An Adaptive Reversible Image Watermarking Scheme Based on Integer Wavelet Coefficients

Chien-Chang Chen
Department of Computer Science and Information Engineering, Tamkang University
Email: ccchen34@mail.tku.edu.tw

Zeng-Chuang Jiun
Department of Computer Science and Information Engineering, Tamkang University
Email: u9310632@gmail.com

Chien-Hsiang Hsu
Department of Information Management, Hsuan Chuang University
Email: vitamin1115@hotmail.com

***Abstract*—This paper presents an integer wavelet coefficients based reversible image watermarking scheme. A reversible image watermarking approach extracts the embedded watermarks from a watermarked image and recovers the watermarked image to the original image simultaneously. The proposed approach first applies the host image to 3-layered integer wavelet transform. Nine subimages are acquired from the 3-layered integer wavelet transform. Each subimage is then segmented to blocks of size $2^L \times 2^L$, where L is determined by structure of the subimage. Then, reversible watermarks are embedded into differences between central ordered pixel and other pixels in each block. Largest difference in each block determines the embedded quantity in each difference. Experimental results show that the proposed adaptive block size scheme has high capacity and quality ratio. A better selection on subimage embedding is also provided.**

***Index Terms*—reversible image watermarking, integer wavelet transform, difference expansion, adaptive block size**

I. INTRODUCTION

Digital images are easily transmitted, modified and reproduced. Therefore, preserving the ownership of a digital image is an important issue. Watermarking has recently become important in solving this image authentication problem [4]. Reversible watermarking techniques adjust an image by embedding user authentication information and extract watermarks from the watermarked image with recovering to the original image. This application is essential in some special images, like medical or military images.

Research on reversible watermarking approaches is briefly reviewed. Alattar [1] presented difference expansion method to embed reversible watermarks. Kamstra and Heijmans [6] improved the DE scheme on lossless compression. Thodi and Rodriguez [10] embedded the location map that the DE scheme required using histogram shifting technique. They then proposed a prediction-error expansion method to embed reversible watermarks. Celik et al. [2] compressed the remainder of an image to embed reversible watermarks in the saved space of the remainder. Ni et al. [9] presented a histogram scheme of embedding a watermark bit 1 by adding the histogram peak-pixel value and embedding a 0 with no modification. Lin et al. [8] extended Ni et al. method to absolute numbers to improve embedded quantity under the difference between 4×4 blocks. Tsai et al. [12] presented a difference calculation in a 5×5 block to improve the embedded capacity that Ni et al. method has. Hong et al. [5] employed the prediction errors and the histogram modulation method to embed watermarks. Tian [11] presented the method to embed one watermark bit into the LSB of the difference between two pixels. Lee et al. [7] segmented a host image to 2×2 blocks and embedded reversible watermarks to each block by Tian's method. Chen and Tsai [3] further improved Lee et al.'s work [7] to segment an image to adaptive blocks.

This study presents adaptive image watermarking scheme under integer wavelet transform. The transform segments a host image to subimages and this segmentation reduces each subimage's distortion. Therefore, the embedded quantity can be improved in high-frequency coefficients, like three 1-layered subimages. Two different experiments are performed in this work. First, a comparison between the proposed integer wavelet based reversible image watermarking scheme and two related literatures [3, 7] are provided to demonstrate the performance of proposed scheme. Second, the embedded capacities in different combination of subimages are also compared.

The rest of this paper is organized as follows. Section 2 reviews related works of the proposed scheme. Section 3 introduces the proposed integer wavelet transform based reversible image watermarking method. Section 4 presents the experimental results of the proposed method.

doi:10.4304/jcp.8.7.1677-1682

Comparisons with other related works are also provided. Finally, Section 5 draws a brief conclusion.

II. RELATED WORKS REVIEW

The proposed reversible watermarking method embeds reversible watermarks into integer wavelet coefficients using the proposed adaptive block sized modulation method. The related works are difference expansion method [11], fixed block sized modulation method [7], and adaptive block sized modulation method [3] and these works are briefly introduced in Sections 2.1, 2.2, and 2.3, respectively.

2.1 Review of different expansion method

The difference expansion method [11] embeds a watermark bit into difference between a pair of pixels. Assume that (x, y) denote two pixels in the host image. First, difference and mean of these two pixels is calculated from Equation (1). Then, extend the difference d' by $2\times d$ and embed a watermark bit b by $2\times d+b$ by Equation (2). At last, recover to embedded pair of pixels (x', y') by Equation (3).

$$d = x - y\,,\; m = \left\lfloor \frac{x+y}{2} \right\rfloor \tag{1}$$

$$d' = 2\times d + b \tag{2}$$

$$x' = m + \left\lfloor \frac{d'+1}{2} \right\rfloor \text{ and } y' = m - \left\lfloor \frac{d'}{2} \right\rfloor \tag{3}$$

In the difference expansion method, a watermark bit b is embedded into LSB of calculated difference d'. Equation (2) multiplies 2 for embedding one bit. Multiplying 4 lets embed capacity to 2 bits. The watermark can be extracted by the embedding similar steps.

2.2 Review of fixed block sized modulation method

Lee et al. [7] proposed fixed block sized modulation method that is an improvement of above difference expansion method [11]. The fixed block sized method first segments the host image to combination of non-overlapping 2×2 blocks. Assume that $t(i)$ ($1\le i\le 4$) denotes each ordered pixel in a 2×2 block with $t(i)\le t(j)$ of $i<j$ and $1\le i,j\le 4$. Then, watermarks are embedded into $t(k)$-$t(2)$, k=1,3,4, by Tian’s difference embedding method [11]. The embedded quantity (0, 1, or 2 bits in each difference of a block) is determined by the block structure, which is calculated by the largest difference d between the middle pixel $t(2)$ and all pixels in a block defined as $max\{|t(k)-t(2)|\}$. Four embedding strategies are adopted by the difference d and user predefined threshold T as follows.

1. If $d<T$, then embed 2 bits to each pixel difference.

2. If $T \leqq d<4T$, then embed 1 bit to each pixel difference.

3. If $4T \leqq d<8T$, then embed 1 bit to LSB of each pixel difference and record the original difference LSB bit.

4. If $d \geqq 8T$, then embed nothing.

2.3 Review of adaptive block sized modulation method

Chen and Tsai [3] further improved Lee et al. method [7] to present adaptive block sized modulation method. The adaptive strategy partitions the host image to different size of blocks and this partition is based on variance of block in an image.

Given an image of size $2^N\times 2^N$, the proposed method segments it adaptively to blocks of size $2^L\times 2^L$, where L starts from a user-defined number L_{max} to 1, and then embeds watermark by the following strategy. Denote $t(i)$, $1\le i\le 2^{2L}$ as ordered pixels in an $2^L\times 2^L$ block satisfying $t(i)\le t(j)$ of $i<j$, $1\le i\le 2^{2L}$ and $1\le j\le 2^{2L}$. Watermarks are embedded into $t(k)$-$t(2^{2L-1})$, $1\le k\le 2^{2L}$ excepting 2^{2L-1}, by Tian’s difference embedding method [11].

A simple example is provided for explaining the adaptive block sized modulation method. Assume L_{max} is set to 4 and $t(i)$ represent ordered pixels in a block. The block segmentation scheme to acquire blocks of size 16×16, 8×8, 4×4, or 2×2 is described as follows.

1. If the maximum difference $max\{|t(k)-t(128)|\}$ in an 16×16 block is larger than predefined threshold T, then the 16×16 block is segmented to four 8×8 blocks. Otherwise, 2 watermark bits are embedded into each pixel difference $t(k)$-$t(128)$.

2. For each 8×8 block, if the maximum difference $max\{|t(k)-t(32)|\}$ in this block is larger than T, then the 8×8 block is segmented to four 4×4 blocks. Otherwise, 2 watermark bits are embedded into each pixel difference $t(k)$-$t(32)$.

3. For each 4×4 block, if the maximum difference $max\{|t(k)-t(8)|\}$ in this block is larger than T, then the 4×4 block is segmented to four 2×2 blocks. Otherwise, 2 bits are embedded into each pixel difference $t(k)$-$t(8)$.

4. For each 2×2 block, applying the following steps.

4.1 If the maximum difference $max\{|t(k)-t(2)|\}$ is smaller than T, then embeds 2 bits to each $t(k)$-$t(2)$, k=1,3,4.

4.2 Else, if the maximum difference $max\{|t(k)-t(2)|\}$ is smaller than $4T$, then embeds 1 bit to each $t(k)$-$t(2)$, k=1,3,4.

4.3 Else, if the maximum difference $max\{|t(k)-t(2)|\}$ is smaller than $8T$, then embeds 1 bit to each $t(k)$-$t(2)$, k=1,3,4, and record the LSB of each difference $t(k)$-$t(2)$.

4.4 Else, embeds nothing.

5. At last, combine all processed blocks to acquire the watermarked image. The image segmentation format according with the block embedded type for recovering the watermarked image to the original host image also needs to be recorded.

III. PROPOSED METHOD

This section introduces the proposed reversible image watermarking method. The proposed method first applies the host image to integer wavelet transform to acquire subimages. Then, each subimage is segmented to adaptive blocks according to their structures and user-predefined thresholds. The segmentation is based on the measurement of the largest difference between the middle pixel and all other pixels in a sorted block pixels. The proposed embedding and extracting algorithms are introduced in Section 2.1 and 2.2, respectively. A property analysis of the proposed method is discussed in Section 3.3.

3.1 Watermark Embedding Algorithm

This section introduces the embedding algorithm of the proposed reversible image watermarking method. Given a host image of size $2^N \times 2^N$, the proposed method first applies the image to 3-layered integer wavelet transform to acquire 3 subimages with a size of $2^{N-1} \times 2^{N-1}$, 3 subimages with a size of $2^{N-2} \times 2^{N-2}$, and 4 subimages with a size of $2^{N-3} \times 2^{N-3}$. Each subimage then segments to adaptive blocks of size $2^L \times 2^L$ according to their block structures, where L starts from a user-defined number L_{max} to 1. The segmentation is performed by threshold G, in which values G are varied according to the subimage level. Watermarks are embedded into $t(k)$-$t(2^{2L-1})$, where $t(i)$ denotes ordered pixels in an $2^L \times 2^L$ block and $1 \le k \le 2^{2L}$ excepting 2^{2L-1}. Max{$|t(k)$-$t(2^{2L-1})|$} and user-defined thresholds $T1$, $T2$, $T3$ determine the embedded quantity (0, 1, or 2 bits in each $2^L \times 2^L$ block).

The proposed watermark embedding algorithm is described as follows.

1. Apply the host image to 3-layered integer wavelet transform.

2. Perform each subimage to the following steps to partition to blocks.

2.1 Denote B be the image block and calculate var(B) that is defined by variance calculation of B.

2.2 If var(B) > G, partition B to four new blocks B' with size $\frac{B}{2} \times \frac{B}{2}$ and apply each B' to step 2.1.

2.3 If var(B) ≤ G, block B is a segmented block.

2.4 Collect all segmented blocks in step 2.3 to from the segmented format of a subimage.

3. Apply the following steps to each block $2^L \times 2^L$ to embed watermarks.

3.1 Sort a $2^L \times 2^L$ block and denote by $t(i)$, $1 \le i \le 2^{2L}$.

3.2 Calculate the maximum difference d, which is defined by max{$|t(1)$-$t(2^{2L-1})|$, $|t(2^{2L})$-$t(2^{2L-1})|$}.

3.3 If d ≦ $T1$, then embeds 2 bits to the difference between $t(2^{2L-1})$ and all other pixels, and perform next block acquired from Step 2.4.

3.4 Else if d ≦ $T2$, then embeds 1 bit to the differences between $t(2^{2L-1})$ and all other pixels, and perform next block acquired from Step 2.4.

3.5 Else if d ≦ $T3$, then embeds 1 bit to the differences between $t(2^{2L-1})$ and all other pixels, record the LSB, and perform next block acquired from Step 2.4.

3.6 Else embed nothing.

4. Combine all processed blocks and apply inverse wavelet transform to acquire the watermarked image. Each subimage segmentation format accompanying with each block's embedded type needs be recorded.

Step 2 segments each subimage to blocks, in which three G values are assigned for different subimage levels. For example, G=[2, 6, 14] representing G values for 3-level, 2-level, 1-level subimages are 2, 6, 14, respectively. Step 3 embeds watermarks into each difference in blocks by three thresholds. For example, our experiments adopt $T1=2$, $T2=8$, $T3=20$ corresponding with 3-level, 2-level, 1-level subimages, respectively. The subimage segmentation format records the segmented block size of each subimage. The block embedded type records the embedded quantity being 2 bits, 1 bit, or 0 bit in a block's difference. Both subimages segmentation format and block embedded type need to be recorded for watermark extraction and original host image recovery.

3.2 Watermark Extracting Algorithm

This section demonstrates the watermark extracting algorithm, in which the original host image is obtained after extracting the embedded watermarks. The image segmentation format and the block embedded type generated from embedding algorithm are both needed in watermark extracting algorithm. The proposed watermark extracting algorithm is described as follows.

1. Apply the host image to 3-layered integer wavelet transform.

2. Use subimage segmentation format to partition each subimage to blocks.

3. For each block, denoted by $2^L \times 2^L$, apply the following steps to extract watermarks.

3.1 Extract watermarks embedded in this block by LSB or LSB2 bits of each difference based on the block embedded type.

3.2 Recover the block from above watermark-extracted block differences by recovery procedure of difference expansion method.

4. Combine all watermarks as the extracted watermarks.

5. Combine all recovery blocks to acquire all subimages and perform inverse wavelet transform to acquire the original host image.

In Step 3.2, the extracted steps of difference expansion based reversible watermarking scheme [11] are applied to extract embedded watermarks and each difference pair is recovered to original values.

3.3 Property Analysis

The proposed scheme use integer wavelet transform to partition a host image to ten subimages that include one low-frequency subimage $L3$ and nine high-frequency subimages $H_{i,j}(1 \le i \le 3, 1 \le j \le 3)$ as shown in Figure 1. Two important subimage properties determine the embedding strategy. First, the low-frequency subimage is ignored in watermark embedding algorithm because any modification on $L3$ subimage leads to large image distortion. Therefore, the watermarks are only embedded in high-frequency subimages H. Second, modification on different levels of subimage causes different visual distortion. Modifying subimages $H_{1,j}(1 \le j \le 3)$ has lower visual distortion than modifying subimage $H_{3,j}(1 \le j \le 3)$ since the $H_{1,j}$ are highest frequency subimages. Therefore, assigning different thresholds according to their subimage levels are applied in our experiments. Different threshold combinations are chosen for finding better results in the proposed scheme.

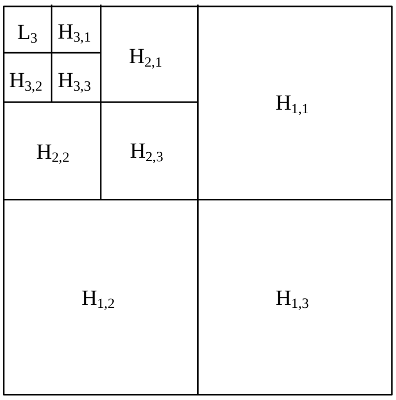
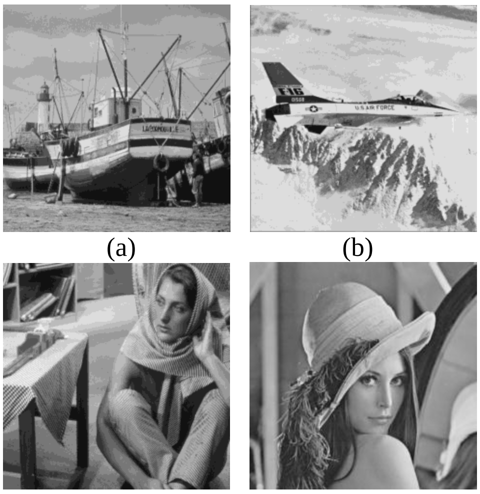


Figure 1. Subimages under integer wavelet transform.

IV. Experimental Results

This section presents the experimental results of the proposed method. Figure 2 shows four test images with size of 512×512. The embedded watermarks are randomly generated and the basis of integer wavelet transform is Haar. Each subimage is segmented to blocks of sizes from 8×8, 4×4, to 2×2. Largest block size is 8×8 since no partitioned block of size larger than 8×8 is found in all subimages. The experimental results are compared with Lee et al. [7] and Chen and Tsai [3]. The threshold *T* is empirical determined and listed in Table 1.



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

Figure 2. Test images: (a) Boat; (b) Airplane; (c) Barbara; (d) Lena.

Table 1 shows the experimental results comparing with Chen's [3] and Lee's [7] method on the test image. The adopted thresholds (G, T) are empirical determined by $G1$=2, $G2$=6, $G3$=18, and $T1$=2, $T2$=8, $T3$=32 under subimages $H_{1,j}$, $H_{2,j}$, $H_{3,j}$, respectively. The PSNR and Capacity changes are compared with Lee's method. In test image Boat, comparing with Lee's 255789 bits, Chen's improves the capacity of 7.22% and the watermarked image quality reduces 0.74%. Furthermore, the proposed method improves the capacity of 14.62% with reducing image quality of 5.83%. The similar results also acquired from the other three test images. Therefore, from Table 1, the proposed scheme acquires a good balance between capacity and PSNR.

Table 1. Comparisons with two works on test images with thresholds $G1$=2,$G2$=6,$G3$=18 and $T1$=2,$T2$=8,$T3$=32.

Test Image	Methods	PSNR	PSNR rate	Capacity	Capacity rate
Boat	Lee's [7]	34.31		255789	
Boat	Chen's [3]	34.06	-0.74%	274266	7.22%
Boat	Proposed method	32.30	-5.83%	293193	14.62%
Airplane	Lee's [7]	34.71		282147	
Airplane	Chen's [3]	34.33	-1.10%	308016	9.17%
Airplane	Proposed method	32.26	-7.04%	326463	15.71%
Barbara	Lee's [7]	35.07		214833	
Barbara	Chen's [3]	34.95	-0.34%	219990	2.4%
Barbara	Proposed method	32.09	-8.51%	243906	13.53%
Lena	Lee's [7]	34.26		253536	
Lena	Chen's [3]	34.14	-0.33%	259992	2.55%
Lena	Proposed method	31.04	-9.40%	293403	15.72%

Table 2. Performance of applying different set of thresholds on test image Boat.

PSNR/Capacity	$G1$=2,$G2$=6,$G3$=14	$G1$=2,$G2$=6,$G3$=18
$T1$=2,$T2$=8,$T3$=20	33.9849/260553	33.1198/278355
$T1$=2,$T2$=8,$T3$=24	33.7439/265413	32.9476/283215
$T1$=2,$T2$=8,$T3$=26	33.2790/272592	32.5889/288957
$T1$=2,$T2$=8,$T3$=32	32.9233/276828	32.3013/293193

Table 2 provides performance of the proposed method applying different sets of thresholds on test image Boat. The experimental results show that smaller thresholds acquire lower capacity and lower image distortion. Larger thresholds acquire higher capacity with higher image distortion. Therefore, the embedded capacity can be efficiently adjusted by selecting thresholds. Comparing with previous methods [3, 7], the proposed scheme has better capacity than Lee's method [7] or Chen's method [3] with limited image distortion. However, the proposed scheme can easily embed watermarks to whole image by embedding into one subimage.

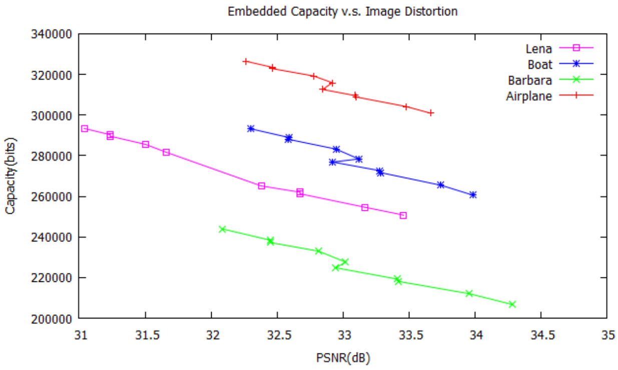


Figure 3. Capacity v.s. image distortion in four test images.

A comparison on four test images under different thresholds is depicted in Figure 3. In this figure, the rate between embedded capacity (bits) and image distortion (PSNR) is highly related to the image structure. For example, a smooth image Airplane has better capacity/PSNR rate than image Barbara which is not as smooth as Jet. Acquiring a covered image with certain PSNR value, the capacities are ordered as Airplane>Boat>Lena>Barbara. Relatively, the embedded image capacities are ordered as Barbara<Lena< Boat<Airplane for acquiring nearly capacity.

Table 3. PSNR/Capacity ratio of embedding into different combinations of subimages.

PSNR/ Capacity	Boat	Babara	Airplane	Lena
$H_{1,1}$	36.43/84276	36.11/70992	34.41/97875	34.89/86298
$H_{1,2}$	40.72/82722	40.93/67695	41.25/97158	41.27/87825
$H_{1,3}$	37.11/90786	36.29/81912	36.94/94749	34.79/91851
$H_{1,1}+H_{2,1}$	35.84/96627	35.85/78411	35.65/111069	34.62/95625
$H_{1,2}+H_{2,2}$	40.34/92010	40.74/73917	40.81107346	41.06/94137
$H_{1,3}+H_{2,3}$	36.51/102426	35.92/90684	36.39/106050	34.48/102078
$H_{1,1}+H_{2,1}+H_{3,1}$	35.79/97542	35.66/78708	35.63/111876	34.59/96171
$H_{1,2}+H_{2,2}+H_{3,2}$	40.30/92526	40.65/74100	40.63/107862	41.04/94428
$H_{1,3}+H_{2,3}+H_{3,3}$	36.39/103125	35.88/91098	36.33/106725	34.44/102804

As we have discussed in Section 2.3, watermarks can be only embedded into some high-frequency subimages for limited embedding with reducing visual distortion as decreasing PSNR value. Embedding watermarks into each subimage has the property of embedding into the whole image. Table 3 shows the experimental results of embedding into different combinations of high-frequency subimages. Figure 4 plots the capacity v.s. image distortion in four test images. Among these four images, Airplane has better capacity/PSNR ratio.

To provide a clear comparison on embedding watermarks into different combinations of $H_{i,j}(1\le i\le 3)$, Figures 5-7 depict the capacity v.s. PSNR value ratio between embedding watermarks into different j of subimages $H_{i,j}(1\le i\le 3)$. For example, $H_{1,1}$, $H_{1,1}+H_{2,1}$, and $H_{1,1}+H_{2,1}+H_{3,1}$ are three combinations of subimage $H_{i,1}(1\le i\le 3)$ and experimental results of embedding into them are depicted in Figure 5. Furthermore, experimental results of embedding into $H_{1,2}$, $H_{1,2}+H_{2,2}$, $H_{1,2}+H_{2,2}+H_{3,2}$ and embedding into $H_{1,3}$, $H_{1,3}+H_{2,3}$, $H_{1,3}+H_{2,3}+H_{3,3}$ are depicted in Figure 6 and Figure 7, respectively.

From these figures, some properties are listed as follows. Considering only embedding into highest-frequency subimages $H_{1,j}(1\le j\le 3)$, embedding into $H_{1,2}$ is better choice than only embedding into $H_{1,1}$ or $H_{1,3}$, both in capacity and PSNR value. Embedding into $H_{1,3}$ has higher capacity and better PSNR value than embedding into $H_{1,1}$. Embedded capacities in $H_{1,1}$ and $H_{1,2}$ are close, but in $H_{1,2}$ preserves better PSNR value. Therefore, the better selection locates on embedding into $H_{1,2}$ or $H_{1,3}$. When only embedding into highest-frequency subimages. At last, the best selection for requiring high capacity should be $H_{1,3}$ and the best selection for requiring better image distortion should be $H_{1,2}$. The same properties existed in that $H_{1,2}+H_{2,2}$ is better than $H_{1,1}+H_{2,1}$ and $H_{1,2}+H_{2,2}+H_{3,2}$ is better than $H_{1,1}+H_{2,1}+H_{3,1}$. Consequently, $H_{i,2}$ are better PSNR/Capacity selection between $H_{i,j}(1\le j\le 3)$.

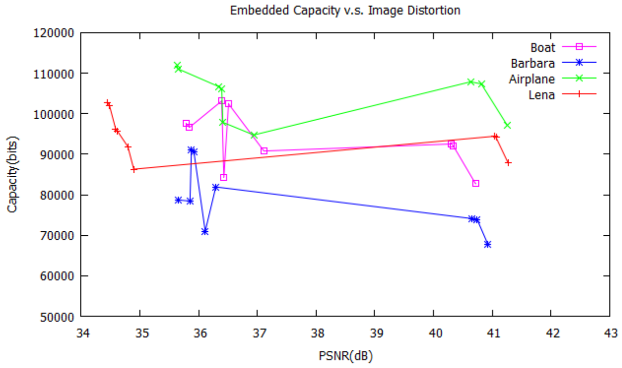


Figure 4. Capacity v.s. image distortion in four test images.

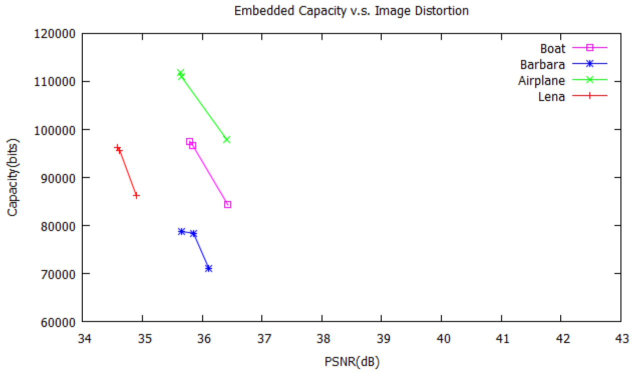


Figure 5. Capacity v.s. image distortion in four test images of embedding watermarks into $H_{i,1}$ subimages.

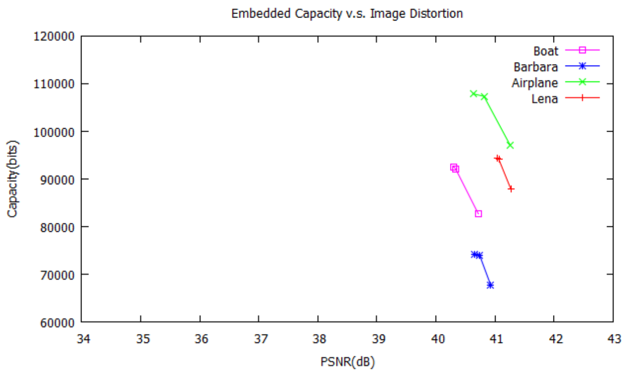


Figure 6. Capacity v.s. image distortion in four test images of embedding watermarks into $H_{i,2}$ subimages.

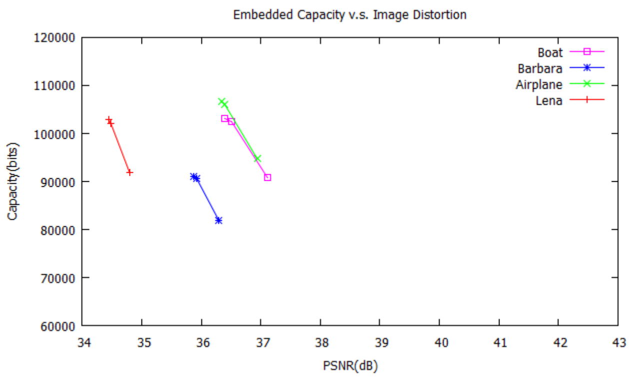


Figure 7. Capacity v.s. image distortion in four test images of embedding watermarks into $H_{i,3}$ subimages.

From above discussion, the proposed scheme has the following properties. First, the proposed scheme has high capacity and good image visual distortion of high PSNR value. Second, embedding into any subimage performs embedding into the whole image. At last, a good selection on embedding into subimages are $H_{i,2}$ between $H_{i,j}$(1≤j≤3).

V. CONCLUSIONS

This study presents an integer wavelet coefficients based reversible image watermarking approach. The proposed scheme allows users to assign different thresholds on subimage segmentation for increasing embedded quantity and preserving good image quality. Experimental results show that the proposed approach preserves high embedded capacity to 27K bits or high image quality to PSNR 33dB under different parameters selection. Experimental results also show that subimages $H_{i,2}$ are better PSNR/Capacity selection between $H_{i,j}$($1 \leq j \leq 3$). Comparisons with two literatures demonstrate that the proposed scheme has excellent embedded quantity v.s. image quality property. A trade-off search between parameters selection and the embedded quantity merits out future study.

REFERENCES

[1] A.M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. on Image Processing*, vol. 13, no.8, 1147–1156, 2004.

[2] M.U. Celik, G. Sharma, A.M. Tekalp, and E. Saber, "Lossless generalized-LSB data embedding," *IEEE Trans. on Image Processing*, vol. 14, no. 2, pp. 253–266, 2005.

[3] C.C. Chen and Y.H. Tsai, "Adaptive reversible image watermarking scheme", *The Journal of Systems and Software*, vol. 84, no.3, 428–434, 2011.

[4] I. Cox, M. Miller and J. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, 2002.

[5] W. Hong, T.S. Chen and C.W. Shiu, "Reversible Data Hiding for High Quality Images Using Modification of Prediction Errors", *The Journal of Systems and Software*, vol. 82, no.11, 1833–1842, 2009.

[6] L. Kamstra and H. Heijmans, "Reversible data embedding into images using wavelet techniques and sorting", *IEEE Trans. on Image Processing*, vol. 14, no.2, 2082–2090, 2005.

[7] C.C. Lee, H.C. Wu, C.S. Tsai and Y.P. Chu, "Adaptive Lossless Steganographic Scheme with Centralized Difference Expansion", *Pattern Recognition*, vol. 41, pp. 2097–2106, 2008.

[8] C.C. Lin, W.L. Tai and C.C. Chang, "Multilevel reversible data hiding based on histogram modification of difference images", *Pattern Recognition*, vol. 41, no.12, 3582–3591, 2008.

[9] Z. Ni, Y.Q. Shi, N. Ansari and W. Su, "Reversible Data Hiding", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 16, no.3, 354–362, 2006.

[10] D.M. Thodi and J.J. Rodríguez, "Expansion embedding techniques for reversible watermarking", *IEEE Trans. Image Processing*, vol. 16, no.3, 721–730, 2007.

[11] J. Tian, "Reversible Data Embedding Using a Difference Expansion," *IEEE Trans. on Circuits Systems and Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.

[12] P. Tsai, Y.C. Hu and H.L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting", Signal Processing, vol. 89, no.6, 1129–1143, 2009.

Chien-Chang Chen received the B.S. degree from Department of Computer and Information Science at Tung Hai University, Taichung, in 1991, and the Ph.D. degree in Computer Science from National Tsing Hua University in 1999 He is currently an Associate Professor at the Department of Computer Science and Information Engineering, Tamkang University. His research interests include secret image sharing, watermarking, and texture analysis.

Zeng-Chuang Jiun is pursuing the master degree from Department of Computer Science and Information Engineering, Tamkang University. His research interests include image watermarking and secret image sharing.

Chien-Hsiang Hsu received the master degree from Department of Information Management, Hsuan Chuang University in 2011. His research interests include image watermarking.