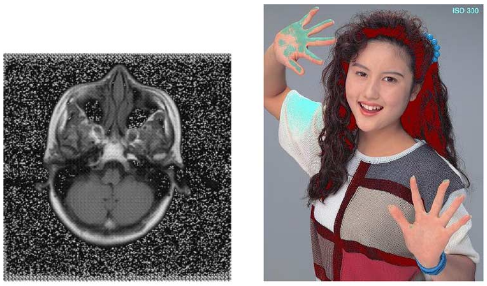
A Semi-Fragile Lossless Digital Watermarking Scheme Based on Integer Wavelet Transform

Dekun Zou, *Member, IEEE*, Yun Q. Shi, *Fellow, IEEE*, Zhicheng Ni, *Member, IEEE*, and Wei Su, *Senior Member, IEEE*

***Abstract*—In this paper, a new semi-fragile lossless digital watermarking scheme based on integer wavelet transform is presented. The wavelet family applied is the 5/3 filter bank which serves as the default transformation in the JPEG2000 standard for image lossless compression. As a result, the proposed scheme can be integrated into the JPEG2000 standard smoothly. Different from the only existing semi-fragile lossless watermarking scheme which uses modulo-256 addition, this method takes special measures to prevent overflow/underflow and hence does not suffer from annoying salt-and-pepper noise. The original cover image can be losslessly recovered if the stego-image has not been altered. Furthermore, the hidden data can be retrieved even after incidental alterations including image compression have been applied to the stego-image.**

***Index Terms*—Digital watermark, integer wavelet transform.**



(a) (b)

Fig. 1. Salt-and-pepper artifact examples. (a) Marked medical image (b) Marked N1A image.

I. Introduction

DIGITAL watermarking, or data hiding, a more general alternative term, is a process to hide data into cover media such as images, audio clips or video streams. It can be used as a way to transport information secretly or to protect the integrality of the cover medium itself [1]. In this paper, we mainly focus on data hiding into images. Digital image watermarking can be visible or invisible. In the case of invisible digital watermarking, no visual artifact is expected in the marked ones (which are referred to as stego-images as well). Although the difference between a stego-image and its original one is imperceptible by human eyes, the stego-images are different to the original ones because of the manipulations exerted to the original image during the watermark embedding process. However, in some scenarios, for example, satellite images or medical images, the original ones are so precious or subtle that people wish the original pictures can be recovered without any distortion after the hidden data have been extracted. In the literature, the techniques satisfying this criterion have been referred to as lossless or invertible watermarking [2]–[4]. All these methods can losslessly recover the original cover media with no distortion. However, any minor alteration applied to the stego-image will fail the correct retrieval of the hidden data. This group of lossless data hiding schemes is called fragile lossless data hiding. This property can be used to detect alteration in fragile authentication. However, for many applications, it is too strict if image compression is not permitted. It is desired that the hidden data will be robust to unintentional changes like image compression and sometimes unavoidable addition of random noise below a certain level which does not change the content of an image. Algorithms with this property are referred to as semi-fragile lossless data hiding methods. In the literature, to our best knowledge, there is only one lossless data hiding method that is somewhat robust to compression.

In [5], a semi-fragile lossless data hiding scheme is proposed. One big problem with this method is that modulo-256 addition is used to prevent overflow/underflow. Consequently, white pixels may be flipped to black ones after modulo-256 addition and black ones could become white ones, resulting in salt-and-pepper noise. Fig. 1 provides two examples. It is noted that an alternative scheme has been proposed in [5] in order to avoid salt-and-pepper noises. However, as the authors stated in [5], this alternative scheme is not semi-fragile.

Recently, JPEG2000 has emerged as the new standard for still image compression because of its superior efficiency. It supports both lossy and lossless compressions. For the lossless part, the 5/3 wavelet filter is used [6]. In this paper, we propose a new semi-fragile lossless data hiding scheme based on the 5/3 integer wavelet transform (IWT). Data are embedded into the IWT coefficients of a selected high frequency subband. The subband accommodating the hidden data is called the *carrier subband*.

The rest of this paper is organized as follows. In Section II, the basic idea of our proposed algorithm is presented. Section III focuses on the overflow/underflow issue and the details of the proposed data embedding algorithm are elaborated. Performance

Manuscript received June 26, 2005; revised May 9, 2006. This work was supported in part by NJCST via NJWINS and in part by the Digital Data Embedding Technologies Group of the Air Force Research Laboratory, Rome Research Site, Information Directorate, Rome, NY, under Contract F30602-03-1-0264. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Air Force Research Laboratory, or the U.S. Government.

D. Zou was with the New Jersey Institute of Technology, Newark, NJ 07102 USA. He is now with Thomson Inc., Princeton, NJ 08536 USA (e-mail: dekun.zou@thomson.net).

Y. Q. Shi, Z. Ni, and W. Su are with New Jersey Institute of Technology, Newark, NJ 07102 (e-mail: shi@njit.edu).

Color versions of Figs. 1, 2, 4, and 9 are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2006.881857

1051-8215/$20.00 © 2006 IEEE

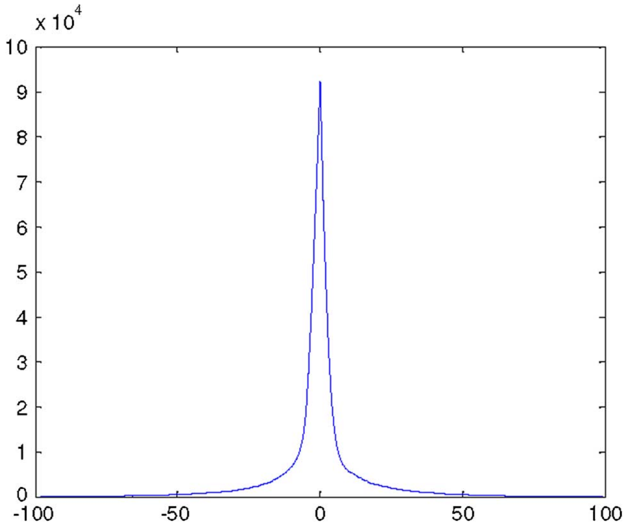


Fig. 2. Histogram of the IWT coefficients in the HL_1 subband of JPEG2000 N1A (red color plane). The size of N1A is 1536 × 1920. For the HL_1 subband, the size is 768 × 960, i.e., having 737 280 coefficients.

analysis of the proposed method is provided in Section IV. Finally, summary is made in Section V.

II. Foundation of Proposed Data Embedding Algorithm

We have studied the features of image wavelet transforms and found out that coefficients of the high frequency subbands $(\mathrm{HL}_n, \mathrm{LH}_n,$ or $\mathrm{HH}_n)$ have a zero-mean and Laplacian-like distribution. One such an example is depicted in Fig. 2, where the X axis represents the IWT coefficient values, while the Y axis the numbers of coefficients that assume the corresponding value. We further divide this subband into nonoverlapping square blocks with the side length B and calculate the mean of the coefficients values in each block.

$$\text{mean}^{(t)} = \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} C_{ij}^{(t)} \tag{1}$$

where the superscript (t) indicates the tth block. $C_{ij}^{(t)}$ denotes the IWT coefficients of the tth block. We can find that the mean values also have a zero-mean and Laplacian-like distribution.

The basic idea of the proposed algorithm is based on this property. At first, we scan all the blocks and find out the maximum absolute mean value of coefficients, denoted by $m_{\max}$. A threshold T is set to be the smallest integer number which is greater than $m_{\max}$. As a result, the absolute mean values of all blocks will not be larger than T. Now, one bit can be embedded into each of the nonoverlapping blocks by manipulating the mean value of the block. Consider a block, if a bit "1" is to be embedded, the mean value of the block will be shifted away from 0 by a quantity S, which is equal to or larger than T. If a bit "0" is to be embedded, this block is left unchanged, so is the mean value of the IWT coefficients in this block. Therefore, in the hidden data extraction, the mean value of a block with its absolute value larger than T represents a hidden binary bit "1" in the block. A block mean with value smaller than T represents an embedded binary bit "0."

On the one hand, since the shift quantity S is fixed for all blocks, the original coefficients can be recovered by conducting the reverse operation, i.e., subtracting S from the IWT coefficients in these blocks, where bit "1" is embedded. In other words, the embedding operation is reversible. As a result, the original cover media can be restored exactly. On the other hand, since we embed data by controlling the mean value of the IWT coefficients in one block, minor changes to the image caused by unintentional attacks such as JPEG/JPEG2000 compression will not cause the mean value change much. Therefore, the correct detection of the hidden data is expected even after the stego-image has been altered.

It should be pointed out that lossless watermarking is based on such a principle of information theory that all information of the original cover media should be preserved in the marked image in the lossless data hiding process. Therefore, if the marked images are subjected to lossy compression, the original cover media can no longer be recovered since some information about the original image will be impaired. It is also noticed that we only choose HL_1 or LH_1 to embed data. That is, though the mean value distribution of blocks of the IWT coefficients of higher level high frequency subbands, HL_n or $\mathrm{LH}_n (n > 1)$, also obey our observation described in this section, we do not embed data in these high level IWT coefficients because the alterations to those higher-level subbands will degrade the image quality more and make the prevention of underflow/overflow more complicated.

III. Proposed Lossless Data Hiding Algorithm

In this section, the issue of how to prevent overflow/underflow is first addressed which is the key issue in lossless data hiding. The proposed data embedding procedure is then presented in detail.

A. *Origin of Overflow/Underflow*

If stego images stay in JPEG2000 format, the proposed method would be good enough. Assuming an image in the JPEG2000 format, all the IWT coefficients can be derived from the image file through tier 2 decoding followed by tier1 decoding [6]. Then, the hidden data can be embedded into the IWT coefficients. The changed coefficients can be encoded again to form the marked JPEG2000 file for transmission or storing. At the receiver end, the hidden data could be extracted from the marked JPEG2000 image file and all the modified coefficients can be inverted back to the original ones. The recovered image should be exactly the same as the original one. Since JPEG2000 encoding ensures no truncation to the wavelet coefficients [7], overflow/underflow will not take place.

However, the stego-image may be converted into formats other than JPEG2000. File format change should be considered permissible and the original cover media should be preserved after the file format change. Nevertheless, we discovered that, sometimes, the recovered media may differ from the original one after the format changes such as from the JPEG2000 to the bitmap. This is because the pixel grayscale values in spatial domain are represented by a fixed length of bits (mostly 8 bits, sometimes 16 bits). In color image, each fundamental color component in a pixel is also often represented by 8 bits. After the IWT coefficients have been altered by the embedding operation, some pixels' grayscale values will fall out of the permitted range, say, 0–255 for an 8-bit grayscale image. This phenomenon is referred to as overflow/underflow.

B. Prevention of Overflow/Underflow

From the above discussion, we can see that as long as the pixel grayscale values of the marked image in the spatial domain remain in the permitted range, the overflow/underflow problem can be avoided. In the rest of this paper, we assume the pixel grayscale value will be represented by 8 bits and the permitted range is [0,255]. If the permitted range is not [0,255] for some images, the proposed principle is still valid and the corresponding solution can be easily derived.

As said, our proposed scheme is block based, specifically, it splits the selected high frequency subband into nonoverlapping blocks, and the data hiding is carried out block by block. But the modification of the IWT coefficients in the adjacent blocks of a block under consideration will interfere with the grayscale values of the pixels corresponding to this block. Therefore, it is much easier to handle if we can remove this type of interference among neighboring blocks. Therefore, a mask is introduced so that the outer elements on the boundary of the mask are marked by "0"s and the inner elements by "1"s. At the bit embedding stage, in order to change the mean value of an IWT-coefficient block, we only change those coefficients whose positions are associated with "1" in the mask and keep the coefficients intact in the outer bound of the block. For the 5/3 filter bank, it can be proved that, when we change the IWT coefficients in one block by using this manner, the pixels grayscale values in adjacent blocks will not be affected.

After the interference among adjacent blocks has been removed, we can focus on one single block. Consider an IWT-coefficient block B_w and its corresponding block in the spatial domain B_s. Here, by corresponding, it is meant the group of pixels whose grayscale values are associated with the IWT-coefficient block. This correspondence relation can easily be determined. We assume the maximum absolute pixel grayscale value change is $S_{\max}$. In B_s, overflow may occur if there are pixels with their grayscale values greater than (255-$S_{\max}$) and the grayscale values need to be increased in the embedding process. Underflow may take place when there are pixels with their grayscale values less than $S_{\max}$ and the grayscale values need to be decreased. To avoid overflow and underflow, the above two scenarios should not occur. We call [0, $S_{\max}$) as *0-zone* and (255-$S_{\max}$, 255] as *255-zone*. In the spatial domain, according to the presence and absence of those two different zones, we can classify a block into one of the following four categories. The best case is that none of the above two zones are present, as is illustrated in Fig. 3(a), in which no overflow/underflow will occur. If in the B_s, there exist pixels in *0-zone* but no pixel in *255-zone*, as is illustrated by the solid line in Fig. 3(b), it is safe if all changes to the pixel grayscale values in this group are restricted to increase only. The resultant histogram after data hiding is represented by the dotted line in Fig. 3(b). Under the circumstances, we can see that no pixel will have the grayscale value greater than 255 or small than zero, indicating no overflow/underflow will occur. Fig. 3(c) depicts the opposite case to the previous one. If there is no pixel of this block in the *0-zone*, it will be safe if all changes to the pixels in this group are decreases. The worst case is described by Fig. 3(d) where pixels in *0-zone* and pixels in *255-zone* are both presented in one block. In this case, we do not change the coefficients of this block to avoid overflow and underflow.

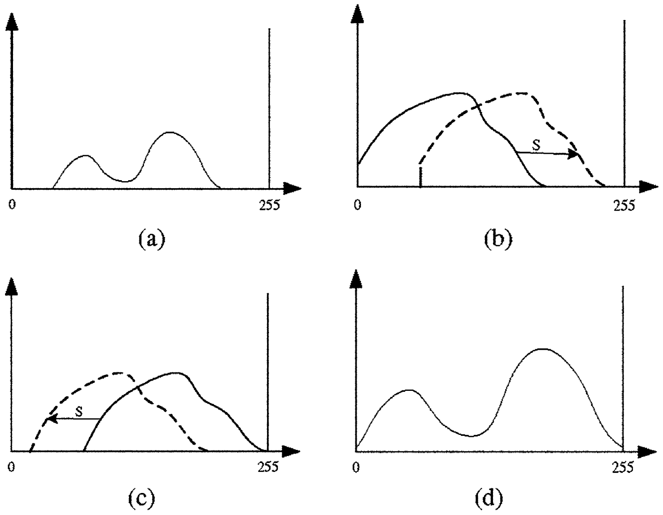


Fig. 3. Four categories of blocks. (a) Type A. (b) Type B. (c) Type C. (d) Type D.

TABLE I
5/3 FILTER COEFFICIENTS

n	Analysis Filter Coefficients		Synthesis Filter Coefficients	
	Low-pass Filter $h_L(n)$	High-pass Filter $h_H(n)$	Low-pass Filter $g_L(n)$	High-pass Filter $g_H(n)$
0	6/8	1	1	6/8
±1	2/8	-1/2	1/2	-2/8
±2	-1/8	0	0	-1/8

In the above, we have proposed a mechanism to embed bits into the IWT coefficients and have discussed how to avoid overflow/underflow by considering the tendency of grayscale change. The idea is that if we can find a way to embed one bit into one block of HL_1 or LH_1 subband and all the affected pixels will only move their grayscale values toward the desired direction, the overflow/underflow problem can be avoided.

As mentioned, in JPEG2000 standard, the 5/3 wavelet filter is used as the default filter for reversible image encoding [6]. The coefficients of the 5/3 wavelet filter are given in Table I [6].

Now, the effects of changing the wavelet coefficients on the spatial domain counterpart need to be investigated. From Table I, we can easily deduce the unit step response in the spatial domain for a unit step change occurred to an IWT coefficient located at the (i, j) position in the HL_1 subband. Specifically, an array of pixel grayscale values in the spatial domain centered at $(2i - 1, 2j)$ with a size of 3×5 (refer to Fig. 4) will change by the mount specified by the elements of this matrix if the IWT coefficient (i, j) in the HL_1 subband is increased by 1. The corresponding unit step responses can be found for the coefficients in LH_1 and LL_1 subbands.

Assuming the HL_1 is the carrier subband, from Table I, we know that the pixels affected by wavelet coefficients change will

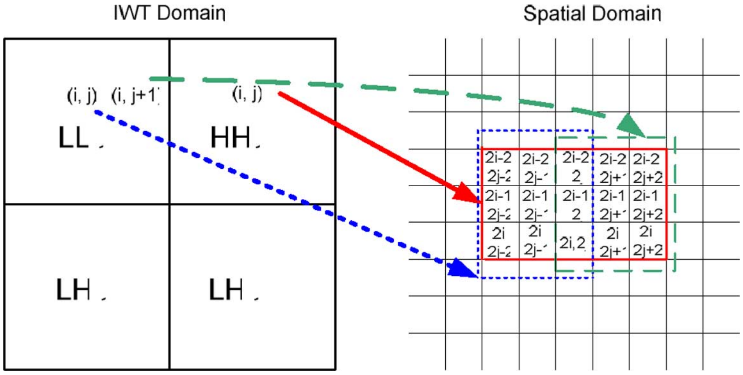


Fig. 4. Relationship between IWT coefficients and pixel in spatial domain.

change their grayscale values towards both increasing and decreasing directions. Similarly, this is true if the LH_1 is used as the carrier subband. But for changes in LL subband, the pixel grayscale values will move towards only one direction. The core idea of combating overflow/underflow here is to manipulate the coefficients in the LL subband to force the resultant pixel grayscale value change toward only one desired direction.

Now we study the unit response of LL coefficients. It can be deduced that for an LL_1 coefficient at (i,j), the center of the unit step response is at $(2i-1,2j-1)$ in the spatial domain, while the center of unit step response will be at $(2i-1,2j+1)$ for LL_1 coefficient $(i,j+1)$. Fig. 4 illustrated the relationship of an IWT coefficients and its unit step response in the spatial domain. The solid arrow and the solid box are for the coefficient (i,j) in HL_1 subband; the dotted ones are for LL_1 coefficient (i,j); the dashed ones are for LL_1 coefficient $(i,j+1)$.

It is observed that the combined 2-D arrays corresponding to the unit step responses of LL_1 coefficients (i,j) and $(i,j+1)$ have the size of 3×5 and cover the exactly same group of pixels corresponding to the unit step response of the IWT HL_1 coefficient (i,j). We use $C_{\mathrm{HL}}(i,j)$ to denote the coefficient at (i, j) of the HL_1 subband and $C_{\mathrm{LL}}(i,j)$, the coefficient at (i,j) of the LL_1 subband. Then, it is straightforward to verify that the combined effect of the unit step response of $C_{\mathrm{HL}}(i,j)$, the 1/4 of the unit step response of $C_{\mathrm{LL}}(i,j)$ and the 1/4 of the unit step response of $C_{\mathrm{LL}}(i,j+1)$ can be represented by

$$R = \begin{bmatrix} 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 \end{bmatrix}. \qquad (2)$$

which indicates that all the change in the spatial domain will have the same sign. In other words, if we decide to embed data in the HL_1 subband, and if we change an HL_1 coefficient at (i,j) by S and change both LL_1 coefficients at (i,j) and $(i,j+1)$ by $S/4$, the pixel grayscale value change in the spatial domain will have the same sign as that of the S. It is noted that after the embedding, the maximum pixel grayscale value change in the spatial domain is also S. Similarly, if the LH_1 serves as the carrier subband, the same goal can be achieved by changing the LH_1 coefficient at (i,j) by S and both the LL_1 coefficients at (i,j) and $(i+1,j)$ by S/4.

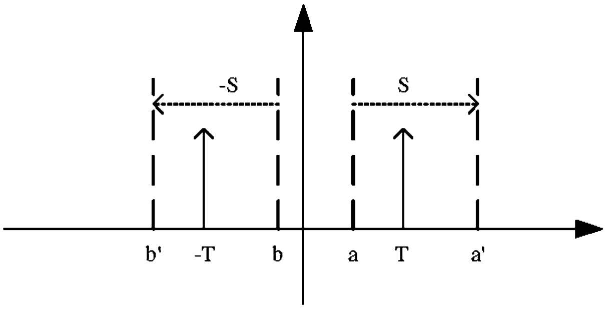


Fig. 5. Embedding rule for Type A block.

C. *Semi-Fragile Lossless Watermarking Algorithm*

If the payload bit is “0,” we do not change this block. If it is “1,” different embedding strategies are applied to blocks of different categories showed in Fig. 3.

1) For Blocks of Type A: In this case, overflow and underflow will not take place. We can either increase the mean value or decrease the mean value of the carrier subband (HL_1 or LH_1) coefficients in the block. The detail is illustrated in Fig. 5. Actually, “Lena” image, the most popular picture in image processing community, has only this type of blocks due to its lack of extreme white and extreme black regions.

2) For Blocks of Type B: To embed a bit “1,” the absolute value of the mean should be increased by S. If the original mean value is greater or equal to zero, as is the case of a in Fig. 6, the mean can be shifted to a'. Besides, we modify the corresponding coefficients in the LL subband by $S/4$. As the result, the resultant pixel values in the spatial domain will only increase after changes are made. Therefore, underflow is avoided. If the mean value is negative, say, b, no change to this block. This is equal to embedding a bit “0.” We rely on the error correction coding (ECC) to correct the error bits.

3) For Block of Type C: This case is similar to the previous one except everything is opposite to Type “b.” For bit “1,” only the blocks with a negative mean can be used to hide bit “1.” Fig. 7 illustrated the detail.

4) For Block of Type D: This type of blocks is illustrated in Fig. 3(d). Changes on the mean value in either direction will cause either overflow or underflow. No changes to this type of blocks. We rely on the ECC to rectify the possible error.

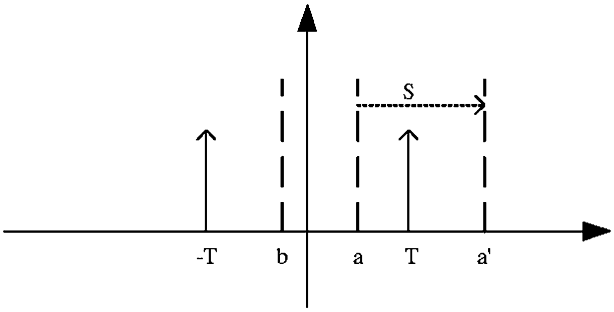


Fig. 6. Embedding rule for Type B block.

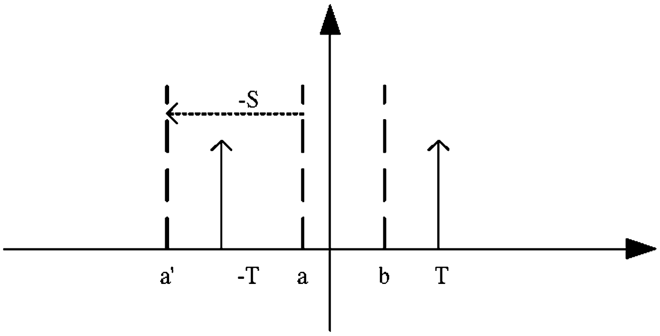


Fig. 7. Embedding rule for Type C block.

It is noted that for frequently used images, most blocks belong to Type A. For example, all blocks are Type A blocks for Lena image. For other types of images, the ECC is necessary to be used to correct the errors. In some cases, the errors may muster together. This is called burst error. At this circumstance, ECC cannot efficiently correct them. At this circumstances, permutation or interleaving technique is needed to covert the burst errors into random-like errors, hence making ECC rather effective [8]. Chaotic mixing [9] is used in our experiments to combat burst errors. Fig. 8 shows the block diagram of the data embedding. Block size B, threshold T, shifting value S, and which subband is chosen as the carrier are saved as side information which will be needed for data extraction and cover media recovering later. From a different perspective, the side information can be treated as a key and is needed to be transmitted through a separate secure channel. Only authorized person can extract the hidden data with the provided side information. An alternative way of our proposed algorithm is to use the same B, T, S, and subband for all images. As a result, the side information will not be needed anymore. The drawback of doing so is the parameters will not be the optimal for a given particular image.

D. Data Extraction and Original Image Recovering

At the receiver side, a stego-image is the input. Data extraction and original cover image restoration are performed.

For Hidden data extraction, the integer wavelet coefficients of the stego-image are first obtained. Then the carrier subband is divided into nonoverlapping blocks with size B. From each block, we extract one bit of data. At this stage, we do not need to care about which category this block falls into. The comparison of the mean value with the threshold T will decide whether bit “0” or bit “1” is extracted.

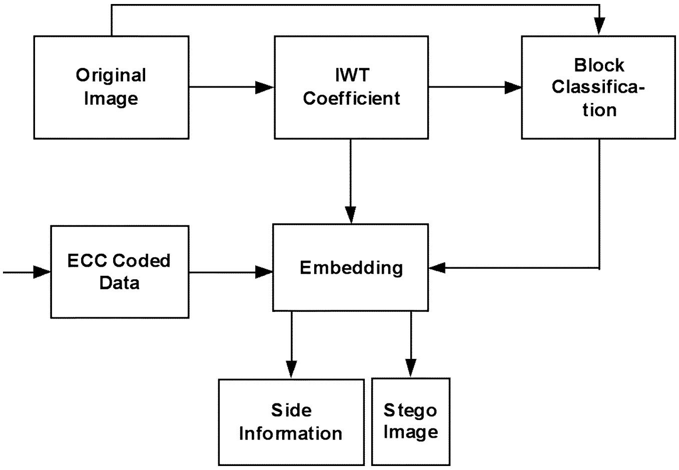


Fig. 8. Block diagram of data embedding.

After all hidden data have been extracted we can possibly recover the original cover media. Two possibilities exist here. One is the stego-image has not been subjected to any alteration. The other is that some alteration has been applied to the stego-image. In the former case, the original image can be recovered with no error. In the latter case, the original image may not be able to be recovered with no error, the hidden data may still be able to be recovered correctly.

IV. Experimental Results and Performance Analysis

For the sake of generality, the bit stream to be embedded is randomly generated. We applied our algorithm to various images including those in the USC SIPI image database[1] and JPEG2000 testing images. For all of the test images, our proposed algorithm works well. No salt-and-pepper noises exist for all test images and the visual quality of stego-images produced by our proposed algorithm is much higher than the method in [5]. Specifically, the peak signal-to-noise ratio (PSNR) of our stego images are all over 38 dB.

To analyze the performance of the proposed algorithm, we presents the testing results in detail with the Lena image, one of the most commonly used images in the image processing community, which is an 8-bit grayscale image with a size of 512×512.

In our algorithm, the capacity is determined by the block size B. The smaller the B is, the larger the number of blocks will be, consequently, more bits can be embedded. Table II gives the relationship between the block size, its capacity and PSNR of the marked image embedded using minimum shift values. For ECC, we use BCH(15,11) [8] in experimental investigation.

From Table II, it is obvious that smaller block size can provide larger capacity. However, the $m_{\max}$—the maximum absolute mean value of IWT coefficient in a block—will be possibly larger. A bigger shift quantity S is needed thus degrading the quality of the marked image and causing lower PSNR. The “minimum shift value” is defined as the smallest integer that is greater than $m_{\max}$.

[1] USC Image Database. [Online] Available: http://sipi.usc.edu/services/database/Database.html

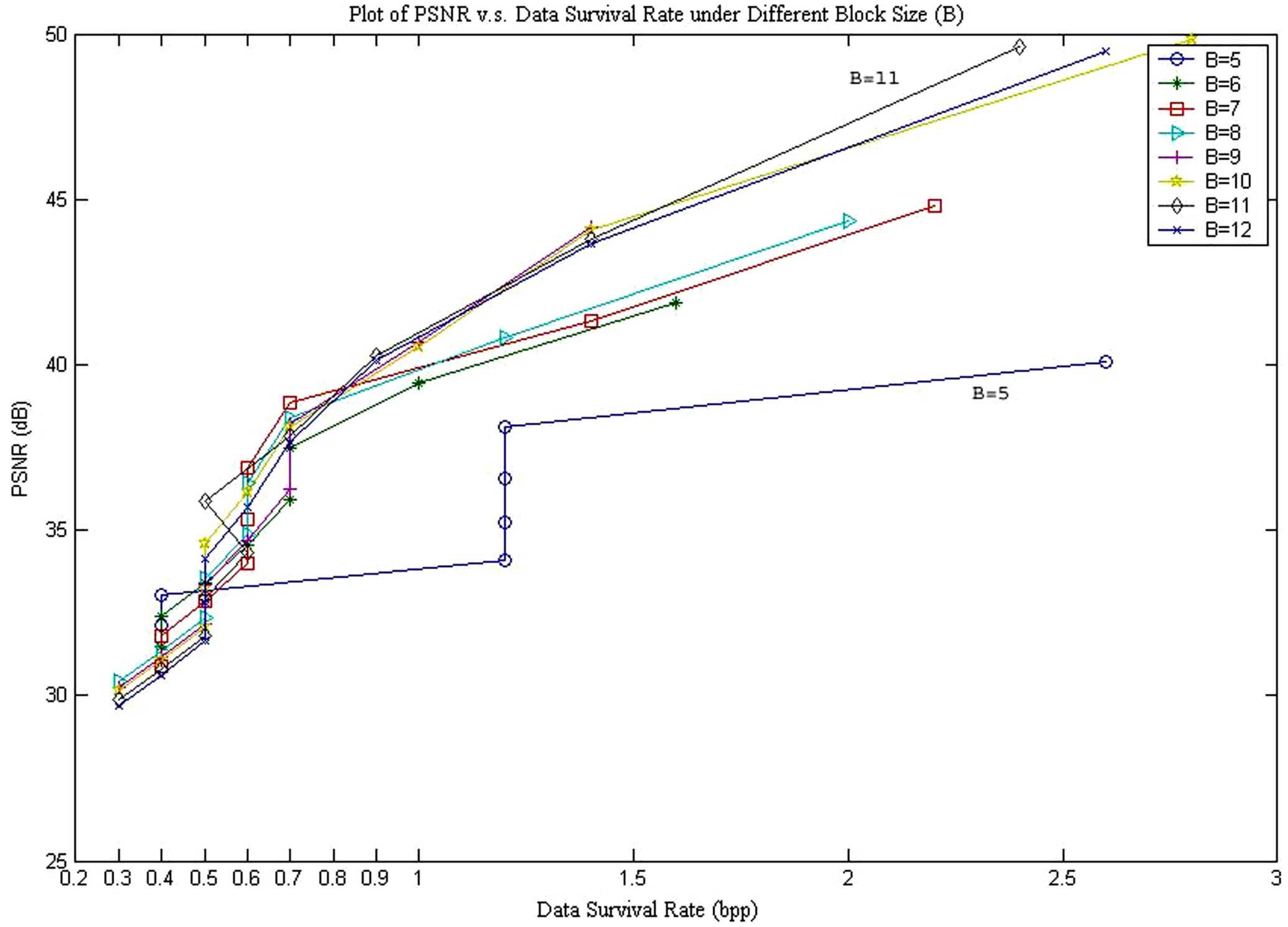


Fig. 9. Data survival rate versus PSNR under different block sizes.

TABLE II
BLOCK SIZE VERSUS CAPACITY (LENA: 512 × 512 × 8)

Block size	ECC Scheme	Capacity	Minimum shift values	PSNR (dB)
5	(15,11)	1907	8	40.09
6	(15,11)	1293	6	41.87
7	(15,11)	950	4	44.81
8	(15,11)	750	4	44.36
9	(15,11)	574	4	44.18
10	(15,11)	458	2	49.86
11	(15,11)	387	2	49.62
12	(15,11)	323	2	49.46

TABLE III
TEST RESULTS AGAINST ADDITIVE GAUSSIAN NOISE

Bit Error Rate % (BER)		Block Size							
		5	6	7	8	9	10	11	12
Shift Value	2	-	-	-	-	-	30.57	35.40	37.46
	4	-	-	21.05	16.00	8.01	5.90	2.58	0
	6	-	24.21	11.68	4.27	1.39	0.87	0	0
	8	34.45	18.25	9.16	2.40	0.52	0	0	0
	10	16.36	12.30	2.00	1.20	0	0	0	0
	12	2.41	2.17	0.42	2	0	0	0	0
	14	0.73	0.46	0	0	0	0	0	0
	16	0	0	0	0	0	0	0	0
	18	0	0	0	0	0	0	0	0
	20	0	0	0	0	0	0	0	0

Different from most lossless watermarking algorithms, the proposed IWT based lossless watermarking algorithm is robust to incidental alterations. This is discussed below.

At first, it is robust to JPEG2000 compressions. In experimental study, the marked images are JPEG2000 lossy compressed with increasing compression ratios. The robustness against JPEG2000 compression is measured by data survival rate, meaning that when the resultant data rate after compression is above or equal to the data survival data rate, the hidden data can be reliably extracted without error. In other words, the lower the data survival rate is, the more robust the hidden data are. Fig. 9 shows the relationship between stego-image PSNR and data survival rate under different block sizes. It can be observed that in general, the lower the PSNR, the lower the data survive rate is, and thus the more robust the hidden data are. For the same data survive rate, generally, the larger the block size, the higher the PSNR is, at the same time, the lower the capacity will be. It is obvious that the stego-image visual quality, the hidden data robustness and the data embedding capacity interact with each other. Compromise needs to be made for different purposes in the different applications.

Additive Gaussian noise is the noise frequently encountered in communication systems. In our experiments, a Gaussian noise with zero mean and variance of 0.001 is added to the marked images. The bit-error rate in hidden data extraction is counted and listed in Table III. The following observation can be made that the larger the block size, the larger the shift value, and the more robust the hidden data will be.

V. Conclusion

In this paper, a new semi-fragile lossless data hiding scheme is proposed. A preliminary report can be found at [10]. Data are embedded into the integer wavelet domain. It can be integrated into JPEG2000 standard seamlessly. The issue of overflow and underflow is addressed. The original cover media can be obtained after hidden data extraction if the stego-image has not been lossily altered. The hidden data are robust to nonmalicious attacks, e.g., lossy compression, to a certain degree. Performance analysis is conducted which shows the advantages of the proposed algorithm. The visual quality of the marked image is dramatically improved compared with the prior-art [5]. Specially, no salt-and-pepper noise and other artifacts are presented in the stego-images. It can be applied to content based image authentication and some other applications. In fact, this technique has been adopted by a joint proposal by Institute of Infocomm Research, Singapore, and New Jersey Institute of Technology, Newark [11].

References

[1] D. Zou, C. W. Wu, G. Xuan, and Y. Q. Shi, "A content-based image authentication system with lossless data hiding," in *Proc. ICME*, Baltimore, MD, Jul. 2003, vol. 2, pp. 6–9.

[2] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Proc. SPIE Security Watermarking Multimedia Contents*, San Jose, CA, Jan. 2001, vol. 4314, pp. 197–208.

[3] G. Xuan, J. Chen, J. Zhu, Y. Q. Shi, Z. Ni, and W. Su, "Distortionless data hiding based on integer wavelet transform," *IEEE Electron. Lett.*, vol. 38, no. 25, pp. 1646–1648, Dec. 2002.

[4] Z. Ni, Y. Q. Shi, N. Ansari, and S. Wei, "Reversible data hiding," in *Proc. Int. Symp. Circuits Systems*, Bangkok, Thailand, May 2003, vol. 2, pp. 25–28.

[5] C. De Vleeschouwer, J. F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 97–105, Mar. 2003.

[6] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consum. Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[7] *JPEG 2000 Part I Final Committee Draft Version 1.0.*, ISO/IEC JTC 1/SC 29/WG 1 N1646R, Mar. 2000 [Online]. Available: http://www.jpeg.org/public/fcd15444-1.pdf

[8] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.

[9] G. Voyatzis and I. Pitas, "Chaotic mixing of digital images and applications to watermarking," *Proc. ECMAST*, pp. 687–695, May 1996.

[10] D. Zou, Y. Shi, and Z. Ni, "A semi-fragile lossless digital watermarking scheme based on integer wavelet transform," *Proc. MMSP2004*, pp. 195–198.

[11] *A Unified Authentication System for JPEG 2000 Images*, FDIS ISO/IEC JTC 1/SC 29/WG 1N3853, Security Part of JPEG2000 (JPSEC), Nov. 2005.