# Near Real Time Enhancement Compression Technique for Aerial and Remote Sensing Images

Dr. Emad S. Othman, Dr. Mohammed M. Sakre

**Abstract**— On-board image compression systems aim to increase the amount of data stored in the on-board mass memory and transmitted to the ground station. These Generally, the best known accepted image compression techniques - such as wavelet based technique- have some obvious weakness to act as on-board real time practical compression technique due to their large processing units and multistage floating coefficients. It is difficult to implement them in compact reliable hardware architecture and achieve high compression performance results suitable for on-board compression systems. On the other hand, high image fidelity of decoded image and its hardware simplicity are important features for on-board real time compression systems. The introduced Resampling Block Coding (RBC) algorithm is a step forward to fulfill some of the features. RBC compression results are significant when compared to the Set Partitioning In Hierarchical Trees (SPIHT) wavelet based image compression technique within the near-lossless region. In this article, the results are declared concerning the image fidelity, compression ratio (CR), speed and the maximum grayscale error of the decoded image.

**Index Terms**— Resampling Block coding, wavelet transform, image compression, adaptive image coding, Huffman coding, image fidelity.

## 1 Introduction

IT is difficult to implement techniques in compact reliable hardware architecture and achieve high compression performance results suitable for on-board compression systems. On the other hand, high image fidelity of decoded image and its hardware simplicity are important features for on-board real time compression systems. The compression algorithms based on frequency domain transform such as JPEG, the joint expert group for image coding and compression and discrete wavelet transform [1] and others are not quite suitable to fulfill the requirements of on-board compression systems for remote sensing and aerial images [2].

## 2 Wavelet Weakness

Although wavelet based image compression method is accepted as one of the best image compression methods, it has some obvious weakness to work as on-board real time compression method such as:

- The wavelet compression algorithm is based on using large areas of the image while processing leading to more computational consuming time. On the other side, by decreasing the used processing areas, the restored image quality will be defected rapidly [3].
- In practical implementation, it is more complicated to use the large processing units to be integrated in a real-time compression system performing high image fidelity and high speed rates [4].
- In order to get high fidelity decoded images, more real number in the wavelet basis functions are necessary. This increases the wavelet transform calculations processing time leading to large volume device with low reliability and high power consumption [5].
- Due to the limitations of computer systems word-length, the sequence of the coefficients in the wavelet transform has to be truncated to a finite length. This causes the contours and edges distortion in the image [6].

These negative defects operating on the aerial and remote sensing images causes severe defects on the small details of information in the image [7].

## 3 RBC and On Board System Requirements

Several special features are required for on-board aerial and remote sensing image compression technique such as: High speed compression time, high fidelity of the decoded image (visually lossless) and low complexity of the applied algorithm. Simple algorithm can be easily implemented as on-board real time compression system with compact structure, low weight, low power consumption and high reliability.

As an approach to the goal, RBC a new compression algorithm -working in the space domain of the image- is introduced. The RBC algorithm is a step forward to fulfill some of the features [8]. RBC compression results are significant compared to the Set partitioning In Hierarchical Trees (SPIHT) wavelet based image compression technique within the near-lossless region. The results are declared concerning the image fidelity, compression ratio (CR), speed and the algorithm encoding complexity.

The image redundancy types and the RBC algorithm theory are discussed in the next section. Another section describes the RBC algorithm process main steps. The compression performance and conclusion of the results are included in sections the last two sections respectively.

- *Dr. Emad S. Othman, Senior Member IEEE - Region 8, High Institute for Computers and Information Systems, AL-Shorouk Academy, Cairo – Egypt, PH- 0020-01121024270. E-mail: emad91@hotmail.com*
- *Associate Professor. Mohammed M. Sakre, Senior Member IEEE - Region 8, High Institute for Computers and Information Systems, AL-Shorouk Academy, Cairo – Egypt, PH- 0020-01006525776. E-mail: m_sakre2001@yahoo.com*

## 4 DIGITAL IMAGE REDUNDANCY

There exist three basic data redundancy types in the image: coding, interpixel and psyhvisual redundancy.

The interpixel redundancy stems from the fact that there is a correlation between the pixels values in general. A real image (not containing pure noise) can be modeled as a Markov process where the value of the current pixel has some dependency with the previous pixels [9].

Since the human perception is not a constant pixel oriented sensor then every area in the visual field is not processed with the same amount of sensitivity. These areas - which do not contribute with valuable visual details in the image- can possibly be removed without major loss in quality for the human perceiver. This invisible image details are called psychovisual redundant information [10].

## 5 SAMPLING AND QUANTIZATION

A two dimensional image can be represented by a light intensity function *f(x,y)* where *x* and *y* denote the spatial coordinates. The value of the function *f(x,y)* is proportional to the brightness of the image at the pixel *(x, y)*. The digital image is formed by digitizing both spatial resolution and brightness function of the image [11]. The two digitizing processes for the spatial resolution and brightness function values are called sampling and quantization processes respectively. A continuous function *f(x, y)* can be sampled using a discrete grid of sampling points in the plane. A second possibility is to expand the image function using some orthonormal function as a base like the Fourier transform. The coefficients of this expansion represent the digitized image [12]. It is easy to explain the sampling and quantization definitions from signal processing point of view as in the next sections.

## 6 DESCRIPTION OF RBC ALGORITHM

Based on these criteria, a three-stages sampling and quantized compression algorithm denoted by RBC is implemented.

### 6.1 Classifying Each Unit Block

During the sampling process, the rectangular sampling regions of the input image are divided into $2^n \times 2^n (n = 0 \sim N)$ as large as the original regions. If the sampling area represents high frequency image region then the value of *n* approaches zero while it approaches maximum limited size if it represents low frequency regions.

Based on the theory of the sampling, three-stage for sampling and quantization processes are required before the image coding. First, the image is divided into *4x4* units from top to bottom, and from left to right. Depending on its information contents and the gray level features, each unit in the image is oriented to one of three different frequency modes: low (*n=2*), medium (*n=1*) and high (*n=0*) mode. The mean square error of the *4x4* unit $\bar{E}^2$ is defined as:

$$\bar{E}^2 = \sum_{0 \le i \le 15} (r_i - \bar{R})^2 / \sum_{0 \le i \le 15} r_i \quad (1)$$

Where: $r_i$ is the gray value of the each pixel in the unit block, *R* is the mean value of the unit block. Depending on the required compression ratio given by the user, the threshold values $T_1$ and $T_2$ are selected where: $T_1 < T_2$. During the coding process the threshold values $T_1$ and $T_2$ are adaptively adjusted. According to these threshold values and the unit mean square error *E*, each unit block in the image is classified to one of the three modes. The first mode is for low frequency regions where $\bar{E}^2 \le T_1$. The second mode is for the medium frequency regions where $T_1 < \bar{E}^2 \le T_2$ and the last mode is for the high frequency regions, where $T_2 < \bar{E}^2$.

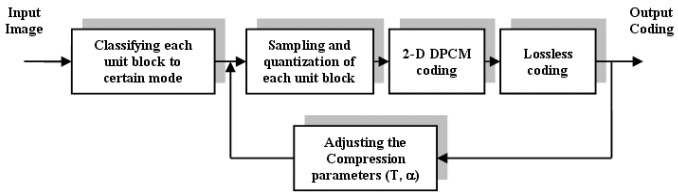


Fig. 1. The flow of compression process

After classifying all the unit blocks in the image, the sampling and quantization processes take place as shown in Fig. 1.

### 6.2 Sampling and quantization of each unit block

Different sampling and quantization levels are used with each unit block depending on its mode. In the low frequency regions, the sampling area is *4x4* and the quantization gray level is $D_1$. In the medium frequency regions, the sampling area is *2x2* and the quantization gray level is $D_2$. In the high frequency regions, the sampling area is *1x1* (as the original) and the quantization gray levels are $D_3$. The quantization parameter $\alpha$ is selected with the thresholds $T_1$ and $T_2$ according to the required compression ratio given by the user before starting the compression process, where:

$$D_1 = 256/\alpha \,,\; D_2 = 256/2\times\alpha \,,\; D_3 = 256/4\times\alpha \quad (2)$$

### 6.3 Image coding with 2-D Differential Pulse Code Modulation (DPCM)

After sampling and quantization processes, the image error differences are coded by the 2D DPCM to improve the coding performance and become more efficient.

### 6.4 Lossless coding with multiple Huffman coding tables

Multiple Huffman coding tables are adaptively used to code the result of DPCM. Depending on the previous statistical record coding and due to the correlation between the image strips, different Huffman tables are selected (static or dynamic tables) at the end of each strip coding.

### 6.5 Adjusting compression parameters

To achieve the required compression ratio given by the user, the compression parameters $(T_i, \alpha)$ are adjusted adaptively at the end of each strip coding. Thus, according to the required compression ratio of the whole image and the instance actual compression results achieved from the previous image strips, the compression parameters $(T_i, \alpha)$ are adjusted adaptively during the compression process.

## 7 Compression Performance

In order to demonstrate the compression performance of RBC, we compare it with SPIHT algorithm. SPIHT algorithm is one of the best known compression methods based on wavelet transform. A few years ago, SPIHT has become the benchmark state-of-the-art algorithm for image compression. Two typical images City and Lenna are compressed with RBC and SPIHT. Decoded images are compared in objective and subjective aspects respectively.

The objective performance is declared by compression time, Peak Signal and Noise Ratio (PSNR) and maximum gray level error MaxErr, which are defined as:

$$PSNR = 10\log_{10}\frac{MSE}{255^2} \quad \text{(dB)} \tag{3}$$

$$MaxErr = \max_{1\le i\le M}(r_i - d_i) \tag{4}$$

MSE is the Mean Square Error value between original image and the decoded image, $r_i$ is the gray value of the original image, $d_i$ is the gray value of the decoded images and $M$ is the total number of the image pixels. From the comparison results of RBC and SPIHT as shown in Table 1, the following remarks are significant:

Using a spatial method needs less complex calculations than the frequency transformation method. This is the reason why coding speed of RBC is five times faster than the wavelet based algorithm SPIHT. Only few and simple integer operations are needed for RBC algorithm. On the other side, SPIHT method uses large floating point during operation process.

This important feature leads to a faster compression algorithm, and makes RBC fits the main requirements for high-speed compression system.

TABLE 1
A comparison of the compression results between RBC and SPIHT within the near-lossless region

| Compression Ratio (CR) | Compression method | Compression time(sec) | MaxErr | PSNR |
|---|---|---|---|---|
| 8.00 | SPIHT | 3.74 | 62 | 28.45 |
| | RBC | 0.80 | 56 | 28.70 |
| 7.00 | SPIHT | 5.32 | 56 | 29.34 |
| | RBC | 1.21 | 50 | 29.72 |
| 6.00 | SPIHT | 8.33 | 47 | 30.59 |
| | RBC | 1.21 | 43 | 30.69 |
| 5.00 | SPIHT | 8.64 | 35 | 32.01 |
| | RBC | 1.38 | 34 | 32.24 |
| 4.00 | SPIHT | 9.61 | 30 | 34.11 |
| | RBC | 1.48 | 27 | 34.33 |

The subjective results for both algorithms can be declared using the decoded 768×768 City image. SPIHT and RBC compress City image at the same CR (8:1).

(a) Original 768×768 City Image
(b) RBC algorithm, PSNR= 28.70 dB
(c) SPIHT algorithm, PSNR= 28.45 dB

Fig. 2. Decoded 768×768 City at CR=8 using both algorithms

In order to protect the edges and details in the image, the RBC algorithm adaptively changes its compression parameters by using different sampling sizes during the image coding process. The decoded images from RBC and SPIHT are shown in Fig. 2 (b) and (c) respectively. It can clear that SPIHT less maintains the image edges, details and may be blurred at higher compression as shown from the Fig. 2 (c).

The small processing units used in RBC compression system makes it easy to be realized and accepted as a near real-time on board compression system. On the other hand, large image areas -sometimes the whole image- are used in the wavelet transformation based compression algorithms. The large transformed areas are hard to be realized and accepted as a real-time coding for on board compression systems.

## 8 Conclusion

This paper in hands introduces the RBC compression method based on the spatial sampling. Three sampling and quantization stages are applied for each unit block in the image. A comparison between RBC and SPIHT is shown with respect to time complexity, maximum error and image quality within near-lossless (4-8:1) compression ratio. The results show that, within this range of near-lossless vision (compression ratio less than 10) and under the condition of about equal quality of decoded images, the RBC is not only leading in subject evaluation but in the objective calculations as well. Its processing speed and image fidelity put it as a step forward to implement a real-time onboard compression system suitable for aerial and remote sensing images.

## Acknowledgment

The authors wish to thank Prof. Dr. Shilong Ma in Beijing University of Aeronautics and Astronautics (BUAA), China. This work was supported in part by Dr. Ahmed Gohar, Egypt.

## REFERENCES

[1] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Prentice Hall; 3rd edition, August 31, 2007, ISBN-10: 013168728X.

[2] Campbell, J. B., Introduction to Remote Sensing , 3rd edition, The Guilford Press. ISBN 1-57230-640-8, 2002.

[3] Jensen, J. R., Remote sensing of the environment: an Earth resource perspective, 2nd edition, Prentice Hall.ISBN 0-13-188950-8, 2007.

[4] Jensen, J. R., Digital Image Processing: a Remote Sensing Perspective (3rd ed.). Prentice Hall, 2005.

[5] Lentile, Leigh B.; Holden, Zachary A., Falkowski, Michael J.; Hudak, Andrew T.; Morgan, Penelope; Lewis, Sarah A.; Gessler, Paul E.; Benson, Nate C.. Remote sensing techniques to assess active fire characteristics and post-fire effects. International Journal of Wildland Fire;3(15):319–345, 2010.

[6] Lillesand, T. M.; R. W. Kiefer, and J. W. Chipman. Remote sensing and image interpretation, 5th edition, Wiley.ISBN 0-471-15227-7, 2003.

[7] Richards, J. A.; and X. Jia. Remote sensing digital image analysis: an introduction, 4th edition, Springer. ISBN 3-540-25128-6, 2006.

[8] Datla, R.U.; Rice, J.P.; Lykke, K.R.; Johnson, B.C.; Butler, J.J.; Xiong, X.. Best practice guidelines for pre-launch characterization and calibration of instruments for passive optical remote sensing. Journal of Research of the National Institute of Standards and Technology, March-April; 116(2):612-646, 2011.

[9] http://en.wikipedia.org/wiki/ wavelet transform _coding

[10] http://en.wikipedia.org/wiki/Run-length_encoding

[11] http://en.wikipedia.org/wiki/Huffman_coding

[12] http://en.wikipedia.org/wiki/Entropy_coding