Reversible Data Hiding in Block Compressed Sensing Images

Ming Li, Di Xiao, and Yushu Zhang

Block compressed sensing (BCS) is widely used in image sampling and is an efficient, effective technique. Through the use of BCS, an image can be simultaneously compressed and encrypted. In this paper, a novel reversible data hiding (RDH) method is proposed to embed additional data into BCS images. The proposed method is the first RDH method of its kind for BCS images. Results demonstrate that our approach performs better compared with other state-of-the-art RDH methods on encrypted images.

Keywords: Reversible data hiding, block compressed sensing, encrypted image.

I. Introduction

Reversible data hiding (RDH) is a technology that embeds a secret message into a cover image in a reversible manner. In recent years, to protect the privacy of a cover image during an RDH process, attentions have been drawn to the research of RDH in encrypted images. In [1], Zhang proposed an RDH method for use with an image that has been fully encrypted using a standard stream cipher. Given such an encrypted image, we first divide it into several blocks. Then, by flipping the three least significant bits (LSBs) of every eight in each pixel for half of the pixels in each block, additional data can be embedded. Data extraction and image recovery is then carried out through a determination of which pixels have been flipped within each block. To separate data extraction from image decryption, Zhang [2] emptied out space for data embedding based on an idea for compressing encrypted images [3], [4].

Ma and others [5] proposed an RDH method for encrypted images, and within this method, space meant for data embedding is reserved prior to any image encryption. In the methods of [2]–[5], an extra processing step (that is, image compression) is required. Zhang's method [1] is a classical block division–based method — one that was improved upon later by [6]–[8]. However, the extracted bit-error rate of the improved methods in [6]–[8] is still unsatisfactory. In addition, an encrypted image cannot be compressed — a fact that can result in high communication overheads.

By using compressed sensing (CS) [9], both compression [10] and encryption [11] can be achieved simultaneously. For natural image sampling, block compressed sensing (BCS) [12] is widely used for its high computing speed and ease of implementation. BCS images may have broader application prospects compared with stream cipher–encrypted images due

Manuscript received Feb. 27, 2014; revised Sept. 11, 2015; accepted Sept. 25, 2015.

The work was supported by the National Natural Science Foundation of China (Grant Nos. 61173178, 61272043, 61302161, 61472464, 61502399, and 61572089), the Fundamental Research Funds for the Central Universities (Grant Nos. XDJK2015C077, 106112013CDJZR180005, and 106112014CDJZR185501), the Natural Science Foundation Project of Chongqing CSTC (Grant Nos. cstc2012jjA40017, cstc2013jcyjA40017, cstc2013jjB40009, and cstc2015jcyjA40039), the Key Program of the Higher Education Institutions of Henan Province (Grant No. 15A520020), the PhD Scientific Research Foundation of Henan Normal University (Grant No. qd14134), and the Science Foundation for Young Scholars of Henan Normal University (Grant No. 2014QK20).

Ming Li (liming@htu.edu.cn) is with the College of Computer and Information Engineering, Henan Normal University, Xinxiang, China.

Di Xiao (corresponding author, xiaodi_cqu@hotmail.com) is with the Key Laboratory of Dependable Service Computing in Cyber Physical Society (Chongqing University) of Ministry of Education, College of Computer Science, Chongqing University, China.

Yushu Zhang (mishuzhang@foxmail.com) is with the School of Electronics and Information Engineering, Southwest University, Chongqing, China.

to the fact that the communication overhead of BCS images is lower. However, so far, there has been no RDH method proposed for BCS images. Therefore, we offer a solution to this kind of application in this paper.

II. Proposed Method

As illustrated in Fig. 1, an original image is first sampled by a content owner using BCS to obtain a BCS image according to a Gaussian measurement matrix generated from a secret seed. Then, additional data can be embedded into the obtained BCS image by a data hider in accordance with a data-hiding key. When the receiver possesses both the data-hiding key and the secret seed (both of which are required to generate a measurement matrix), then the image can be recovered and any embedded data may be extracted at the same time.

1. BCS

At the initial stage of our proposed architecture, an original image is divided into non-overlapping blocks, each of which is of the same size. For each block, let $X \in \mathbb{R}^n$ represent a vectorized signal of a block, and let $Y \in \mathbb{R}^m$ ($m < n$) be a number of linear random projections (measurements) obtained from

$$Y = AX . \quad (1)$$

The measurement matrix $A \in \mathbb{R}^{m \times n}$ must satisfy a restricted isometry property (RIP) of order k ($k \ll n$). It has been shown that the entries of such a matrix can be chosen according to a Gaussian distribution [13], generated using a random seed. Figure 2 gives an illustration of CS for a block. The obtained measurement, Y, is denoted as a dot. Its square is less than that of the original signal, X. When an image is transformed into multiple measurements after BCS, the amount of data is reduced (see Fig. 3). Finally, the obtained measurements are assembled to form a BCS image. Thus, the size of the original image is decreased, and any content is concealed. The BCS applied here is secure due to the fact that the underlying measurement matrix used in each block is believed to be secure [11].

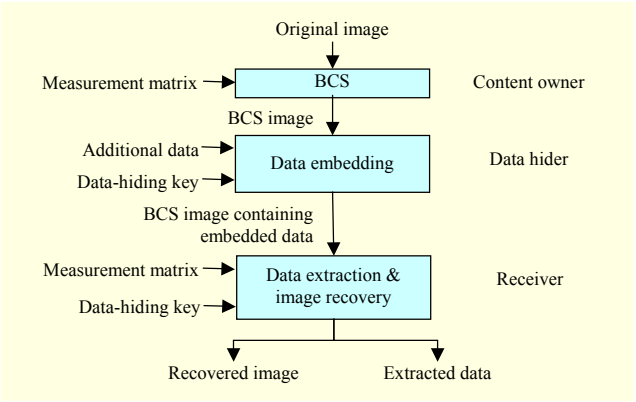


Fig. 1. Proposed architecture.

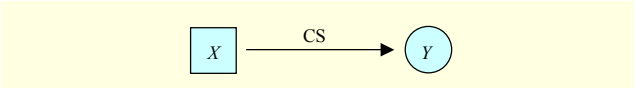


Fig. 2. CS performed on one block.

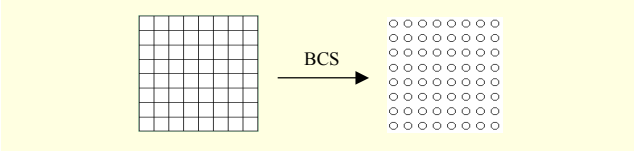


Fig. 3. BCS of image.

2. Data Embedding

In Fig. 3, each dot (measurement) corresponds to an image block. To embed data into a BCS image, we first divide the measurements into two categories — embeddable and un-embeddable. In Fig. 4, the measurements denoted by the black dots are used for data embedding, and the white dots are set to be un-embeddable. Here, we assume that one dot can carry one bit, and the number of bits of any additional data is smaller than that of the black dots. Obviously, the embedding capacity varies with the block size.

At the beginning of data embedding, a data hider encrypts any additional data to be embedded with a data-hiding key. It then embeds the ith bit of encrypted additional data into the ith embeddable measurement (that is, the black dots). If the bit is 0, then it does nothing; else, it replaces measurement Y with Y' using

$$Y' = A \times \overrightarrow{255} - Y, \quad (2)$$

where $\overrightarrow{255}$ denotes a block in which the values of its pixels are all set to be 255. It is noted that only the result of $A \times \overrightarrow{255}$ can be known to the data hider. The measurement matrix A is secret.

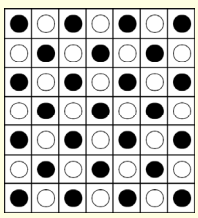


Fig. 4. Division of BCS image by measurements.

ETRI Journal, Volume 38, Number 1, February 2016
http://dx.doi.org/10.4218/etrij.16.0114.0242

3. Data Extraction and Image Recovery

Knowing the measurements, Y, the measurement matrix, A, and that X is k-sparse, we can reconstruct the original signal, X, approximately by solving the following ℓ_1 minimization problem [13]:

$$\begin{aligned} &\text{minimize} \quad \| X \|_1 \\ &\text{s.t.} \quad Y = AX. \end{aligned} \tag{3}$$

For natural images, the original signal X is not sparse, but it has a sparse representation in some orthonormal bases. Let $\Phi \in \mathbb{R}^{n \times n}$ denote an orthonormal matrix whose columns are the basis vectors. Then, X can be represented as $\Phi\alpha$, where α is nearly k-sparse. Given the measurements $Y = AX$, the original signal X can be recovered by solving the following minimization problem:

$$\begin{aligned} &\text{minimize} \quad \| \alpha \|_1 \\ &\text{s.t.} \quad Y = A\Phi\alpha. \end{aligned} \tag{4}$$

In the case of modified measurements, Y', where

$$\begin{aligned} Y' &= A \times \overrightarrow{255} - Y \\ &= A \times \overrightarrow{255} - A \times X \\ &= A \times (\overrightarrow{255} - X), \end{aligned} \tag{5}$$

the recovered signal should be $\overrightarrow{255} - X$; that is, a flipped X.

Due to the spatial correlation of natural images, any border pixels between two unflipped blocks are smoother than those that have been flipped. Since the neighboring blocks of X are all unflipped (as shown in Fig. 4; the blocks with a white dot inside cannot be flipped by data embedding), a flipped X can be identified according to the following side match method.

Suppose there are four neighboring blocks around X. Let $p_{u,v}$ denote the pixel value at position (u, v) in the block represented by X of size $s_1 \times s_2$. The fluctuation function of the block sides can then be defined as

$$\begin{aligned} f = &\sum_{v=1}^{s_2} (| p_{0,v} - p_{1,v} | + | p_{s_1,v} - p_{s_1+1,v} |) \\ &+ \sum_{u=1}^{s_1} (| p_{u,0} - p_{u,1} | + | p_{u,s_2} - p_{u,s_2+1} |), \end{aligned} \tag{6}$$

where $p_{0,v}$, $p_{s_1+1,v}$, $p_{u,0}$, and p_{u,s_2+1} fall into the neighboring blocks.

Similarly, the fluctuation function of a flipped X can be computed from

$$\begin{aligned} f' = &\sum_{v=1}^{s_2} (| p_{0,v} - \overline{p}_{1,v} | + | \overline{p}_{s_1,v} - p_{s_1+1,v} |) \\ &+ \sum_{u=1}^{s_1} (| p_{u,0} - \overline{p}_{u,1} | + | \overline{p}_{u,s_2} - p_{u,s_2+1} |), \end{aligned} \tag{7}$$

where $\overline{p}_{u,v}$ denotes a flipped pixel value of X.

If $f < f'$, which means the block sides of the recovered X are smoother than that of the flipped X, then the recovered X is considered unflipped, and the bit embedded in it is 0; otherwise, X is considered flipped during the data embedding procedure, and the bit embedded in it is then 1. Moreover, X can then be flipped again to obtain the final recovered signal.

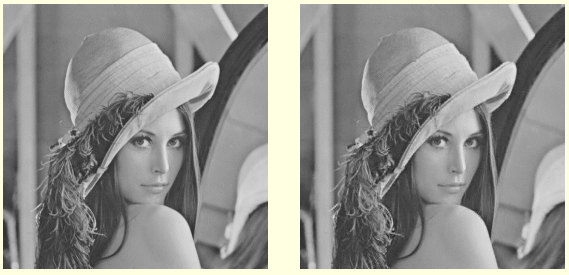
At last, all the recovered signals are collected to form the recovered image, and all the extracted bits are concatenated and then decrypted using the data-hiding key to obtain the final extracted data.

III. Experimental Results

A standard test image, named Lena and of size 512 × 512, is tested first. Assume the size of a divided block to be 8 × 8. Then, the original signal of the block (that is, X) can be orthogonally decomposed into sparse discrete cosine transformation coefficients. The orthogonal matching pursuit is used to reconstruct the image. We set the compression ratio to be 0.8; thus, the size of the original image is decreased to 80%. The results are shown in Fig. 5, where (a) is the original image and (b) is the final recovered image after 2,048 bits are embedded (its PSNR value is 33.90). The embedded bits are extracted perfectly, and the recovered image is exactly the same as that obtained directly from a BCS image with no additional data embedded. For the next image, named Baboon, the settings are the same as above, and the experimental results show almost the worst case of image recovery using the proposed method. In Fig. 6(b), although it seems the same as Fig. 6(a), there exist 50 incorrect recovered blocks; that is, about 1.2% of the blocks have been flipped compared with the original BCS image, and the PSNR is 24.44. However, if the block is 16 × 16, then correct data-extraction and perfect image recovery can both be ensured.

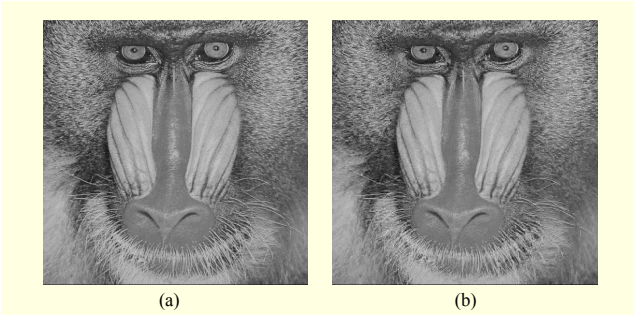
In further experiments, we randomly selected 50 natural images sized 512 × 512 from [14] and presented the average exacted bit-error rate comparison.

BCS is a lossy compression method, but the compression is



(a) (b)

Fig. 5. Lena: (a) original image and (b) final recovered image.



(a) (b)

Fig. 6. Baboon: (a) original image and (b) final recovered image.

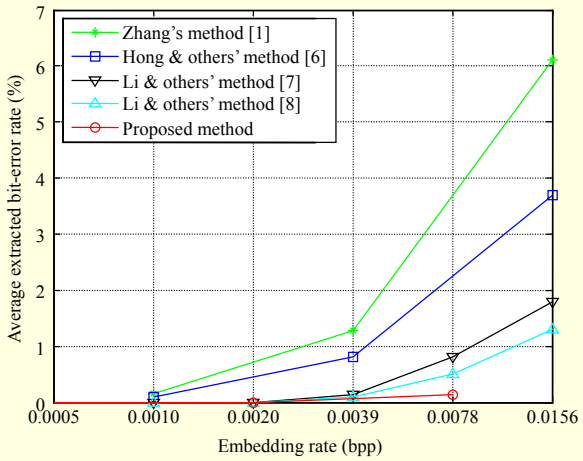


Fig. 7. Average extracted BER comparison.

based on blocks. Each block corresponds to one measurement, and each measurement can carry one bit of any additional data to be embedded. Similarly, in the block division–based methods [1] and [6], which are for uncompressed images, one image block is also responsible for carrying one bit. Therefore, [1], [6], and the recent improved version [7], [8] are compared with our method. For a fair comparison, the average exacted bit-error rates are computed under different embedding rates (bits per pixel (bpp)) rather than data embedding capacity (bits).

It is noticed that in [1] and [6]–[8], the content owner encrypts but does not compress the original image I to obtain $E(I)$, and the data hider embeds data into $E(I)$. A user can extract data from the embedded $E(I)$ and try to losslessly restore the original image I. However, in the proposed method, the original image I is encrypted and compressed by BCS, and from a BCS image one can only obtain an estimated version of I, denoted by I'. The data hider embeds data into the BCS image. A user can extract data from the embedded BCS image (that is, a BCS image containing embedded data) and try to restore I' rather than the original image I. Therefore, the embedding rate of our method should be "bpp of I'" rather than "bpp of I," but the results of the two are equal to each other.

The comparison results are shown in Fig. 7. When the side length of a block is set to be 32 (respectively, 16, 8), the embedding rates of Zhang's method [1] and Hong and others' method [6] are both 0.0010 (respectively, 0.0039, 0.0156), and it is 0.0005 (respectively, 0.0020, 0.0078) using the proposed method. Since Li and others' methods, [7] and [8], break the idea of block division and can achieve arbitrarily assigned embedding rates, we calculate the average extracted bit-error rates under each given embedding rate on the horizontal axis. The error rates are calculated as the ratio of the number of incorrectly recovered embedded bits to the total number of embedded bits. As shown in Fig. 7, it is clear that the error rates of the proposed method are significantly lower than that of [1] and [6]–[8]. The main reason is that the correctness of the border-concatenated pixels of the unflipped blocks can be ensured, which can help with the recovery of blocks containing any additional data, as well as with the extraction of any embedded bits. Furthermore, the proposed method shows more steady performance.

IV. Conclusion

The first ever RDH method for BCS images is proposed in this paper. Experiments show that the proposed method can outperform existing RDH methods in encrypted images.

References

[1] X. Zhang, "Reversible Data Hiding in Encrypted Image," *IEEE Signal Process. Lett.*, vol. 18, no. 4, Apr. 2011, pp. 255–258.

[2] X. Zhang, "Separable Reversible Data Hiding in Encrypted Image," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, Apr. 2012, pp. 826–832.

[3] W. Liu et al., "Efficient Compression of Encrypted Grayscale Images," *IEEE Trans. Image Process.*, vol. 19, no. 4, Apr. 2010, pp. 1097–1102.

[4] X. Zhang et al., "Efficient Reversible Data Hiding in Encrypted Images," *J. Vis. Commun. Image Representation*, no. 25, no. 2, Feb. 2014, pp. 322–328.

[5] K. Ma et al., "Reversible Data Hiding in Encrypted Images by Reserving Room Before Encryption," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 3, Mar. 2013, pp. 553–562.

[6] W. Hong, T.-S. Chen, and H.-Y. Wu, "An Improved Reversible Data Hiding in Encrypted Images Using Side Match," *IEEE Signal Process. Lett.*, vol. 19, no. 4, Apr. 2012, pp. 199–202.

[7] M. Li et al., "A Modified Reversible Data Hiding in Encrypted Images Using Random Diffusion and Accurate Prediction," *ETRI J.*, vol. 36, no. 2, Apr. 2014, pp. 325–328.

[8] M. Li et al., "Improved Reversible Data Hiding for Encrypted Images Using Full Embedding Strategy," *Electron. Lett.*, vol. 51,

no. 9, Apr. 2015, pp. 690–691.
[9] D.L. Donoho, "Compressed Sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, Apr. 2006, pp. 1289–1306.
[10] Z. Gao et al., "Image Representation Using Block Compressive Sensing for Compression Applications," *J. Vis. Commun. Image Representation*, vol. 24, no. 7, Oct. 2013, pp. 885–894.
[11] Y. Rachlin and D. Baron, "The Secrecy of Compressed Sensing Measurements," *Annual Allerton Conf. Commun., Contr. Comput.*, Urbana-Champaign, IL, USA, Sept. 23–26, 2008, pp. 813–817.
[12] L. Gan, "Block Compressed Sensing of Natural Images," *Int. Conf. Digital Signal Process.*, Cardiff, UK, July 1–4, 2007, pp. 403–406.
[13] Y.C. Eldar and G. Kutyniok, "Compressed Sensing: Theory and Applications," NY, USA: Cambridge University Press, 2012, pp. 226–232.
[14] R. Rodríguez-Sánchez et al., *Miscellaneous Gray Level Images (512×512)*, Computer Vision Group. University of Granada, 2014. Accessed Nov. 6, 2015. http://decsai.ugr.es/cvg/dbimagenes/g512.php

Ming Li received his MS degree in science from the College of Physics and Information Engineering, Henan Normal University, Xinxiang, China, in 2010 and his PhD degree in engineering from the College of Computer Science, Chongqing University, China, in 2014. Since 2014, he has been with the College of Computer and Information Engineering, Henan Normal University, where he is now an associate professor. His main research interests include multimedia security, information hiding, and compressed sensing.

Di Xiao received his PhD degree in computer software and theory from Chongqing University, China, in 2005. From 2006 to 2008, he conducted postdoctoral research at Chongqing University, Chongqing, China. From 2008 to 2009, he was a visiting scholar, funded by the Chinese government, with the Department of Computer Science, New Jersey Institute of Technology, Newark, USA. Currently, he is a professor at the College of Computer Science, Chongqing University. His main research interests include image processing, chaos-based cryptography, image watermarking, and compressed sensing. He is a member of IEEE and ACM.

Yushu Zhang received his BS degree in science from the Department of Mathematics, North University of China, Taiyuan, China, in 2010 and his PhD degree in engineering from the College of Computer Science, Chongqing University, China, in 2014. Since 2014, he has been with the School of Electronics and Information Engineering, Southwest University, Chongqing, China, where he is now an associate professor. His main research interests include multimedia security, multimedia coding, and compressed sensing.