IET Image Processing / Volume 8, Issue 11 / p. 662-670

Article | Free Access

Capacity improved robust lossless image watermarking

Rasha Thabit, Bee Ee Khoo

First published: 01 November 2014
https://doi.org/10.1049/iet-ipr.2013.0862

Abstract

Nowadays, the lossless watermarking methods that can resist attacks have attracted more attention. Obtaining a robust lossless watermarking was at the cost of reducing the capacity and the watermarked image quality. This study presents a new robust lossless watermarking scheme in the transform domain where the Slantlet transform (SLT) has been applied to transform the host image and blocks of the SLT coefficients have been selected for the embedding process. The histograms of the selected blocks are modified according to a predefined threshold value to carry the watermark bits. The overflow/underflow of the pixel values have been avoided by using a pixel adjustment method as a post-processing step. In the proposed scheme, the original host image can be recovered without any distortion after the hidden watermark has been extracted and the watermark can withstand different kinds of attacks. In comparison with the previous methods, the proposed scheme has higher embedding capacity, better robustness against unintentional attacks and improved visual quality. The results of the experiments prove the efficiency of the proposed scheme.

1 Introduction

Data hiding schemes can be classified into two main groups: (a) the steganography schemes where the interest is directed towards the secret message and (b) the watermarking schemes where the interest is directed towards the original image. In the watermarking applications, the distortion must be as small as possible; however, in some fields (e.g. medical imaging and military imaging), any distortion in the original image is not acceptable; the reversible watermarking (RW) schemes [1-5] have been proposed to be applied in these fields.

The application of the RW methods has been extended to many types of media; however, most of these methods required a lossless environment to transmit the media [6]. The lossless environment is not available in practical applications, so the robust reversible watermarking (RRW) methods have been proposed lately [7-11]. Robust (or semi-fragile) RW techniques have been proposed to overcome the noise generated in the carrier channel and the lossy compression. In [7], a histogram rotation based scheme has been proposed where a block in the host image is divided into two equal sets of pixels; then the pixel values in each set are mapped to a circle. The watermark bit can be embedded by rotating the vectors pointing from the centre of a circle to the centres of mass of the two sets [7]. After embedding watermark bits, some of the pixel values suffer from the underflow/overflow problem (i.e. for the 8 bit greyscale images the pixel values may become less than '0' or more than '255'); additions and subtractions with modulo 256 have been used as a solution to this problem, but this causes salt-and-pepper noise in the watermarked image [7].

Aiming to solve the problem of the salt-and-pepper noise, RRW methods have been proposed either in the transform domain [8, 9] or in the spatial domain [10, 11]. The transform domain based methods [8, 9] depend on the properties of the integer wavelet transform (IWT) coefficients in the high-frequency subbands. In these methods, the original host image is transformed using IWT. Then, one of the high-frequency subbands is chosen to carry the watermark bits and divided into non-overlapping blocks. In [8], the watermark bits are embedded by shifting the mean value of the block, which can cause underflow/overflow of the pixel values in the spatial domain. To solve this problem, classification of the spatial domain blocks has been used; but in some cases, a watermark bit must be changed from '1' to '0' and thus the use of error correction coding (ECC) became necessary. This scheme solved the problem of the salt-and-pepper noise, but the use of the ECC reduced the capacity. In [9], after obtaining the transform domain blocks, the histogram shifting and clustering for watermark embedding and extraction processes have been used. This scheme obtained better capacity in comparison with the transform-based scheme in [8], but the capacity has been controlled by a threshold value (i.e. to improve the visual quality of the watermarked image the capacity has been reduced).

In the spatial domain based schemes [10, 11], the original host image is divided into blocks and the pixel values in each block are divided into two groups. The arithmetic average difference (α) of greyscale values of pixel pairs is calculated and used to embed the watermark; α is shifted by a specific threshold value to embed a binary bit. To avoid the overflow/underflow of the pixel values, the blocks have been classified into different categories; in some cases, the watermark bits have been changed from '1' to '0' and the ECC has been used to correct the errors in the extracted watermark. This scheme solved the problem of the salt-and-pepper noise, but it has two problems [11]: (a) the conditions for classifying the blocks have not been defined appropriately and (b) the shift value that has been used to adjust the difference (α) was not enough to ensure the correct recovery of the embedded watermark. These two problems lead to incomplete reversibility, therefore the solutions have been proposed in [11], where a block skipping scheme (BSS) has been proposed to solve the first problem and a new shift value has been selected to solve the second problem [11]. The method in [11] solved the problem of irreversibility, but the capacity has been reduced because of using the BSS. The algorithms in [10, 11] have been implemented to automatically choose the block size, threshold value and the ECC in order to obtain the best visual quality, which makes the data hiding capacity changeable for every image.

Considering the problems that have been mentioned, we developed a new robust lossless watermarking method based on SLT matrix to transform image and selecting sets of SLT coefficients to carry the watermark bits. The watermark bits are embedded in each block by modifying the histograms of the selected sets. We used a pixel adjustment method to keep the pixel values in the acceptable range. The aims of the proposed method are: (a) obtaining higher embedding capacity in a completely RW scheme and (b) improving the image quality and the watermark robustness. The rest of this paper is organised as follows. Section 2 briefs the related works to better understand the proposed scheme. In Section 3, we detail the proposed scheme. The experimental results are in Section 4. The final section contains the conclusions of this paper.

2 Related works

In this section, we explain previous robust watermarking methods and discuss their useful inspirations to our novel method. Thereafter, a wavelet-like transform called SLT is presented to lay the groundwork for the proposed scheme.

2.1 Robust watermarking schemes

For robust watermarking, some of the researchers depend on extracting the feature points of the host image. In [12], the feature points of the host image have been extracted using scale-invariant feature transform (SIFT), then circular patches have been generated from the host image. The watermark has been embedded into multiple patches in the transform domain. The embedding process starts by extracting a rectangle patch from the circular patch and applying the discrete cosine transform (DCT) to the rectangle patch. To embed a watermark of length (L), they selected L pair of mid-frequency DCT coefficients following the zig-zag scan order. Consider the two coefficients are c_1 and c_2, to embed a watermark bit $w = \{w_1, w_2, \ldots, w_L\}$ in the selected coefficients the rules are as follows [12].

If $w_i = 1$ and $D = c_1 - c_2 < T$ (where T is a threshold to control the visual quality), then increase c_1 and decrease c_2 as follows

Else if $D = c_1 - c_2 > T$, do nothing.

If $w_i = 0$ and $D = c_2 - c_1 < T$, the same process is done as follows

Else if $D = c_2 - c_1 > T$, do nothing.

After embedding the whole watermark bits, the inverse DCT is applied to obtain the watermarked region.

For the watermark extraction process, similar to the steps of the embedding process the DCT coefficients of the selected region are calculated and the L pairs of the coefficients are extracted. Then, the watermark bit is extracted by comparing the coefficients c_1 and c_2 as follows

$$w_i^* = \begin{cases} 1 & c_1 \geq c_2 \\ 0 & c_1 < c_2 \end{cases}$$

In [13], another scheme based on SIFT has been presented where the circular patches have been transformed using discrete wavelet transform (DWT), then the horizontal and vertical high-frequency coefficients have been selected to carry the watermark bits. The original watermark is resized to match the available space in the circular patch and each watermark bit is embedded using the same rules that have been applied in [12]. These two schemes obtained good robustness results, but the watermarking process is not reversible.

Obviously, the main idea of the watermark embedding process in [12, 13] depends on making the first selected coefficient greater than the second selected coefficient when the watermark bit is ‘1’ and the reverse when the watermark bit is ‘0’. This main idea will be adopted to embed the watermark bits in the proposed scheme, but the methods of selecting the coefficients, defining the threshold value and applying the modifications are different. In addition, the reversibility of the watermarking process has been taken into consideration while implementing the proposed algorithm. All the details will be explained in Section 3.

2.2 Slantlet transform and its matrix

An improved version of DWT called SLT has been presented in [14]. This transform has been applied in different applications [15-19] and obtained better performance in comparison with other schemes. In the conventional SLT, the moment vectors have been computed by using recursive algorithm, which resembles a sequence of convolutions and downsampling. The Slantlet-based applications in [15-19] used the conventional SLT to transform the input signal.

In [14], Selesnick presented the method of calculating the Slantlet matrix to prove that the SLT is an orthogonal transform. In the proposed method, we suggested a different way to find the SLT of the image's blocks using Slantlet matrix. According to the method of image transformation that was explained in [20], the image can be transformed by applying rows transformation and columns transformation. So that, we suggested using the SLT matrix that has been implemented in [14] to transform image's blocks by the matrix multiplication process.

Assuming a single block of the host image (*B*), the complete SLT can be represented in matrix format by

(1)

where *B* is the original block, TB is the transformed block and SLT_N is an $N \times N$ Slantlet matrix. Note that *B*, TB and SLT_N have the same size.

The inverse SLT (ISLT) in matrix format can be obtained by

(2)

A preliminary study has been conducted to show the difference between using the conventional SLT and the proposed matrix multiplication process. To evaluate the performance, Lena image (size 512 × 512) has been divided into non-overlapping blocks (size 64 × 64). First, the SLT and ISLT have been applied to a single block using the two methods (i.e. the conventional SLT and the matrix multiplication process). The difference between the original block and the recovered block has been calculated and the time required by each method has been recorded. Then, the SLT has been applied to whole image blocks using the two methods and the time has been recorded. The experiments have been applied in a computer with the following properties: 2.40 GHz Intel® Core™ i5 central processing unit (CPU) and 4 GB memory.

To record the programmes run-time in seconds, the tic and toc commands in MATLAB (R2009b) have been used. The results are shown in Table 1. The matrix multiplication process and the recursive algorithm are different. The matrix multiplication process has less difference in the recovered block and it requires less time. Since our target is a RW scheme, this small difference between the original block and the recovered block will not be accepted. To avoid this difference, the recovered values after ISLT will be rounded to their integer values. Thus, the difference between the original block and the recovered block will be zero. We will make use of this rounding process in the implementation of the proposed algorithm.

Table 1. Comparison between the conventional SLT and the matrix multiplication process

Parameters	Conventional SLT	Matrix multiplication
the difference between the original block and the recovered block	1.136868×10^{-12}	3.410605×10^{-13}
the time of SLT and ISLT for a single block, s	0.074829	0.009022
the time of SLT for the whole image blocks, s	0.062996	0.008536

3 Proposed robust reversible watermarking

In this section, we introduce a new robust lossless watermarking scheme in the transform domain. The proposed scheme can be divided into four main components as follows.

3.1 Host image blocking, transformation and choosing coefficients

The host image I_m is divided into non-overlapping blocks (B_i, where i = 1, 2, …, K). The total number of image blocks K depend on the original image size and the block size. If the host image is of size $H \times W$ and the side length of a square block is L_S, then K can be calculated as follows

(3)

Each image block B_i is transformed using (1) and the transformed blocks are named as (TB_i, where i = 1, 2, …, K) as shown in Fig. 1.

Fig. 1

Open in figure viewer | PowerPoint

Image blocking, transformation and choosing SLT coefficients

The SLT coefficients in matrix (***TB**$_i$*) are divided into seven-sets (i.e. seven-subbands) as shown in Fig. 1; four sets are chosen for the watermark embedding as follows

where *x* and *y* are the coordinates of the coefficients in the (***TB**$_i$*) matrix and L_S is the side length of the (***TB**$_i$*) matrix.

3.2 Embedding watermark bits

For the watermark embedding, the mean values of the SLT subbands (HL_1, LH_1, HL_2 and LH_2) are calculated to be used as controlling factors of the embedding process. The mean value (μ) for each subband can be calculated using the following equation

(4)

where the superscript (t) indicates the name of the subband, B_S is the side length of the subband and is the SLT coefficients in the specified subband.

The (HL_1 and LH_1) subbands are used to carry one watermark bit and (HL_2 and LH_2) subbands are used to carry the second watermark bit. The process of embedding the watermark bit in (HL_1 and LH_1) subbands can be explained in the following steps (note: the same steps can be used to embed a watermark bit in (HL_2 and LH_2):

Step 1. An integer threshold value (T) is defined. The threshold value must be integer to ensure the reversibility and this threshold has been used as a factor to control the watermark robustness and invisibility. The threshold value is the same for all image blocks.

Step 2. Take the watermark bit.

If the watermark bit is ‘1’ then calculate the difference $\varepsilon_1 = \mu^{HL1} - \mu^{LH1}$ and compare it with the threshold value T. If the difference $\varepsilon_1 \geq T$, then no change will happen to the carrier subbands. If the difference $\varepsilon_1 < T$, then the histograms of the carrier subbands will be shifted to make μ^{HL1}

more than μ^{LH1}. The shift value $S_1 = (T - \varepsilon_1)/2$ is added to the coefficients in $HL1$ and subtracted from the coefficients in $LH1$.

If the watermark bit is ‘0’ then calculate the difference $\varepsilon_2 = \mu^{LH1} - \mu^{HL1}$ and compare it with the threshold value T. If the difference $\varepsilon_2 \geq T$, then no change will happen to the carrier subbands. If the difference $\varepsilon_2 < T$, then the histograms of the carrier subbands will be shifted to make μ^{LH1} more than μ^{HL1}. The shift value $S_2 = (T - \varepsilon_2)/2$ is subtracted from the coefficients in $HL1$ and added to the coefficients in $LH1$.

Owing to the reversibility requirements, the difference values ε_1 or ε_2 will be saved as a side information when the mean values are changed to embed the watermark bit. Fig. 2 shows a detailed example of the histogram modification process, where the spatial domain block size is (32 × 32), subband size is (16 × 16), the threshold value is 3 and the watermark bit is ‘0’.

Fig. 2

Open in figure viewer | PowerPoint

Example of the histogram modification process to embed watermark bit ‘0’

3.3 Obtaining the watermarked image

After embedding all watermark bits, the original subbands in the SLT matrix are replaced by the modified subbands. Then ISLT is applied [using (2)]. Thereafter, to ensure the reversibility, the resultant values are rounded to integer numbers.

After the watermark embedding process, some of the pixel values may suffer from underflow/overflow. Previously in the RW methods, different techniques have been applied to avoid this problem; these techniques can be summarised as shown in Table 2.

Table 2. Avoiding overflow/underflow problem in the previous methods

References	The method of avoiding overflow/underflow problems
[1, 4]	• the histogram modification process has been applied as a pre-processing step • the histogram has been narrowed from both sides before embedding the data • always applied at the beginning although sometimes it is not necessary • the pixel values that have been modified most be saved as bookkeeping information

References	The method of avoiding overflow/underflow problems
	been narrowed from each side separately • applied only when it is necessary • the pixel values that have been modified most be saved as bookkeeping information
[7]	• additions and subtractions with modulo 256 have been applied • the scheme suffers from salt-and-pepper noise
[8]	• blocks classification has been applied and the watermark bits have been changed • the scheme suffers from incomplete reversibility
[9]	• the histogram modification process has been applied as a pre-processing step • applied only when it is necessary • the modified pixels and their locations must be saved as side information
[10]	• blocks classification has been applied and the watermark bits have been changed • the scheme suffers from incomplete reversibility
[11]	• blocks classification has been applied and the watermark bits have been changed • BSS has been used to avoid the irreversibility in [8] • location map has been generated from the BSS and saved as

The approach of the pixel adjustment that has been used in [1, 4, 5] cannot be applied directly to the transform domain methods because the adjustment scale is undefined. In addition, this process will degrade the visual quality of the watermarked image. The method in [7] cannot be used because of the salt-and-pepper noise and the methods in [8, 10] suffer from incomplete reversibility. The method in [9] successfully avoids the overflow/underflow; however, it degrades the visual quality of the watermarked image because of shifting some of the pixel values in the host image more than required as a pre-processing step. In [11], the use of the BSS reduced the capacity.

In our method, we suggested the use of the pixel adjustment as a post-processing step instead of using it as a pre-processing step, and thus the visual quality of the watermarked image can be improved. Each individual pixel is checked and if the pixel value is more than ‘255’ this means overflow happens then that pixel value is changed to ‘255,’ if the pixel value is less than ‘0’ this means underflow happens then the pixel value is changed to ‘0,’ this process can be summarised as follows

(5)

where I_w is the watermarked image before pixel adjustment, (i, j) are the coordinates of the value in the image and $I'_w(i, j)$ is the modified pixel value.

The pixel values that are suffering from overflow/underflow and their locations are saved as part of the side information. The side information together with the block size and the threshold value must be sent with the watermarked image to the receiver side. Fig. 3 shows the block diagram of the watermark embedding scheme.

Fig. 3

Open in figure viewer | PowerPoint

Block diagram of the watermark embedding scheme

(info. ≡ information)

3.4 Watermark extraction scheme

At the receiver, after reading the watermarked image and the side information, the pixel values that were adjusted must be returned back to their locations in the watermarked image. Then the image is divided into non-overlapping blocks. Each block is transformed using (1) and the coefficients are divided into subbands as shown in Fig. 1. For each block, the mean values of the chosen coefficients are calculated and the watermark bit is extracted according to the following

(6)

where is the extracted bit, is the mean value of the SLT coefficients in HL_x subband and is the mean value of the SLT coefficients in LH_x subband. The subscript x is either 1 or 2.

According to the extracted watermark value and the difference value that was already saved in the side information, the original mean values in each block can be recovered by applying the inverse of the process that was applied in the watermark embedding side. Each subband will be recovered by shifting back its histogram, and then the ISLT [using (2)] and the rounding process must be applied. Thus, the original image is recovered by re-arranging the recovered blocks. Fig. 4 shows the block diagram of the watermark extraction scheme.

Fig. 4

Open in figure viewer | PowerPoint

Block diagram of the watermark extraction scheme

(hist. ≡ histograms)

4 Experimental results and discussion

The proposed algorithm has been implemented in MATLAB (R2009b) and the computer properties are 2.40 GHz Intel® Core ™ i5 CPU and 4 GB memory. The algorithm has been tested for 300 different images [i.e. 100 general images [21], 100 medical images [22] and 100 synthetic aperture radar (SAR) images collected from different websites] of size 512 × 512 8 bit. The following subsections contain the results of the experiments; the proposed method has been compared with four previous methods [8-11] depending on the results that have been reported in these previous papers.

4.1 Capacity test

The pure capacity of the proposed method depends on the size of the original image and the size of the block. Two watermark bits can be embedded in each spatial domain block. In the previous transform domain schemes [8, 9], the term 'block size' is referring to the size of the transform domain carrier subband. Therefore the capacity of the proposed method is calculated according to the transform domain block size as follows.

Consider an image I_m with size $M\times N$ and a spatial domain block with size $X \times Y$, the block size in the transform domain (B_{size}) means the size of one subband (HL_1 or LH_1) of the SLT coefficients (i.e. $B_{size}=m \times n=X/2 \times Y/2$). The pure capacity ($C$) can be calculated by

(7)

The previous transform domain methods [8, 9] used only HL_1 or LH_1 to carry the watermark bit and they do not embed data in the high-level IWT coefficients because the modification of those higher-level subbands will degrade the watermarked image quality more and make the avoidance of underflow/overflow more complicated [8]. Since our method can use the higher-level subbands (HL_2 and LH_2), the capacity has been improved. The improvement of the capacity in our method and the method in [9] is at the cost of increasing the amount of side information in comparison with the method in [8]. In [8], the pixel adjustment process has not been used, therefore it requires less side information. The amount of the side information required by our method and the method in [9] depend on the contents of the image, the block size and the threshold value.

As an example, the capacity is calculated for different block sizes when the original image size is (512 × 512) as shown in Table 3; the capacity decreases with the increase of the block size.

Table 3. Comparison of capacity

B_{size} = $m \times n$	Capacity, bits			Capacity methods in [10, 11]
	Method in [8]	Method in [9]	Proposed method	
4 × 4	3003	≤4096	8192	capacity is changeable for every image (depends on the block size, threshold value and ECC)
8 × 8	750	≤1024	2048	
16 × 16	187	≤256	512	

4.2 Invisibility test

To test the invisibility, the peak signal-to-noise ratio (PSNR) between the original image and the watermarked image has been calculated. To compare the invisibility, the proposed algorithm has been applied to the same test images that have been used in the previous methods; these test images are shown in Fig. 5.

Fig. 5

Open in figure viewer | PowerPoint

Some test images

a General $image_1$

b Medical $image_1$

c SAR $image_1$

d Lena

e Baboon

f Boat

g $Mpic_1$

h $Mpic_2$

The test images shown in Figs. 5a–*c* have been used in [9] to evaluate the visual quality, therefore these images have been watermarked using the proposed method at the same parameters that have been used in [9] (i.e. B_{size} = 4 × 4, and the watermark strength T = 8). To embed the same payload, one watermark bit has been embedded in each block and the results are shown in Table 4. As shown in the results, the proposed scheme has better visual quality in comparison with the previous methods except for the watermarked medical $image_1$ using the method in [8].

Table 4. Comparison of invisibility with the methods in [8, 9] [PSNR(dB)]

Methods	General $image_1$	Medical $image_1$	SAR $image_1$
method in [8]	30.3	41.2	30.3
method in [9]	33.1	29.5	37.6
proposed method	38.2263	38.7298	38.6403

The test images shown in Figs. 5d–*h* have been used in [10, 11] to evaluate the visual quality, therefore these images have been watermarked using the proposed method at the same payload and the same threshold values. The results are shown in Table 5 where the PSNR values are the same for the two previous methods, except for one medical image (i.e. $Mpic_2$). As shown in the results, the proposed method has better visual quality in comparison with the previous methods.

Table 5. Comparison of invisibility with the methods in [10, 11] [PSNR(dB)]

Images	Lena	Baboon	Boat	$Mpic_1$	$Mpic_2$
payload, bits	792	560	528	100	100
at T=2					
method in [10]	45.12	45.47	45.57	56.14	51.42
method in [11]	45.12	45.47	45.57	56.14	51.42
proposed method	50.3251	52.3352	53.2761	60.8954	65.9604

Images	Lena	Baboon	Boat	$Mpic_1$	$Mpic_2$
at T=4					
method in [10]	39.10	39.45	39.55	50.12	45.40
method in [11]	39.10	39.45	39.55	50.12	50.12
proposed method	45.6579	47.3088	47.7344	55.0902	60.4810

Another experiment has been conducted to evaluate the effect of the threshold value (T) on the visual quality of the watermarked images. Fig. 6 presents the experimental results at $B_{size} = 8 \times 8$ for general, medical and SAR images, which demonstrates that the higher the threshold value is, the lower the average PSNR is, and vice versa.

Fig. 6

Open in figure viewer | PowerPoint

Invisibility test at different threshold values

4.3 Robustness test

For robustness evaluation, the experiments have been conducted at the same values of the parameters that were used in [9] (i.e. $B_{size} = 8 \times 8$ and threshold 16). Similar to the experiments in [9], the major voting on robustness has been used and watermarks with 10 bits have been repeatedly embedded in the host image. The robustness against three types of attacks [i.e. JPEG compression with quality factor 100–20%, JPEG2000 compression with rate 0.2–2.0 and additive Gaussian noise (AGN) with variance 0.001–0.01] has been evaluated using the unified parameter Φ [9]

(8)

where $0 \leq \Phi < 1$, (α) is a normalisation factor (which is 1 for JPEG/JPEG2000 and 0.1 for AGN), (Ψ) is the benchmark of the unintentional attack strength (which is 100 for JPEG, 2.0 for JPEG2000 and 0.001 for AGN) and (Λ) is the tolerated unintentional attack strength [9]. The embedded watermarks can be recovered correctly if the unintentional attack strength is less than (Λ) [9].

To clarify how to calculate (Λ), two examples are explained:

For the JPEG compression, when the embedded watermark can be extracted correctly if the quality factor is more than 30%, then (Λ = 30) and Φ = 1 × [abs(100 − 30)/100] = 0.7.

For the JPEG2000 compression, when the embedded watermark can be extracted correctly if the rate is more than 0.2, then (Λ = 0.2) and Φ = 1 × [abs(2 − 0.2)/2] = 0.9.

The robustness is better with the larger values of Φ and vice versa. Table 6 contains the results, which demonstrate that the proposed method has better robustness as compared with the previous methods.

Table 6. Comparison of robustness with methods in [8, 9] (Φ)

Attacks	Image types	Method in [8]	Method in [9]	Proposed method
JPEG compression	general images	0.2	0.6	0.7
	medical images	0	0.6	0.8
	SAR images	0.2	0.6	0.8
JPEG2000 compression	general images	0.4	0.7	0.8
	medical images	0.5	0.9	0.9
	SAR images	0.2	0.6	0.8
AGN	general images	0.6	0.9	0.9
	medical images	0.3	0.9	0.9
	SAR images	0.6	0.9	0.9

For comparison with the method in [10], the robustness has been evaluated by calculating the data survival rate (DSR) in [bit-per-pixel (bpp)] after JPEG2000 compression. DSR is the minimum bit rate that can be applied and the watermark still can be extracted correctly [10], which means the lower the DSR the better the robustness. Table 7 contains the results of this experiment; as shown in the results the proposed method has better robustness at the same PSNR.

Table 7. Comparison of robustness with method in [10], the DSR in bpp

Images	Payloads, bits	Method in [10]	Proposed method

Images	Payloads, bits	PSNR, dB	DSR, bpp	PSNR, dB	DSR, bpp
		PSNR, dB	DSR, bpp	PSNR, dB	DSR, bpp
Lena	792	40.2	0.8	40.2	0.6
Baboon	585	38.7	1.6	38.7	0.9
Boat	560	40.5	1	40.5	0.5
$Mpic_1$	100	37.6	0.4	37.6	0.01
$Mpic_2$	100	37.7	0.8	37.8	0.01

Another experiment has been conducted to evaluate the effect of the threshold value (*T*) on the robustness of the proposed scheme. Fig. 7 presents the robustness results in terms of DSR at payload 1000 bits for three test images, which demonstrate that the higher the threshold value is, the better the robustness is, and vice versa.

Fig. 7

Open in figure viewer | PowerPoint

Robustness test at different threshold values

4.4 Run-time test

The time required for the watermark embedding process and the watermark extraction process has been calculated at different block sizes. To record the programmes run-time in seconds, the tic and toc commands in MATLAB have been used. The average run-time for each type of the 300 test images has been calculated and the results are shown in Table 8. Since we did not re-implement the previous algorithms, this section has no comparison with the previous methods.

Table 8. Run-time (embedding/extraction) (in seconds)

Block sizes	General images	Medical images	SAR images
4 × 4	0.590/0.358	0.629/0.385	0.584/0.355

Block sizes	General images	Medical images	SAR images
8 × 8	0.176/0.103	0.210/0.112	0.172/0.102
16 × 16	0.076/0.042	0.119/0.044	0.076/0.042

4.5 Reversibility test

To test the reversibility, we employ the image error rate (IER), which is the ratio of the number of the images recovered with errors to the total number of test images for each type [9]. The proposed scheme is completely reversible for all test images and the results are shown in Table 9.

Table 9. Comparison of reversibility based on IER

Images	Method in [8]	Method in [9]	Method in [10]	Proposed method
general images	0.51	0	0.84	0
medical images	0.94	0	0.96	0
SAR images	0.58	0	0.89	0

4.6 Practical application

To test the performance of the proposed method in a practical application, two different binary watermarks [each of size (156 × 122)] have been embedded in one medical image of size (1248 × 976) at block size (4 × 4). When the threshold value is 15, the PSNR is 30.1387 dB. The extracted watermarks after different attacks are shown in Fig. 8.

Fig. 8

Open in figure viewer | PowerPoint

Extracted watermarks after different attacks

5 Conclusions

The purpose of the robust lossless watermarking is embedding data in such a way that the host image can be recovered correctly after the hidden data extraction and at the same time the hidden data have robustness against attacks. In the proposed scheme, the host image is divided into small blocks; each block is transformed using SLT matrix, which is helpful for the robustness and the invisibility because of the properties of the SLT coefficients. Two bits are embedded in each block, without degrading the visual quality a lot and without affecting the reversibility, and thus the capacity has been improved. The pixel adjustment process is applied as a post-processing step, which is helpful in improving the image quality and obtaining a completely RW scheme. The experimental results prove the efficiency of the proposed method. Obtaining higher embedding capacity for medical images will be the future work.

6 References

1 Xuan G., Shi Y.Q., Zheng Y., Zou D., and Chai P.: ' Lossless data hiding using integer wavelet transform and threshold embedding technique'. IEEE Int. Conf. Multimedia and Expo, July 2005, pp. 1520– 1523
Google Scholar

2 Ni Z., Shi Y., Ansari N., and Su W.: 'Reversible data hiding', *IEEE Trans. Circuits Syst. Video Tech.*, 2006, **16**, (3), pp. 354– 362 (doi: http://doi.org/10.1109/TCSVT.2006.869964)
Crossref | Web of Science® | Google Scholar

3 Hwang J., Kim J., and Choi J.: ' A reversible watermarking based on histogram shifting'. IWDW 2006, 2006 (*LNCS*, **4283**), pp. 348– 361
Google Scholar

4 Xuan G., Yao Q., and Yang C. *et al*: ' Lossless data hiding using histogram shifting method based on integer wavelet'. IWDW 2006, 2006 (*LNCS*, **4283**), pp. 323– 332
Google Scholar

5 Xuan G., Shi Y.Q., Chai P., Cui X., Ni Z., and Tang X.: ' Optimum histogram pair based image lossless data embedding'. IWDW 2007, 2008 (*LNCS*, **5041**), pp. 264– 278
Google Scholar

6 An L., Gao X., Deng C., and Ji F.: ' Robust lossless data hiding: analysis and evaluation'. Proc. Int. Conf. High Performance Computing and Simulation (HPCS 2010), Caen, France, 2010, pp. 512– 516
Google Scholar

7 De Vleeschouwer C., Delaigle J., and Macq B.: 'Circular interpretation of bijective transformations in lossless watermarking for media asset management', *IEEE Trans. Multimed.*, 2003, **5**, (1), pp. 97– 105

(doi: http://doi.org/10.1109/TMM.2003.809729)
Crossref | Web of Science® | Google Scholar

8 Zou D., Shi Y., Ni Z., and Su W.: 'A semi-fragile lossless digital watermarking scheme based on integer wavelet transform', *IEEE Trans. Circuits Syst. Video Tech.*, 2006, **16**, (10), pp. 1294– 1300 (doi: http://doi.org/10.1109/TCSVT.2006.881857)
Web of Science® | Google Scholar

9 An L., Gao X., Li X., Tao D., Deng C., and Li J.: 'Robust reversible watermarking via clustering and enhanced pixel-wise masking', *IEEE Trans. Image Process.*, 2012, **21**, (8), pp. 3598– 3611 (doi: http://doi.org/10.1109/TIP.2012.2191564)
PubMed | Web of Science® | Google Scholar

10 Ni Z., Shi Y.Q., Ansari N., Su W., Sun Q., and Lin X.: 'Robust lossless image data hiding designed for semi-fragile image authentication', *IEEE Trans. Circuits Syst. Video Tech.*, 2008, **18**, (4), pp. 497– 509 (doi: http://doi.org/10.1109/TCSVT.2008.918761)
Crossref | Web of Science® | Google Scholar

11 Gao X., An L., Li X., and Tao D.: 'Reversibility improved lossless data hiding', *Signal Process.*, 2009, **89**, (10), pp. 2053– 2065 (doi: http://doi.org/10.1016/j.sigpro.2009.04.015)
Web of Science® | Google Scholar

12 Li L., Guo B., and Guo L.: 'Rotation, scaling and translation invariant image watermarking using feature points', *J. China Univ. Posts Telecommun.*, 2008, **15**, (2), pp. 82– 87 (doi: http://doi.org/10.1016/S1005-8885(08)60089-8)
Google Scholar

13 Li L., Qian J., and Pan J.: 'Characteristic region based watermark embedding with RST invariance and high capacity', *Int. J. Electron. Commun.*, 2011, **65**, (5), pp. 435– 443 (doi: http://doi.org/10.1016/j.aeue.2010.06.001)
Web of Science® | Google Scholar

14 Selesnick I.W.: 'The Slantlet transform', *IEEE Trans. Signal Process.*, 1999, **47**, (2), pp. 1304– 1313 (doi: http://doi.org/10.1109/78.757218)
Google Scholar

15 Abbas S.M., Abbas M.N., and Mohammed S.A.: 'Slantlet transform-based OFDM scheme', *J. Eng.*, 2006, **13**, (3), pp. 1638– 1647
Google Scholar

16 Maitra M., and Challerjee A.: 'A Slantlet transform based intelligent system for magnetic resonance brain image classification', *Elsevier Biomed. Signal Process. Control*, 2006, **1**, (4), pp. 299– 306

(doi: http://doi.org/10.1016/j.bspc.2006.12.001)
Crossref | Web of Science® | Google Scholar

17 Mutt S., and Kumar S.: ‘ Secure image steganography based on Slantlet transform’. Proc. Int. Conf. Methods and Models in Computer Science, 2009, pp. 1– 7
Google Scholar

18 Abou-Loukh S.J., and Gatea S.M.: ‘Spoken word recognition using Slantlet transform and dynamic time warping’, *Nahrain Univ. Coll. Eng. J. (NUCEJ)*, 2011, **14**, (1), pp. 34– 45
Google Scholar

19 Mohammed R.T., and Khoo B.E.: ‘ Image watermarking using Slantlet transform’. Proc. 2012 IEEE Symp. Industrial Electronics and Applications (ISIEA2012), Bandung, Indonesia, 2012, pp. 285– 290
Google Scholar

20 Mulcahy C.: ‘Image compression using the Haar wavelet transform’, *Spelman Sci. Math. J.*, 1997, **1**, pp. 22– 31. Available at http://www.cs.toronto.edu/~arnold/320/Readings/haarWavelets.pdf
Google Scholar

21 CVG-UGR Image database. Available at http://www.decsai.ugr.es/cvg/dbimagenes/index.php
Google Scholar

22 OsiriX. Available at: http://www.osirix-viewer.com/Downloads.html
Google Scholar

ABOUT THE IET

IET PRIVACY STATEMENT

CONTACT IET

About Wiley Online Library

Privacy Policy
Terms of Use
Cookies
Accessibility
Publishing Policies

Help & Support

Contact Us
Training and Support
DMCA & Reporting Piracy

Opportunities

Subscription Agents
Advertisers & Corporate Partners

Connect with Wiley

The Wiley Network
Wiley Press Room