Research Article

Performance Evaluation of Data Compression Systems Applied to Satellite Imagery

Lilian N. Faria,[1] Leila M. G. Fonseca,[1] and Max H. M. Costa[2]

[1] *Image Processing Division, National Institute for Space Research (INPE), 12227-001 São José dos Campos, SP, Brazil*
[2] *School of Electrical and Computer Engineering, University of Campinas (Unicamp), 13083-852 Campinas, SP, Brazil*

Correspondence should be addressed to Lilian N. Faria, lilian@dpi.inpe.br

Onboard image compression systems reduce the data storage and downlink bandwidth requirements in space missions. This paper presents an overview and evaluation of some compression algorithms suitable for remote sensing applications. Prediction-based compression systems, such as DPCM and JPEG-LS, and transform-based compression systems, such as CCSDS-IDC and JPEG-XR, were tested over twenty multispectral (5-band) images from CCD optical sensor of the CBERS-2B satellite. Performance evaluation of these algorithms was conducted using both quantitative rate-distortion measurements and subjective image quality analysis. The PSNR, MSSIM, and compression ratio results plotted in charts and the SSIM maps are used for comparison of quantitative performance. Broadly speaking, the lossless JPEG-LS outperforms other lossless compression schemes, and, for lossy compression, JPEG-XR can provide lower bit rate and better tradeoff between compression ratio and image quality.

1. Introduction

High-resolution cameras onboard remote sensing satellites generate data at a rate on the order of hundreds of Mbits/s. Data compression [1] is used in many space missions to reduce onboard storage and telemetry bandwidth requirements. Particularly in remote sensing applications, in which data are acquired at high cost and the information contained in the image data is important for scientific exploration, a lossless or near-lossless compression is desirable to preserve image data.

In 1988, the Brazilian National Institute for Space Research (INPE) and the China Academy of Space Technology (CAST) signed a cooperation agreement for the development of remote sensing satellites, known as CBERS (China-Brazil Earth Resources Satellite) [2]. Due to the success of CBERS 1 and 2 and CBERS-2B, the cooperation was expanded to include the satellites CBERS 3 and CBERS 4.

The CBERS-2B satellite, launched in 2007, carried onboard three cameras—WFI (Wide Field Imager), CCD (CCD medium resolution camera), and HRC (High-Resolution Camera) [3]. The new satellites CBERS 3 and 4 will carry onboard four cameras—WFI, IRMSS (Infrared Multispectral Scanner), PANMUX (Panchromatic and Multispectral

cameras), and MUXCAM (Multispectral Camera) [4]. Only the HRC and PANMUX cameras onboard CBERS-2B and CBERS 3 and 4, respectively, have coders developed by CAST in China.

Given the importance of compression in space missions, a working group at INPE has evaluated some compression systems. The objective is to select an appropriate compression scheme to be implemented in FPGA hardware to meet the minimum requirements of compression ratio and PSNR (peak signal-to-noise ratio) of 4 and 50 dB, respectively. Within this context, we have studied some compression algorithms, and conducted their performance evaluation using quantitative rate-distortion measurements.

To compare the algorithms we assembled a dataset with a hundred of test images acquired by CCD camera (Table 2) onboard CBERS-2B (CBERS-2B CCD). This dataset is representative of a variety of content relevant to remote sensing applications including agriculture, forest, urban areas, and surface water with different cloud cover.

The selected compression system must meet certain requirements for real-time hardware compression onboard a spacecraft, such as nonframe (push-broom) data processing, a high decoded image quality, and packet loss effects that

are limited to a small region of the image. In particular, the algorithm complexity must be sufficiently low to make high-speed hardware implementation feasible.

In this paper, we present the rate-distortion comparison of some compression systems suitable for remote sensing images such as DPCM, JPEG-LS, CCSDS-IDC, and JPEG-XR. The paper is structured as follows. In Section 2, we show an overview of image compression systems. Prediction- and transform-based compression systems are described in Sections 3 and 4, respectively. The performance results and analysis are presented in Section 5, and the conclusions appear in Section 6.

2. Image Compression Systems

According to Gonzalez and Woods [5], data compression refers to the process of reducing the amount of data required to represent a given quantity of information. Image compression schemes are divided into two broad categories: *lossless* and *lossy*. Lossless image compression allows an image to be compressed and decompressed without any information loss. In lossy compression schemes, some controlled loss of data is tolerated. Although it is impossible to reconstruct the original image using lossy image compression, it provides a much higher compression ratio than lossless compression.

Digital images generally contain a significant amount of redundancy, thus image compression techniques take advantage of these redundancies to reduce the number of bits required to represent the image. There are two main kinds of data redundancy on digital images: spatial redundancy and coding redundancy [6].

(a) *Spatial Redundancy.* Means that, due to the interpixel correlations within the image, the value of any pixel can be partially predicted from the value of its neighbors. To reduce the spatial redundancy, the image is usually modified into a more efficient format using spatial decorrelation methods, such as prediction or transforms.

(b) *Coding Redundancy.* Utilizes the probability distribution associated with the occurrence of the symbols. To reduce coding redundancy, a variable-length coding assigns the shortest codewords to the most frequently occurring symbols and longer codewords to low-probability symbols. This process is also called *entropy coding*. The main entropy coding schemes are arithmetic coding [7], Huffman [8], and Golomb coding [9].

Generally, a compression system model consists of two distinct structural blocks: an *encoder* and a *decoder*. The encoder creates a codestream from the original input data. After transmission over the channel, the decoder generates a reconstructed output data.

A typical model of an encoder consists of three functional modules, as depicted in Figure 1: a *prediction* module (for prediction-based compression systems) or a *forward transform* module (for transform-based compression systems) that performs the spatial decorrelation, a *quantization* module that reduces the dynamic range of the errors, and

an *entropy encoding* module that reduces the coding redundancy. When lossless compression is desired, the quantization step is omitted because it is an irreversible operation.

Basically, the decoder consists of two functional modules: an *entropy decoding* and an *inverse prediction or transform*. The quantization step results in irreversible information loss, and reconstruction of quantized data is based on the midpoints of each quantization interval.

Different schemes can be used for spatial decorrelation. We consider schemes based on prediction and on transforms. Prediction techniques are used to predict the current pixel value from the values of neighboring pixels. The prediction-based compression methods include DPCM (Differential Pulse Code Modulation) [10], lossless JPEG (Joint Photographic Experts Group) [11], and JPEG-LS [12]. Transform-based systems perform a mapping of the image from the spatial (pixel) domain into a rotated system of coordinates in signal space by applying transforms, such as DCT (discrete cosine transform) and DWT (discrete wavelet transform). The baseline JPEG standard [11] is a DCT-based compression scheme. DWT-based compression schemes include JPEG2000 [13], ICER [14, 15], and CCSDS-IDC (Consultative Committee for Space Data Systems-Image Data Compression) [16, 17]. The new standard JPEG-XR (JPEG Extended Range) uses another transform [18–20].

As reported by Yu et al. [6], prediction and DCT-based compression are the methods most commonly used onboard satellites. Although prediction-based methods have a low compression ratio, they are still popular in space missions due to their efficacy at achieving lossless data compression and the low complexity of the algorithm. Even though lossy DCT-based compression methods have undesirable block artificial effects, they have been used for long time. However, in recent years, DWT-based compression schemes are being more used in space missions because they can provide higher image quality.

We evaluated the performances of some image compression systems suitable for space missions, such as DPCM, JPEG-LS, CCSDS-IDC, and JPEG-XR. The considered compression methods are one or bidimensional; notwithstanding, the considered data sets are multispectral. The details of these prediction- and transform-based compression systems are introduced in Sections 3 and 4, respectively.

All the studies were performed considering that the selected compression scheme is going to be developed in hardware and to be used onboard satellites. Thus, the evaluation must consider practical aspects of the systems, such as processing complexity (speed), and ease of implementation in FPGA hardware.

3. Prediction-Based Compression Systems

This section reviews the two prediction-based compression techniques evaluated in this paper: DPCM and JPEG-LS. The basic idea behind these schemes is to predict the value of a pixel based on the correlation between certain neighboring pixel values, using certain prediction coefficients. The number of pixels used in the prediction is called the order of the predictor. The difference between the predicted value and the
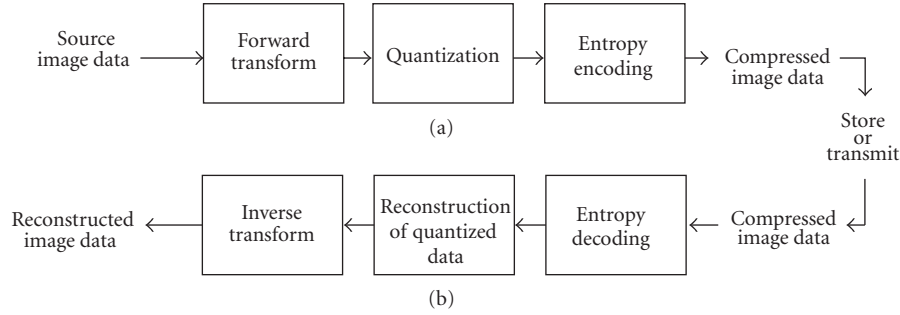
FIGURE 1: A general block diagram of image compression systems: (a) encoder and (b) decoder.
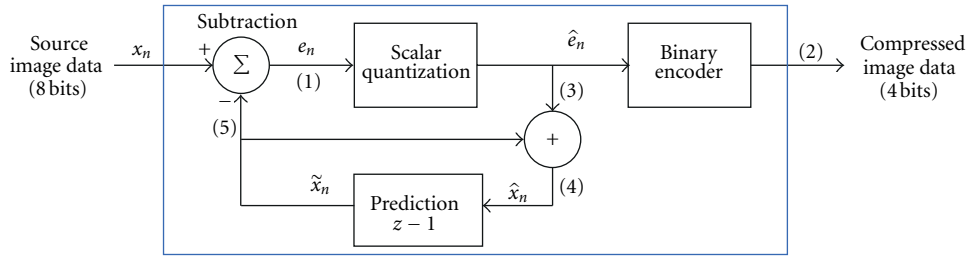

FIGURE 2: A diagram of the DPCM encoding method [21].

actual value of the pixels gives the difference (residual) image, which is much less spatially correlated than the original image. The difference image is then quantized and encoded. The basic function of the quantization is to map a large range of values onto a relatively smaller set of values.

Predictive methods do not require much storage and present a good tradeoff between complexity and efficiency. The main drawback of prediction methods is their susceptibility to error propagation.

3.1. DPCM. Differential Pulse Code Modulation (DPCM), the most common approach to predictive coding, offers the advantages of computational simplicity and ease of parallel implementation in hardware [10].

In this work, we evaluated a particular lossy DPCM compression method used by CAST/China in the PANMUX camera onboard CBERS 3 and 4 satellites [21]. The encoding process of this first-order predictor has five steps: subtraction, scalar quantization, binary encoding, summation, and prediction, as shown in Figure 2. Lookup tables are used in this particular one-dimensional DPCM algorithm in the prediction, quantization, and binary encoding, as shown in Figure 3, simplifying implementation. This is a lossy algorithm with fixed compression ratio of 2, because the quantization introduces error and lowers the bit rate from eight bits to four bits wide.

To ensure that the predictions at both the encoder and decoder are identical, the encoder uses the preceding reconstructed pixel value to predict the next one; that is, $\tilde{x}_n = \rho \cdot \hat{x}_{n-1}$, where ρ is the prediction coefficient equal or smaller than one and reduces the diffusion of error code. This particular algorithm actually turns the multiplication function into a lookup table (data in the lookup table makes

FIGURE 3: Lookup tables of the DPCM encoding implemented by CAST [21]: (1) prediction error, (2) binary code of the prediction error, (3) quantized prediction error, (4) preceding reconstructed prediction value, and (5) predicted pixel value using the predictive encoding lookup table.

up a common subtraction function). The prediction error results are rounded up into integer and then encoded.

The typical image distortions caused by lossy DPCM coding are granular noise (random fluctuations in the flat areas of the picture), slope overload (blurring in the high-contrast edges), and Moiré patterns (in periodic structures).

3.2. JPEG-LS. JPEG-LS [12] provides simple, efficient, and low-complexity lossless and near-lossless image compression. The core of JPEG-LS is based on the LOCO-I (Low Complexity Lossless Compression for Images) algorithm [22], which relies on prediction, error modeling, and context-based encoding of the errors. In the near-lossless mode, the maximum absolute error can be controlled by the encoder. Basically, JPEG-LS consists of two independent stages called *modeling* and *encoding*. The main procedures for

the lossless and near-lossless encoding processes are shown in Figure 4.

The modeling approach is based on the notion of "context." In context modeling, each sample value is conditioned on a small number of neighboring samples. The template used for context modeling and prediction is shown in Figure 4. The context is determined from four neighborhood-reconstructed samples at positions a, b, c, and d. From the values of the reconstructed samples at a, b, c, and d, the context first determines if the information in the sample x should be encoded in the *regular* or *run mode*. The run mode is selected when the context estimates that successive samples are likely to be either identical (for lossless coding) or nearly identical within the tolerances required (for near-lossless coding); otherwise, the regular mode is used [12].

Regular Mode: Prediction and Error Encoding. In the regular mode, the context determination procedure is followed by a *prediction* procedure (decorrelation) and *error encoding*. The predictor combines the reconstructed values of the three neighborhood samples at positions a, b, and c to predict the sample at position x.

(a) *Prediction.* The prediction approach is a variation of median adaptive prediction, in which the predicted value is the median of the a, b, and c pixels. In the LOCO-I algorithm, primitive detection of horizontal or vertical edges is achieved by examining the pixels neighboring the current pixel. The pixel x is predicted according to the following algorithm: the b pixel is used in cases were a vertical edge exists left of x, the a pixel is used in cases of a horizontal edge above x, and $a + b - c$ is used if no edge is detected. This simple predictor is called the median edge detection (MED) predictor or LOCO-I predictor [22]. The initial prediction is then refined using the average value of the prediction error in that particular context.

(b) *Error Encoding.* The prediction error is computed as the difference between the actual sample value at position x and its predicted value. This prediction error is then corrected by a context-dependent term to compensate for biases in prediction. In the case of near-lossless coding, the prediction error is quantized. The prediction errors are then encoded using a procedure derived from Golomb coding [9] that is optimal for sequences with a geometric distribution. The context modeling procedure determines a probability distribution that is used to encode the prediction errors. During the encoding process, shorter codes are assigned to the more probable events.

Run Mode. If the reconstructed values of the samples at a, b, c, and d are identical (for lossless coding) or if the differences between them are within set bounds (for near-lossless coding), the context modeling procedure selects the run mode and the encoding process skips the prediction and error encoding procedures. In the run mode, a sequence of consecutive samples with identical values (or values within
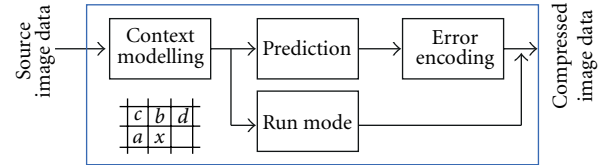

FIGURE 4: A simplified diagram of the JPEG-LS encoder [12].

the specified bound in the case of near-lossless coding) is encoded.

4. Transform-Based Compression Systems

Transform-based compression systems are based on the insight that the decorrelated coefficients of a transform can be coded more efficiently than the original image pixels. The transform typically results in some energy compaction (i.e., an energy redistribution of the original image into a smaller set of coefficients). Even though the energy is compacted into fewer coefficients, the total energy is conserved, resulting in a significant number of coefficients with values of zero or near zero.

Several kinds of transforms, with different efficiencies, energy compacting methods, and computational complexities, are useful in compression systems. The most common data compression transforms are DCT (discrete cosine transform) and DWT (discrete wavelet transform).

4.1. DCT-Based Compression System. The JPEG (Joint Photographic Experts Group) committee published the JPEG standard (ITU-T T.81 | ISO/IEC 10918-1) in 1992 [23]. The JPEG baseline, a typical DCT compression technique, has been widely used for digital imaging, including digital photography and images on the Internet. JPEG and other DCT-based compression techniques have been employed in many space missions, as discussed in [6].

JPEG is a standard for continuous tone still images that allows lossy and lossless coding. Lossless JPEG [11] is an independent predictive coding compression technique that includes differential coding, run length, and Huffman code. Lossless JPEG is not largely used in space mission due to the low compression ratio.

There are several modes defined for JPEG, including baseline, progressive, and hierarchical. The baseline mode, which supports only lossy compression using the DCT, is the most popular. The process flow is shown in Figure 5. The JPEG-baseline encoder starts with an 8×8 block-based DCT, quantization, zigzag ordering, and entropy coding using Huffman tables. A quality factor is set using quantization tables. When quality adjustments are applied to an image, block artifacts are induced by the encoding process and serious quality degradation becomes evident.

4.2. DWT-Based Compression System. The wavelet transform decomposes the original image into a sum of spatially and frequency localized functions, in a way that is similar
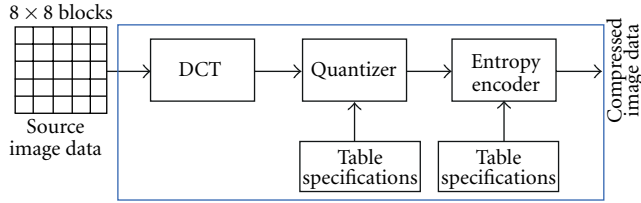
FIGURE 5: A block diagram of the JPEG-baseline encoder [23].

FIGURE 6: A general diagram of the CCSDS-IDC encoder [16].

to subband decomposition. The most important visual information tends to be concentrated into a reduced number of components (coefficients); therefore, the remaining coefficients can be quantized coarsely or truncated to zero with little image distortion. Compression methods based on wavelets avoid the block artifacts that occur in compression methods based on DCT. This is one of the reasons why wavelet-based compression schemes tend to produce superior image quality.

The CCSDS (Consultative Committee for Space Data Systems) is an international group dedicated to providing technical solutions to common problems faced by member space agencies, such as NASA (National Aeronautics and Space Administration), CAST/China, and INPE/Brazil. Two compression algorithms have been proposed by CCSDS. The first method is CCSDS Lossless Data Compression (CCSDS-LDC), which has been widely used in many missions [24, 25]. The second is the CCSDS Image Data Compression (CCSDS-IDC) algorithm.

The CCSDS-IDC [16, 17] is a new image compression recommendation suitable for space applications, which was established in 2005. The compression technique described in this recommendation can be used to produce lossy and lossless compression.

The CCSDS-IDC compressor consists of two main functional parts: a *discrete wavelet transform* (DWT) module that performs the decomposition of image data and a *bit-plane encoder* (BPE) that encodes the transformed data, as shown in Figure 6. This architecture is similar to the JPEG2000 structure, but it differs from the JPEG2000 in certain aspects: (a) it specifically targets the high-rate instruments used onboard in space missions; (b) a tradeoff has been performed between compression performance and complexity; (c) the lower complexity supports a low-power hardware implementation; (d) it has a limited set of options. According to the literature [17, 26], CCSDS-IDC can achieve performance similar to JPEG2000.

The algorithm decomposes the image with a three-level, two-dimensional, separable DWT, as shown in Figure 7(a). Two wavelets are specified with the recommendation: the 9/7 biorthogonal DWT, referred to as the *float DWT*, and a nonlinear integer approximation to this transform, referred to as the *integer DWT*. The float DWT cannot provide lossless compression; therefore, the integer DWT must be used in applications that require perfect image reconstruction. The integer DWT requires only integer arithmetic, while the float DWT requires floating-point calculations. Thus, the integer DWT may be preferable in some applications for complexity
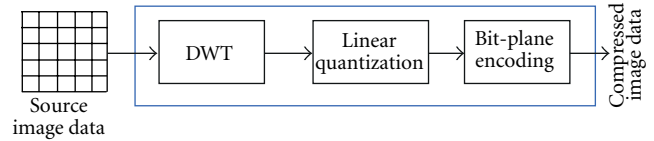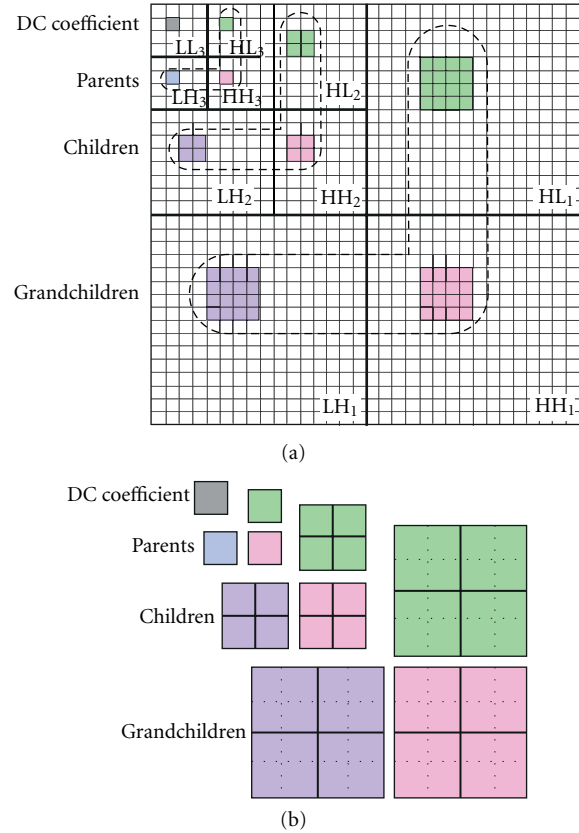

FIGURE 7: (a) A three-level, two-dimensional DWT decomposition of an image; (b) a block of 64 coefficients [17].

reasons. At low bit rates, the float DWT often provides better compression efficacy than the integer DWT [16, 17].

The BPE processes wavelet coefficients in groups of 64 coefficients referred to as a block. As shown in Figure 7(b), each block consists of a single coefficient from the lowest spatial frequency subband (referred to as the DC coefficient) and 63 AC coefficients. A segment is defined as a group of S consecutive blocks. The BPE encodes the DWT coefficients segment by segment, and each segment is coded independently of the others.

4.3. Other Transform Models.
JPEG-XR (ITU-T T.832 | ISO/IEC 29199-2) is the newest image coding standard from the JPEG committee [18, 19]. It is a block-based image coder that is similar to the traditional image-coding paradigm and that includes transformation and coefficient-encoding stages. JPEG-XR employs a reversible integer-to-integer mapping, called *lapped biorthogonal transform* (LBT),

as its decorrelation tool [20]. The main operations of JPEG-XR are transform, scalar quantization, and entropy coding, as shown in Figure 8.

JPEG-XR begins with a transform stage that maps the pixel information from the spatial domain to the frequency domain. Then, a quantization stage divides each coefficient by some integer value, rounding to the nearest integer. For lossless transforms, a quantization parameter of 0 will not affect the transformed coefficients. Next, the transform coefficients are scanned in order of increasing frequency and (finally) entropy coded.

The lifting-based reversible hierarchical lapped biorthogonal transform (LBT) converts the image data from the spatial domain to a frequency domain representation. The transform requires only a small number of integer processing operations for both encoding and decoding. It is exactly invertible in integer arithmetic and hence supports lossless image representation [20].

The two-stage transform is based on two basic operators: the *photo core transform* (PCT) operator and the optional *photo overlap filtering* (POT) operator, as shown in Figure 9. The core transform is similar to the widely used discrete cosine transform (DCT) and can exploit spatial correlations within a block-shaped region. The overlap filtering is designed to exploit the correlation across block boundaries and to mitigate blocking artifacts. It can be switched on or off by the encoder.

The smallest element of an image is a pixel. Each 4×4 set of pixels is grouped into a block. Then, each 4×4 set of blocks is grouped into a macroblock. A set of macroblocks can then be grouped into a tile, although the number of macroblocks included along the width and height may vary between tiles. At the highest level of the hierarchy, the tiles come together to form the complete image. An illustrative example of this partitioning is shown in Figure 10.

The image data is represented in the frequency domain, and the transform coefficients associated with each macroblock are split into three frequency bands: DC coefficients, lowpass coefficients, and highpass coefficients. According to this hierarchy, each macroblock contains 256 transform coefficients: one is DC, 15 are lowpass, and 240 are highpass. This hierarchical division supports direct decompression at three different resolutions.

The bitstream of a JPEG-XR compressed image is structured as shown in Figure 11. The image may be compressed into either the spatial or the frequency format. The spatial structure stores the coefficients of each macroblock together and stores the macroblocks sequentially in raster scan order (left to right, then top to bottom). The frequency structure groups the coefficients according to frequency band. Each band of coefficients is stored in raster scan order.

To enable the optimization of the quantization, JPEG-XR uses a flexible coefficient quantization approach that is controlled by *quantization parameters* (QPs). Adaptive coefficient scanning is used to convert the two-dimensional array of transform coefficients within a block into a one-dimensional vector to be encoded. Finally, the transform coefficients are entropy encoded. For this purpose, a variable-length coding (VLC) look-up table approach is
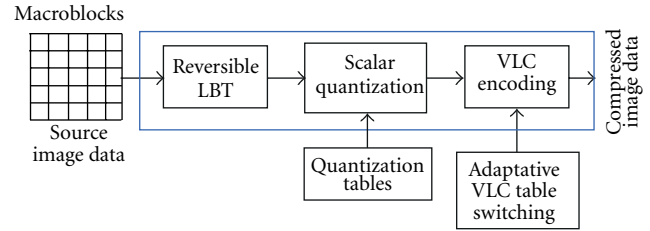

Figure 8: A block diagram of the JPEG-XR encoder [18].
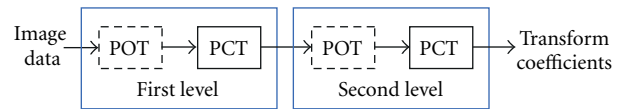

Figure 9: Two levels of the lapped biorthogonal transform [18].

used; a VLC table is selected from a small set of fixed predefined tables, with the table being selected adaptively based on the local statistics.

JPEG-XR supports a wide range of color formats, including *n*-channel encodings using fixed and floating point numerical representations; the bit depth varieties allow for a wide range of data compression scenarios.

5. Performance Comparison

5.1. Reference Softwares. In this work, we used reference C++ software implementations as listed in Table 1. The DPCM software was implemented by the Brazilian company AMS Kepler, which is based on the documentation provided by CAST (China) [21]. The JPEG-LS Reference Encoder—v.1.00 was originally developed by Hewlett-Packard Laboratories [27]. The CCSDS-IDC reference software [28] described in [16] was developed by the University of Nebraska, Lincoln. A reference software implementation of JPEG-XR has been published as the ITU-T Recommendation T.835 and ISO/IEC International Standard 29199-5 [29].

The C++ source codes have been slightly modified to support the monochromatic raw image format. Certain encoding parameters must be specified in the command line. The selection of compression options and parameters affects the compression efficacy and implementation complexity. Naturally, two different sets of compression parameters yield different rate-distortion results for the same test image. An appropriate selection of compression parameters must be determined for each application.

The main JPEG-LS parameter is *Near* (the maximum absolute error), which takes values of 0 (for the lossless mode) or 1, 2, 3, and 4 (for the near-lossless mode). The CCSDS-IDC method uses the following main parameters: *BitsPerPixel* (Bpp) (the desired bit rate in bits/pixel), which takes values of 0.25, 0.5, 1, 2, or 4; the *S* value, the number of blocks per segment; the *TypeDWT*, which takes values of 0 (for the float DWT) or 1 (for the integer DWT). JPEG-XR uses the following main parameters: *Mode*, which takes values of 0 (for All coefficients), 1 (for NoFlexbits), 2 (for
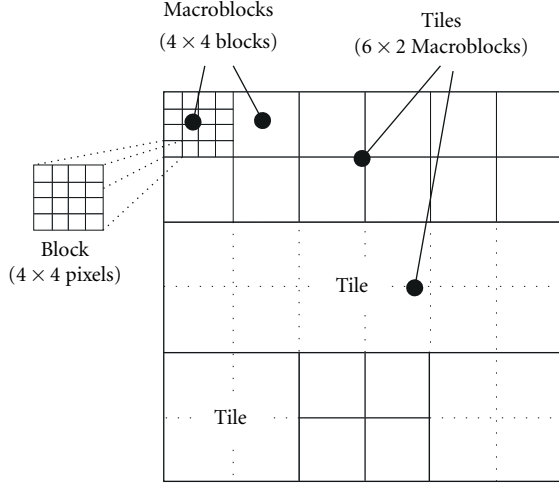
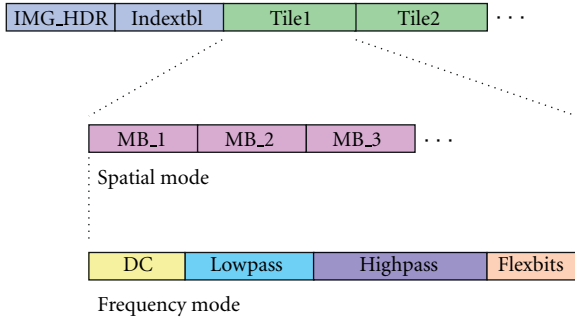FIGURE 10: An overview of image partitioning.

FIGURE 11: JPEG-XR bitstream structure [18].

TABLE 1: Reference software.

Algorithm	Software developer
DPCM	AMS Kepler (http://www.amskepler.com/)
JPEG-LS	HP Labs
CCSDS	University of Nebraska, Lincoln
JPEG-XR	Microsoft (ITU-ISO-IEC)

NoHighpass), and 3 (for DC-Only); the *Quantization Parameter* (QP), which can take values from 0 (no quantization) up to 255. To get lower complexity, the JPEG-XR method was tested with no overlap filter (POT).

5.2. Metrics. We considered compression ratio and distortion measures to compare the various compression algorithms. Although the processing time is also very important, we did not consider it in this work because it is highly dependent on the processing engine and the implemented algorithm.

Compression ratio (CR) is defined as the number of bits in the original image divided by the number of bits used to represent the compressed image.

In lossy image compression, one common reproduction distortion measure is the *mean squared error* (MSE), defined as

$$\text{MSE} = \frac{1}{w \cdot h} \sum_i \sum_j \left(x_{i,j} - \hat{x}_{i,j} \right)^2, \tag{1}$$

where $x_{i,j}$ and $\hat{x}_{i,j}$ are the original and reconstructed pixel values in the ith row and jth column and w and h denote the image width and height, respectively. More commonly, (objective) image quality is evaluated in terms of *peak signal-to-noise ratio* (PSNR), measured in dB and defined as

$$\text{PSNR} = 10 \log_{10} \frac{(2^B - 1)^2}{(\text{MSE} + 1/12)}, \tag{2}$$

where B is the dynamic range (in bits) of the original image. The term 1/12 eliminates the infinite value for PSNR when MSE approaches 0 and represents the MSE associated with the quantization of the original analog data.

Since image signals are highly structured, the pixels dependency carry important structural information about the image content. Although the PSNR is a practical and useful image quality measure, the structural information might not be well measured by the PSNR [30]. Therefore, we used another quality measure known as *structural similarity* (SSIM) index [31] and defined as

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{\left(2\mu_x\mu_y + C_1 \right)\left(2\sigma_{xy} + C_2 \right)}{\left(\mu_x^2 + \mu_y^2 + C_1 \right)\left(\sigma_x^2 + \sigma_y^2 + C_2 \right)}, \tag{3}$$

where μ_x and μ_y are the local sample means of the windows $\mathbf{x}$ and $\mathbf{y}$ of common size of N × N, σ_x and σ_y are the local sample standard deviations of $\mathbf{x}$ and $\mathbf{y}$, and σ_{xy} is the sample cross-correlation of $\mathbf{x}$ and $\mathbf{y}$ after removing their means. The constant values C_1 and C_2 are included to avoid instability when the values in the denominator of the equation are very close to zero.

The SSIM index is locally computed within a sliding window (e.g., 11 × 11 pixels) that moves pixel by pixel across the image, resulting in an SSIM map [31]. This measure takes into account the luminance, contrast, and structure information in the image. We also use a mean SSIM index to evaluate the overall image quality

$$\text{MSSIM}(\mathbf{X}, \mathbf{Y}) = \frac{1}{M} \sum_{j=1}^{M} \text{SSIM}\left(\mathbf{x}_j, \mathbf{y}_j \right), \tag{4}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are the reference and the distorted images, respectively; $\mathbf{x}_j$ and $\mathbf{y}_j$ are the image contents at the jth local window; M is the number of local windows of the image. A MATLAB and C++ implementation of the SSIM index algorithm is available on the Internet [32].

5.3. Test Image Dataset. The scenes captured by CBERS-2B CCD camera have five spectral bands, as shown in Table 2. Each spectral band is a monochromatic raw data, with 5812 × 5812 pixels and eight bits per pixel. The algorithms

TABLE 2: CBERS-2B: CCD camera [3].

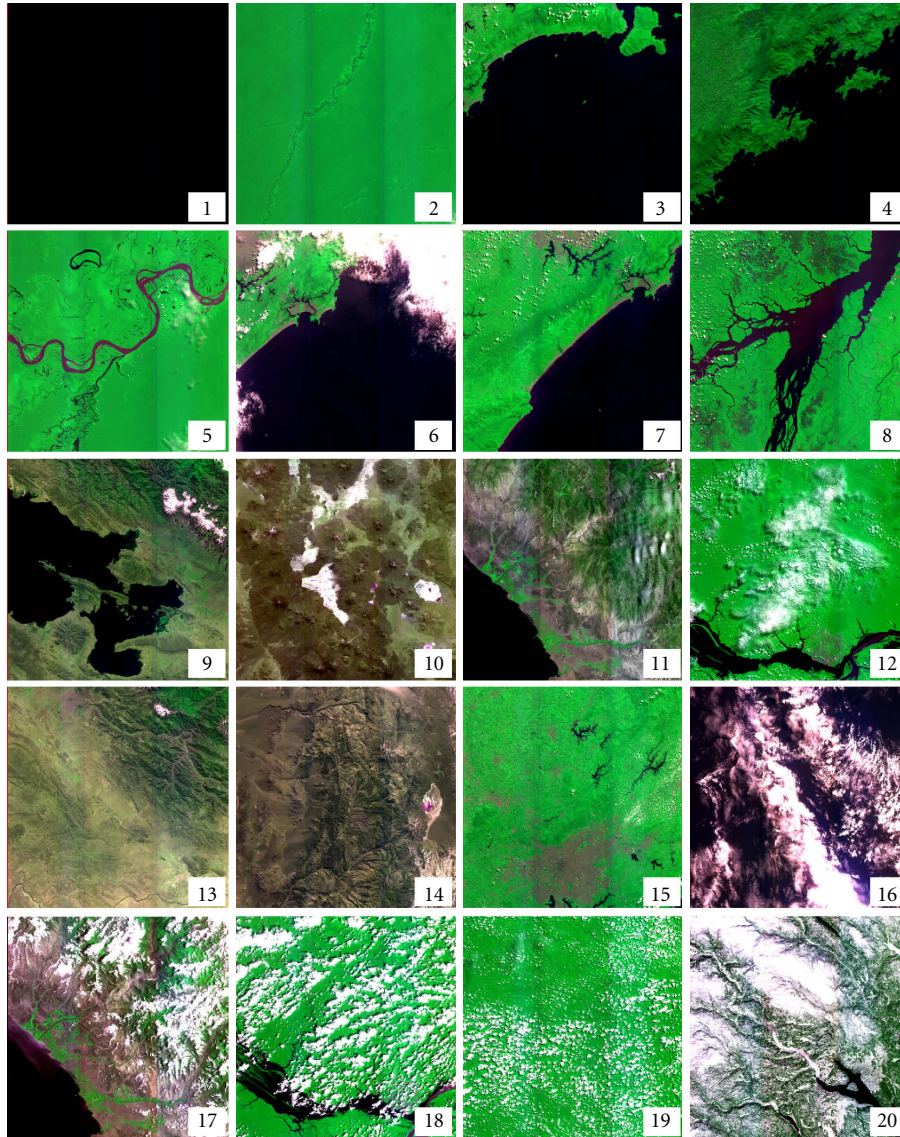| Band	Name	Wavelength (μm)	Spatial resolution (m)/ temporal Resolution (days)	Dimensions ($w \times h$) pixels	Bit depth (bits/pixel)
B1	Blue	0,45– 0,52	$20 \times 20/26$	5812×5812	8
B2	Green	0,52–0,59	$20 \times 20/26$	5812×5812	8
B3	Red	0,63–0,69	$20 \times 20/26$	5812×5812	8
B4	Near infrared	0,77–0,89	$20 \times 20/26$	5812×5812	8
B5	Pan (Panchromatic)	0,51–0,73	$20 \times 20/26$	5812×5812	8

FIGURE 12: CBERS-2B CCD scenes: 3R4G2B color composition [33].

performance evaluation was carried out using 20 scenes described in Table 3.

The CBERS-2B CCD images available from the INPE catalog public website [33] were not used in this test because they have radiometric and geometric calibration. The test image set consisted of twenty 5-band images without any calibration processing. These images were chosen to represent a variety of content relevant to different remote sensing applications, such as agriculture, forest, urban areas, surface water, and other scenes with different cloud cover. The
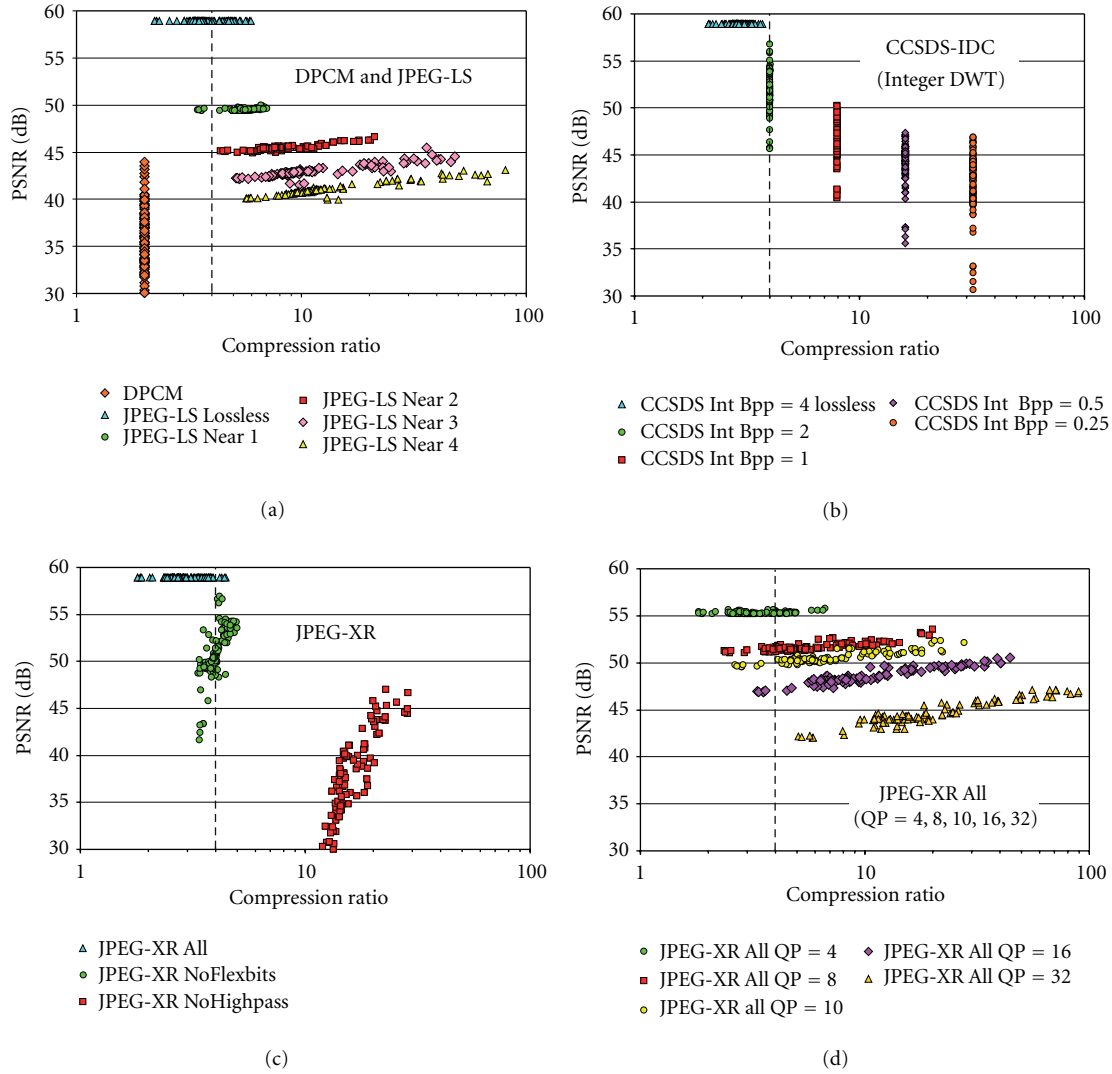
(a)

(b)

(c)

(d)

Figure 13: Dispersion charts of PSNR × compression ratio over 100 test images.

3R4G2B color compositions (bands 3, 4, and 2) are shown in (1) through (20) of Figure 12, in descending order of compression ratio.

5.4. Results. To evaluate lossy compression performance, the PSNR metric was measured at several bit rates for each image in the test set. Table 4 shows the compression ratio and PSNR results for DPCM, JPEG-LS, CCSDS-IDC, and JPEG-XR. The values highlighted in bold have not met the minimum constraints of a compression ratio of 4 and a PSNR of 50 dB.

The average CR is calculated from the average bit rate. Similarly, as the image sizes and bit depths are the same for all original images, the average CR can be calculated as the ratio of the average image sizes before and after compression. This is more meaningful than the direct average of the obtained CRs, because this second estimate tends to underestimate the compressed sizes as the harmonic mean of the achieved results. The average PSNR is calculated as the PSNR corresponding to the average MSE (added to the 1/12 correction term), to avoid biasing the estimate with the concavity of the log function.

Every PSNR versus compression ratio results over a hundred images are plotted in (a) through (d) of Figure 13. Figure 14(a) presents the average PSNR value calculated over all scenes, except scenes 1 and 2. The average MSSIM versus compression ratio values are plotted in Figure 14(b).

The CBERS-2B CCD images have low contrast, so Figure 15 shows band 2 (scene 5) enhanced by linear contrast stretch for visualization. The raw images of CBERS-2B present artifacts as striping effects due to the disparities between the odd and even pixels, as shown in Figure 15(c). This problem will not occur in the images of CBERS 3 and 4 satellites; therefore, we also gathered compression results on filtered images. This dataset smoothed using median filter has inflated the compression ratio, especially for lossy JPEG-LS and JPEG-XR using All coefficients mode. For this reason, in this paper we only present compression results on raw data.
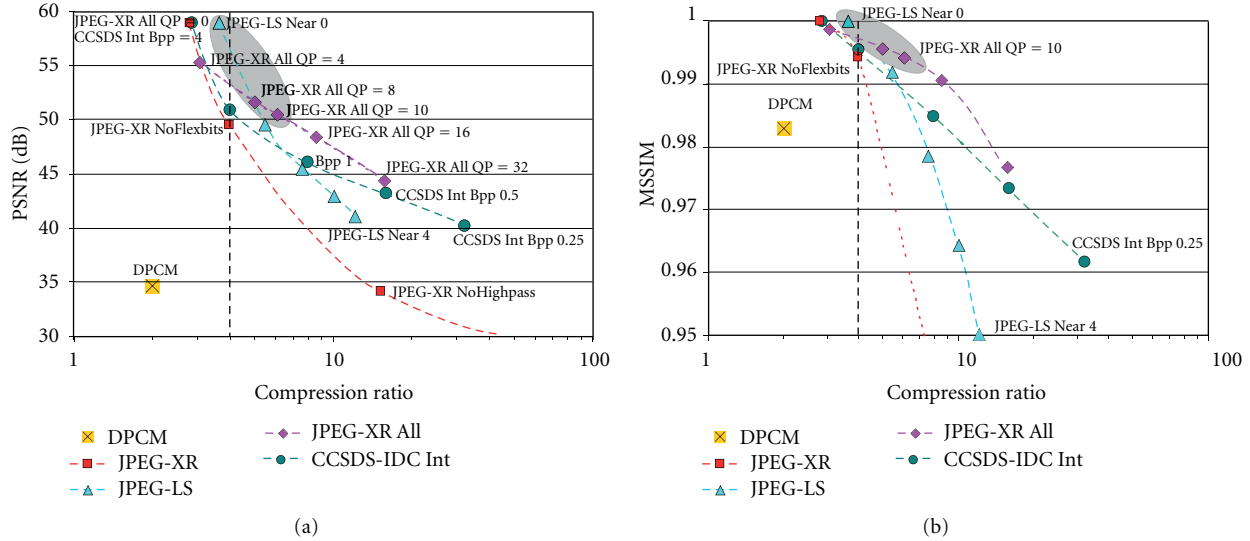
(a)

(b)

FIGURE 14: (a) PSNR × compression ratio chart. (b) MSSIM × compression ratio chart. Average values over 90 images for DPCM; JPEG-LS Near 0, 1, 2, 3, and 4; CCSDS-IDC "Int Bpp = 0.25, 0.5, 1, 2, and 4"; JPEG-XR All, NoFlexbits and NoHighpass (out of the MSSIM chart), with no quantization; and JPEG-XR "All, QP = 4, 8, 10, 16, and 32".

(a) Original image (5812 × 5812 pixels)

(b) Original image cropped to 400 × 400

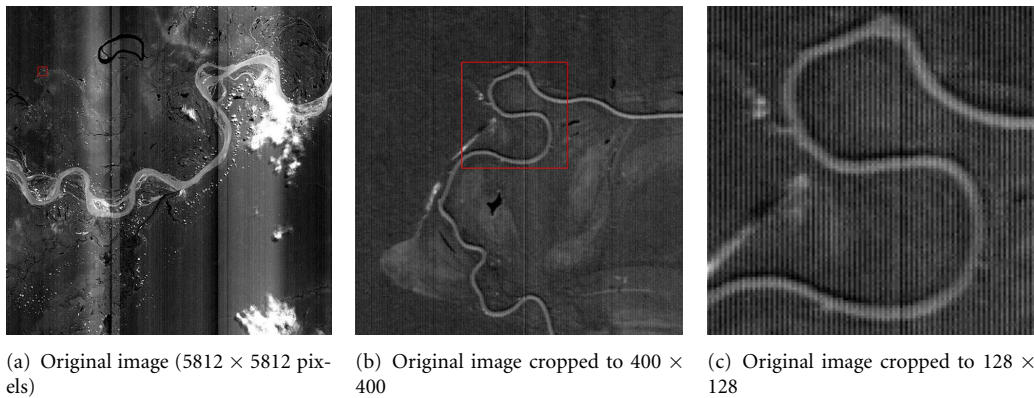(c) Original image cropped to 128 × 128

FIGURE 15: Band 2 of the CBERS-2B CCD scene 5 (enhanced by linear contrast stretch for visualization).

Some SSIM maps are shown on the left and center columns in Figure 16, applying the color map of Figure 17. To give an idea of the visual artifacts produced by the compression and decompression processes, some reconstructed images of scene 5 (band 2) are shown in detail on the right column of Figure 16. The histograms of the SSIM maps are shown in Figure 17(e).

5.5. Performance Analysis. The quantitative performance of the algorithms was evaluated using the dataset listed in Table 3. Table 4 shows the compression ratio and PSNR values averaged over five bands for each scene. The PSNR values taking into account 100 images are plotted for each band in Figure 13. To avoid distortions at lower bit rate, the PSNR and MSSIM average values for each algorithm, plotted in Figure 14, are calculated over all raw images except scenes 1 and 2.

A fixed-rate coder is a desirable feature to guarantee that the memory will never overflow and that the data is always transmitted through the channel in a fixed time. However, some variable-rate schemes can yield higher compression ratios than fixed-rate coders of comparable complexity. In (a) through (d) of Figure 13 we can see that some compression schemes, such as the JPEG-LS and the newly adopted JPEG-XR standard, are variable-rate coders. Although Figure 13(b) shows that CCSDS-IDC provides fixed-rate compression, it can also compress with variable rate (quality limited) using other parameters choices.

(a) *DPCM Performance Analysis.* The PSNR charts in Figures 13(a) and 14(a) show that DPCM has poor compression ratio and PSNR performances in comparison with other algorithms. Due to the basic prediction algorithm of this particular DPCM, the compression ratio is fixed at 2, and the PSNR quality measurement is lower than 50 dB. The MSSIM chart in Figure 14(b) shows that DPCM can reconstruct images with better structural similarity than some lossy JPEG-LS (Near up to 1) and some CCSDS-IDC (Bpp
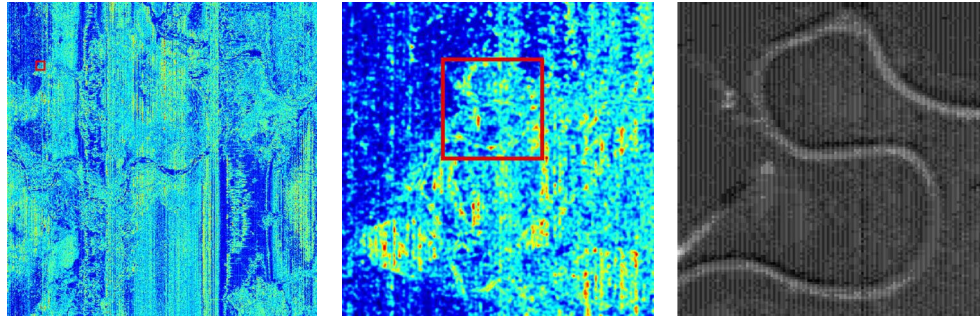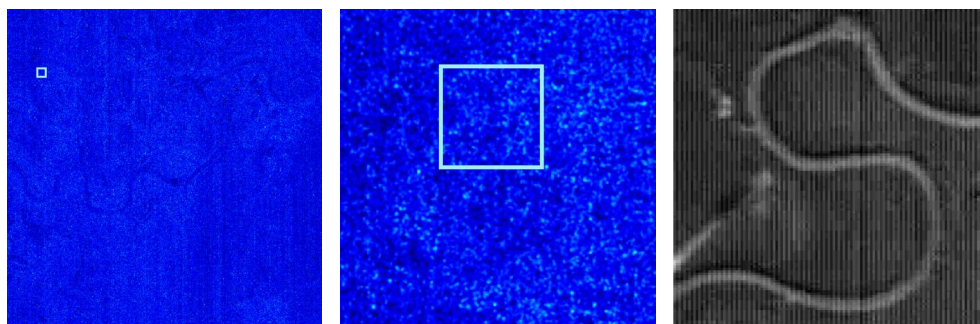
(a) DPCM decompressed image. Compression ratio 2.0, PSNR = 38.58 dB, MSSIM = 0.9852
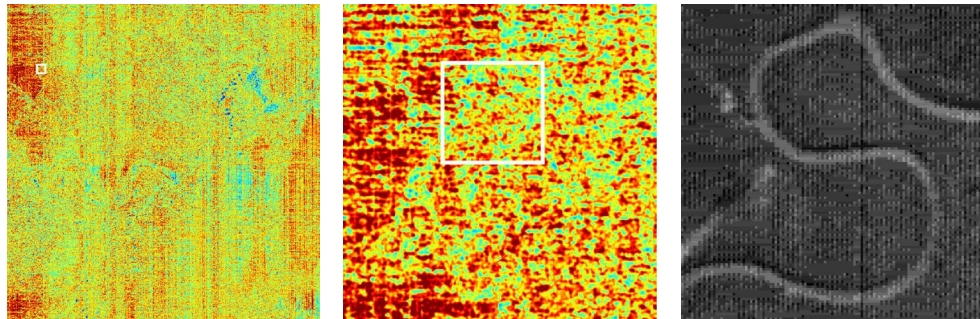
(b) JPEG-LS (Near 2) decompressed image. Compression ratio 8.21, PSNR = 45.34 dB, MSSIM = 0.9705
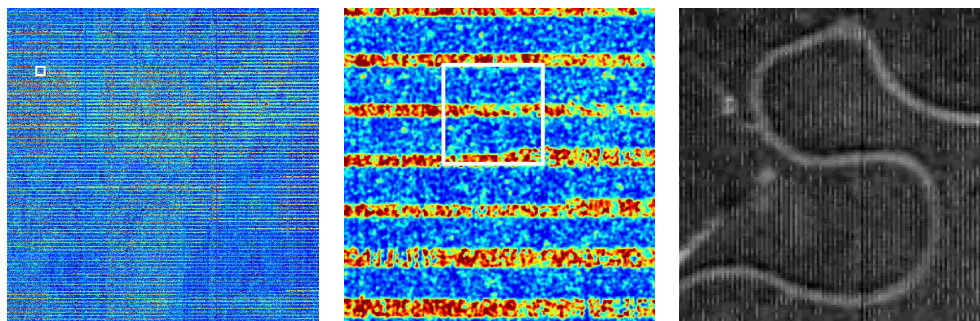
(c) CCSDS-IDC Integer (Bpp = 1) decompressed image. Compression ratio 7.99, PSNR = 47.75 dB, MSSIM = 0.9854

(d) JPEG-XR "All, QP = 10" decompressed image. Compression ratio 9.64, PSNR = 51.89 dB, MSSIM = 0.9937

FIGURE 16: Some reconstructed images compressed by different compression algorithms. The left (original size) and center (cropped to 400×400) images show SSIM maps of the compressed images, where color indicates the local SSIM magnitude. The right image (cropped to 128×128) shows details of the decompressed image (enhanced by linear contrast-stretch for visualization).

up to 1) algorithms. Figure 16(a) and the SSIM histogram in Figure 17(a) confirm that the original raw image and the DPCM reconstructed image have poorer structural similarity than CCSDS-IDC Int Bpp = 1 and JPEG-XR QP = 10.

(b) *JPEG-LS Performance Analysis.* The lossless JPEG-LS outperforms the lossless CCSDS-IDC and lossless JPEG-XR. This is confirmed with PSNR results in Figures 13(a) and 14(a).

TABLE 3: Test scenes.

| Scenes	Source data (path_row_date)	Satellite/sensor
1	P151_R127_20080719	CBERS-2B/CCD
2	P178_R105_20080716	CBERS-2B/CCD
3	P153_R127_20080305	CBERS-2B/CCD
4	P152_R126_20080525	CBERS-2B/CCD
5	P180_R103_20080805	CBERS-2B/CCD
6	P153_R127_20071027	CBERS-2B/CCD
7	P154_R127_20080302	CBERS-2B/CCD
8	P162_R102_20080712	CBERS-2B/CCD
9	P179_R118_20080713	CBERS-2B/CCD
10	P177_R124_20080814	CBERS-2B/CCD
11	P189_R112_20081212	CBERS-2B/CCD
12	P173_R103_20080826	CBERS-2B/CCD
13	P178_R119_20080811	CBERS-2B/CCD
14	P177_R130_20080719	CBERS-2B/CCD
15	P154_R126_20080328	CBERS-2B/CCD
16	P146_R137_20090201	CBERS-2B/CCD
17	P189_R112_20090202	CBERS-2B/CCD
18	P173_R103_20090103	CBERS-2B/CCD
19	P172_R100_20080829	CBERS-2B/CCD
20	P121_R31_20080323	CBERS-2B/CCD

Due to the run mode, the lossy JPEG-LS method can degrade images generating line effects that increase with the *Near* parameter. It must be noted that the vertical stripes in the raw images avoided higher degradation in run mode. However, JPEG-LS can generate more intense horizontal line effects in images free of vertical stripes, as we observed in other evaluation tests performed with smoothed images.

The MSSIM chart in Figure 14(b), the SSIM map in Figure 16(b) (left and center), and its histogram in Figure 17(b) show that the lossy JPEG-LS algorithm has lower image fidelity than other algorithms with comparable bit rate.

The advantage of the JPEG-LS is its low complexity in comparison with the transform-based algorithms.

(c) *CCSDS-IDC Performance Analysis.* We compared the integer DWT of the CCSDS-IDC for "push-broom" sensors. The integer DWT is used for applications requiring lossless compression and is preferable in lossy compression for complexity reasons to avoid floating point operations in the DWT calculation.

An image with width w and height h generates $\lceil w/8 \rceil \cdot \lceil h/8 \rceil$ DWT coefficient blocks. The entire image is compressed as a single segment (full-frame compression) defining $S = \lceil w/8 \rceil \cdot \lceil h/8 \rceil$. When $S = \lceil w/8 \rceil$, each image segment corresponds to a thin horizontal strip of the image (strip compression). We defined a segment of blocks that correspond to the image width. By setting $S = w$, each segment consists of eight strips. The advantage of the strip compression is that there is no need to store a complete frame of image or DWT data; thus, it can lead to a memory-efficient implementation that is convenient to push-broom sensors.
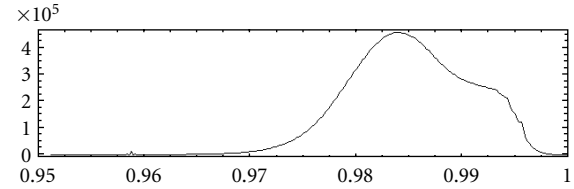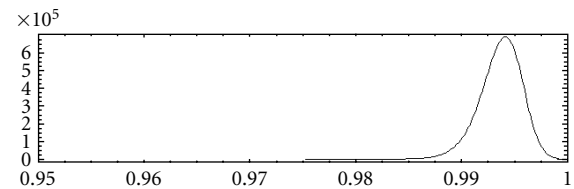

(a) DPCM, MSSIM = 0.9852
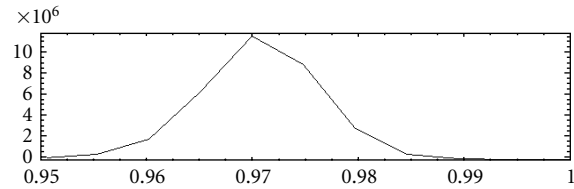
(b) JPEG-LS Near 2, MSSIM = 0.9705
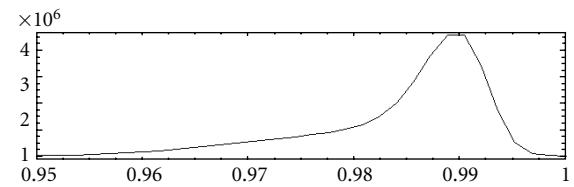
(c) CCSDS-IDC Int Bpp = 1.0, MSSIM = 0.9854

(d) JPEG-XR "All, QP = 10", MSSIM = 0.9937

(e)

FIGURE 17: ((a)–(d)) The histograms of the SSIM maps. (e) Color map used to show the SSIM maps, where blue color indicates SSIM indexe qual to 1.0 and red color indicates SSIM index lower than 0.95. The MSSIM index achieves value 1.0 if and only if the two images being compared are equal.

We also imposed fixed rate constraints on each compressed segment, evaluating the performance at bit rate set to 0.25, 0.5, 1, 2, and 4 bits per pixel.

The CCSDS-IDC strip-based compression at fixed rate can distort some image portions because the segments may not be completely encoded within the segment byte limit, as illustrated by the horizontal stripes in the SSIM map in Figure 16(c) (center image).

There are certain parameters specified by CCSDS recommendation, as the integer and the float DWTs, strip and frame based, as well as rate-limited and quality-limited compression. We are not intended to test the full range of available options. More detailed performance comparison of CCSDS-IDC can be seen in the document [17, 26].

TABLE 4: Compression ratio and PSNR values for raw CBERS-2B CCD images.

	DPCM	JPEG-LS			CCSDS-IDC								JPEG-XR		
					Integer DWT strip-based fixed-rate					All			NoFlexbits	NoHighpass	DC-only
	Near = 0 Lossless	Near = 1	Near = 2	Near = 3	Bpp = 4 Lossless	Bpp = 2	Bpp = 1	Bpp = 0.5	Bpp = 0.25	Qp = 0 Lossless	QP = 4	QP = 10	QP = 0	QP = 0	QP = 0
Scenes							*Compression ratio								
1	*2.0	5.6	6.5	12.3	28.4	3.5	4.0	8.0	16.0	32.0	4.2	5.9	22.4	4.3	27.8	353.5
2	2.0	4.5	6.0	8.8	18.1	3.1	4.0	8.0	16.0	32.0	3.4	4.1	15.6	4.8	20.6	264.5
3	2.0	4.8	6.4	13.2	34.8	3.3	4.0	8.0	16.0	32.0	3.7	4.5	15.1	4.5	21.5	258.2
4	2.0	4.6	6.0	11.0	19.6	3.2	4.0	8.0	16.0	32.0	3.4	4.1	12.0	4.3	19.8	245.3
5	2.0	4.2	5.9	9.9	18.7	3.0	4.0	8.0	16.0	32.0	3.3	3.8	12.4	4.5	18.4	222.4
6	2.0	4.7	6.7	8.5	12.0	3.2	4.0	8.0	16.0	32.0	3.4	4.0	9.3	3.9	18.9	222.3
7	2.0	4.0	5.8	10.3	16.3	3.0	4.0	8.0	16.0	32.0	3.1	3.6	9.2	4.3	17.6	215.4
8	2.0	4.2	5.8	8.1	13.7	3.0	4.0	8.0	16.0	32.0	3.1	3.6	9.8	4.3	16.9	197.8
9	2.0	3.9	5.7	8.1	10.6	2.9	4.0	8.0	16.0	32.0	3.0	3.4	7.4	4.0	16.3	199.9
10	2.0	3.8	5.6	7.5	10.2	2.9	4.0	8.0	16.0	32.0	3.0	3.2	7.1	4.1	15.0	180.4
11	2.0	3.6	5.7	7.5	9.3	2.9	4.0	8.0	16.0	32.0	2.9	3.1	6.7	4.0	14.9	182.2
12	2.0	3.5	5.3	7.6	9.4	2.8	4.0	8.0	16.0	32.0	2.7	2.9	5.8	4.0	14.5	174.6
13	2.0	3.3	5.3	6.3	7.9	2.7	4.0	8.0	16.0	32.0	2.7	2.8	5.6	4.0	14.1	178.0
14	2.0	3.5	5.3	6.4	8.2	2.7	4.0	8.0	16.0	32.0	2.7	2.9	5.7	3.9	14.1	174.4
15	2.0	3.2	5.1	6.5	8.1	2.7	4.0	8.0	16.0	32.0	2.6	2.7	5.5	3.9	13.9	175.3
16	2.0	3.5	5.6	7.4	8.9	3.0	4.0	8.0	16.0	32.0	2.7	2.9	5.6	3.9	14.3	170.8
17	2.0	3.4	5.4	7.0	8.2	2.9	4.0	8.0	16.0	32.0	2.6	2.7	5.0	3.7	13.6	165.6
18	2.0	3.5	5.5	7.6	9.0	2.9	4.0	8.0	16.0	32.0	2.5	2.7	4.8	3.5	13.2	156.0
19	2.0	3.2	4.9	6.6	7.9	2.6	4.0	8.0	16.0	32.0	2.3	2.5	4.3	3.5	12.6	151.0
20	2.0	2.3	3.6	4.6	5.3	2.2	4.0	8.0	16.0	32.0	1.8	1.9	2.8	3.5	11.5	149.8
**Average	2.0	3.6	5.4	7.6	10.1	2.9	4.0	8.0	16.0	32.0	2.8	3.1	6.1	4.0	15.0	185.5
Scenes							*PSNR (dB)								
1	*41.0	58.9	49.6	45.6	43.4	58.9	55.3	49.5	45.8	45.3	58.9	55.5	52.0	56.0	45.2	37.4
2	36.2	58.9	49.5	45.5	43.5	58.9	52.2	46.9	44.5	43.5	58.9	55.3	50.9	52.3	42.4	36.0
3	39.3	58.9	49.6	45.8	44.1	58.9	53.5	48.7	46.1	45.2	58.9	55.3	51.2	53.5	41.9	33.9
4	39.4	58.9	49.5	45.5	43.6	58.9	53.0	47.8	45.1	43.8	58.9	55.3	51.0	52.3	40.7	33.6
5	36.0	58.9	49.5	45.7	43.7	58.9	52.0	47.4	45.1	43.8	58.9	55.2	50.7	51.9	40.4	33.0
6	34.3	58.9	49.8	45.7	42.3	58.9	52.9	48.3	42.6	41.2	58.9	55.6	51.2	52.6	37.5	30.5
7	37.2	58.9	49.5	45.6	43.6	58.9	52.0	47.4	45.0	43.1	58.9	55.3	50.7	51.3	38.3	30.9
8	36.9	58.9	49.5	45.4	43.1	58.9	51.7	46.3	44.2	42.7	58.9	55.3	50.8	52.0	38.4	30.8
9	37.1	58.9	49.4	45.3	42.9	58.9	51.6	46.1	44.1	42.6	58.9	55.3	50.5	51.1	37.5	30.8
10	34.8	58.9	49.4	45.3	42.9	58.9	51.8	47.1	45.3	43.7	58.9	55.2	50.3	51.4	39.3	32.0
11	35.0	58.9	49.4	45.4	42.9	58.9	51.7	47.2	45.3	43.3	58.9	55.2	50.2	51.0	39.0	31.5
12	34.3	58.9	49.4	45.4	42.9	58.9	51.1	46.3	44.3	42.1	58.9	55.2	50.3	50.0	35.3	27.0
13	35.7	58.9	49.4	45.2	42.6	58.9	50.6	45.9	43.7	41.7	58.9	55.2	50.0	49.7	38.3	31.8
14	36.6	58.9	49.4	45.1	42.6	58.9	50.8	45.5	43.4	41.4	58.9	55.2	50.2	49.6	35.2	28.3
15	36.1	58.9	49.4	45.2	42.6	58.9	50.1	45.6	43.2	40.6	58.9	55.2	50.0	49.1	36.2	29.2
16	33.2	58.9	49.5	45.4	42.8	58.9	51.9	47.1	44.8	42.2	58.9	55.4	50.3	50.0	34.2	25.5
17	32.6	58.9	49.6	45.4	42.7	58.9	51.3	46.6	44.1	40.7	58.9	55.4	50.3	49.5	32.6	24.1
18	32.0	58.9	49.6	45.6	42.9	58.9	51.3	46.5	43.7	40.0	58.9	55.5	50.5	49.6	30.6	21.8
19	33.1	58.9	49.5	45.3	42.7	58.9	50.2	45.4	42.7	38.9	58.9	55.3	50.2	48.6	30.1	21.3
20	31.2	58.9	49.5	45.1	42.2	58.9	45.8	41.0	36.6	32.1	58.9	55.3	49.6	42.7	27.1	20.6
**Average	34.7	58.9	49.5	45.4	42.9	58.9	50.9	46.1	43.2	40.2	58.9	55.3	50.4	49.5	34.2	26.4

* Compression ratio and PSNR values are averaged over five bands of each scene.
** Average values are calculated over all images, except scenes 1 and 2. Due to the small set of scenes and the high compression ratio on scenes 1 and 2, their values were not considered to avoid distorted average values for some algorithms.

According to this document, CCSDS-IDC has performance similar to the JPEG2000 standard.

We noticed that, in comparison with other algorithms as DPCM, lossy JPEG-LS, and nonquantized JPEG-XR, CCSDS-IDC can achieve a good tradeoff between compression efficiency and image quality, as shown in Figures 13(b) and 14. However, this coder presents the highest complexity of all tested algorithms.

(d) *JPEG-XR Performance Analysis.* The JPEG XR encoding and decoding are performed using only basic integer operations to simplify the compression processing. Some researchers have reported that the compression performance of JPEG-XR is similar to that of JPEG2000, depending on various implementation details [19]. In addition, both JPEG-XR and JPEG2000 outperform JPEG. It can be seen from the results that, depending on the parameters used, JPEG-XR can achieve both good compression ratio and image quality results compared to the other lossy compression methods, as shown in Figures 13(d) and 14.

The nonquantized JPEG-XR with All and NoFlexbits option have presented similar efficiency of the CCSDS-IDC. The nonquantized JPEG-XR with NoHighpass and DC-Only option have presented high compression ratio, but poor image quality with intense blocking effects.

JPEG-XR with All coefficients quantized by QP parameter set to 8 or 10 value can give a good tradeoff (the best observed) between compression efficiency and image quality, as shown in Figures 14, 16(d), and 17(d). Whether the optional overlap filtering is enabled, JPEG-XR can reduce block artifacts, which can be seen in the right image of Figure 16(d). However, the overlap filtering has not enabled in this test to target the low complexity implementation for embedded applications.

JPEG-LS and CCSDS-IDC support a bit depth of 8 up to 16 bit per channel. However, more advanced cameras may require greater bit depth. This feature is addressed in JPEG-XR, which was designed to support compression of HDR (high dynamic range) formats, including 16- and 32-bit float, and 16-bit and 32-bit signed and unsigned integer.

We have confirmed that prediction-based compression systems, such as DPCM and JPEG-LS, are generally faster than transform-based compression systems, such as CCSDS-IDC, and JPEG-XR. Although the processing time is also important for embedded applications, we did not consider it in this analysis because it is highly dependent on the processing engine. Moreover, some softwares used in this test were implemented only to demonstrate compression performance, with no optimization in speed.

6. Conclusions

In this paper, we presented a comparison among different compression systems suitable for remote sensing images. We performed various experiments to compare the compression performance of the DPCM, JPEG-LS, CCSDS-IDC and JPEG-XR methods. The test-image dataset used in the experiments was acquired from CBERS-2B CCD camera and represented a variety of content commonly found in remote sensing images. The comparative analysis used objective metrics given by compression ratio, PSNR, and SSIM values.

The low complexity of the DPCM algorithm is an advantage for real-time applications. The predictor performs only basic integer operations, and the encoder is implemented by lookup tables. Consequently, each pixel is coded in real time as soon as it is captured by the sensor. Moreover, this fixed-rate coder transmits the data through the channel in a fixed time, simplifying implementation. All these features make this DPCM coder easily implementable for embedded applications. However, its performance results did not meet the minimum requirements of compression ratio and PSNR (4 and 50 dB, resp.) desirable for some space missions.

JPEG-LS performs well for lossless compression and is suitable for cost-sensitive and embedded applications that do not require any additional functionality, such as progressive bit streams and error resilience. Lossy JPEG-LS, on the other hand, did not achieve the minimum requirements of image quality, mainly due to the line artifacts induced by this scheme.

The CCSDS-IDC method presents better image quality than the lossy JPEG-LS and nonquantized JPEG-XR methods. The drawback of the strip-based compression is the lower quality in portions of each segment that are not completely encoded due to the segment byte limit. Furthermore, the algorithm has the highest complexity for embedded environments.

Depending on the parameter choices, the lossy JPEG-XR can achieve better image quality than other algorithms. In addition, JPEG-XR algorithm achieves higher compression performance than the predictive algorithms. It also has lower computational complexity and memory requirements than CCSDS-IDC.

As DPCM, JPEG-LS, CCSDS-IDC, and JPEG-XR are based on different technologies, the artifacts induced by the compression systems are also very different. Both peak signal-to-noise ratio (PSNR) and mean structural similarity (SSIM) metrics were useful to measure the image quality. With all this information, we can select the algorithms that have presented the best tradeoff between compression ratio and image quality. For lossless image compression, we have selected the JPEG-LS algorithm, as it achieves the best performance. For lossy image compression, the JPEG-XR "All, QP = 8 or 10" gives the best tradeoff between compression efficiency and image quality.

Our results indicate that for high bit rates, the JPEG-LS, CCSDS-IDC, and JPEG-XR coders have approximately the same overall performance. However, at lower bit rates, JPEG-XR "All, QP = 10" performs better than the lossy JPEG-LS and CCSDS-IDC in strip-based compression mode for every tested image.

As a conclusion, the results obtained from objective quality metrics show that lossless JPEG-LS and lossy JPEG-XR "All, QP = 8 or 10" successfully meet the minimum requirements of compression ratio and PSNR of 4 and 50 dB, respectively, as shown in the highlighted area of the charts in Figure 14.

For future work, we plan to analyze the lossy JPEG-XR with different parameters, using a more diverse set of test images (captured by other satellite sensors) with a wider range of bit rates, better spatial, and radiometric resolutions (bit depth). We are also planning to conduct a series of subjective tests, in which a group of remote sensing interpreters will be asked to rank lossy decompressed images according to their perceived qualities.
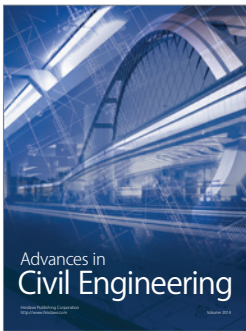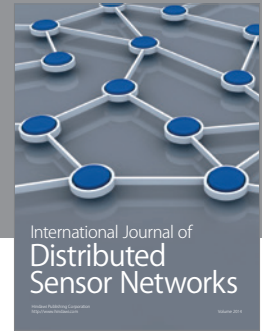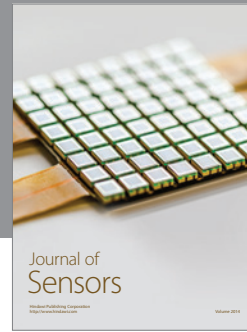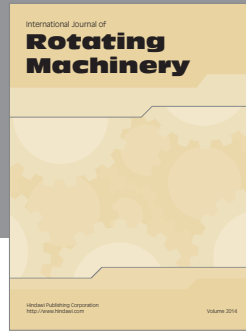
This study aimed to evaluate different image compression methods to choose a compression scheme suitable to be implemented in FPGA hardware and used onboard the next generation of satellites of the Brazilian Space Program. The next phase is to evaluate practical aspects of the systems, such as processing complexity (speed), and ease of implementation in FPGA hardware.

Acknowledgments

References

[1] D. Salomon, *Data Compression: The Complete Reference*, Springer, New York, NY, USA, 4th edition, 2007.

[2] National Institute for Space Research, "CBERS—China-Brazil earth resources satellite: history," 2011, http://www.cbers.inpe.br/?hl=en&content=historico/.

[3] National Institute for Space Research, "CBERS—China-Brazil earth resources satellite: cameras, CBERS 1, 2, and 2B," 2011, http://www.cbers.inpe.br/?hl=en&content=cameras1e2e2b/.

[4] National Institute for Space Research, "CBERS—China-Brazil earth resources satellite: cameras, CBERS 3 and 4," 2011, http://www.cbers.inpe.br/?hl=en&content=cameras3e4/.

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, New York, NY, USA, 3rd edition, 2007.

[6] G. Yu, T. Vladimirova, and M. N. Sweeting, "Image compression systems on board satellites," *Acta Astronautica*, vol. 64, no. 9-10, pp. 988–1005, 2009.

[7] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[8] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, pp. 1098–1101, 1952.

[9] S. Golomb, "Run-length encoding," *IEEE Transactions on Information Theory*, vol. 12, no. 3, pp. 399–401, 1966.

[10] N. Moayeri, "A low-complexity, fixed-rate compression scheme for color images and documents," *The Hewlett-Packard Journal*, vol. 50, no. 1, pp. 46–52, 1998.

[11] W. B. Pennebaker and J. L. Mitchell, *JPEG: Still Image Data Compression Standard*, Springer, New York, NY, USA, 1st edition, 1992.

[12] ISO/IEC FCD 14495-1, "Lossless and near-lossless coding of continuous tone still images (JPEG-LS), ISO international standard," July 1997.

[13] D. Taubman and M.W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Springer, New York, NY, USA, 1st edition, 2001.

[14] A. Kiely and M. Klimesh, "The ICER progressive wavelet image compressor," *The Interplanetary Network Progress Report*, vol. 42–155, pp. 1–46, 2003.

[15] A. Kiely, M. Klimesh, H. Xie, and N. Aranki, "ICER-3D: a progressive wavelet-based compressor for hyperspectral images," *The Interplanetary Network Progress Report*, vol. 42–164, pp. 1–21, 2006.

[16] Consultative Committee for Space Data Systems, "CCSDS 122.0-B-1: image data compression, report concerning space data system standards, blue book," November 2005.

[17] Consultative Committee for Space Data Systems, "CCSDS 120.1-G-1: image data compression, report concerning space data system standards, green book," June 2007.

[18] ITU-T Recomendation T.832 | ISO/IEC 29199-2, Information technology—JPEG XR image coding system—Image coding specification, ITU-T CCITT Recommendation, 2009.

[19] F. Dufaux, G. J. Sullivan, and T. Ebrahimi, "The JPEG XR image coding standard [Standards in a Nutshell]," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 195–204, 2009.

[20] C. Tu, S. Srinivasan, G. J. Sullivan, S. L. Regunathan, and H. S. Malvar, "Low-complexity hierarchical lapped transform for lossy-to-lossless image coding in JPEG XR/HD Photo," *Proceedings of SPIE Applications of Digital Image Processing XXXI*, vol. 7073, 2008.

[21] China Academy of Space Technology, "Introduction to DCPM encoding algorithm in data transmission sub-system of PAN-MUX_IRMSS onboard CBERS 3&4 satellites," ∗[S.l.]: CAST, (Wx CBERS03/04DPS.SM01), 2010.

[22] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1309–1324, 2000.

[23] ITU-T T.81—ISO/IEC10918-1, digital compression and coding of continuous-tone images, requirements and guidelines, ITU CCITT recommendation, September 1992.

[24] Consultative Committee for Space Data Systems, "CCSDS 120.0-G-2: lossless data compression, recommendation for space data systems standards, green book," December 2006.

[25] Consultative Committee for Space Data Systems, "CCSDS 121.0-B-1: lossless data compression, blue book," May 1997.

[26] P.-S. Yeh, P. Armbruster, A. Kiely et al., "The new CCSDS image compression recommendation," in *Proceedings of the IEEE Aerospace Conference*, pp. 4138–4145, March 2005.

[27] "LOCO-I/JPEG-LS reference encoder—v.1.00," May 2011, http://www.hpl.hp.com/loco/software.htm/.

[28] "An implementation of CCSDS 122.0-B-1 recommended standard," May 2011, http://hyperspectral.unl.edu/.

[29] ISO/IEC FCD 29199-5, "Information technology—JPEG XR image coding system—part 5: reference software," [ISO/IEC JTC 1/SC 29/WG 1 N 5020], May 2011.

[30] Z. Wang and A. C. Bovik, "Mean squared error: lot it or leave it? A new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, 2009.

[31] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[32] Z. Wang, "The SSIM index for image quality assessment," http://www.cns.nyu.edu/lcv/ssim/.

[33] National Institute for Space Research, "Image catalog," May 2011, http://www.dgi.inpe.br/CDSR/.