Application of Bit-Plane Decomposition Steganography to JPEG2000 Encoded Images

Hideki Noda, Jeremiah Spaulding, Mahdad N. Shirazi, and Eiji Kawaguchi

Abstract—**This letter presents a steganography method based on a JPEG2000 lossy compression scheme and bit-plane complexity segmentation (BPCS) steganography. It overcomes the lack of robustness of bit-plane-based steganography methods with respect to lossy compression of a dummy image: a critical shortcoming that has hampered deployment in a practical scenario. The proposed method is based on a seamless integration of the two schemes without compromising their desirable features and makes feasible the deployment of the merits of a BPCS steganography technique in a practical scenario where images are compressed before being transmitted over the network. Embedding rates of around 15% of the compressed image size were achieved for preembedding 1.0-bpp compressed images with no noticeable degradation in image quality.**

Index Terms—**Bit plane, data hiding, information hiding, JPEG2000, steganography.**

I. Introduction

STEGANOGRAPHY is the practice of hiding or camouflaging secret data in an innocent looking dummy container. This container may be a digital still image, audio file, video file, or even a printed image. Once the data have been embedded, it may be transferred across insecure lines or posted in public places. Therefore, the dummy container should seem innocent under most examinations.

In previous steganographic algorithms, bit-plane decomposition (e.g., an n-bit image can be decomposed into a set of n binary images by bit-slicing operations) was commonly used [1], [2] combined with the simple approach of replacing the binary data in the least significant bit planes with secret binary data [1]. We presented a sophisticated steganography method, called bit-plane complexity segmentation (BPCS) steganography, which makes use of bit-plane decomposition and the characteristics of the human vision system [3]. Noting that the human eye cannot perceive any shape information in a very complicated binary pattern, we can replace noise-like regions in the bit planes of the dummy image with secret data without deterioration of image quality. BPCS steganography has proven to be very effective in embedding data into many classes of dummy files including eight-bit gray images [3], 24-bit true-color images [4], eight-bit indexed color images [5], and digital audio files [6]. The benefits of this technique over traditional steganography are the very large percentage (30% to 50%) of the dummy file that can be replaced with secret data and the lower occurrence of visual artifacts in the postembedding image.

However, BPCS steganography is not robust with respect to lossy compression of the dummy image, as are all other bit-plane-based steganography methods. To deploy the merits of the BPCS steganography technique, in a practical scenario where the dummy image should be compressed before being transmitted, we propose a steganography technique based on the JPEG2000 lossy compression scheme [7] and BPCS steganography. In JPEG2000 compression, wavelet coefficients of an image are quantized into a bit-plane structure, and BPCS steganography can, therefore, be applied in the wavelet domain. The proposed JPEG2000-BPCS steganography provides a significant integration of JPEG2000 lossy compression scheme and BPCS steganography and a solution to the aforementioned problem associated with bit-plane-based steganography methods.

II. BPCS Steganography

When an image is decomposed into bit planes, the complexity of each region can be measured. Areas of low complexity such as homogenous color or simple shapes appear as uniform areas with very few changes between one and zero. Complex areas would appear as noise-like regions with many changes between one and zero. These seemingly random regions in each bit plane can then be replaced with hidden data, which is ideally also noise-like. Because it is difficult for the human eye to distinguish differences between the two noise-like areas, we are able to disguise the changes to the image. In BPCS steganography, the complexity of each subsection of a bit plane is defined as the number of nonedge transitions from one to zero and zero to one, both horizontally and vertically. For any square of $n \times n$ pixels, the maximum complexity is $2n(n-1)$, and the minimum is of course zero.

A typical procedure for data hiding in BPCS steganography is summarized as follows.

1) Segment each bit plane of a dummy image into small size, e.g., 8×8 blocks. Then classify these blocks into informative and noise-like blocks using a threshold of the complexity α_0. A typical value of α_0 is $0.3\alpha_{\max}$, where $\alpha_{\max}$ is the maximum possible complexity value.
2) Segment a secret file into a series of blocks each containing eight bytes of data. These blocks, which we call secret blocks, are regarded as 8×8 binary images.

Manuscript received April 23, 2002; revised August 8, 2002. This work was supported in part by the Okawa Foundation for Information and Telecommunications, Japan. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Amir Aisf.

H. Noda, J. Spaulding, and E. Kawaguchi are with the Department of Electrical, Electronic and Computer Engineering, Kyushu Institute of Technology, Kitakyushu, 804-8550 Japan (e-mail: noda@know.comp.kyutech.ac.jp; jerry@know.comp.kyutech.ac.jp; kawaguch@know.comp.kyutech.ac.jp).

M. N. Shirazi is with the Keihanna Human Info-Communications Research Center, Communications Research Laboratory, Kyoto, 619-0289 Japan (e-mail: mahdad@crl.go.jp).

Digital Object Identifier 10.1109/LSP.2002.806056

3) If a secret block is less complex than the threshold α_0, conjugate it to make it more complex. Here the process called conjugation, which guarantees that any secret data can be embedded, is the exclusive-or operation with a checkerboard pattern. The relation $\alpha^* = \alpha_{\max} - \alpha$ holds true [3], where α and α^* are the complexity of a given image and that of the conjugated image, respectively.
4) Replace each noise-like block in the bit planes with a block of secret data. If the block is conjugated, then record this fact in a conjugation map. The location order of block replacement can be determined using a random-number generator.
5) Also embed the conjugation map in the same way as the secret blocks (since the conjugation map is generally compressed before embedding, its complexity is high enough; therefore conjugation is generally unnecessary for the conjugation map). However, in case it is necessary and applied, that information should be given to the receiver. The conjugation map is usually embedded into first noise-like blocks in the replacement order.

The decoding procedure to extract the embedded secret data is just the reverse of the embedding procedure. In the decoding process, the embedding threshold α_0 and amount of secret data need to be known. The amount of secret data can be embedded into a specific place in the dummy file.

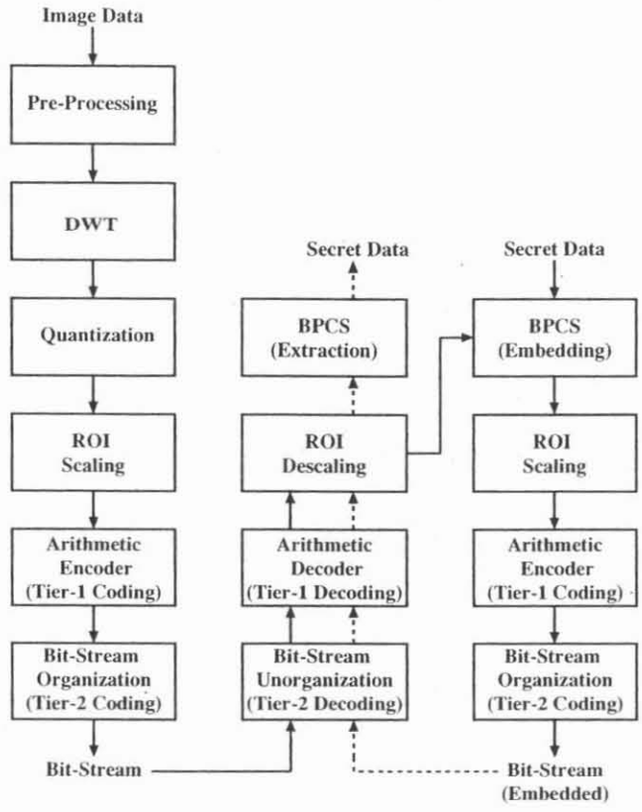

Fig. 1. Flowchart of data embedding and extraction in JPEG2000-BPCS steganography.

III. JPEG2000 Compression Standard

JPEG2000 encoder consists of several fundamental components: preprocessing, discrete wavelet transform (DWT), quantization, arithmetic coding (tier-1 coding) and bit stream organization (tier-2 coding) [7] (see the left part of Fig. 1). Preprocessing includes intercomponent transformation for multicomponent images, typically color images. After the DWT is applied to each component, wavelet coefficients are quantized uniformly with deadzone. After the quantization step, an optional step to realize a functionality called region of interest (ROI) can be added. The ROI is realized by scaling up the wavelet coefficients in the relevant regions. The quantized wavelet coefficients are then bit-plane-encoded by arithmetic coding.

In JPEG2000, each subband of the wavelet transformed image is encoded independently of the other subbands. Furthermore, each subband is partitioned into small blocks called codeblocks, and each codeblock is independently encoded by an embedded block-coding-with-optimized-truncation algorithm [8]. This procedure is absolutely different from other well-known embedded wavelet coders such as embedded zerotree wavelet [9] and set partitioning in hiearchical trees [10]. The independent encoding of codeblocks provides many advantages such as localized random access into an image, improved error resilience, efficient rate control, and flexible bit stream ordering. The quantized wavelet coefficients in a codeblock are bit-plane-encoded by three passes with arithmetic coding: significance propagation pass, refinement pass, and cleanup pass.

The compressed data from the codeblocks are organized into units called packets and layers in tier-2 coding. A precinct is a collection of spatially contiguous codeblocks from all subbands at a resolution level. The compressed data from the codeblocks in a precinct constitutes a packet. A collection of packets from all resolution levels constitutes a layer. Therefore, a layer corresponds to one quality increment for the entire full-resolution image. In JPEG2000, different types of progression orders are possible, and a typical one is layer-resolution-component-position progression. Once the entire image has been compressed, a rate-distortion-optimized bit stream is generated for a target file size (bit rate).

IV. JPEG2000-BPCS Steganography

Basically, secret data can be embedded in the bit-plane representation of the quantized wavelet coefficients after the quantization step, provided that the rate-distortion optimization of JPEG2000 is bypassed. However, this procedure, which determines the optimal number of bit planes for a given bit rate, is an essential part of the codec, which contributes to its high compression efficiency. Thus, to avoid compromising the compression efficiency of JPEG2000, data embedding by BPCS is decided to be performed right after ROI descaling in the decoding process where the optimal bit-plane structure for a given bit rate is available. The procedure for data embedding and extraction in JPEG2000-BPCS steganography is shown in Fig. 1.

The entire process to embed data in JPEG2000-BPCS steganography follows the solid line arrows shown in Fig. 1. An image is encoded into a JPEG2000 bit stream, whose size can be met almost exactly to a target bit rate (bit per pixel). The encoding process is shown in the left part of Fig. 1; from

preprocessing to bit stream organization. The JPEG2000 bit stream (compressed image file) is then decoded, but decoding is halted right after ROI descaling. The information at this point is used to construct the bit planes of quantized wavelet coefficients and then used to embed secret data with BPCS steganography (see the top box of the right part in Fig. 1). The quantized wavelet coefficients modified by embedding are then subjected to JPEG2000 encoding again, which produces a secret-data-embedded JPEG2000 bit stream. Data embedding into an already compressed JPEG2000 file is also possible. In this case, the process starts with a JPEG2000 compressed image, i.e., a bit stream from the bottom of the middle part in Fig. 1, and follows the same process as the aforementioned one.

The data extraction procedure follows the dashed arrows in the middle part of Fig. 1. JPEG2000 decoding of the secret-data-embedded bit stream starts from bit stream unorganization and is halted right after ROI descaling. At this point, extraction of secret data is carried out by the BPCS method using the bit planes of quantized wavelet coefficients. We assume that the data extraction starts after the entire file of the bit stream has been received.

V. Experimental Results

The JPEG2000-BPCS steganography was implemented using JJ2000 Java software of JPEG2000 compression [11], with which the program module for BPCS steganography was integrated. It was tested on several standard images including "Lena," "Barbara," and "Mandrill." Lena and Barbara are 8-bpp gray images, and Mandrill is a 24-bpp true-color image, all of which were 512×512 pixels in size. Here a 4×4 patch size was used as an embedding unit, and random binary data was used as secret data.

In the implementation of JPEG2000-BPCS steganography, an error correction scheme was devised to decrease the distortion of a data-embedded image. As data are being embedded into the wavelet coefficients, each bit that is used for embedding is recorded. After all the data have been embedded, the bits of each coefficient that have not been used are changed to bring the new value of the coefficient as close to the original value as possible. The change of bits is only allowed unless the change makes the complexity value for the relevant patch larger than the complexity threshold for embedding. The PSNR with the error correction increased by about 1.7 dB. In the following experiments, the error correction was always applied.

Results of embedding experiments are shown in Fig. 2. The least significant bit plane and the two least significant bit planes were used to embed data for 0.5-bpp and 1.0-bpp compressed images, respectively. Nine data points within each line in Fig. 2 were obtained by changing the complexity threshold α_0 from two to ten. In Fig. 2, data points for no data embedding are also included. Note that the compression rate for the color Mandrill image is in fact three times less than those for the other gray images. Generally, the JPEG2000-BPCS steganography was able to achieve embedding rates of around 9% of the final compressed image size for preembedding 0.5-bpp images and 15% for preembedding 1.0-bpp images with no noticeable degradation in image quality. These results were derived with the complexity threshold $\alpha_0 = 8$, which corresponds to $0.33\alpha_{\max}$. Fig. 3 shows experimental results for Mandrill.

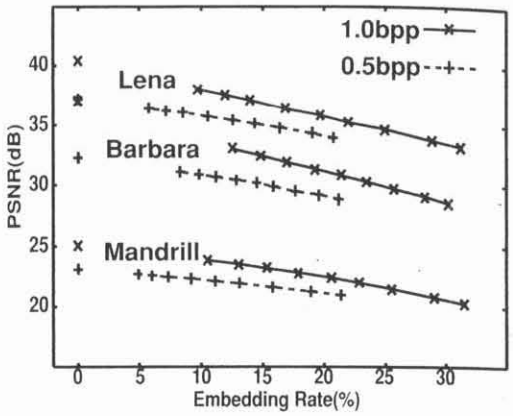

Fig. 2. Results of embedding experiments by JPEG2000-BPCS steganography.

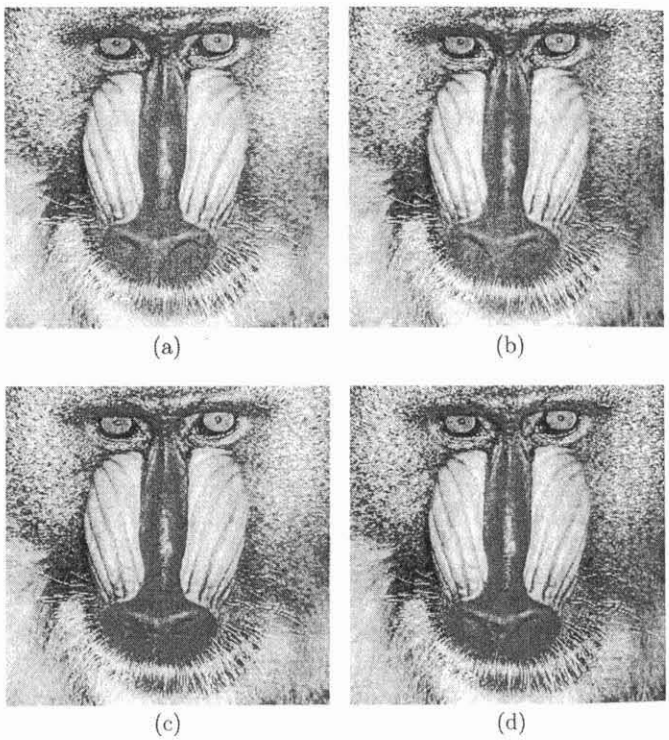

Fig. 3. Experimental results for "Mandrill." (a) JPEG2000 compressed image (0.5 bpp). (b) Seven percent embedded into (a). (c) JPEG2000 compressed image (1 bpp). (d) Fifteen percent embedded into (c).

Data embedding increases the postembedding file size significantly. For Barbara, as an example, 16 588 bytes (file size of preembedding a 0.5-bpp image) increased to 21 532 bytes by embedding 2412-byte data and 32 581 bytes (file size of preembedding a 1.0-bpp image) to 43 283 bytes by embedding 7332 bytes. However, the proposed method shows a high performance, as evidenced by comparing its embedding rate $2412/21\,532 = 11.2\%$ with 2.2% (reported for Barbara) by a conventional method [12] that embeds data into a JPEG bit stream by modifying the quantized discrete cosine transform coefficients.

VI. Conclusion

This letter presented a solution to the problem of hiding data into compressed image files in bit-plane-based steganography methods. The proposed scheme is based on a seamless integration of BPCS steganography with the JPEG2000 image compression standard. Embedding rates of around 15% of the compressed image size were achieved for preembedding 1.0-bpp compressed images with no noticeable degradation in image quality. The proposed JPEG2000-BPCS steganography will allow many more people access to the benefits of BPCS steganography due to many desirable features of the JPEG2000 standard.

References

[1] S. Katzenbeisser and F. A. P. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*. Norwood, MA: Artech House, 2000.

[2] R. Z. Wang, C. F. Lin, and J. C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognit.*, vol. 34, pp. 671–683, 2001.

[3] M. Niimi, H. Noda, and E. Kawaguchi, "A steganography based on region segmentation by using complexity measure" (in Japanese), *Trans. IEICE*, vol. J81-D-II, pp. 1132–1140, 1998.

[4] E. Kawaguchi and R. O. Eason, "Principle and applications of BPCS-steganography," in *Proc. SPIE*, vol. 3528, 1998, pp. 464–473.

[5] R. Ouellette, H. Noda, M. Niimi, and E. Kawaguchi, "Topological ordered color table for BPCS steganography using indexed color images," *IPSJ J.*, vol. 42, pp. 110–113, 2000.

[6] I. Kusatsu, M. Niimi, H. Noda, and E. Kawaguchi, "A Large capacity steganography using acoustic dummy data," (in Japanese), EA98-69-78, 1998.

[7] M. Rabbani and R. Joshi, "An overview of JPEG 2000 still image compression standard," *Signal Process.: Image Commun.*, vol. 17, pp. 3–48, 2002.

[8] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, pp. 1158–1170, July 2000.

[9] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

[10] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, June 1996.

[11] JJ2000 Partners. JJ2000 Web site. [Online]. Available: http://jj2000.epfl.ch/index.html.

[12] H. Kobayashi, Y. Noguchi, and H. Kiya, "A method of embedding binary data into JPEG bitstreams" (in Japanese), *Trans. IEICE*, vol. J83-D-II, pp. 1469–1476, 2000.