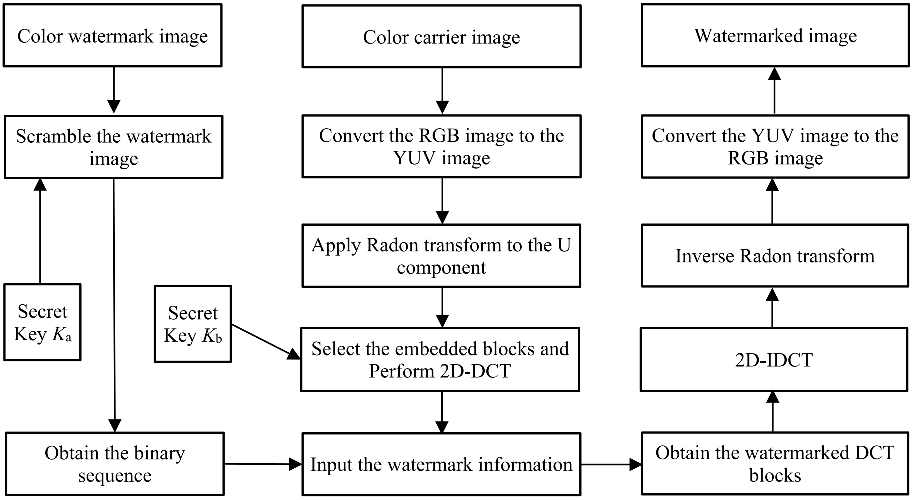

Color watermark image
Color carrier image
Watermarked image
Scramble the watermark image
Convert the RGB image to the YUV image
Convert the YUV image to the RGB image
Apply Radon transform to the U component
Inverse Radon transform
Secret Key Ka
Secret Key Kb
Select the embedded blocks and Perform 2D-DCT
2D-IDCT
Obtain the binary sequence
Input the watermark information
Obtain the watermarked DCT blocks

Fig. 1 Diagram of the watermark embedding process

Considering the tradeoff between robustness and imperceptibility, the watermark is embedded into the DCT midfrequency coefficients of the selected blocks. In the selected DCT block, two pairs of DCT midfrequency coefficients with fixed positions (1,7) and (7,1), (1,5) and (5,1) are selected.

Two binary watermark bits w_i are selected from S successively, and Eqs. (9) and (10) are used to modify DCT midfrequency coefficients pairs (c_{i1}, c_{i2}) to ($c^*_{i1}c^*_{i2}$,) for embedding the watermark bit w_i, where $i = 1$,2,respectively. The modified pairs (c^*_{i1} , c^*_{i2}) are calculated as follow:

$$c^*_{i1} = \begin{cases} signs\left(c_{i1}\right) \times (avg - 0.25 \times T), & if\ w = 1\ and\ abs\left(c_{i2}\right) - abs(c_{i1})) < \alpha \times T \\ signs\left(c_{i1}\right) \times (avg + 0.25 \times T), & if\ w = 0\ and\ abs\left(c_{i1}\right) - abs(c_{i2})) < \alpha \times T \\ c_{i1}, & otherwise \end{cases} \quad (9)$$

$$c^*_{i2} = \begin{cases} signs\left(c_{i2}\right) \times (avg + 0.25 \times T), & if\ w = 1\ and\ abs\left(c_{i2}\right) - abs(c_{i1})) < \alpha \times T \\ signs\left(c_{i2}\right) \times (avg - 0.25 \times T), & if\ w = 0\ and\ abs\left(c_{i1}\right) - abs(c_{i2})) < \alpha \times T \\ c_{i2}, & otherwise \end{cases} \quad (10)$$

where, $avg = (abs\left(c_{i1}\right) + abs\left(c_{i2}\right))/2$, $abs(.)$ function returns the absolute value of a number, T is called the quantization step which is obtained through several experiments, signs(.) is defined as follow:

$$signs(x) = \begin{cases} 1, & if\ x \geq 0 \\ -1, & otherwise \end{cases} \quad (11)$$

Step 5: Finally, all the modified midfrequency coefficients are merged into the corresponding selected DCT blocks. A 2D-IDCT is performed on each selected block, and then the inverse Radon transform is used to obtain the embedded U component. Three components of the YUV color space are combined, and the YUV is converted to the RGB color space to obtain the watermarked image.

3.2 Watermark extraction process

Figure 2 shows the watermark extract process. It first checks whether the synchronization of watermark is destroyed by geometric attacks, followed by the meaningful watermark extraction. The specific steps are described below:

Step 1: Determine whether there is a geometric attack on a watermarked image. If yes, then correct the watermarked image; otherwise, perform the Radon transform on the U chrominance of the YUV color space of the watermarked image to obtain Radon domain data.

Step 2: Obtain the extracted blocks from the Radon domain data using the random permutation generated by function randperm(n,k) and then apply DCT to these blocks.

Step 3: According to the relationship between the absolute values of coefficients pair (c^*_{i1}, c^*_{i2}), the watermark bit is extracted according to Eq. (12):

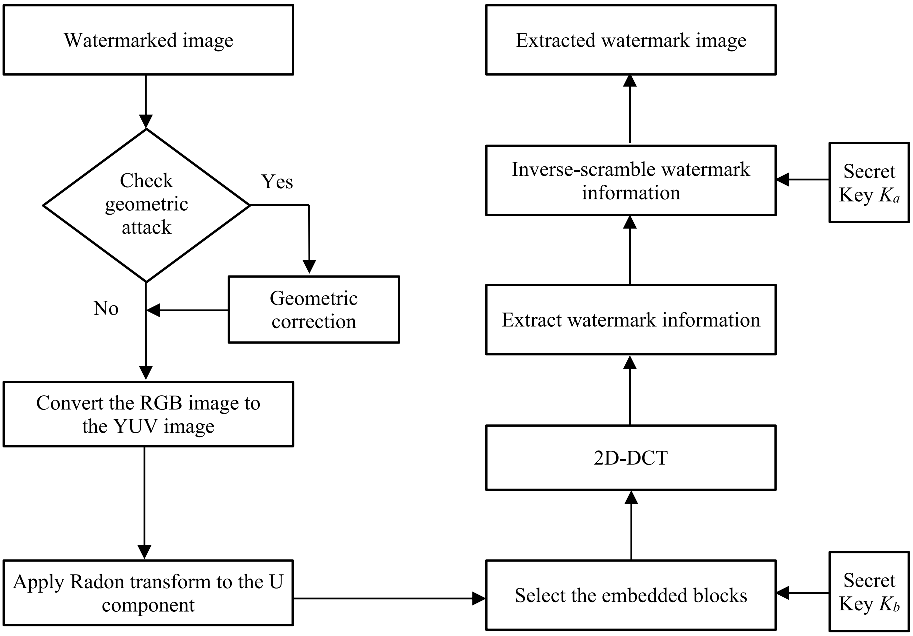

Watermarked image
Check geometric attack
Yes
No
Geometric correction
Convert the RGB image to the YUV image
Apply Radon transform to the U component
Select the embedded blocks
Secret Key K_b
2D-DCT
Extract watermark information
Inverse-scramble watermark information
Secret Key K_a
Extracted watermark image

Fig. 2 Diagram of the watermark extraction process

$$w_i^* = \begin{cases} 1, & if\ abs\left(c_{i1}^*\right) < abs\left(c_{i2}^*\right) \\ 0, & otherwise \end{cases} \tag{12}$$

Then, the scrambled watermark image is produced.

Step 4: Obtain the color watermark image through the inverse Arnold transform on the scrambled watermark image with secret key K_a.

3.3 Geometric correction

After the watermarked image is translated, scaled, or rotated, the synchronization of the embedded watermark is changed. To extract the watermark, it is necessary to know the pixels and direction of translation, the scale factor, and the rotation angle. Using these parameters, we can recover the image to its original size or orientation before watermark extraction.

Different from the invariant methods [28, 29], we calculate the rotation feature of an image based on the positions of the first nonzero pixels of the image. After an image is rotated, the four corners of the image are filled with black pixels with a pixel value of 0. Then the image is cropped, and its size remains unchanged. Figure 3b shows the rotated image after rotating the Lena image shown in Fig. 3a counterclockwise to a size of 512×512. Therefore, the rotation angle of the rotated image can be calculated according to the positions of the nonzero pixels.

As shown in Fig. 3b, the first nonzero pixel point (x_1, x_2) in the first row and the first nonzero pixel point (x_2, y_2) in the first column can be obtained according to the pixel position index, and the rotation angle θ is given as follows:

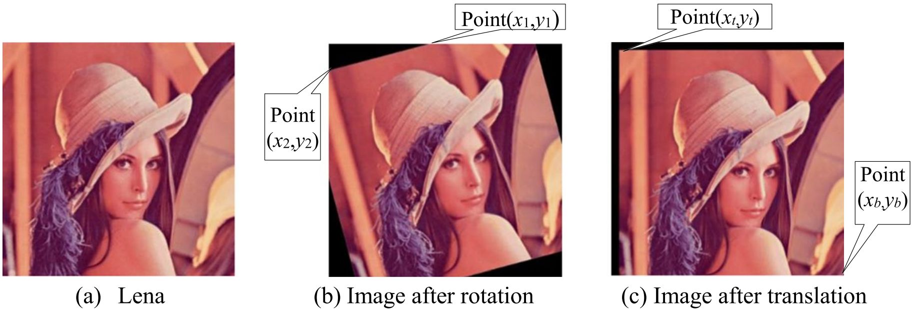


Fig. 3 Image rotation, translation

$$\theta = arctan\frac{x_2 - x_1}{y_2 - y_1} \tag{13}$$

Rotating the image clockwise by θ degrees, the image can be corrected.

There are typically eight ways to translate an image: right, left, up, down, top left, bottom left, top right, and bottom right. The translated image maintained its original size. Figure 3c is the "Lena" image in Fig. 3a, translated by 20 pixels along the horizontal direction, and 20 pixels along the vertical direction.

For an image $I(x,y)$ of size $N \times N$, if the translated image is represented by $I'(x,y)$, we can obtain the position (x_t, y_t) of the upper-left corner and the position (x_b, y_b) of the lower-right corner in the nonzero pixel area of $I'(x,y)$. If $x_t = y_t = 1$ and $x_b = y_b = N$, the image is not translated; if $x_t > 1$, it is translated down; if $y_t > 1$, it is translated right; if $x_b < N$, it is translated up; if $y_b < N$, it is translated left. Therefore, we can calculate the direction and distance of the image translation and then correct $I'(x,y)$.

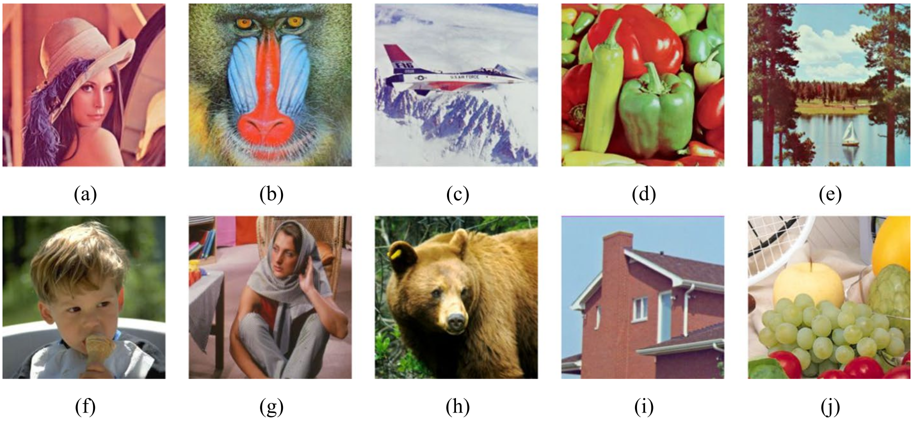
After the watermarked image is scaled, we only need to recover it to its original size. The scaling attack has very little effect on the embedded watermark in Radon domain. Therefore, the watermark can be effectively extracted from the watermarked image using the proposed algorithm.

4 Experimental results and discussion

Usually, the performance of watermarking algorithm is evaluated from two aspects of imperceptibility and robustness. To evaluate the robustness and imperceptibility of the proposed watermarking algorithm, ten 24-bit 512×512 color images (Lean, Baboon, Airplane, Peppers, Lake, Kid, Barbara, Bear, House, and Fruits) shown in Fig. 4a–j are used as the carrier images in this study. These carrier images were selected from standard digitized image databases (USC-SIPI [30] and CVG-UGR [31]). Two color images with a size of 32×32, as shown in Fig. 5a and b, were used as watermark images.

The watermark imperceptibility reflects the ability of watermarking algorithm to hide the watermark in the carrier image. In general, the peak signal-to-noise ratio (PSNR) [18] and structural similarity index measurement (SSIM) [10, 32] were used to assess the imperceptibility of digital watermark.

PSNR is used to calculate the pixel difference between the original image and the degraded image and is a widely used objective assessment instrument for watermarked image quality. The higher the PSNR value, the better the quality of the watermarked


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 4 Carrier images: **a** Lena, **b** Baboon, **c** Airplane, **d** Peppers, **e** Sailboat, **f** Kid, **g** Barbara, **h** Bear, **i** House **j** Fruits

Fig. 5 Watermark images: **a** watermark 1, **b** watermark 2

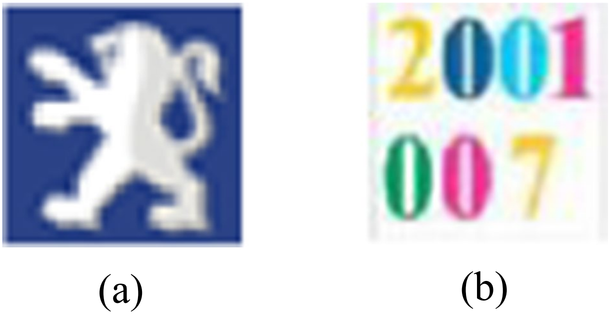


image and the better the watermarking algorithm. The PSNR for color digital images is defined as follows:

$$PSNR = 10lg\frac{N_1 \times N_2 \times M \times 255^2}{\sum_{x=1}^{N_1}\sum_{y=1}^{N_2}\sum_{j=1}^{M}\left[I(x,y,j) - I^*(x,y,j)\right]^2} \tag{14}$$

where N_1 and N_2 are the number of rows and columns of original carrier image, M is the number of image' components, $I(\mathrm{x},y,j)$ represents the pixel value of the component j of the carrier image at the (x,y) location, and $I^*(x,y,j)$ represents the pixel value of the component j of the watermarked image at the (x,y) location.

The SSIM is a newer measurement tool for the similarity between two images. It was designed based on the characteristics of the human visual system. The value of SSIM is between 0 and 1, and it can effectively measure the quality of images. The higher the SSIM value is, the better the quality of the watermarked image. The SSIM is defined as follows:

$$SSIM(I,I^*) = l(I,I^*)c(I,I^*)s(I,I^*) \tag{15}$$

where

$$\begin{cases} l(I,I^{*}) = \left(2\mu_I\mu_{I^{*}} + C_1\right)/\left(\mu_I^2 + \mu_{I^{*}}^2 + C_1\right) \\ c(I,I^{*}) = \left(2\sigma_I\sigma_{I^{*}} + C_2\right)/\left(\sigma_I^2 + \sigma_{I^{*}}^2 + C_2\right) \\ s(I,I^{*}) = \left(\sigma_{II^{*}} + C_3\right)/\left(\sigma_I\sigma_{I^{*}} + C_3\right) \end{cases} \tag{16}$$

$l(I,I^{*})$, $c(I,I^{*})$ and $s(I,I^{*})$ are functions that compare the luminance, contrast, and structures of image I and image I^{*}, respectively.

In addition, normalized cross-correlation (NC) is an objective criterion for evaluating the robustness of the watermarking algorithm. The value of NC is between 0 and 1.The higher the NC value, the higher the similarity between two images. The NC value is calculated as follows:

$$NC = \frac{\sum_{j=1}^{3}\sum_{x=1}^{M}\sum_{y=1}^{N}\left(W(x,y,j) \times W'(x,y,j)\right)}{\sqrt{\sum_{j=1}^{3}\sum_{x=1}^{M}\sum_{j=1}^{N}\left[W(x,y,j)\right]^2}\sqrt{\sum_{j=1}^{3}\sum_{x=1}^{M}\sum_{j=1}^{N}\left[W'(x,y,j)\right]^2}} \tag{17}$$

where W' represents the extracted color digital watermark image, W represents the original color digital watermark image, M and N represent the size of the watermark image, and j represents the index of the components of the color image.

In the watermark embedding process, the quantization step T is an important parameter that has a great impact on the imperceptibility and robustness of watermarking algorithm. In order to select the parameter T, we conduct many experiments. Experiments show that when α is 1 and T is in the range of [300,400], the trade-off between imperceptibility and robustness can be achieved. Therefore, the value of T is set to 350 in this algorithm.

4.1 The imperceptibility test and analysis

An effective watermarking algorithm must ensure high imperceptibility and robustness. To test the imperceptibility of the proposed algorithm, the watermark images in Fig. 5a and b are respectively embedded into the carrier image in Fig. 4a–j. Some test results of the watermarked images and the extracted watermark images are shown in Fig. 6. Table 1 lists the PSNR and SSIM values of the watermarked carrier images and the original carrier images.

It can be seen from Fig. 6 and Table 1 that the average value of PSNR is greater than 37db, and the average value of SSIM is greater than 0.93. Moreover, all the watermarks are completely extracted. This shows that the original carrier images are similar to the watermarked images, and the watermark cannot be perceived by the human perception system. Thus, the proposed algorithm exhibits good imperceptibility.

To further evaluate the imperceptibility of our algorithm, the color digital watermark 1 in Fig. 5a is respectively embedded into the carrier image "Lena" in Fig. 4a and carrier image "Peppers" in Fig. 4d, the color digital watermark 2 in Fig. 5b is respectively embedded into the carrier image "Sailboat" in Fig. 4e and carrier image "Kid" in Fig. 4f. Then, our watermarking algorithm was compared with other watermarking algorithms. The comparative results of watermark imperceptibility between different watermarking algorithms are listed in Table 2. The comparative results of NC between different algorithms without the attack are listed in Table 3. From Tables 2 and 3, we can see that this watermarking algorithm achieves a good trade-off between the imperceptibility and robustness.

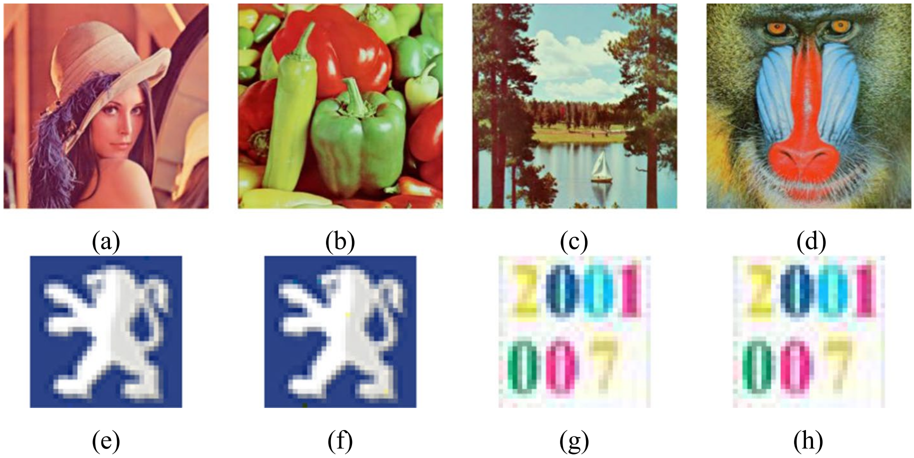


Fig. 6 Watermarked images and extracted watermarks: (**a**–**d**) watermarked image of Lena, Peppers, Sailboat, Baboon (**e**–**f**) extracted watermark of Lena, Peppers, Sailboat, Baboon.

Table 1 The test results (PSNR/ SSIM) of watermark imperceptibility

	watermark 1	watermark 2
Lena	38.7824/0.9417	39.1794/0.9511
Baboon	37.5208/0.9676	38.1026/0.9747
Airplane	38.0927/0.9025	38.8441/0.9463
Peppers	38.2924/0.9284	39.1626/0.9491
Sailboat	38.0173/0.9360	38.7124/0.9512
Kid	38.7549/0.9266	39.6340/0.9456
Barbara	38.6107/0.9518	39.4036/0.9662
Bear	38.1322/0.9186	38.9768/0.9366
House	38.1654/0.9204	38.9408/0.9443
Fruits	38.7491/0.9380	39.6832/0.9561

Table 2 The comparative results (PSNR/SSIM) of imperceptibility between different algorithms

	Algorithm [6]	Algorithm [9]	Algorithm [10]	Algorithm [24]	Proposed Algorithm
Lena	38.0535/0.9414	48.6761/0.9956	37.5851/0.9350	39.4385/0.9656	38.7824/0.9417
Peppers	37.6262/0.9231	40.7425/0.9708	38.0702/0.9241	38.3367/0.9299	38.2924/0.9284
Kid	36.8241/0.9231	41.6958/0.9857	37.6455/0.9250	34.4638/0.8650	39.6340/0.9456
Sailboat	37.7440/0.9458	45.7757/0.9933	38.4066/0.9414	34.2340/0.8766	38.7124/0.9512

4.2 The robustness test and analysis

The robustness of watermark means that the watermark can still be extracted under geometric attacks or other malicious attacks, which is crucial for watermarking technology.

For the purpose of measuring the robustness of the proposed watermarking algorithm, we embedded two watermark images in Fig. 5 into the carrier images "Lena" and "Peppers"

Table 3 The comparative results of NC between different algorithms

	Algorithm [6]	Algorithm [9]	Algorithm [10]	Algorithm [24]	Proposed Algorithm
Lena	1.0000	0.9475	1.0000	0.9937	1.0000
Peppers	1.0000	0.8568	1.0000	0.9826	1.0000
Kid	1.0000	0.8441	0.9997	0.9771	1.0000
Sailboat	1.0000	0.8746	1.0000	0.9817	1.0000

in Fig. 4, respectively, and then performed different types of attacks on the watermarked images, such as geometric attacks, adding noise, median filtering, low-pass filtering, JPEG compression, sharpening, and JPEG2000 compression. NC can be is used to determine the degree of similarity between the extracted and original watermark images. The higher the NC value, the higher the extracted watermark image quality. Figures 7 and 8 show the extracted watermarks and their NC values under a series of attacks. It can be noticed that the majority of extracted watermarks are of high quality.

To demonstrate the robustness of our algorithm further, the watermark image 1 in Fig. 5 is embedded into the carrier image “Lena” in Fig. 4a, and the robustness of this algorithm is compared with related algorithms [6, 9, 10, 24] under various attacks. The results of the simulation experiments are shown in Figs. 9, 10, 11 and 12.

Figure 9 presents the comparative results under JPEG and JPEG2000 compression attacks. This indicates that the proposed algorithm effectively resists compression attacks.

Figure 10 presents the comparative results of applying the salt and pepper noise attack with 0.002 of intensity and the Gaussian noise attack with mean 0 and variance 0.001. It is evident that the proposed algorithm shows better robustness against in resisting noise attacks than other related algorithms. Figure 11 shows the comparison results of applying the Butterworth low-pass filtering attack with filter order 10 and the 3×3 median filtering attack on the watermarked images. This indicates that the proposed algorithm effectively resists these filtering attacks.

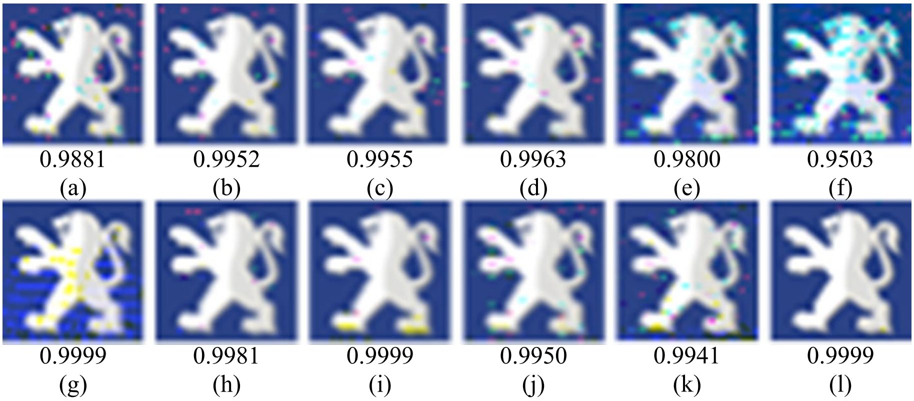


Fig. 7 Robustness comparison results under different types of attacks: (**a**) JPEG (Q = 60) (**b**) JPEG2000 (4:1) (**c**) Salt & Peppers noise (1%) (**d**) Gaussian noise (0, 0.003) (**e**) Median filtering (3×3) (**f**) Butterworth low-pass filter (100,6) (**g**) Cropping (25%) (**h**) Zoom-in (400%) (**i**) Translation (30, 20) (**j**) Rotation (30°) (**k**) Rotation (45°) (**l**) sharpening (1,0.8)

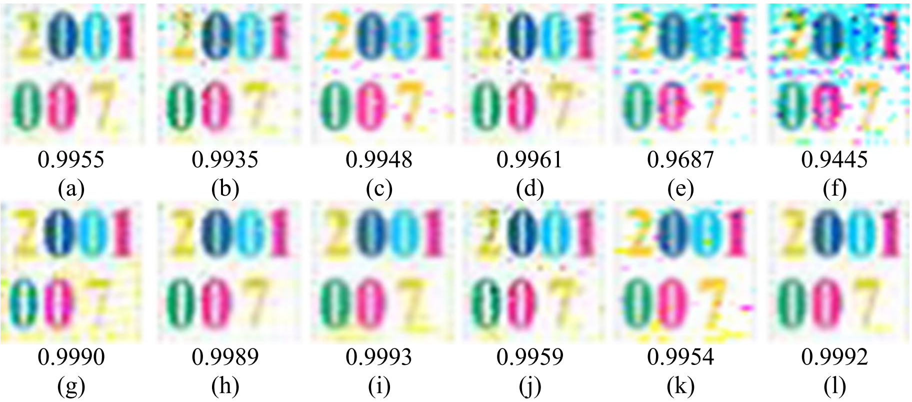


Fig. 8 Robustness comparison results under different types of attacks: (**a**) JPEG (Q = 60) (**b**) JPEG2000 (5:1) **c** Salt & Peppers noise (1%) (**d**) Gaussian noise (0, 0.003) (**e**) Median filtering (3 × 3) (**f**) Butterworth low-pass filter (100,6) (**g**) Cropping (25%) (**h**) Zoom-in (4:1) (**i**) Translation (30, 20) (**j**) Rotation (30°) (**k**) Rotation(45°) (**l**) sharpening(1,0.8)

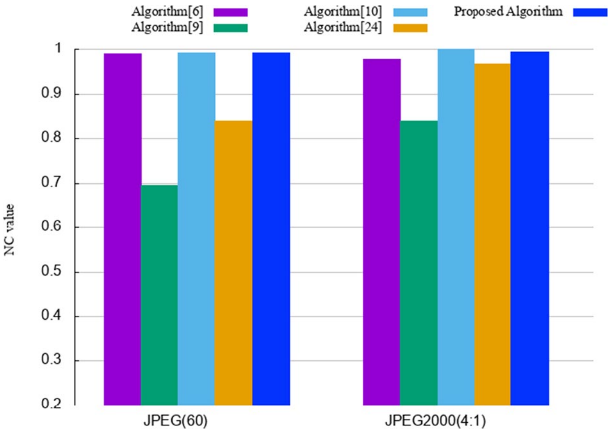


Fig. 9 Robustness comparison results under compression attacks

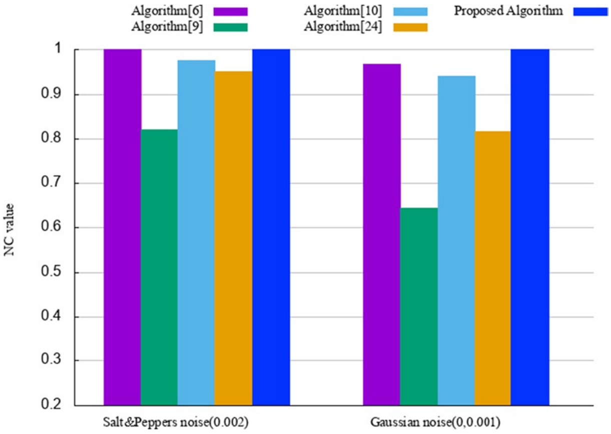


Fig. 10 Robustness comparison results under noise attacks

Fig. 11 Robustness comparison results under filtering attacks

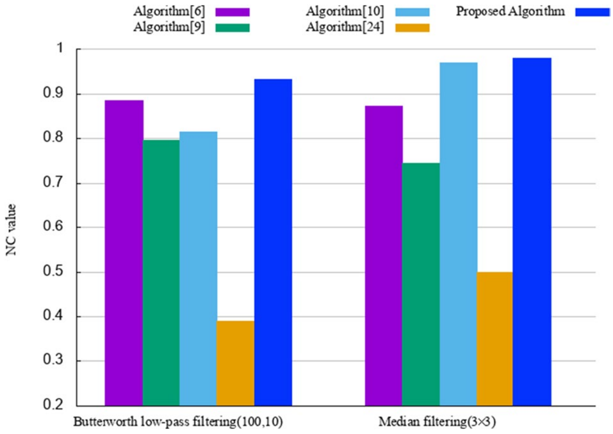


Fig. 12 Robustness comparison results under cropping attacks

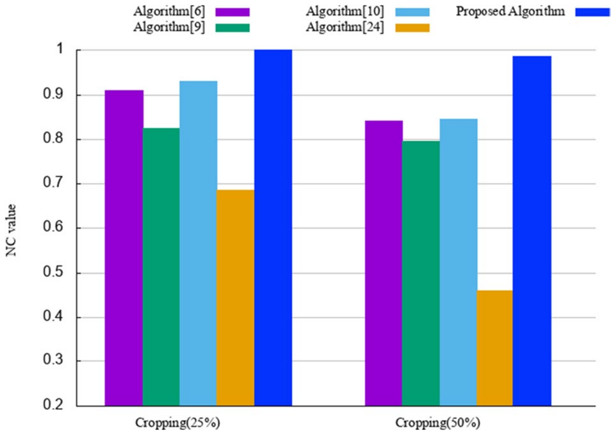


Image geometric attacks are to distort or manipulate an image. After geometric attacks, the quality of copyright images will greatly decline. Image cropping attack is one of common geometric attacks. Figure 12 shows the NC values of the extracted watermark images of different algorithms under attacked by cropping attacks with cropping ratios of 25% and 50%. Figure 13 shows the comparative results under scaling attacks with scaling ratios of 400% and 50%. It can be clearly observed that the presented algorithm has higher robustness for resisting cropping and scaling attacks than other related algorithms.

The translation or rotation attack is one of the common geometric attacks which can also damage the watermark information embedded in the image. Table 4 lists the experimental results when the watermarked images are intentionally attacked by translation with 30 pixels along the horizontal direction and 40 pixels along the vertical direction, (-30, 40), and (30, 40), respectively, and by rotation with 40 degrees clockwise, 75^0 clockwise and 40° counterclockwise, respectively. Because the Radon transform is a line integral transform and has geometric invariability, in the face of noise and geometric attacks, the Radon domain information is less affected. Therefore, the watermark can be completely extracted.

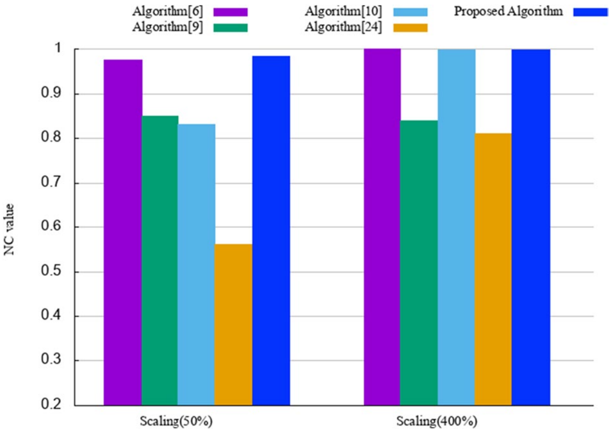


Fig. 13 Robustness comparison results under scaling attacks

Table 4 The extracted watermark (NC) under translation and rotation attacks

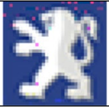
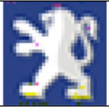
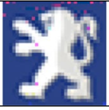
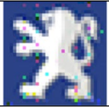
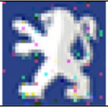
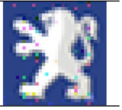
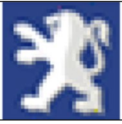
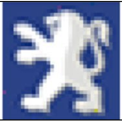
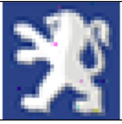
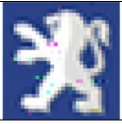
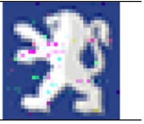
Attacks	Translation (30,40)	Translation (-30, -40)	Translation (30, -40)	Rotation (40^0)	Rotation (75^0)	Rotation (-40^0)
Extracted watermark						
NC	0.9945	0.9973	0.9933	0.9941	0.9919	0.9935

Table 5 The attacked image and extracted watermark (NC) under sharpening attacks

Attacked image					
	sharpening (1,0.8)	sharpening (2,2)	sharpening (3,3)	sharpening (4,4)	sharpening (5,5)
Extracted watermark					
NC	0.9999	0.9998	0.9995	0.9987	0.9971

As shown in Figs. 10, 12, and 13 and Table 4, this algorithm is highly robust against noise and geometric attacks.

Image sharpening is a common image processing technique that enhances the edges and contrast of an image. It is also commonly used to attack the watermark images. Table 5 shows the watermarked image and its corresponding extracted watermark under sharpening attacks with different values of the radius and amount parameters. It can be seen that the watermarked image is greatly changed by the sharpening, and the proposed algorithm can still effectively extract the watermark, which shows that our algorithm has strong resistance to sharpening attacks.

Table 6 The comparison of average NC values with different algorithm against different types of attacks

Algorithms	Algorithm [6]	Algorithm [9]	Algorithm [10]	Algorithm [24]	Proposed Algorithm
Average NC	0.9423	0.7694	0.9430	0.7156	0.9769

Table 7 The comparison of NC values with different algorithm against various attacks

Attacks	Algorithm [15]	Algorithm [11]	Algorithm [18]	Proposed Algorithm
PSNR	51.15	39.88	40.00	40.69
No Attacks	0.9992	1.000	1.000	1.000
Salt &Peppers noise(density = 0.01)	0.9715	0.9406	0.9660	0.9934
Gaussian noise(variance = 0.0005)	0.9462	0.8796	0.8585	0.9756
JPEG compression(QF = 75)	-	0.9956	-	0.9891
JPEG compression(QF = 50)	0.9831	0.9759	0.7950	0.7862
Cropping 1/4	0.9561	0.8939	0.9779	0.9747
Scaling 0.8	0.9677	0.9960	0.9358	0.9731
Gaussian filter(3 × 3)	0.9611	0.9971	0.9912	0.9983
Median filter(3 × 3)	0.9797	0.9425	0.9047	0.9662
Average filter(3 × 3)	0.9711	0.9422	0.9833	0.9810
Histogram equalization	0.9233	0.9653	-	0.9977
Average NC	0.9659	0.9571	0.9347	0.9668

Furthermore, a comparison of the average NCs of the algorithms under different types of attacks in Figs. 9, 10, 11, 12 and 13 is shown in Table 6. It is clear that the proposed algorithm achieves the highest average NC value compared to the other algorithms [6, 9, 10, 24].

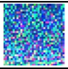
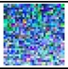
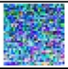
In order to further verify the robustness of the algorithm, we compare our results with algorithms [11, 15, 18] with the color Lena image as carrier image and the binary image of size 64 × 64 as watermark image under different types of attacks. Table 7 shows the comparison results. Compared with algorithms [11, 15, 18], the proposed algorithm has better resistance to the Salt&Peppers noise, Gaussian noise, Gaussian filter, and histogram equalization attacks. Compared with their methods, this algorithm has weak performance in JPEG compression (QF = 50) attack but better overall performance in resisting various attacks. Therefore, the comparisons show that the proposed algorithm is as robust as the state-of-art approaches.

4.3 The security analysis

4.3.1 Key space analysis

In this paper, in order to enhance the security of the watermarking algorithm, color watermark image is divided into three components: red, green and blue. Arnold transform is applied to scramble each component using private key K_a. Because color watermark image has three

Table 8 NC of extracted watermark with wrong keys

Key change	False K_a, Correct K_b	Correct K_a, False K_b	False K_a, False K_b
Watermark			
NC	0.5783	0.6103	0.6023

components, and the key space of each K_{a_i} is 15-bit, thus the key space of this scrambling transform is 45-bit.

In addition, in the watermark embedding, the randperm(n,k) function with the private key K_b is used to generate a random permutation to select randomly the DCT blocks for embedding watermark. The private key K_b is determined by two integers n, k and the maximum value of them is limited by the number of blocks. Therefore the key space of each of n and k is 14-bit. Thus, the key space of the key K_b is 28-bit.Thus, the total key space of the proposed algorithm determined by the keys K_a and K_b is 73-bit, which is strong enough to withstand brute force attack.

4.3.2 Key sensitivity analysis

A secure watermark algorithm should also have good key sensitivity performance. When a false key is used to decrypt the encrypted image, the decrypted image should be completely different from the original watermark. The keys K_a and K_b were changed to test the key sensitivity of this algorithm. The simulation results are shown in Table 8. It can be seen from Table 8 that the watermarks extracted with the false keys are fuzzy and have very low NC values. This experiment indicates that this algorithm has good key sensitivity performance and high security.

5 Conclusion

In this study, we presented a robust watermarking algorithm for color images based on the Radon transform and DCT transform. It has three main features: (1) In order to improve the robustness against common attacks, especially geometric and noise attacks, the Radon transform is applied to embed watermarks. (2) 24-bit color images are used as watermark images instead of the grayscale images. (3) The Arnold transform and the random permutation function are applied to enhance the security performance of the proposed algorithm. The experimental results prove that the presented algorithm has good imperceptibility and high security, and it has also strong robustness against different types of attacks. But there are still some limitations in this algorithm. When there are black pixels at the edge of a watermarked image subjected to rotation attack, the rotation angle calculated based on black pixels is not accurate enough. In future research, authors will focus on the methods of obtaining accurate rotation angle. Moreover, the application and performance enhancement of this algorithm are also the focus of later works.

Acknowledgements This work was supported by the Shanghai Committee of Science and Technology, China (No. 20392001300). The authors would like to thank anonymous referees for their valuable comments and suggestions, which lead to substantial improvements of this paper.

Data availability The authors declare data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Makbol NM, Khoo BE, Rassem TH, Loukhaoukha K (2017) A new reliable optimized image watermarking scheme based on the integer wavelet transform and singular value decomposition for copyright protection. Inf Sci 417:381–400. https://doi.org/10.1016/j.ins.2017.07.026
2. Sangeetha N, Anita X (2018) Entropy based texture watermarking using discrete wavelet transform. Optik 160:380–388. https://doi.org/10.1016/j.ijleo.2018.01.136
3. Chopra J et al (2018) An efficient watermarking for protecting signature biometric template. 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN). IEEE. https://doi.org/10.1109/SPIN.2018.8474269
4. Gawande U, Golhar Y, Hajari K (2017) Biometric-based security system: Issues and challenges. Intelligent Techniques in Signal Processing for Multimedia Security, pp 151–176. https://doi.org/10.1007/978-3-319-44790-2_8
5. Abraham J, Paul V (2019) An imperceptible spatial domain color image watermarking scheme. J King Saud Univ Comput Inf Sci 31(1):125–133. https://doi.org/10.1016/j.jksuci.2016.12.004
6. Su Q et al (2019) New rapid and robust color image watermarking technique in spatial domain. IEEE Access 7:30398–30409. https://doi.org/10.1109/access.2019.2895062
7. Garg M, Ubhi JS, Aggarwal AK (2019) Steganography and its advancements in spatial domain. No. 2177. EasyChair. https://easychair.org/publications/preprint/t5Cr
8. Ernawan F, Kabir MN (2018) A robust image watermarking technique with an optimal DCT-psychovisual threshold. IEEE Access 6:20464–20480. https://doi.org/10.1109/access.2018.2819424
9. Su Q, Wang G, Jia S, Zhang X, Liu Q, Liu X (2015) Embedding color image watermark in color image based on two-level DCT. Signal Image Video Process 9(5):991–1007. https://doi.org/10.1007/s11760-013-0534-2
10. Yuan Z, Liu D, Zhang X, Su Q (2019) New image blind watermarking method based on two-dimensional discrete cosine transform. Optik 204:164152. https://doi.org/10.1016/j.ijleo.2019.164152
11. Zhang H et al (2022) Robust image watermarking algorithm based on QWT and QSVD using 2D Chebyshev-Logistic map. J Frankl Inst 359(2):1755–1781. https://doi.org/10.1016/j.jfranklin.2021.11.027
12. Ernawan F, Kabir MN (2018) A block-based RDWT-SVD image watermarking method using human visual system characteristics. Vis Comput 36(1):19–37. https://doi.org/10.1007/s00371-018-1567-x
13. Jane O, Elbaşi E (2014) A new approach of nonblind watermarking methods based on DWT and SVD via LU decomposition. Turkish J Elect Eng Comput Sci 22:1354–1366. https://doi.org/10.3906/elk-1212-75
14. Maini S, Aggarwal AK (2018) Camera position estimation using 2D image dataset. Int J Innov Eng Technol 10:199–203. https://doi.org/10.21172/ijiet.102.29
15. Roy S, Pal AK (2019) A hybrid domain color image watermarking based on DWT–SVD. Iran J Sci Technol Trans Electr Eng 43:201–217. https://doi.org/10.1007/s40998-018-0109-x
16. Li Z et al (2021) Blind and safety-enhanced dual watermarking algorithm with chaotic system encryption based on RHFM and DWT-DCT. Digit Signal Process 115:103062. https://doi.org/10.1016/j.dsp.2021.103062

17. Yuan Z, Liu D, Zhang X, Wang H, Su Q (2020) DCT-based color digital image blind watermarking method with variable steps. Multimedia Tools Appl 79(41–42):30557–30581. https://doi.org/10.1007/s11042-020-09499-w
18. Ouyang J, Coatrieux G, Chen B, Shu H (2015) Color image watermarking based on quaternion Fourier transform and improved uniform log-polar mapping. Comput Elect Eng 46:419–432. https://doi.org/10.1016/j.compeleceng.2015.03.004
19. Su Q, Niu Y, Wang G, Jia S, Yue J (2014) Color image blind watermarking scheme based on QR decomposition. Signal Process 94:219–235. https://doi.org/10.1016/j.sigpro.2013.06.025
20. Khan SMH, Hussain A, Alshaikhli IFT (2012) Comparative study on content-based image retrieval (CBIR). 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), pp 61–66. https://doi.org/10.1109/ACSAT.2012.40
21. Mohammed AA, Abdullah MAM, Awad SR, Alghareb(2022) A novel FDCT-SVD based watermarking with radon transform for telemedicine applications. Int J Intell Eng Syst 15(1). https://doi.org/10.22266/ijies2022.0228.07
22. Dhar PK, Hasan R, Shimamura T (2018) Color image watermarking based on radon transform and jordan decomposition. Digital Image and Video Watermarking and Steganography. IntechOpen. https://doi.org/10.5772/intechopen.80407
23. Zhang JG, Qi HB (2020) A robust digital watermarking algorithm based on finite-set discrete radon transform tight frame. J Comput Commun 8:123–133. https://doi.org/10.4236/jcc.2020.812012
24. Goléa NE-H, Seghir R, Benzid R (2010) A bind RGB color image watermarking based on singular value decomposition. In: ACS/IEEE Int. Conf. Comput. Syst. Appl., Hammamet, Tunisia, pp 1–5. https://doi.org/10.1109/AICCSA.2010.5586967
25. Ariatmanto D, Ernawan F (2020) An improved robust image watermarking by using different embedding strengths. Multimedia Tools Appl 79(17–18):12041–12067. https://doi.org/10.1007/s11042-019-08338-x
26. Narawade NS, KanphadeTaiyue R (2012) DCT based robust reversible watermarking for geometric attack. Int J 1(2):27–32
27. Kingston A, Svalbe I (2003) Mapping between digital and continuous projections via the discrete Radon transform in Fourier space. Proceedings of VIIth digital image computing: techniques and applications, pp 263–272
28. Zhu H, Liu M, Li Yu (2010) The RST invariant digital image watermarking using Radon transforms and complex moments. Digit Signal Process 20(6):1612–1628. https://doi.org/10.1016/j.dsp.2010.01.010
29. Aggarwal AK (2015) Autonomous navigation of intelligent vehicles using vision based method. Int J Res Electron Commun Technol 3(2):1–10
30. University of Southern California and Signal and Image Processing Institute (n.d.) The USC-SIPI image database. http://sipi.usc.edu/database/. Accessed 20 Oct 2023
31. University of Granada and Computer Vision Group (n.d.) CVG-UGR image database. http://decsai.ugr.es/cvg/dbimagenes/c512.php. Accessed 20 Oct 2023
32. Xiao J et al (2023) Deep learning-based spatiotemporal fusion of unmanned aerial vehicle and satellite reflectance images for crop monitoring. IEEE Access. https://doi.org/10.1109/ACCESS.2023.3297513

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

Bin Bao was born in 1973. He received his master's degree from the East China Normal University, Shanghai, China, in 2006. He is currently an instructor with the Department of Intelligent Agricultural Engineering, Shanghai Vocational College of Agriculture and Forestry, Shanghai, China. His research interests include digital watermarking and image processing.

Yu Wang was born in 1984. He received his M.S. degree and Ph.D. degree in Computer Science and Technology from State Key Laboratory of Mathematical Engineering and Advanced Computing in China. His current research interests include network information security, internet routing and complex networks.