A Low-complexity Lossless Image Compression for Small Spacescrafts' On-board Computers

Hieu V. Dang[1], Son C. Chu[2], Nhung T. T. Nguyen[1], and Can C. Chu[1]
[1] Department of Communications Engineering
University of Transport and Communications, Hanoi, Vietnam
[2] Faculty of Art, Computing, Engineering and Science
Sheffield Hallam University, Sheffield, United Kingdom
Emails: {h.v.dang@ieee.org|schucong@gmail.com|nhungoinhung@gmail.com|chucongcan.ati@gmail.com}

***Abstract*—In this paper, we present a novel and low-complexity lossless compression for gray-scale images. The gray-scale image is first separated into bit-planes. These bit-planes are then performed a binary wavelet transform (BWT) to obtain an efficient representation for compression. The BWT bits of significant bit-planes are then encoded by the run-length coder that uses Golomb-Rice codes for run-encoding. The experimental results show that the algorithm obtained efficiency in image compression, and low-complexity in implementation that is highly applicable for image compression systems on small spacecraft's on-board computers.**

***Index Terms*—Lossless image coding; bit-plane coding; binary wavelet transform; Golomb-Rice coding; small spacecrafts' on-board computers;.**

I. Introduction

Small satellites are recently increasing interest all over the world for their attractive applications. An advantage of small satellites is the fast and low-cost development, which makes them a suitable platform for evaluating and demonstrating rapid new technologies in space. Because of the limitation of available size, mass, and power of small satellites, demanding on high computational capabilities on small satellites is a challenge to engineers. The future need for small satellites is to process high resolution payload data (e.g., imaging payload), and complex control algorithms.

Satellite imaging payloads mostly operate a store-and-forward mechanism, whereby captured images are stored on board and transmitted to ground stations later on [1]. With the demanding of high spatial resolution imaging, space missions are faced with the complexity of processing and conveying an extensive amount of imaging datas. Thus image compression becomes an important procedure in payload processing on-board computers of small satellites. Image compression reimburses for the limited on-board computing resources, such as mass memory and downlink's bandwidth.

Image compression tries to exploit and remove redundancies in image to obtain a high compression ratio and an acceptable quality of the reconstructed image. There are different classes of redundancies in an image, such as spatial, statistical, and human vision redundancies. Image compression techniques can be classified into two classes, lossless and lossy image compressions. In lossless compression, the reconstructed image is identical to the original one (there is no loss of information). In an opposite manner, lossy image compression methods reconstruct the image with an information lost. Lossless image compression is required for applications that cannot tolerate any degradation of original image. For instance, satellite images or geographical map images, where it cannot be tolerated by distortion caused by compression techniques.

One approach for exploiting efficiently spatial correlations for compression is to decompose the image into a set of binary layers (bit-planes), and then compress these layers by a binary image compression technique [2], [3], [4], [5]. The decompression is an inverse process of the compression, where the compressed file is decompressed into a set of layers which are then combined back into the gray-scale image. The less significant bit-planes are typically difficult to predict the structures to be compressed well. This is because bit-plane separation destroys the gray-level correlations of the original image. Thus, single lossless coding methods such as dictionary-based, run-length codings are not efficient to encode these insignificant bit-planes.

Recently, wavelet transforms have been applied to reduce the entropy of the data source for lossless image compressions [6], [7], [8]. Multiresolution image representation with high coding efficiency are the most attractive attributes of wavelet-based coding methods. The wavelet transforms are almost for real-valued and complex-valued functions (i.e., the data to be analyzed, the basis functions, and the arithmetic operators are in the real or complex fields) [9]. These transforms have a degree of computational complexity. Swanson and Tewfik [9] have introduced the theory of *binary wavelet transform* (BWT) for binary images over the finite Galois field of order 2, GF(2). BWT shares many of the important characteristics of the real wavelet transform. Furthermore, this BWT has several distinct advantages over the real wavelet transforms, including: (i) the entire decomposition process is performed in GF(2), which means that the intermediate and transformed data produced by BWT are binary (This leads to no quantization effects introduced and the decomposition is completely invertible); (ii) the algorithm is extremely fast and much simpler since the data remains in GF(2) and the transform uses modulo-2 arithmetic operators which can be performed using simple Boolean operations.

In this paper, we present a lossless image compression using bit-plane coding, BWT, and run-length/Golomb-Rice codes. The gray-scale iamge is first decomposed into bit-planes by bit-plane separation methods. These bit-planes are then sequentially performed BWT. The binary wavelet bits are scanned and then applied run-length coding with Golomb-Rice codes to obtain the compressed bit-stream. The proposed method has some important advantages listed as follows.

1. It is a reversible coding method, in which the reconstructed image is identical to the original one.
2. The algorithm is extremely fast since all the data processing flow are in binary computation in the GF(2).
3. The algorithm is simple and applicable for the simple implementations on microprocessors.

The rest of the paper is organized as follows. Sec. II gives a background on bit-plane coding, BWT, and Golomb-Rice codes. Sec. III describes the proposed compression method. Sec. IV discusses experimental results. Finally, the paper ends with the conclusions.

II. BACKGROUNDS

A. Bit-plane Coding

Bit-plane coding is a technique whereby a group of bits is divided into subgroups so that some of the subgroups can be summarily described [2]. Since data is commonly stored in a binary format in most electronic computing devices, one natural approach to implement an embedded coding system is through sequential bit-plane coding, where the input data are sequentially scanned and coded by bit-planes, usually from the most significant to the least, to generate the compressed bit-stream [3].

The foremost step in bit-plane coding is data decomposition where data is decomposed into different bit-planes for later encoding. There are three common decomposition methods: (i) *binary-coded separation* (BCS); (ii) *gray-coded separation* (GCS); and (iii) *prediction-error separation* (PES). In this work, we consider the BCS and GCS for bit-plane coding.

The BCS is a straightforward bit-plane separation. With a gray-scale image, pixels have values varying from 0 to 255, which can be represented by 8 bits binary data from the most significant bit MSB to the least significant bit LSB. These binary data is separated into 8 bit-planes from MSB bit-plane which contains MSB bits of all pixels to LSB bit-plane which correspondingly contains LSB bits of pixels. The eight bit-planes of gray-scale image Cameraman with the size of 256-by-256 are displayed in Fig. 1. The main disadvantage of BPS is that pixels which differ by 1 or 2 in decimal values differ in many bit positions in binary values.

The second separation method is a gray-code separation, in which the pixel intensities are represented by gray codes so that the change of pixel value by +1 or -1 causes the change of only one bit in the corresponding bit-planes. The gray-codes can be drawn by converting a binary number to a gray number. Thus, the gray-coded separation can be implemented from the binary-coded separation as follows. The gray-scale pixels are first represented by binary codes. The binary codes are then converted into gray codes. The bit-planes are created as the same procedure of BCS that include eight bit-planes from MSB to LSB bit-planes. An example of gray-coded separation for the gray image Cameraman is shown in Fig. 1.

B. Binary Wavelet Transform

1) Binary Field Transform: A binary field, also called the Galois Field of order 2 (GF(2)), has only two symbols 0 and 1. Operations of addition, subtraction, multiplication, and division are defined over these two symbols only. Real field transforms can be applied to the binary field, but it is complex in computation. To overcome this difficulty, the *binary field transform* (BFT) have been proposed by Swanson and Tewfik [9]. For finite sequences, the BFT takes the form of a square symmetric matrix. The construction of the BFT matrix and its inverse is as follows.

Let us define

$$\boldsymbol{B}_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \tag{1}$$

and

$$\boldsymbol{B}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \tag{2}$$

For $N \geq 6$ and even, the BFT matrix $\boldsymbol{B}_N$ is constructed by

$$\boldsymbol{B}_N = \begin{bmatrix} \boldsymbol{B}_N^{ul} & \boldsymbol{B}_N^{ur} \\ \boldsymbol{B}_N^{ll} & \boldsymbol{B}_N^{lr} \end{bmatrix} \tag{3}$$

The upper-left submatrix with the size of $(N-2)$x$(N-2)$ is given by

$$\boldsymbol{B}_N^{ul} = \begin{bmatrix} 1\mathrm{s}_{2\mathrm{x}2} & 1\mathrm{s}_{2\mathrm{x}(N-4)} \\ 1\mathrm{s}_{(N-4)\mathrm{x}2} & \bar{\boldsymbol{B}}_{N-4} \end{bmatrix} \tag{4}$$

where $1\mathrm{s}_{N\mathrm{x}M}$ is an NxM matrix of number 1. Matrix $\bar{\boldsymbol{B}}_{N-4}$ is the result of applying the logical-not operation to each element of the BFT matrix $\boldsymbol{B}_{N-4}$.
The upper-right submatrix $\boldsymbol{B}_N^{ur}$ with the size of (N-2)x2 is defined by

$$\boldsymbol{B}_N^{ur} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \tag{5}$$

The lower-left submatrix $\boldsymbol{B}_N^{ll}$ with the size of 2x(N-2) is defined as the transpose of $\boldsymbol{B}_N^{ur}$, given by

$$\boldsymbol{B}_N^{ll} = \boldsymbol{B}_N^{ur} \tag{6}$$

For example, 8x8 BFT matrix is constructed as follow.

$$\boldsymbol{B}_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{7}$$

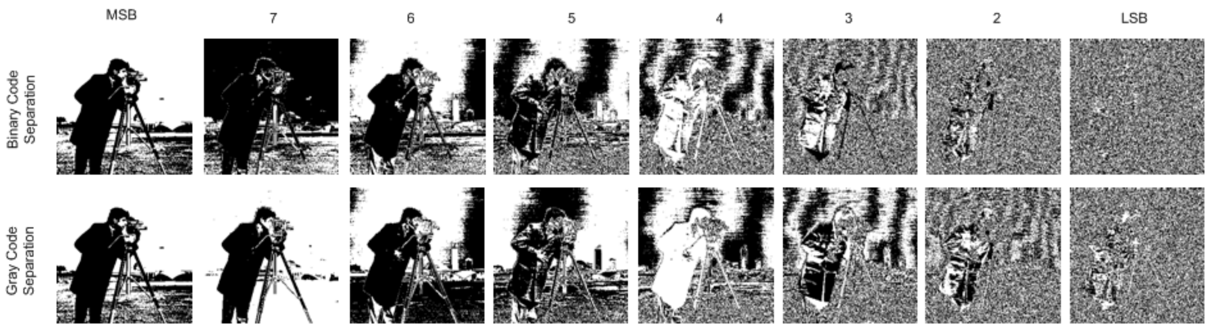

Fig. 1. Image deomposition into bit-planes for gray-scale image Cameraman.

Since det($\boldsymbol{B}_N$)=1 for all $N \geq 2$, $\boldsymbol{B}_N$ is invertible over the binary field. The inverse $\boldsymbol{B}_N^{-1}$ can be evaluated using a simple recursive formula for $N \geq 6$ as follows.

$$\boldsymbol{B}_N^{-1} = \begin{bmatrix} \boldsymbol{A}_{(N-2)\text{x}(N-2)} & \boldsymbol{C}_{(N-2)\text{x}2} \\ \boldsymbol{C}^T_{(N-2)\text{x}2} & \boldsymbol{D}_{2\text{x}2} \end{bmatrix} \tag{8}$$

where

$$\boldsymbol{A}_{(N-2)\text{x})(N-2)} = \left[\begin{array}{cc|cccccc} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 & 1 \\ 0 & 1 & & & & & \\ 0 & 0 & & & & & \\ \vdots & \vdots & & & B^{-1}_{N-4} & & \\ 0 & 0 & & & & & \\ 1 & 1 & & & & & \end{array} \right] \tag{9}$$

$$\boldsymbol{C}_{(N-2)\text{x}2} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \tag{10}$$

$$\boldsymbol{D}_{2\text{x}2} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{11}$$

For the 8x8 BFT, the inverse BFT matrix is

$$\boldsymbol{B}_8^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{12}$$

We can see that the constructions of both transform and inverse transform matrices do not require any matrix computations.

The BFT of vector $\mathbf{x}$ with the length of N is performed by applying $\boldsymbol{B}_N^{-1}$ to all circular shifts of $\mathbf{x}$. To compute the NxN matrix BFT $\hat{\boldsymbol{X}}$ of $\mathbf{x}$, we begin by forming the equivalent one-circulant matrix $\boldsymbol{X} = 1 - \text{circ}(\mathbf{x})$, then evaluate the matrix by matrix product by

$$\hat{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{B}_N^{-1} \tag{13}$$

The filter BFT (FBFT) of a vector $\mathbf{x}$ is performed by

$$\tilde{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{B}_N \tag{14}$$

2) Binary Wavelet Transform: The theory of BWT is developed from the principle of BFT and parallels with the theory of wavelet transform developed over the real field. The construction of two-band discrete orthonormal binary wavelets as equivalent to the design of a two-band perfect reconstruction filter bank with added vanishing moments as shown in Fig. 2. A filter bank is often cascaded with one or more additional filter banks to provide further resolutions of the input signal.

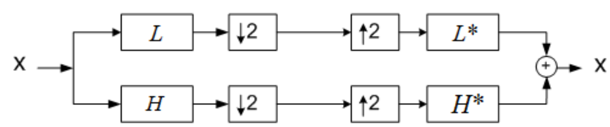

Fig. 2. Two-band perfect reconstruction filter bank.

To guarantee that the binary multiresolution decomposition inherits the important characteristics of the real wavelet decomposition, and still be able to reconstruct the original signal perfectly, the filters must satisfy three constraints: (i) the bandwidth constraint that restrict the bandwidths of the low-pass and high-pass filters to guarantee that no information is lost after the outputs of the two filters are downsampled by a factor of 2; (ii) the vanishing moment constraint to guarantee that the BWT of slowly varying binary sequences are very sparse; and (iii) the perfect reconstruction constraint to guarantee that the BWT is invertible. From the above three constraints, the binary filters for BWT are designed as follows.

Suppose the N-tap low-pass filter $\boldsymbol{l}$ and the N-tap high-pass filter $\boldsymbol{h}$ are formulated as $\boldsymbol{l} = \begin{bmatrix} l_0 & l_1 & \cdots & l_{N-1} \end{bmatrix}$ and $\boldsymbol{h} = \begin{bmatrix} h_0 & h_1 & \cdots & h_{N-1} \end{bmatrix}$. To satisfy the bandwidth, the vanishing moment, and the perfect reconstruction constraints, the filters $\boldsymbol{l}$ and $\boldsymbol{h}$ must follow the conditions given by [9]

$$\begin{cases} \displaystyle\sum_{i=0,even}^{N-2} l_i = 0; \text{ and } \sum_{i=1,odd}^{N-1} l_i = 1 \\ \displaystyle\sum_{i=0,even}^{N-2} h_i = 1; \text{ and } \sum_{i=1,odd}^{N-1} h_i = 1 \end{cases} \tag{15}$$

From the designed filters $\boldsymbol{l}$ and $\boldsymbol{h}$, we then formulate two-circulant matrices $\boldsymbol{L}_2 = 2 - \text{circ}(l)$ and $\boldsymbol{H}_2 = 2 - \text{circ}(h)$. The BWT matrix $\boldsymbol{T}$ is then setup by

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{L}_2 \\ \boldsymbol{H}_2 \end{bmatrix} \tag{16}$$

The 2-D BWT of an image $\boldsymbol{F}$ with the size NxN is performed by

$$\hat{\boldsymbol{F}} = \boldsymbol{T}\boldsymbol{F}\boldsymbol{T}^T \tag{17}$$

The transform in Eq. (17) corresponds to passing the image $\boldsymbol{F}$ through a low-pass 2-D separable filter and three band-pass 2-D separable filters, and decimating by 2 in each direction as shown in Fig. 3.

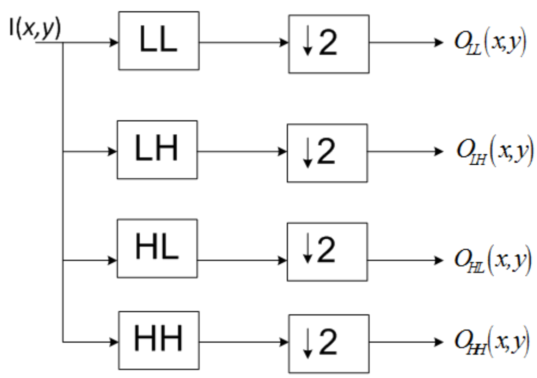

Fig. 3. Two-band 2-D perfect reconstruction filter banks.

The inverse BWT is given by

$$\tilde{\boldsymbol{F}} = \boldsymbol{T}^{-1}\hat{\boldsymbol{F}}\left(\boldsymbol{T}^T\right)^{-1} \tag{18}$$

Since the BWT is perfect reconstruction, $\tilde{\boldsymbol{F}}$ and $\boldsymbol{F}$ are identical. To obtain multiresolution decomposition, we can successively applying the decimated output of each LL filter as the input to the next stage.

C. Golomb-Rice Codes

Golomb-Rice coders have been applied widely in image compression systems [10]. Golomb-Rice coders are optimal or nearly optimal for integer sources with two-sided geometric distributions, which approximate quite closely the distribution of uniformly quantized Laplacian sources [11]. The main advantage of Golomb-Rice codes is that the output codewords are easily computed for the corresponding input symbols by changing a single integer parameter k, so that no explicit tables are actually required. This makes the computation of Golomb-Rice coders much faster than memory access based coding. The theory and implementation of Golomb-Rice coders are briefly introduced as follows.

For an integer number n, the principle of Golomb-Rice coder with parameter k is defined by the encoding rule in Fig. 4 [12], [13], [11].

The number n can be represented as a function of q and k by

$$n = q * 2^k + r = \lfloor \frac{n}{2^k} \rfloor * 2^k + r \tag{19}$$

Input Value	Output Codeword
n	$G(n,k) = \underbrace{11 1 \cdots 11}_{\substack{prefix,\\ q\ bits}} 0\ \underbrace{b_{k-1} b_{k-1} \cdots b_0}_{\substack{suffix,\\ k\ bits}}$

Fig. 4. Golomb-Rice coding principle with parameter k.

The prefix q bits is also called the quotient part which consists of q unary code bits. The remainder r is the fixed-length code bits, k LSB of the number n. For example, with $k = 3$, Golomb-Rice codes for number $n = 1, 4, 8, 11, 16, 20$ are shown in Table I.

TABLE I
EXAMPLE OF GOLOMB-RICE CODES WITH $k = 3$.

n	q	r	Codeword
1	0	1	0001
4	0	4	0100
8	1	0	10000
11	1	3	10011
16	2	0	110000
20	2	4	110100

III. PROPOSED ALGORITHMS

In this section, the proposed lossless compression of gray-scale image is presented. The proposed technique is described in Fig. 5.

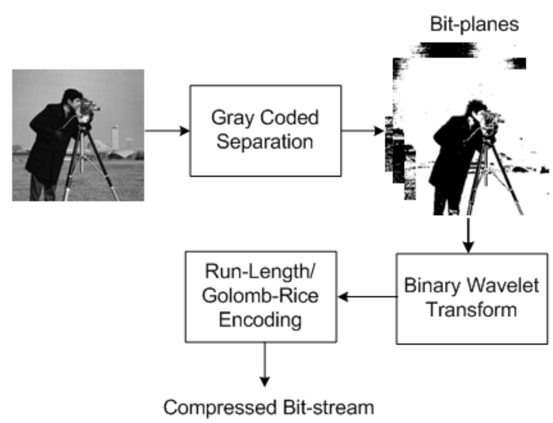

Fig. 5. The proposed lossless image compression.

The proposed algorithm is described as follows. The gray-scale image is first decomposed into bit-planes by using gray coded separation (GCS) method. The bit-planes are then decomposed in 3 levels by the binary wavelet transform (BWT) sequentially from the MSB bit-plane to the LSB bit-plane. The binary wavelet coefficients of each significant bit-planes (e.g., bit-planes MSB, 7, 6, and 5) are scanned to bit sequences for run-length/Golomb-Rice encoding with the parameter $k = 3$. The scanning procedure used in this work is shown in Fig. 7.

The decompression algorithm is the inverse procedure of the compression algorithm.

The scaling filters used for the 3-level BWT are the 8-tap low-pass filter $\boldsymbol{l} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$, and the 8-tap high-pass filter $\boldsymbol{h} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$. The BWT matrix $\boldsymbol{T}$ is then setup according to Eq. 16 by

$$\boldsymbol{T} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{20}$$

The 3-level BWT decomposition of the bit-plane MSB of image Cameraman decomposed by gray-coded separation in Fig. 1 is shown in Fig. 6.

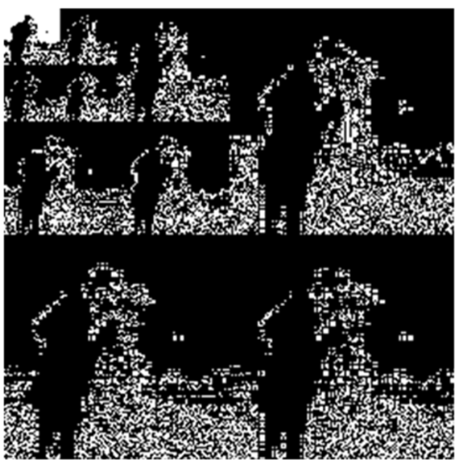

Fig. 6. 3-level decomposition of bit-plane MSB of image Cameraman using BWT.

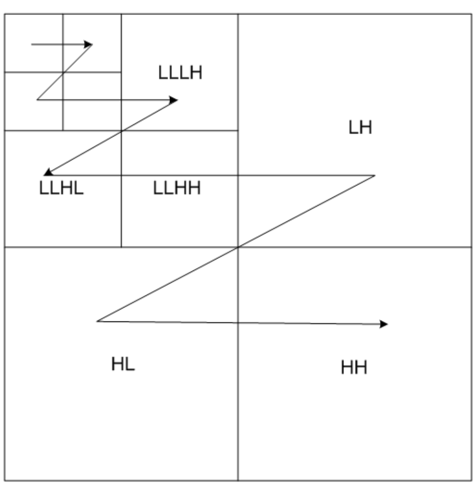

Fig. 7. Scanning procedure of binary wavelet coefficients for run-length/Golomb-Rice encoding.

IV. Experimental Results and Discussions

In this section, we use four different nature images shown in Fig. 8 to evaluate the effectiveness of the proposed coding algorithm. We first evaluate the compactness of the binary image representation resulting from the binary wavelet transform. We then evaluate the compression efficiency of the proposed compression.

Fig. 8. Test images: (a) Cameraman; (b) Clock; (c) Lena; (d) House.

To measure the compactness of the binary image representation in 3-level BWT, we use the entropy function calculated by

$$H(p) = -p\log_2(p) - (1-p)\log_2(1-p) \tag{21}$$

where p is the probability of pixel value 1 in the binary bit-plane images. The measure takes values between 0 and 1. If the entropy is small, it indicates more efficient compression representation, and vice versus. The entropy results for four different input images are displayed in Tables II, III, IV, and V.

TABLE II
ENTROPY RESULTS FOR INPUT IMAGE 'CAMERAMAN'.

Bit-planes	Original Entropy	BWT Entropy
MSB	0.9732	0.6784
7	0.8305	0.4898
6	0.9845	0.7502
5	0.9999	0.8811
4	0.9820	0.9093
3	0.9998	0.9515
2	1	0.9998
1	1	1

From the entropy results in Tables II, III, IV, and V, it can be seen that the entropy is significantly reduced at bit-planes MSB, 7, 6, and 5 after applying 3-level BWT. It indicates

TABLE III
ENTROPY RESULTS FOR INPUT IMAGE 'CLOCK'.

Bit-planes	Original Entropy	BWT Entropy
MSB	0.7061	0.4645
7	0.8616	0.5208
6	0.9964	0.7159
5	0.8245	0.7672
4	0.9569	0.8782
3	0.9820	0.9584
2	0.9954	0.9909
1	1	0.9995

TABLE IV
ENTROPY RESULTS FOR INPUT IMAGE 'LENA'.

Bit-planes	Original Entropy	BWT Entropy
MSB	0.9994	0.6461
7	0.7653	0.6607
6	0.9988	0.7501
5	0.9947	0.8920
4	0.9999	0.9710
3	1	0.9952
2	1	0.9999
1	1	1

that the 3-level BWT is an efficient representation for the compression of significant bit-planes. At the lower bit-planes, the increases of BWT entropy imply that the BWT become less efficient for the less significant bit-planes.

To evaluate the compression efficiency, we adopt the *compression ratio* (CR) and the *reduction percentage* (RP) given by

$$CR = \frac{V}{V^*} \tag{22}$$

$$RP = \frac{V - V^*}{V} 100\% \tag{23}$$

where V is the data volume of the original signal in binary bits, and V^* is the data volume of the encoded signal (i.e., length of the compressed bit-stream). The CR and RP results for the compressions of four different test images are described in Table VI.

The results in Table VI show that the proposed algorithm obtains a moderate-well compression effectiveness in lossless compression landscape.

V. CONCLUSIONS

We have studied efficient and low-complexity lossless compression methods in binary domain. The theory of binary wavelet transform with all computation in GF(2) is simple and low-cost in hardware implementation. The Golomb-Rice coder is also studied for fast and low-cost implementation.

The proposed lossless compression for gray-scale image is presented. The experimental results show that the algorithm obtains good efficiency in compression and less complexity in implementation. The algorithm can be generalized for lossless compression of any real data and can be implemented on a small spacecraft's on-board computer.

TABLE V
ENTROPY RESULTS FOR INPUT IMAGE 'HOUSE'.

Bit-planes	Original Entropy	BWT Entropy
MSB	0.9994	0.7287
7	0.7803	0.5514
6	0.9949	0.5675
5	0.8660	0.7628
4	0.9794	0.9246
3	0.9966	0.9683
2	0.9999	0.9993
1	1	0.9998

TABLE VI
CR AND RP RESULTS.

Test Images	CR	RP
Cameraman	1.2441	19.62 %
Clock	1.3218	24.34 %
Lena	1.2285	18.60 %
House	1.4058	28.87 %

REFERENCES

[1] Guoxia Yu, Tanya Vladimirova, and Martin N. Sweeting, "Image compression systems on board satellites," *Acta Astronautica*, vol. 64, no. 9, pp. 988–1005, May 2009.

[2] Jay W. Schwartz and Richard C. Bakker, "Bit-plane encoding: a technique for souce encoding," *IEEE Trans. Aero. Elect. Syst.*, vol, AES-2, no. 4, pp. 385–392, July 1966.

[3] R. Yu, C. Ko, S. Rahardja, and X. Lin, "Bit-plane Golomb coding for sources with Laplacian distribitions," in *Proceedings of Acoustics, Speech, and Signal Processing, ICASSSP' 03*, vol. 4, pp. 277-280, April 2003.

[4] Alexey Podlasove and Pasi Franti, "Lossless image compression via bit-plane separation and multilayer context tree modeling," *Journal of Electronic Imaging*, vol. 15, no. 4, pp. 1–11, Dec. 2006.

[5] Chris Dunn, "Scalable bitplane runlength coding," in *Audio Engineering Society Convention 120*, May. 2006.

[6] A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy compression," *IEEE Trans. Image Process.*, vol. 5, no. 9, pp. 1303–1310, Sept. 1996.

[7] K. W. Cheung and L. M. Po, "Spatial coefficient partitioning for lossless wavelet image coding," in *IEE Proceedings of Vision, Image and Signal Processing*, vol. 149, no. 6, pp. 365–369, Jan. 2003.

[8] Pen-Shu Yeh, Philippe Armbruster, Aaron Kiely, Bart Masschelein, Gilles Moury, Christoph Schaefer, and Carole Thiebaut, "The new CCSDS image compression recommendation," in *Proceedings of Aerospace Conference*, pp. 4138 – 4145, March 2005.

[9] Mitchell D. Swanson and Ahmed H. Tewfik, "A binary wavelet decomposition of binary images," *IEEE Trans. Image Process.*, vol. 5, no. 12, pp. 1637– 1650, Dec. 1996.

[10] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Process.* vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[11] H. S. Malvar, "Adaptive run-length/Golomb-Rice encoding of quantized generalized Gaussian sources with unknown statistics," in *Proceedings of Data Compression Conference, DCC 2006*, pp. 23–32, March 2006.

[12] Solomon W. Golomb, "Run-length encodings", *IEEE Trans. Infor. Theory*, vol. 12, no. 3, pp. 399–401, Nov. 1966.

[13] Robert F. Rice and James R. Plaunt, "Adaptive variable-length for efficient compression of spacecraft television data," *IEEE Trans. Comm. Tech.*, vol. 19, no. 6, pp. 889–897, Dec. 1971.