http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 10 (3): 681-685, 2011
ISSN 1812-5638 / DOI: 10.3923/itj.2011.681.685

Joint Secret Sharing and Data Hiding for Block Truncation Coding Compressed Image Transmission

[1]Hao Luo, [2]Zhenfei Zhao and [1]Zhe-Ming Lu
[1]School of Aeronautics and Astronautics, Zhejiang University, Hangzhou 310027, China
[2]School of Information Science and Technology, Sun Yat-sen University, GuangZhou 510006, China

Abstract: This study proposed a scheme that incorporates secret sharing and data hiding techniques for block truncation coding compressed image transmission. The bitmap of each compressed block is encrypted and meanwhile two quantization levels are hidden in two share images. The secure transmission system still preserves the properties such as low complexity and acceptable reconstruction image quality of the standard block truncation coding compression. In addition, each share image is half size of the compressed version such that no extra burden is laid on available transmission resources. Experimental results demonstrate the effectiveness of our scheme.

Key words: Block truncation coding, secret sharing, data hiding

INTRODUCTION

In recent years, the security of data storage and transmission has drawn much attention among researchers (Ruan *et al*., 2010). Block Truncation Coding (BTC) (Amarunnishad *et al*., 2008; Guo *et al*., 2009) is a popular lousy image compression method due to low computational complexity and storage demands. It never requires any auxiliary information during the encoding and decoding procedures. However, transmission security is also a problem must be considered in more and more situations. Besides traditional data encryption algorithms (e.g., DES), secret sharing (Ahlswede and Csiszar, 1993; Luo *et al*., 2010; Zhao *et al*., 2010) and data hiding (Pan *et al*., 2007) are another two powerful complementary techniques and also play important roles in information security.

The block truncation coding compression is based on a block-wise thresholding operation. Hence, the task of block truncation coding compressed image transmission is reduced to transmitting binary matrices and thresholds of image blocks. The compression ratio is determined by the partitioned block size. That is, a larger block size partition results to fewer binary matrices and thresholds to be transmitted. However, the compressed image quality is worse than that obtained by a smaller block size partition. Consequently, a good tradeoff is usually achieved between the compression ratio and the reconstructed image quality.

In this study, we aim to develop a secure transmission system for block truncation coding compressed images. The application scenario can be described as follows. Given a gray-level or color image, we need to compress and transmit it over two independent but not secure channels. Besides transmission security, the following three factors must be considered in the system design. (1) The reconstruction image quality should be preserved, i.e., nearly the same as that in block truncation coding compression. (2) Low complexity encoding and decoding, thus the real-time performance of block truncation coding compression is maintained. (3) Small size of shares. This is essential to save transmission time and channel resources. In fact, in most available secret sharing methods, the sizes of shares are expanded compared to the original secret data.

In particular, these three advantages are achieved in the proposed system with secret sharing and data hiding well synthesized. The compressed image can be reconstructed as long as both shares are collected. Otherwise, no meaningful information is decrypted. Besides, the encoding/decoding procedures are quite simple and the reconstructed image quality is nearly the same as that in block truncation coding compression. Furthermore, each share is half size of the compressed image and thus no extra burden is laid on transmission channels.

The standard BTC compression technique is reviewed below. Suppose the original image I is a

Corresponding Author: Hao Luo, School of Aeronautics and Astronautics, Yuquan Campus, Zhejiang University, No. 38 ZheDa Road, Hangzhou 310027, China
Tel: +86-571-87951581-814 Fax: +86-571-87951581-815

gray-level image. In the encoding stage, I is first partitioned into a set of non-overlapping k×k blocks and the mean value p_m of each block is calculated as:

$$p_m = \frac{1}{k \times k} \sum_{x=1}^{k} \sum_{y=1}^{k} p(x,y) \quad (1)$$

where, p(x, y) denotes the pixel value in the position (x, y). Next, the block pixels are thresholded as:

$$b(x,y) = \begin{cases} 1 & \text{if } p(x,y) > p_m \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

resulting in a binary bitmap M, where b(x, y) is the pixel value in the position (x, *y*) of M. Third, two quantization levels p_h and p_l are computed for each block, i.e., the mean values of those pixels with $p(x, y) > p_m$ and $p(x, y) = p_m$ respectively. Thus, each block is represented by a pair of integers p_h, p_l in the range of (0, 255) and a k×k bitmap. The encoding procedure is completed and the compressed content is transmitted.

In the decoding stage, each block can be approximately recovered with p_h, p_l and M as:

$$p'(x,y) = \begin{cases} p_h & \text{if } b(x,y) = 1 \\ p_l & \text{otherwise} \end{cases} \quad (3)$$

where, p'(x, y) is the reconstructed pixel value in the position (x, y) of the current decoded block. In this way, the decoded image can be reconstructed by collecting all the reconstructed blocks. In our context, p_m, p_h and p_l are rounded to their nearest integers.

PROPOSED SCHEME

Our scheme is conducted as a part of the project work on satellite image transmission from Feb, 2009. The design and test duration is completed till Sep., 2010.

Joint data encryption and data hiding model: We start from presenting the joint data encryption and data hiding model. Suppose the secret data is composed by two binary sequences $u = (u_1, u_2, \ldots u_K)$ and $v = (v_1, v_2, \ldots v_{K-1})$. The main idea of our model is to encrypt u and meanwhile *v* is also hidden in two shares $m=(m_1, m_2, \ldots m_K)$ and $n=(n_1, n_2, \ldots n_K)$. The encryption strategy of this model is shown in Fig. 1.

The encoding procedures are described below:

Step 1: Encrypt u_1 in m_1 and n_1. Randomly select a "1" or "0" and assign it to m_1 and then n_1 is determined by m_1 and u_1 as:

$$n_1 = \begin{cases} m_1 & \text{if } u_1 = 1 \\ 1 - m_1 & \text{if } u_1 = 0 \end{cases} \quad (4)$$

Step 2: Hide v_1 in n_1 and m_2. That is, n_1 and v_1 as determine m_2:

$$m_2 = \begin{cases} n_1 & \text{if } v_1 = 1 \\ 1 - n_1 & \text{if } v_1 = 0 \end{cases} \quad (5)$$

Step 3: Encrypt u_2 in m_2 and n_2. That is, m_2 and u_2 as determine n_2:

$$u_2 = \begin{cases} m_2 & \text{if } u_2 = 1 \\ 1 - m_2 & \text{if } u_2 = 0 \end{cases} \quad (6)$$

Step 4: Repeat the above operations until v_{j-1} is hidden in n_{j-1} and m_j as indicated with the dash arrow and u_j is encrypted in m_j and n_j as indicated with the solid arrow.

In this way, the sequence u is encrypted and v is hidden in two produced shares m and n.

In decoding, the sequences u and v can be recovered as:

$$\mu_i = 1 - m_i \oplus n_i \quad 1 \le I \le K \quad (7)$$

$$v_i = 1 - m_{i+1} \oplus n_i \quad 1 \le i \le K-1 \quad (8)$$

where denotes the exclusive-OR operation.

Joint block truncation coding and secret sharing: In the joint BTC and secret sharing scheme, the standard BTC compression and the joint data encryption and data hiding model are applied. In our case, the original image is partitioned into 4×4 blocks.

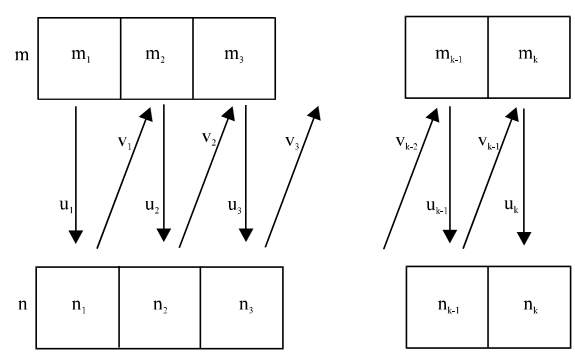

Fig. 1: Encryption strategy of the joint data encryption and data hiding model

In the encoding stage, operations are given below:

- **Step 1:** Perform the standard BTC encoding on the first block and its p_m, p_h and p_l are obtained
- **Step 2:** Compute the difference d between p_m and p_l as:

$$d = p_m - p_l \tag{9}$$

Considering in natural images, pixel values change gradually in a 4×4 block, the difference d is small for most natural image blocks. Therefore, it is reasonable to regard d as an integer belongs to (0, 127), otherwise we set d = 127.

- **Step 3:** Translate p_l and d into 8-bit and 7-bit binary sequences, respectively. Concatenate them to form a 15-bit sequence denoted by $v=(v_1, v_2,\ldots, v_{15})$. Besides, rearrange the bitmap M into a 16-bit binary sequence, represented with $u=(u_1, u_2,\ldots, u_{16})$.
- **Step 4:** Apply the encoding procedure (Eq. 4-6) of the joint data encryption and data hiding model to the block. Thus u is encrypted and at the same time v is hidden in two produced shares $m=(m_1, m_2,\ldots, m_{16})$ and $n = (n_1, n_2,\ldots, n_{16})$.
- **Step 5:** For the first share, translate $(m_1, m_2,\ldots, m_8)$ and $(m_9, m_{10},\ldots, m_{16})$ into two integers in the range of (0, 255), respectively. The similar operations are also performed on n for the second share. In this way, the compressed information of each block corresponds to two shares, each with two "pixels".
- **Step 6:** Repeat the above operations for all the other blocks and the share images S_1 and S_2 are obtained and transmitted independently.

The corresponding operations in the decoding stage are described as follows:

- **Step 1:** Translate the received first two pixels of S_1 and S_2 into two 16-bit binary sequences, respectively.
- **Step 2:** Apply the decoding procedure (Eq. 7-8) of the joint data encryption and data hiding model to these two sequences. Thus p_l, d and the associated bitmap M are recovered. Suppose the number of "0"s and "1"s in M are n_l and n_h=16-n_l, respectively.
- **Step 3:** Compute p_m according to p_l and d as:

$$p_m = p_l + d \tag{10}$$

- **Step 4:** Compute p_h according to p_l and p_m as:

$$p_h = \frac{(16p_m - n_l p_l)}{n_h} \tag{11}$$

- **Step 5:** Reconstruct the BTC-compressed block based on p_l, p_m and M according to Eq. 3.
- **Step 6:** Repeat Steps 1-5 for all the other received shares' pixels and thus the BTC-compressed image is reconstructed.

EXPERIMENTAL RESULTS

The 512×512 gray-level Lena image is selected as the test image and the related experimental results are shown in Fig. 2. Obviously, the two 128×256 share images produced in our scheme look like random noises, thus transmission security of the compressed image can be

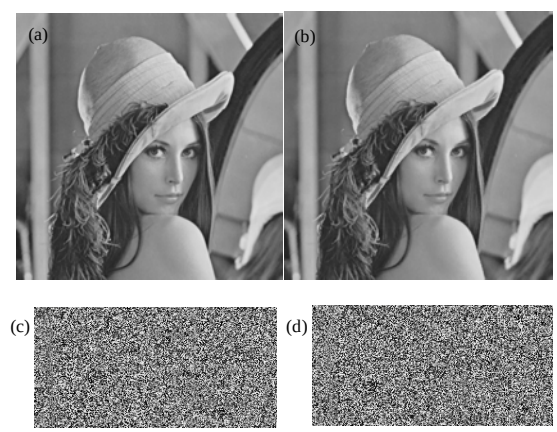

Fig. 2: Experimental results on Lena image, (a) the original image, (b) the constructed image by our scheme, (c) the share image S_1, (d) the share image S_2

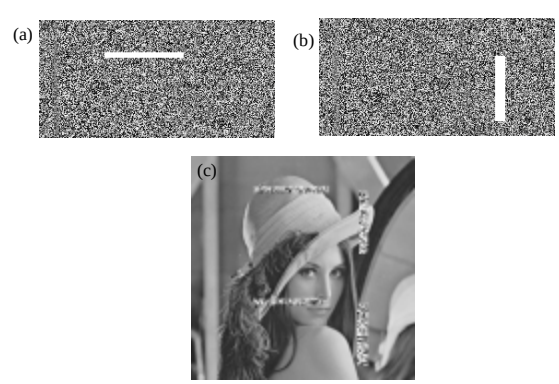

Fig. 3: Experimental results on the received shares suffering transmission errors, (a) the corrupted S_1, (b) the corrupted S_2 (c) the reconstructed image

guaranteed. Furthermore, the total amount of the transmission content is the same as that in the standard BTC compression, i.e., the compression ratio is kept as 4.

Many transmission scenarios, especially wireless transmissions usually suffer some abrupt errors such as packet loss (Lee *et al*., 2007). As our scheme is based on a block-wise processing mechanism, these transmission errors of share images only corrupt the associated decrypted blocks. In other words, errors are not diffused to other reconstructed blocks, as illustrated in another experiment shown in Fig. 3. Figure 3a and b are the corrupted versions of Fig. 2c and d respectively. From Fig. 3c, it is easy to find that only the corresponding blocks are corrupted, while others still reconstructed correctly.

It is necessary to note that, compared with the reconstructed image based on the standard block truncation coding; there is a slight quality degradation of the reconstructed image with the two intact shares. This is due to the computed p_h for image reconstruction in our scheme is not exactly equal to that directly transmitted in the standard BTC. Namely, error is introduced in Eq. 9-11 when a floating-point number is rounded to its nearest integer. A large quantity of experimental results shows that this image quality degradation is acceptable. For example, the PSNR of the reconstructed Lena using the standard BTC is 34.00 dB, while 33.96 dB obtained in our scheme.

As mentioned earlier d is set as 127 when it is larger than 127 such that it can be translated into 7-bit sequence. Thus truncation errors may be produced for the range of d is (0, 255) theoretically. Fortunately, as high correlation exists among pixels in natural image blocks, d is small in most cases. Ten 512×512 gray-level images (Lena, Baboon, Barbara, Bridge, Boat, F16, Peppers, Elaine, Aerial, Truck) are selected to investigate the statistics of d values. Each image is partitioned into 4×4 blocks, thus totally 163840 blocks, i.e., 163840 d values are obtained and examined. The 256-bin histogram of d values is shown in Fig. 4. The bins at the high end (bins from 150-255) are empty and hence only a part (150 out of 256) of the histogram is shown. In this experiment, only 44 d values are larger than 127. In addition, these large d values are in the range of (128, 150). That is, the percentage of the special case is merely about 0.027% and thus the possible errors are acceptable in practical.

In addition, there is another special case in our scheme, i.e., all the pixels in an original image block are of the same value. Suppose this value is denoted by ps. In this case, $p_m = p_s$, $p_l = p_s$, d = 0 and all bits in the bitmap M are 0. Obviously, Eq. 11 cannot be applied any longer for

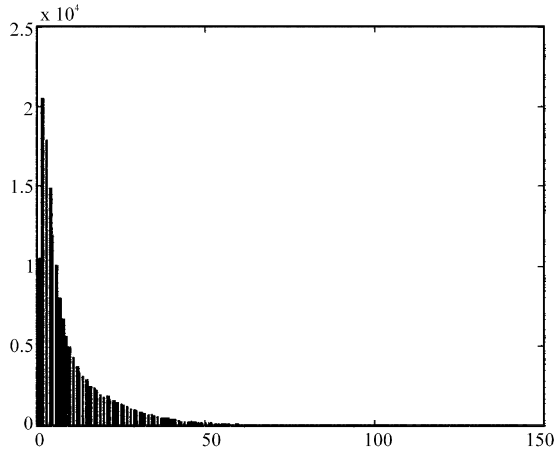

Fig. 4: Statistical histogram of values

both its denominator and numerator equal 0. Thus we set $p_h = p_s$ and remain all the other operations unchanged.

Actually, our scheme is also suitable for BTC-compressed color image transmission. Specifically, the original color image is decomposed into red, green and blue channels and each channel is encoded as a gray-level image. Finally all encoded channels are recomposed into color noise-like shares.

DISCUSSION

In Thien and Lin (2002), Thien and Lin develop a Lagrange interpolation based (r, t)-threshold scheme for image secret sharing. An input image can be encrypted into t random-noise like share images. If r or more than r shares are collected, the original image can be reconstructed. Thien and Lin's scheme is an alternative method to encrypt the BTC-compressed image with r = t =2. For example, a 512×512 input image can be encrypted into two 128×256 shares. However, this is achieved by truncating all compressed "pixel" values (e.g., 4 "pixels" translated from p_h, p_l and M of a block) from the range (0, 255) to (0, 250) before encryption. In other words, to perfectly reconstruct the BTC-compressed image, the truncation errors must be also encrypted additionally. Consequently, two produced shares are usually larger than 128×256 pixels, resulting in more storage space and transmission time. The same mechanism is also adopted in the subsequent work (Chen and Lin, 2005).

As a possible solution, the truncation errors also can be hidden along with the secret data. This can be referred to the side information hiding strategy proposed in several data hiding algorithms (Fridrich *et al*., 2002;

Pan *et al*., 2007; Guo *et al*., 2009) and secret sharing techniques (Luo *et al*., 2010; Zhao *et al*., 2010).

CONCLUSIONS

A joint secret sharing and data hiding system is proposed for BTC-compressed image secure transmission. The compressed content is encrypted and hidden in two share images. Since each share image is half size of the compressed content, the available channel resources are enough for share image transmission. Not only the standard BTC compression properties such as low encoding/decoding complexity and acceptable reconstructed image quality are still preserved, but also transmission security is guaranteed because each block is encrypted individually.

ACKNOWLEDGMENTS

This study is financially supported by National Scientific Fund of China, under the granted No. of 61003255.

REFERENCES

Ahlswede, R. and I. Csiszar, 1993. Common randomness in information theory and cryptography I: secret sharing. IEEE Trans. Inform. Theory, 39: 1121-1132.

Amarunnishad, T.M., V.K. Govindan and A.T. Mathew, 2008. Improving BTC image compression using a fuzzy complement edge operator. Signal Process., 88: 2989-2997.

Chen, S.K. and J.C. Lin, 2005. Fault-tolerant and progressive transmission of images. Pattern Recognit., 38: 2466-2471.

Fridrich, J., M. Goljan and R. Du, 2002. Lossless data embedding-new paradigm in digital watermarking. EURASIP J. Applied Signal Process., 2002: 185-196.

Guo, J.M., M.F. Wu and Y.C. Kang, 2009. Watermarking in conjugate ordered dither block truncation coding images. Signal Process., 89: 1864-1882.

Lee, H.Y., D.H. Im and H.K. Lee, 2007. Error concealment technique of satellite imagery transmission through information hiding. IEICE Trans. Inf. Syst., E90-D: 1881-1884.

Luo, H., F.X. Yu, H. Li and Z.L. Huang, 2010. Color image encryption based on secret sharing and iterations. Inform. Technol. J., 9: 446-452.

Pan, J.S., H.C. Huang, L.C. Jain and W.C. Fang, 2007. Intelligent Multimedia Data Hiding: New Directions. Springer, Berlin-Heidelberg, Germany,.

Ruan, Z., X. Sun, W. Liang, D. Sun and Z. Xia, 2010. CADS: Co-operative anti-fraud data storage scheme for unattended wireless sensor networks. Inform. Technol. J., 9: 1361-1368.

Thien, C.C. and J.C. Lin, 2002. Secret image sharing. Comput. Graphics, 26: 765-770.

Zhao, Z., H. Luo and Z.M. Lu, 2010. Shadow size reduction and multiple image secret sharing based on discrete fractional random transform. Inform. Technol. J., 9: 298-304.