# APPLICATION OF BPCS STEGANOGRAPHY TO WAVELET COMPRESSED VIDEO

*Hideki Noda †, Tomonori Furuta †, Michiharu Niimi †, Eiji Kawaguchi †*

†Kyushu Institute of Technology, Kitakyushu, 804-8550 Japan

## ABSTRACT

This paper presents a steganography method using lossy compressed video which provides a natural way to send a large amount of secret data. The proposed method is based on wavelet compression for video data and bit-plane complexity segmentation (BPCS) steganography. In wavelet-based video compression methods such as 3-D set partitioning in hierarchical trees (SPIHT) algorithm and Motion-JPEG2000, wavelet coefficients in discrete wavelet transformed video are quantized into a bit-plane structure and therefore BPCS steganography can be applied in the wavelet domain. 3-D SPIHT-BPCS steganography and Motion-JPEG2000-BPCS steganography are presented and tested, which are the integration of 3-D SPIHT video coding and BPCS steganography, and that of Motion-JPEG2000 and BPCS, respectively. Experimental results show that 3-D SPIHT-BPCS is superior to Motion-JPEG2000-BPCS with regard to embedding performance.

## 1. 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 has 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 was commonly used combined with the simple approach of replacing the binary data in the least significant bit-planes of a dummy image with secret binary data[1]. We previously 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[2, 3]. Noting that human 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 deteriorating the image quality. The original BPCS steganography could not be applied to lossy compressed images. However it has been developed recently to be applicable to wavelet-based lossy compressed images including JPEG2000 encoded images[4, 5]. The wavelet-based BPCS steganography enables us to use steganography in a practical scenario where images are compressed before being transmitted.

This paper presents a steganography method using lossy compressed video which provides a natural way to send a large amount of secret data. The proposed method is based on wavelet compression for video data and BPCS steganography. In wavelet-based video compression methods such as three-dimensional (3-D) set partitioning in hierarchical trees (SPIHT) algorithm[6] and Motion-JPEG2000[7], wavelet coefficients of video by discrete wavelet transform (DWT) are quantized into a bit-plane structure and therefore BPCS steganography can be applied in the wavelet domain. In this paper, 3-D SPIHT-BPCS steganography and Motion-JPEG2000-BPCS steganography are presented, which are the integration of 3-D SPIHT video coding and BPCS steganography, and that of Motion-JPEG2000 and BPCS, respectively.

## 2. 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 random-seeming 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, a complexity measure is introduced to decide whether a binary image is noise-like or not. The complexity measure currently used is defined based on the length of non-edge border between zero and one. The complexity $\alpha$ for an $m \times m$ size binary image is defined as

$$\alpha = \frac{k}{2m(m-1)}, \quad 0 \leq \alpha \leq 1, \tag{1}$$

This work was partly supported by the International Communications Foundation, Japan.

where $k$ is the total length of the zero-one border in the image and $2m(m-1)$ is the maximum possible border length obtained from an $m \times m$ checkerboard pattern.

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

**(1)** Segment each bit-plane of a dummy image into small size, for example $8 \times 8$, blocks. Then classify these blocks into informative or noise-like blocks using a threshold of the complexity denoted by $\alpha_0$. A typical value of $\alpha_0$ is 0.3.

**(2)** Segment a secret file into a series of blocks each containing 8 bytes of data. These blocks (which we call secret blocks) are regarded as $8 \times 8$ binary images.

**(3)** If a secret block is less complex than the threshold $\alpha_0$, 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 $\alpha^* = 1 - \alpha$ holds true[2], where $\alpha$ and $\alpha^*$ 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.

**(5)** Also embed the conjugation map.

The decoding procedure to extract the embedded secret data is just the reverse of the embedding procedure. In the decoding process, the complexity threshold $\alpha_0$ and the amount of secret data need to be known. The amount of secret data can be embedded into a specific place in the dummy file.

## 3. 3-D SPIHT-BPCS STEGANOGRAPHY

3-D SPIHT algorithm was proposed for video compression by extending 2-D SPIHT algorithm for image compression[8]. 3-D SPIHT has the following characteristics: (1) partial ordering by magnitude of 3-D wavelet transformed video, (2) ordered bit-plane coding, and (3) exploitation of self-similarity across spatio-temporal orientation trees.

The successive approximation method used by 3-D SPIHT algorithm encodes wavelet coefficients one bit-plane at a time, starting with the most significant bit. In 3-D SPIHT compression, each wavelet coefficient $w$ is expressed as

$$w = T(a_0 + a_1 2^{-1} + \cdots + a_{n-1} 2^{-n+1}), \quad a_i \in \{0, 1\}, \quad (2)$$

where $T = 2^{\lfloor \log_2 w_{max} \rfloor}$ ($w_{max}$ is the maximum absolute value among all wavelet coefficients in a 3-D DWT video). Since $(a_0 + a_1 2^{-1} + \cdots + a_{n-1} 2^{-n+1})$ is a binary expression, the 3-D DWT video can be considered to have a bit-plane structure and therefore BPCS steganography can be applied in the wavelet domain.

The wavelet coefficients have many image-like properties, and BPCS steganography is ideal for exploiting them. The main properties leveraged for BPCS steganography are:

· Correspondence: Spatial areas in each section of the subbands correspond directly to areas in the original image.

· Complexity: The bit-planes at corresponding significance levels of the wavelet coefficients and the original image are usually proportionally complex.

· Resilience: Changes in the values of the wavelet coefficients do not create disproportionately large changes in the reconstructed image.

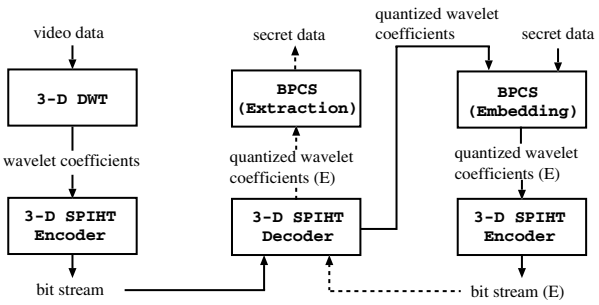


**Fig. 1**. A flowchart of data embedding and extraction in 3-D SPIHT-BPCS steganography.

The procedure for data embedding and extraction in 3-D SPIHT-BPCS steganography is shown in Fig. 1. The entire process to embed data in 3-D SPIHT-BPCS steganography follows the solid line arrows in Fig. 1. After 3-D DWT is applied to an original video, 3-D SPIHT encoder is applied to the wavelet coefficients and a bit-stream (compressed video file) is produced. Then the bit-stream is decoded by 3-D SPIHT decoder and quantized wavelet coefficients are derived[1]. Using these quantized wavelet coefficients, bit-planes for the wavelet coefficients can be constructed and used to embed secret data by BPCS steganography (See the upper box of the right part in Fig. 1). The quantized wavelet coefficients modified by embedding are then subjected to 3-D SPIHT encoding again to produce a secret-data-embedded bit-stream. The mark (E) in Fig. 1 depicts that secret data is embedded. Data embedding in an already compressed video file is also possible. In this case, the process starts with a compressed video file, i.e., a bit-stream from the bottom of the middle part in Fig. 1 and

[1] In principle the two steps of 3-D SPIHT encoding and 3-D SPIHT decoding are unnecessary to obtain the quantized wavelet coefficients. However, the two steps are performed so that the bit-stream may be truncated to meet pre-embedding compression rate requirements.

follows the same process as the aforementioned one.

The data extraction procedure follows the dashed arrows in Fig. 1. 3-D SPIHT decoding of secret-data-embedded bit-stream produces secret-data-embedded quantized wavelet coefficients. Extraction of secret data is carried out by the BPCS method using the quantized wavelet coefficients. We assume that the data extraction starts after the entire file of the bit-stream has been received.

## 4. MOTION-JPEG2000-BPCS STEGANOGRAPHY

Motion-JPEG2000 standard for video coding is described as Part 3 of the JPEG2000 image coding standard[7]. Its file format is designed to contain one or more motion sequences of JPEG2000 coded images with their timing and optional audio annotations. Motion-JPEG2000 is based on intra-frame coding, therefore has reduced complexity and increased resilience to transmission error at the deficit of coding efficiency. Motion-JPEG2000-BPCS steganography can be realized easily since each frame of video is coded independently by JPEG2000 and the BPCS steganography applicable to JPEG2000 coded images (JPEG2000-BPCS steganography) has already been proposed[5].

The procedure for data embedding and extraction in JPEG2000-BPCS steganography is shown in Fig. 2. Considering the intelligent functionality of JPEG2000 on compression rate control, data embedding by BPCS is performed after arithmetic decoding in decoding process.

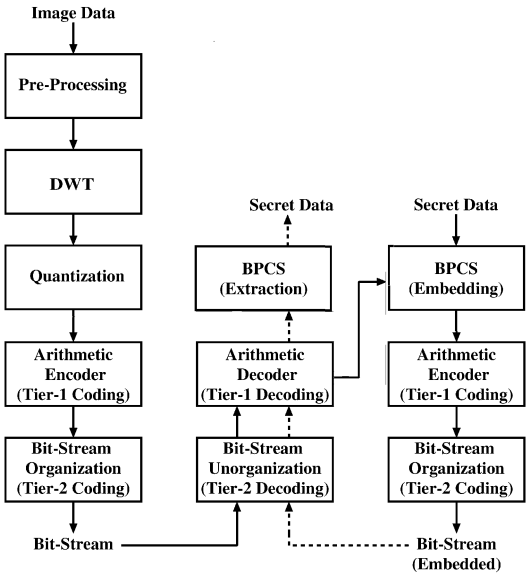


**Fig. 2**. A flowchart of data embedding and extraction in JPEG2000-BPCS steganography.

The entire process to embed data in JPEG2000-BPCS steganography follows the solid line arrows shown in Fig. 2. An image is encoded into JPEG2000 bit-stream, whose size can be set almost exactly to a target bit rate. The encoding process is shown in the left part of Fig. 2; from pre-processing to bit-stream organization. The JPEG2000 bit-stream (compressed image file) is then decoded, but decoding is halted after arithmetic decoding (see steps from bit-stream unorganization to arithmetic decoder in the middle part of Fig. 2). The data at this point is a set of quantized wavelet coefficients. Using these quantized wavelet coefficients, bit-planes for the wavelet coefficients can be constructed and used to embed secret data with BPCS steganography (see the top box of the right part in Fig. 2). The quantized wavelet coefficients modified by embedding are then subjected to JPEG2000 encoding again (see steps from arithmetic encoder to bit-stream organization in the right part of Fig. 2), which produces 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. 2 and follows the same process as the aforementioned one.

The data extraction procedure follows the dashed arrows in the middle part of Fig. 2. JPEG2000 decoding of the secret-data-embedded bit-stream starts from bit-stream unorganization and is halted after arithmetic decoding. At this point, extraction of secret data is carried out by the BPCS method using quantized wavelet coefficients.

## 5. EXPERIMENTAL RESULTS

The 3-D SPIHT-BPCS steganography algorithm was implemented and tested on two standard videos: "Claire" and "Diskus". They consists of 32 frames, each of which is 8 bit per pixel (bpp) gray image and $256 \times 256$ pixels in size. A four-level 3-D DWT with the Daubechies 9/7 filter was applied to videos. The number of bit-planes in the 3-D SPIHT compression was set to 11 and 12. Here $4 \times 4$ patch size was used as an embedding unit and random binary data was used as secret data. The complexity threshold $\alpha_0$ for embedding was set to 0.3.

Table 1 shows results of embedding experiments where degradation in video quality has not been perceived after embedding, and compression results without embedding for reference. The PSNR value in the table is the mean for total 32 frames. The least significant bit-plane and the two least significant bit-planes were used to embed data for the number of bit-planes 11 and 12, respectively. The average embedding rate ((embedded data size)/(compressed video file size)) for two videos was around 18% for 11 bit-planes, and 28% for 12 bit-planes.

Motion-JPEG2000-BPCS steganography was realized using JPEG2000-BPCS steganography and tested at similar compression rates to those by 3-D SPIHT-BPCS. Here an

**Table 1**. Results of embedding experiments using 3-D SPIHT-BPCS steganography

| video | # bit-planes | # bit-planes used for embedding | embedded data size (bytes) | compressed file size (bytes) | PSNR (dB) |
|---|---|---|---|---|---|
| Claire | 11 | - | - | 118175 | 47.1 |
| | 11 | 1 | 21802 | 123678 | 44.4 |
| | 12 | - | - | 249508 | 49.7 |
| | 12 | 2 | 69658 | 264709 | 45.3 |
| Diskus | 11 | - | - | 316656 | 44.2 |
| | 11 | 1 | 58424 | 326881 | 41.7 |
| | 12 | - | - | 540452 | 48.8 |
| | 12 | 2 | 173430 | 567980 | 41.6 |

**Table 2**. Results of embedding experiments using Motion-JPEG2000-BPCS steganography

| video | target bpp for each frame | # bit-planes used for embedding | embedded data size (bytes) | compressed file size (bytes) | PSNR (dB) |
|---|---|---|---|---|---|
| Claire | 0.5 | - | - | 131915 | 45.6 |
| | | 1 | 17322 | 166371 | 41.8 |
| | 1.0 | - | - | 261279 | 49.2 |
| | | 2 | 26306 | 274338 | 39.2 |
| Diskus | 1.2 | - | - | 314018 | 43.6 |
| | | 1 | 49982 | 395071 | 38.2 |
| | 2.0 | - | - | 522726 | 48.4 |
| | | 2 | 81396 | 560327 | 34.4 |

error correction scheme for JPEG2000-BPCS steganography[5] was not applied because of fair comparison with 3-D SPIHT-BPCS. The results are shown in Table 2. The results in Table 1 and 2 show that 3-D SPIHT-BPCS is superior to Motion-JPEG2000-BPCS with regard to embedding performance.

## 6. CONCLUSIONS

This paper has presented a large capacity steganography method applicable to compressed video which is invented based on BPCS steganography and wavelet-based video compression. 3-D SPIHT-BPCS steganography and Motion-JPEG2000-BPCS steganography have been presented, which are the integration of 3-D SPIHT video coding and BPCS steganography, and that of Motion-JPEG2000 and BPCS, respectively. The proposed 3-D SPIHT-BPCS steganography achieved embedding rates of around 28% of the compressed video size for twelve bit representation of wavelet coefficients with no noticeable degradation in video quality.

## 7. REFERENCES


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

[2] M. Niimi, H. Noda and E. Kawaguchi, "A steganography based on region segmentation by using complexity measure," *Trans. of IEICE*, **J81-D-II**, pp.1132-1140, 1998.

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

[4] J. Spaulding, H. Noda, M. N. Shirazi and E. Kawaguchi, "BPCS steganography using EZW lossy compressed images," *Pattern Recognition Letters*, **23**, pp.1579-1587, 2002.

[5] H. Noda, J. Spaulding, M. N. Shirazi, M. Niimi and E. Kawaguchi, "Bit-plane decomposition steganography combined with JPEG2000 compression," *Lecture Notes in Computer Science*, **2578**, pp.295-309, 2003.

[6] B. J. Kim, X. Zixiang and W. A. Pearlman, "Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT)," *IEEE Trans. Circuits and Systems for Video Technology*, **10**, pp.1374-1387, 2000.

[7] "Motion JPEG2000 Final Committee Draft 1.0," ISO/IEC 15444-3 (JPEG2000, Part3), 2001.

[8] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits and Systems for Video Technology*, **6**, pp.243-250, 1996.