Journal of Marine Science and Technology

Volume 20 | Issue 1 Article 6

A COLOR IMAGE AUTHENTICATION AND RECOVERY METHOD USING BLOCK TRUNCATION CODE EMBEDDING

Shu-Chien Huang
Department of Computer Science, National Pingtung University of Education, Pingtung County, Taiwan, R.O.C., schuang@mail.npue.edu.tw

Ching-Fen Jiang
Department of Biomedical Engineering, I-Shou University, Kaohsiung City, Taiwan, R.O.C.

Recommended Citation

Huang, Shu-Chien and Jiang, Ching-Fen (2012) "A COLOR IMAGE AUTHENTICATION AND RECOVERY METHOD USING BLOCK TRUNCATION CODE EMBEDDING," *Journal of Marine Science and Technology*: Vol. 20: Iss. 1, Article 6.
DOI: 10.51400/2709-6998.2421
Available at: https://jmstt.ntou.edu.tw/journal/vol20/iss1/6

A COLOR IMAGE AUTHENTICATION AND RECOVERY METHOD USING BLOCK TRUNCATION CODE EMBEDDING

Shu-Chien Huang[1] and Ching-Fen Jiang[2]

Key words: image authentication, tamper recovery, fragile watermarking, block truncation code.

ABSTRACT

A new method for color image authentication and recovery is proposed in this paper. To achieve this goal, the least significant bit (LSB) and the second LSB of the spatial domain were selected to embed the watermark. The watermark consists of two parts: authentication data and recovery data made by block truncation code and bitmap. According to the experimental results, the image embedded within the authentication data and recovery data can still preserve high image quality. The results also reveal that the tampered images can be successfully recovered with acceptable visual quality.

I. INTRODUCTION

Due to the rapid advancement of computer technology, digital information such as images, audio, and video can be accurately copied and arbitrarily distributed much more easily. On the other hand, the availability of powerful image processing tools has also provided opportunities to manipulate and tamper with digital images for the misuse of intellectual property. Hence, protecting the content of digital images for image authentication is an urgent issue.

In general, robust image watermarking techniques are used to protect ownership of the digital image. In contrast, the purpose of fragile image watermarking techniques is image authentication, that is, to ensure the integrity of the digital image. Many image authentication methods through fragile watermarking have been proposed. These methods can be divided into two types. The first type [1-4, 6, 9, 13] can only determine whether or not the digital image has been tampered with, while the second type [5, 7, 8, 12, 14] not only detects the tampered areas, but also provides recovery ability.

Chen and Wang [6] proposed a fragile watermarking scheme for image authentication and tamper proofing. In their method, the original image is divided into non-overlapping 2×2 blocks, such that a block is regarded as a 4-dimensional vector. The fuzzy c-means clustering is then applied to classify all the blocks into C clusters. A membership matrix U is obtained. For the j-th block, the feature can be generated by $f_j = \lfloor (u_{1j} - u_{cj}) \times 255 \rfloor$, where u_{1j} and u_{cj} represent the maximum and minimum values of the j-th column in U. The authentication data are constructed by $a_j = f_j$ XOR rs_j, where rs_j ($rs_j \in [0, 255]$) is the j-th number of a random sequence created using a pseudorandom number generator seeded with a secret key. The resultant authentication data are embedded into the 8 least significant bits (LSBs) of the corresponding image block. Their method provides accurate tamper detection and localization accuracy, but does not have the ability to recover the tampered region.

Lin *et al.* [8] presented a digital watermarking method for image tamper detection and recovery. The original image is divided into non-overlapping 4×4 blocks, and each block is further divided into four sub-blocks of 2×2 pixels. In their method, the detection of the tampered region is based on a 3-level hierarchical structure. That is, if a tampered block is not detected in level-1 inspection, it will be detected in level-2 or level-3 inspection with a probability of nearly 1. The watermark in each sub-block is a 3-tuple (v, p, r), where v and p are 1-bit authentication watermarks, and r is a 6-bit recovery watermark. The authentication watermarks (v, p) and the recovery watermark r are embedded into the two LSBs of each pixel within the current sub-block and corresponding sub-block of another block, respectively.

In this paper, we propose a new fragile watermarking scheme that is able to detect and recover the tampered regions for color images. In Section II, block truncation coding for color images is briefly reviewed. In Section III, the proposed method is presented. Then, the experimental results and comparison are shown in Section IV, and a conclusion is provided in Section V.

Paper submitted 10/15/09; revised 07/08/10; accepted 08/24/10. Author for correspondence: Shu-Chien Huang (e-mail: schuang@mail.npue.edu.tw).

[1] *Department of Computer Science, National Pingtung University of Education, Pingtung County, Taiwan, R.O.C.*

[2] *Department of Biomedical Engineering, I-Shou University, Kaohsiung City, Taiwan, R.O.C.*

II. BLOCK TRUNCATION CODING

Block truncation coding (BTC) is a well-known image compression technique. In BTC, a grayscale image is divided into 4 × 4 or 8 × 8 nonoverlapping blocks of pixels, and each lock independently needs a two-level quantizer. For each block with size 4 × 4, the mean value *mv* can be calculated and defined as follows:

$$mv = \frac{1}{4\times 4}\sum_{i=1}^{4}\sum_{j=1}^{4} x_{ij}, \tag{1}$$

where x_{ij} indicates the pixel value in the position (i, j) of the block. Then, all pixels within the block are separated into two groups, greater and smaller than or equal to the mean value *mv*, denoted as G_1 and G_0, respectively. A binary bitmap *BM* with the same size as the image block is used to record the output bits of BTC compression. The bit in *BM* is set to 1 if the corresponding pixel value of the image block is greater than *mv* and classified into group G_1; otherwise, it is set to 0. The bitmap is generated using the following rule:

$$BM_{ij} = \begin{cases} 1 & \text{if } x_{ij} > mv, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where BM_{ij} represents the bit in position (i, j) of *BM*.

Next, two mean values of G_1 and G_0, denoted as X_H and X_L, are set as the quantization levels used to reconstruct the image. Therefore, a grayscale image block is decomposed to one binary bitmap *BM* and two quantization levels, X_H and X_L.

In the decoding procedure, the approximate image block can be reconstructed according to the binary bitmap *BM* and two quantization levels X_H and X_L, and the decoded image can be obtained by collecting all the reconstructed image blocks. The reconstruction rule is defined as follows:

$$\tilde{x}_{ij} = \begin{cases} X_H & \text{if } BM_{ij} = 1, \\ X_L & \text{otherwise,} \end{cases} \tag{3}$$

where $\tilde{x}_{ij}$ denotes the reconstructed pixel value in position (i, j) of the current decoded block.

As an example, a block of the image is shown in Fig. 1(a). The average of this block is 116 and the corresponding bitmap for this block is shown in Fig. 1(b). In this example, the value of X_H is 138 and X_L is 88. The reconstructed image block is shown in Fig. 1(c).

The BTC-compression method can also be applied directly to color images. The given color image is divided into a set of blocks. With the BTC-compression method, each block is encoded as a result of three bitmaps and three pairs of quantization levels, denoted as R_H and R_L, G_H and G_L, and B_H and B_L, for R, G, and B, respectively. In Ref. [10], the genetic algorithm is applied to find an approximate optimal common bitmap to replace the original three bitmaps in order to reduce

139	92	84	78
145	120	87	85
145	137	95	94
142	139	140	130

(a) Original image block

1	0	0	0
1	1	0	0
1	1	0	0
1	1	1	1

(b) Bitmap

138	88	88	88
138	138	88	88
138	138	88	88
138	138	138	138

(c) Reconstructed image block

Fig. 1. An example of BTC encoding and decoding.

the number of bitmaps. An initial population of 12 chromosomes is created in Ref. [10]. A chromosome corresponds to a bitmap. For a chromosome corresponding to bitmap *BP*, the fitness value of the chromosome is defined as

$$\begin{aligned} fn &= \frac{1}{MSB_{BP}}, \\ MSB_{BP} &= \frac{1}{m}\Big(\sum_{BP_{ij}=0}\left|x_{ij}-C_1\right|^2 + \sum_{BP_{ij}=1}\left|x_{ij}-C_2\right|^2\Big) \end{aligned} \tag{4}$$

where $x_{ij} = (r_{ij}, g_{ij}, b_{ij})$ denotes the pixel values in the position (i, j) of the block, and the two reconstruction levels C_1, C_2 can be expressed by

$$\begin{aligned} C_1 &= \frac{1}{q}\sum_{BP_{ij}=0} x_{ij}, \\ C_2 &= \frac{1}{m-q}\sum_{BP_{ij}=1} x_{ij}, \end{aligned} \tag{5}$$

where q denotes the number of 0's in the bitmap *BP* and m denotes the block size.

After the genetic operators (reproduction, crossover, and mutation) are performed in the current population, a new generation is generated. If the generation number is greater than a threshold value *th* (for example *th* = 20), the best chromosome is outputted and the corresponding bitmap *BM* is obtained.

In the decoding procedure, the approximate image block can be reconstructed according to the binary bitmap *BM* and

two quantization levels C_1 and C_2, and the decoded image can be obtained by collecting all the reconstructed image blocks. The reconstruction rule is defined as follows:

$$\tilde{x}_{ij} = \begin{cases} C_1 & \text{if } BM_{ij} = 0, \\ C_2 & \text{otherwise,} \end{cases} \tag{6}$$

where $\tilde{x}_{ij}$ denotes the reconstructed pixel value in position (i, j) of the current decoded block.

III. THE PROPOSED METHOD

Suppose that image H is a 24-bit RGB-color image of size $M \times M$ pixels, where M is assumed to be a multiple of 4. Divide H into non-overlapping 4×4 blocks B_{ij} ($1 \le i, j \le M/4$). The first procedure embeds the authentication data and recovery data into the color image to obtain the watermarked image. The second procedure consists of the tamper detection and recovery process. The details of the proposed method are described as follows.

1. Authentication Data Generation

The following steps describe how the authentication data of block B_{ij} are generated.

Step 1: Using only the six most significant bits (MSBs) of all pixels in the block to compute the variance values VA_r, VA_g, and VA_b for the Red, Green, and Blue components, respectively, the VC_r, VC_g, and VC_b are generated by the following equation:

$$\begin{aligned} VC_r &= (\lfloor VA_r \rfloor \bmod 2^9), \\ VC_g &= (\lfloor VA_g \rfloor \bmod 2^9), \\ VC_b &= (\lfloor VA_b \rfloor \bmod 2^9). \end{aligned} \tag{7}$$

The values VC_r, VC_g, and VC_b are between 0 and 511, and it takes 9 bits to store each value.

Step 2: The 5-bit authentication watermark PN is generated as in Eq. (8) below:

$$PN = (i + j) \bmod 2^5. \tag{8}$$

The authentication data of each block are the 4-tuple-watermark (VC_r, VC_g, VC_b, PN). It is obvious that 32 bits are required to store the authentication data.

2. Recovery Data Generation

The single bitmap BTC coding of color images is described in Section II. For a block, it will take 16 bits to store the bitmap BM, and 48 bits to store the two colors C_1 and C_2. Therefore, 64 bits are required to store the recovery data.

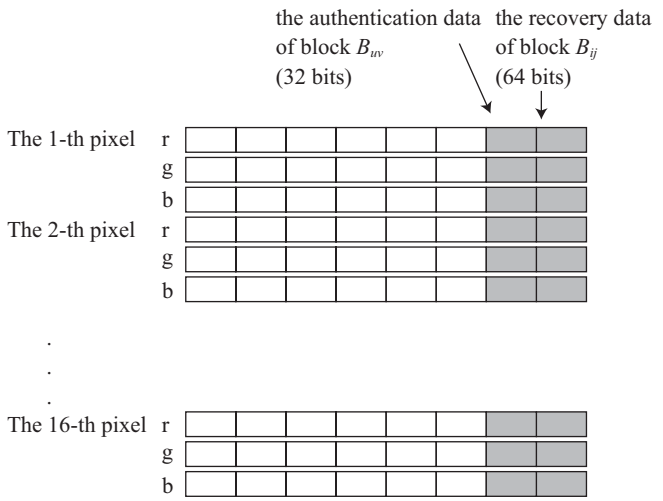


Fig. 2. The authentication data of block B_{uv} and the recovery data of block B_{ij} are embedded into the two LSBs of each pixel within the block B_{uv}.

3. Watermarked Image Generation

In the proposed method, the authentication data of block B_{uv} and the recovery data of block B_{ij} are embedded into the two LSBs of each pixel within the block B_{uv}, as shown in Fig. 2. The authentication and recovery data are embedded in the original image to obtain the watermarked image.

The block-mapping sequence $A \to B \to C \to D \to \cdots \to A$ is used for the recovery data embedding, in which each symbol denotes an individual block. The symbols $A \to B$ indicate that the data for recovering block A will be embedded into block B. To generate the block-mapping sequence, the following 2-D transformation [11] is performed on each block to obtain the one-to-one mapping sequence.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ k & k+1 \end{bmatrix} \times \begin{bmatrix} i \\ j \end{bmatrix} \bmod \frac{M}{4} \tag{9}$$

where B_{ij} is the original block, B_{uv} is the transformed block and $k \in [0, \frac{M}{4} - 1]$.

4. Tamper Detection

The test image is first divided into non-overlapping 4×4 blocks B_{ij} ($1 \le i, j \le M/4$). Perform the following steps to examine whether the block B_{ij} is valid or not.

Step 1: Extract the two LSBs from the two LSBs of each pixel within the block B_{ij}, and obtain the authentication data (VC_r, VC_g, VC_b, PN).

Step 2: Set the two LSBs of each pixel within block B_{ij} to zero and compute the variance value for the R, G, and B components. The three obtained values are denoted by VA_r, VA_g, VA_b, respectively.

Step 3: If (VC_r == ($\lfloor VA_r \rfloor \bmod 2^9$)) and ($VC_g$ == ($\lfloor VA_g \rfloor \bmod$

2^9)) and (VC_b == ($\lfloor VA_b \rfloor$ mod 2^9)) and (PN == (i + j) mod 2^5), mark the block valid; otherwise mark the block invalid and perform the image recovery operation.

5. Tampered Image Recovery

For each block B_{ij} which has been identified as an invalid block, perform the following steps for recovery:

Step 1: Use Eq. (9) and the secret key k to locate block B_{uv}.
Step 2: If the block B_{uv} is valid, go to step 3; otherwise go to step 5.
Step 3: Extract the recovery data from block B_{uv} and obtain the bitmap and two colors C_1 and C_2.
Step 4: Replace the color values of each pixel within block B_{ij}. If the corresponding bitmap value of the pixel is 0, then the pixel is replaced by C_1; otherwise, it is replaced by C_2. Go to step 1 to recover the next tampered block.
Step 5: Compute the average color of valid 8-neighboring blocks of block B_{ij}.
Step 6: Replace the color values of each pixel within block B_{ij} with the average color computed in step 5. Go to step 1 to recover the next tampered block.

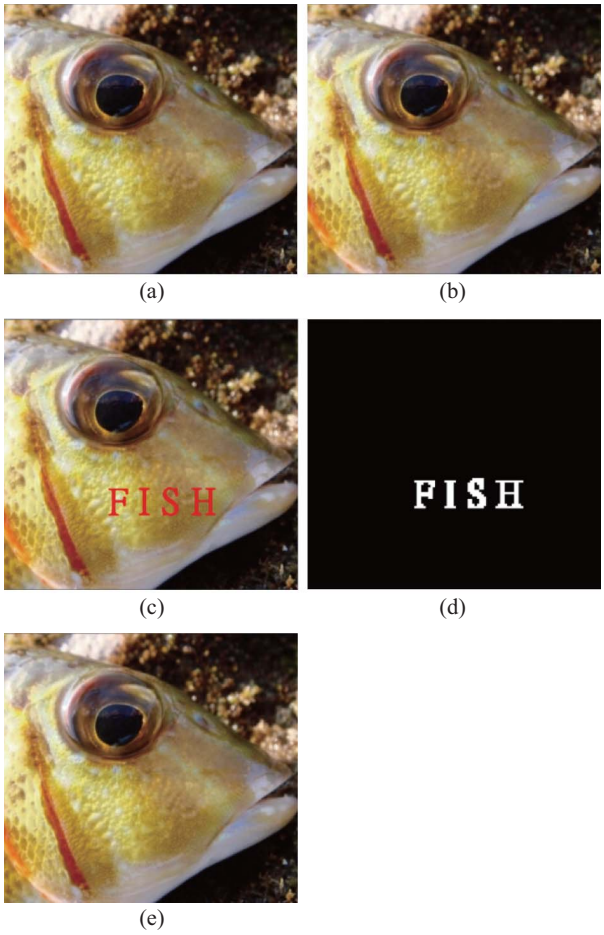
IV. EXPERIMENTAL RESULTS

The proposed algorithm is implemented in *C* language on a Pentium IV PC using the operating system Microsoft Windows XP. The color system is the RGB model. Let the range for valid R, G, B values be [0, 255]. Four 512 × 512 color images, "Fish", "Airplane", "Car", and "Boat", are used to measure the performance.

First, the color image "Fish", shown in Fig. 3(a), is used for testing. The watermarked image is shown in Fig. 3(b). Fig. 3(c) shows the modified image where the tampering is produced by PhotoShop. The detection result of the tampered image is shown in Fig. 3(d). It is observed that the tampered area is almost correctly located. The recovered image is shown in Fig. 3(e). It is seen that the recovered image is visually indistinguishable from the "Fish" image.

Second, the color image "Airplane", shown in Fig. 4(a), is used for testing. The watermarked image is shown in Fig. 4(b). To forge a tampered image of the watermarked image, the sky part in the watermarked image is inserted into the region of one airplane. The tampered image is shown in Fig. 4(c). The tamper detection result and the recovered image are shown in Figs. 4(d) and 4(e), respectively. The experiment reveals that the ability of the proposed method to detect tampering is adequate.

Third, the color image "Car", shown in Fig. 5(a), is used for testing. The watermarked image is shown in Fig. 5(b). Figure 5(c) shows that the important part of the car number plate within Fig. 5(b) is changed. The detection result of the tampered image is shown in Fig. 5(d). The tampered area is correctly identified by the proposed method. The recovered image is shown in Fig. 5(e). The plate number LR-4639 can be easily recognized in the recovered "Car" image.



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

Fig. 3. (a) The "Fish" image, (b) Watermarked image, (c) Tampered image, (d) Tamper detection result of (c), and (e) Recovered image.

Fourth, the color image "Boat", shown in Fig. 6(a), is used for testing. The watermarked image is shown in Fig. 6(b). Fig. 6(c) shows the modified image in which the attack forges an image by copying image blocks from one watermarked image and inserting them into arbitrary positions in the same watermarked image. The tamper detection result and the recovered image are shown in Figs. 6(d) and (e), respectively.

For quantitative evaluation, two measurements, tamper detection rate and peak signal-to-noise ratio (PSNR), were introduced to evaluate the performance of the proposed method. The tamper detection rate TDR is defined as

$$\mathrm{TDR} = \frac{\mathrm{num}_1}{\mathrm{num}_2}, \tag{10}$$

where num_2 is the number of actually altered blocks, and num_1 is the number of altered blocks which are correctly identified. The PSNR is employed to measure the image quality. The PSNR of image H relative to image H' is defined as

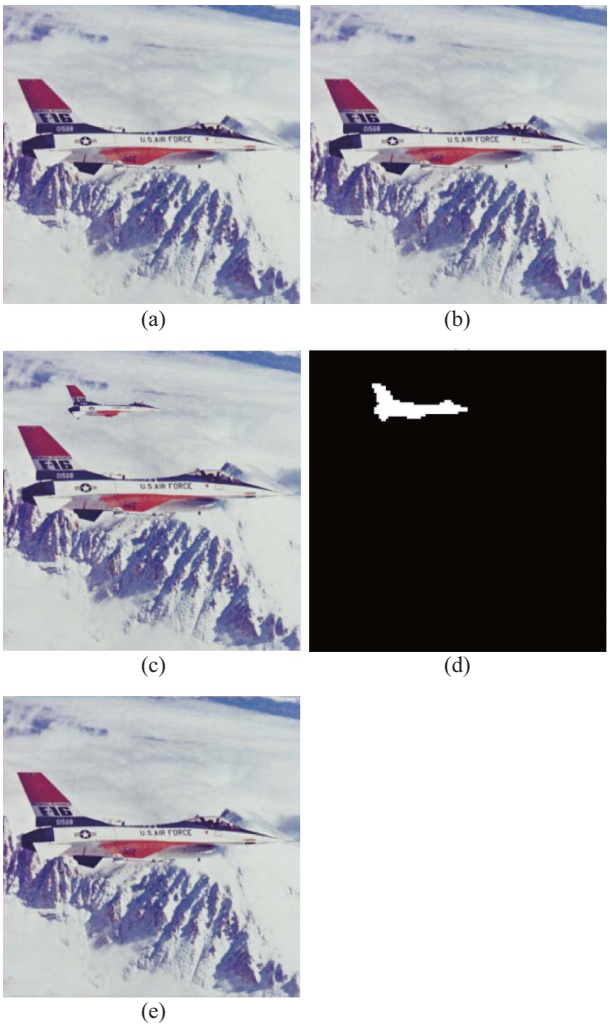


Fig. 4. (a) The "Airplane" image, (b) The watermarked image, (c) Tampered image, (d) Tamper detection result of (c), and (e) Recovered image.

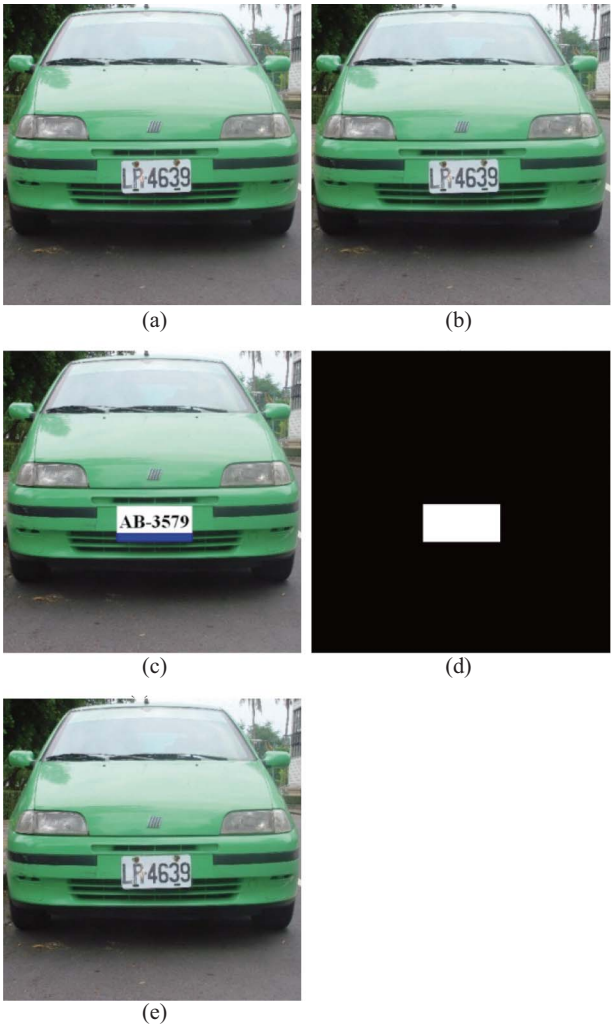


Fig. 5. (a) The "Car" image, (b) The watermarked image, (c) Tampered image, (d) Tamper detection result of (c), and (e) Recovered image.

$$\begin{aligned} \mathrm{PSNR} &= 10\times\log_{10}\frac{255^2}{MSE}, \\ MSE &= \frac{1}{3\times M\times M}\sum_{c\in\{r,g,b\}}\sum_{i=1}^{M}\sum_{j=1}^{M}(H_{i,j}(c)-H'_{i,j}(c))^2, \end{aligned} \tag{11}$$

where r, g and b represent the primary colors red, green, and blue, respectively.

The tamper detection rates of the proposed method and Lin *et al.*'s method [8] are summarized in Table 1. It can be found that the tampered blocks in the tampered "Fish", "Airplane", and "Car" images were completely identified by the proposed method and Lin *et al.*'s method. However, the evaluation results show that Lin *et al.*'s method achieved a low detection rate, 83.8%, for the tampered "Boat" image. Since Lin *et al.*'s method verified the legitimacy of each image block individually, many tampered blocks copied from the same watermarked image in the tampered "Boat" image were misidentified as authentic. On the contrary, the tampered blocks in the tampered "Boat" image were completely identified by the proposed method. If the image is not tampered with, the tamper detection result shows that the number of tampered blocks is zero in our method; however, the recovery data is redundant because no block is needed to be recovered.

Table 2 lists the PSNR of the watermarked image relative to the original image. The quality of the watermarked image is around 44 dB for the proposed method and Lin *et al.*'s method [8]. That is to say, the distortion is imperceptible to the human eye. Table 3 lists the PSNR of the recovered image relative to the watermarked image. The PSNR of a recovered image depends on how well the tampered blocks are recovered. The PSNR values of the recovered "Fish", "Airplane", and "Car" images obtained by the proposed method are about 4 to 7 dB higher than those values obtained by Lin *et al.*'s method. The reason is that the block truncation code is used to recover the tampered block in our method, therefore achieving superior

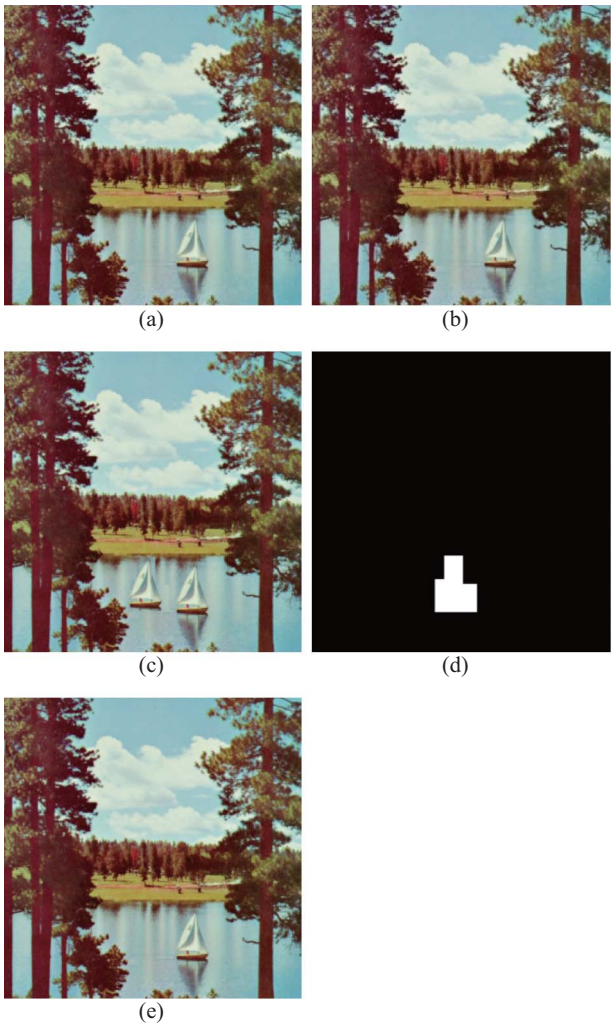


Fig. 6. (a) The "Boat" image, (b) The watermarked image, (c) Tampered image, (d) Tamper detection result of (c), and (e) Recovered image.

recovery ability. Moreover, the PSNR value of the recovered "Boat" image acquired by the proposed method is much higher than the value acquired by Lin *et al.*'s method, since the tampered blocks in the tampered "Boat" image were completely identified by the proposed method, while many tampered blocks in the tampered image were misidentified by Lin *et al.*'s method. Among all the recovered images obtained by the proposed method, the recovered "Car" image has the worst image quality of 42.06 due to the noisy car number plate.

V. CONCLUSION

A new method is proposed for color image tamper detection and tampered image recovery. The proposed scheme embeds a watermark which consists of the authentication data and the recovery data, into each image block. If the watermarked image has been tampered with, the tampered region can be automatically localized, and the recovery data are extracted from another block for recovery by use of the block truncation decoding procedure.

Table 1. Tamper detection rates.

Image	Proposed method			Lin *et al.*'s method		
	num_1	num_2	TDR	num_1	num_2	TDR
Fish	239	239	100%	239	239	100%
Airplane	260	260	100%	260	260	100%
Car	528	528	100%	528	528	100%
Boat	320	320	100%	268	320	83.8%

Table 2. PSNR values of watermarked images relative to original images.

Image	Proposed method	Lin *et al.*'s method
Fish	43.92	43.98
Airplane	44.09	44.34
Car	44.12	44.17
Boat	44.13	44.22

Table 3. PSNR values of recovered images relative to watermarked images.

Image	Proposed method	Lin *et al.*'s method
Fish	53.99	47.15
Airplane	57.83	53.19
Car	42.06	35.65
Boat	49.55	38.17

The experimental results show that the PSNR values of the recovered images obtained by the proposed method are higher than those values obtained by Lin *et al.*'s method. It is seen that the recovered image is visually indistinguishable from the original image. These experiments reveal the superior recovery ability of the proposed method. In conclusion, the proposed scheme is simple and effective for color image authentication and recovery.

REFERENCES

1. Celik, M. U., Sharma, G., Saber, E., and Tekalp, A. M., "Hierarchical watermarking for secure image authentication with localization," *IEEE Transactions on Image Processing*, Vol. 11, No. 6, pp. 585-594 (2002).
2. Chan, C. S. and Chang, C. C., "An efficient image authentication method based on Hamming code," *Pattern Recognition*, Vol. 40, No. 2, pp. 681-690 (2007).
3. Chang, C. C., Hu, Y. S., and Lu, T. C., "A watermarking-based image ownership and tampering authentication scheme," *Pattern Recognition Letters*, Vol. 27, No. 5, pp. 439-446 (2006).
4. Chang, C. C. and Lin, P. Y., "A color image authentication method using partitioned palette and morphological operations," *IEICE Transactions on Information and Systems*, Vol. E91-D, No. 1, pp. 54-61 (2008).
5. Chang, Y. J., Lin, S. J., and Lin, J. C., "Authentication and cross-recovery for multiple images," *Journal of Electronic Imaging*, Vol. 17, No. 4, pp. 043007-1~043007-12 (2008).
6. Chen, W. C. and Wang, M. S., "A fuzzy c-means clustering-based fragile watermarking scheme for image authentication," *Expert Systems with Applications*, Vol. 36, No. 2, pp. 1300-1307 (2009).
7. Lee, T. Y. and Lin, S. D., "Dual watermark for image tamper detection and

recovery," *Pattern Recognition*, Vol. 41, No. 11, pp. 3497-3506 (2008).
8. Lin, P. L., Hsieh, C. K., and Huang, P. W., "A hierarchical digital watermarking method for image tamper detection and recovery," *Pattern Recognition*, Vol. 38, No. 12, pp. 2519-2529 (2005).
9. Queluz, M. P., "Authentication of digital images and video: generic models and a new contribution," *Signal Processing: Image Communication*, Vol. 16, No. 5, pp. 461-475 (2001).
10. Tai, S. C., Chen, W. J., and Cheng, P. J., "Genetic algorithm for single bitmap absolute moment block truncation coding of color images," *Optical Engineering*, Vol. 37, No. 9, pp. 2483-2490 (1998).
11. Voyatzis, G. and Pitas, I., "Chaotic mixing of digital images and applications to watermarking," *Proceeding of the European Conference on Multimedia Applications Services and Techniques*, Vol. 2, pp. 687-689 (1996).
12. Wu, H. K., Chang, R. F., Chen, C. J., Wang, C. L., Kuo, T. H., Moon, W. K., and Chen, D. R., "Tamper detection and recovery for medical images using near-lossless information hiding technique," *Journal of Digital Imaging*, Vol. 21, No. 1, pp. 59-76 (2008).
13. Wu, N. I., Wang, C. M., Tsai, C. S., and Hwang, M. S., "A certificate-based watermarking scheme for coloured images," *The Imaging Science Journal*, Vol. 56, No. 6, pp. 326-332 (2008).
14. Yang, C. W. and Shen, J. J., "Recover the tampered image based on VQ indexing," *Signal Processing*, Vol. 90, No. 1, pp. 331-343 (2009).