Entropy Encoding EBCOT (Embedded Block Coding with Optimized Truncation) In JPEG2000

S-Medouakh[1] , Z-E. Baarir[1]

[1]LESIA Laboratory of Research, Electronic Department, University of Biskra , Algeria .

Abstract

JPEG2000 is the new standard for the compression of images, which succeeds to JPEG. It's has a much higher algorithmic complexity. The complexity of the entropy encoder (EBCOT) is the most significant in JPEG2000. Which alone constitutes about 70% of the overall processing time for compression of an image. This new standard has many features and characteristics (region of interest, several types of decompression). In this paper we are interested on studying the algorithm of JPEG2000 and the most complex in the JPEG2000 compression process, the EBCOT entropy encoder, and its performance are presented.

Keywords

Image compression, JPEG2000, Entropy encoder EBCOT.

1. Introduction

JPEG2000 is the new international standard for image compression [l][8]. The JPEG 2000 standard provides a set of features that are of importance to many high-end and emerging applications by taking advantage of new technologies. Some of the most important features that this standard should possess are the following [1][2][8] : Multiple resolution representation; Embedded bit-stream (progressive decoding and SNR scalability); Region-of-interest (ROI) coding… .

JPEG2000 has a much higher algorithmic complexity. The complexity of the entropy encoder is the most significant in JPEG2000. This alone constitutes about 70% of the overall processing time for compression of an image. The principal advantage of entropy coding: EBCOT is creating a highly scalable compressed bitstream, which can be accessed randomly. As other image compression algorithms such as SPIHT [6] and EZW [7], EBCOT uses the wavelet transform to divide the energy of the original image into subbands.

This paper is organized as follows. In section 2, we explain the encoder engine for the JPEG2000. The most intensive processing JPEG2000 (EBCOT entropy encoder) is

described, Section 3. In Section 4 shows experimental results The Obtained using JPEG2000, a comparison with SPIHT coder, and establishes the results for different characteristics of JPEG2000. Finally, section 5 contains the conclusion.

2. Principle of the JPEG2000

In principle coding standard JPEG2000 [4, 5, 11, 12, 13,14], the original image is partitioned into rectangular non-overlapping tiles if the image is very large [1]. All tiles have exactly the same dimensions, except may be those at the boundary of the image [6]. (Figure 1).then, is followed by the component Transformations step. After component transform, the tile-components are decomposed into different decomposition levels using discrete wavelet transform (DWT). These decomposition levels contain a number of subbands, which consist of coefficients that describe the horizontal and vertical spatial frequency characteristics of the original tile component [8].The transform coefficients are quantized with a Uniform scalar quantization with dead-zone. After quantization, each subband is divided into rectangular blocks, called code-blocks, The code-block is decomposed into bit-planes and they are encoded from the most significant bit-plane to the least significant bit-plane. In JPEG2000 the *embedded block coding with optimized truncation* (EBCOT) algorithm [9] has been adopted for the BPC*(fractional bit-plane coding*). EBCOT encodes each bit-plane in three coding passes. The three coding passes in the order in which they are performed on each bit-plane are *significant propagation pass, magnitude refinement pass,* and *cleanup pass.* and the associated context, are then sent to an arithmetic coder (MQ coder is identical to that of JBIG2 (Joint Bi-Level Image Experts Group)). These coded data are then formatted, respecting the syntax defined in the standard (codestream syntax) to form the final bitstream.

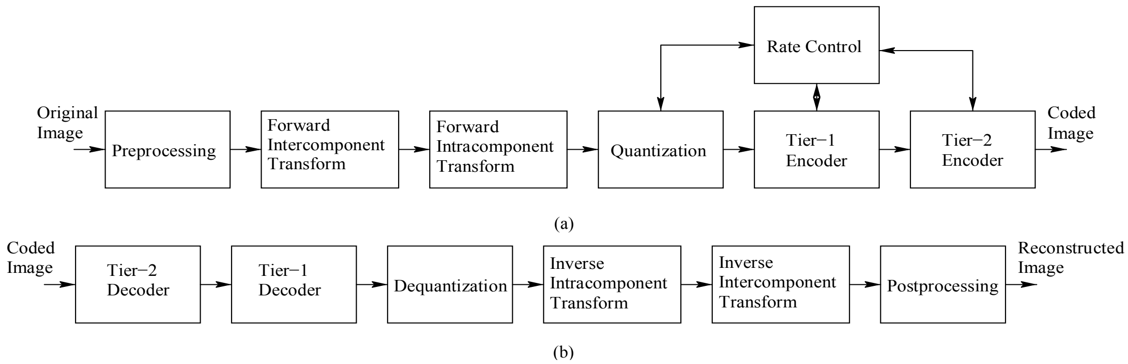


Fig. 1. Codec structure. The structure of the (a) encoder and (b) decoder.

3. Entropy Encoding

The entropy coding and generation of compressed bitstream in JPEG2000 is divided into two coding steps: *Tier-1* and *Tier-2* coding [1].

- *Tier-1* (EBCOT coder [9][10] and Binary Arithmetic Coding-MQ-Coder .
- *Tier-2* (Organization of the bit-stream).

3.1. Divided into blocks

After quantization, each subband is divided into rectangular blocks, called code-blocks (see Figure 2); these code-blocks are encoded independently. The code-block is decomposed into *P* bit-planes and they are encoded from the most significant bit-plane to the least significant bit-plane sequentially (figure 3). Each bit-plane is f irst encoded by a *fractional bit-plane coding* (BPC) mechanism to generate intermediate data in the form of a *context* and a binary *decision* value for each bit position. In JPEG2000 the *embedded block coding with optimized truncation* (EBCOT) algorithm [9] has been adopted for the BPC.

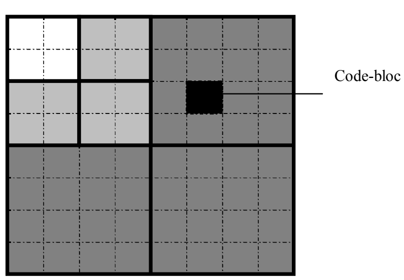


Fig2 : Partitioning into code-blocks

3.2. Tier-1 Coding

In the chain coding JPEG 2000 entropy coding of information is carried by the EBCOT algorithm. This algorithm has been created by David Taubman, The version proposed for JPEG 2000 [9, 10] has very interesting characteristics, especially adapted to the new compression standard.

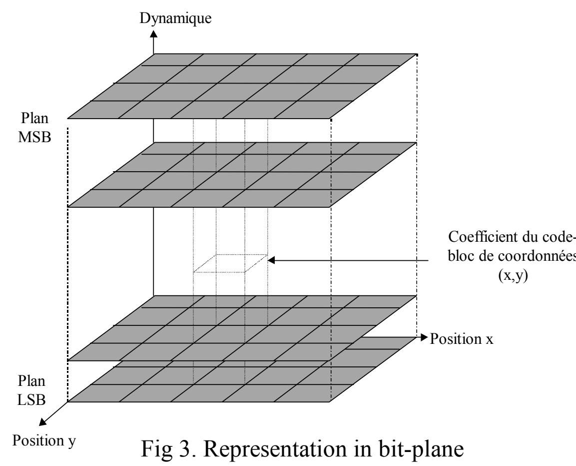


Fig 3. Representation in bit-plane a code-block.

The basic principle of EBCOT is: when coding, EBCOT block receives a set of quantization coefficients together within a code block. The latter is driven bit-plane by bit-plane, starting with the MSB by three coding passes.

Each bit of the code-block is supported by one of these three passes, it sends data to MQ pair to encode the bit. This pair consists of the context determined by the neighbors of the bit, and the value of the symbol to encode. This process is illustrated in Figure 4.

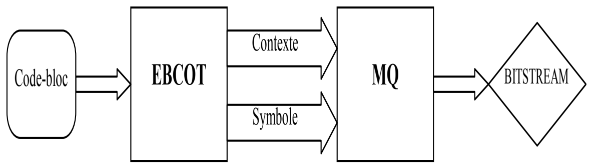


Fig 4. Information flowing between the MQ and EBCOT

3.2.1. EBCOT Coder

EBCOT encodes each bit-plane in three coding passes. The three coding passes in the order in which they are performed on each bit-plane are *significant propagation pass, magnitude refinement pass,* and *cleanup pass.*

All three types of coding passes scan the samples of a code block in the same fixed order shown in Figure 5 The code block is partitioned into horizontal stripes, each having a nominal

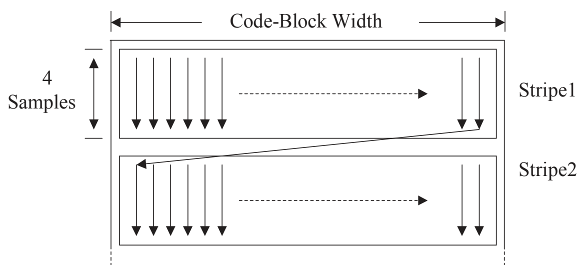


Fig.5. Sample scan order within a code block.

height of four samples. As shown in the diagram, the stripes are scanned from top to bottom. Within a stripe, columns are scanned from left to right. Within a column, samples are scanned from top to bottom [2].

Each coefficient bit in the bit plane is coded in only one of the three coding passes [9], and for each coefficient in a block is assigned a b inary state variable called its significance state that is initialized to zero (insignificant) at the start of the encoding. The significance state changes from zero to o ne (significant) when the first nonzero magnitude bit is found. The context vector for a given coefficient is the binary vector consisting of the significance states of its eight immediate neighbor coefficients [5] For each pass, contexts are created which are provided to the arithmetic coder [8][9][10].
In the following each coding pass is described:

a) Significance propagation pass :
During the significance propagation pass, a bit is coded if its location is not significant, but at least one of its eight-connect neighbors is significant.

b) Magnitude refinement pass :
During this pass, all bits that became significant in a previous biplane are coded. The magnitude refinement pass includes the bits from coefficients that are already significant.

c) Clean-up pass :
The clean-up pass is the final pass in which all bits not encoded during the previous passes are encoded (i.e., coefficients that are insignificant and had the context value of zero during the significance propagation pass).The very first pass in a new code block is always a clean-up pass.

3.2.2. Binary Arithmetic Coding-MQ-Coder

As explained in the previous section [1], the fractional bit-plane coding (EBCOT) produces a sequence of symbols, pairs of context and decision (CX, D), in each coding pass. The context-based adaptive binary arithmetic MQ-coder that is used in JBIG2 is adapted in JPEG2000 standard to encode these symbols.

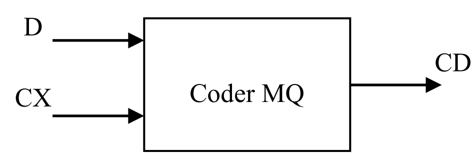


Fig 6. Inputs and outputs of the encoder arithmetic [4].

3.3. Tier-2 Coding

The final operation in a JPEG 2000 compression consists in generating the codestream. This one is a co llection of code-block's bit streams gathered into structures called packets and delimited by a header, containing all the signaling needed to decode. The various natural progression modes of JPEG 2000 codestreams are obtained by ordering packets according to a specific progression. Five major progression modes are actually admissible, depending upon which information is interesting to be reconstructed first [5] :

1). Layer-resolution-component-position-progression (LRCP)
2) .Resolution-layer-component-position-progression (RLCP)
3). Resolution-position-component-layer-progression (RPCL)
4). Position-component-resolution-layer-progression (PCRL)
5). Component-position-resolution-layer-progression (CPRL)

3.4. Rate control

Rate control is responsible for achieving la yer bit-rate targets [2]. This can be achieved through two distinct mechanisms: 1) the choice of quantized step sizes, and 2) the selection of the subset of coding passes to include in the code stream.

When the first mechanism is employed, if the step sizes are in creased, the rate decreases, at the cost of greater distortion. Although this rate control mechanism is conceptually simple. The drawback of such a mechanism is that every time the quantizer step size is changed, entire tire-1 encoding must be performed again. Since tier-1 coding requires a considerable amount of computation, this approach to rate control may not be practical in computationally-constrained encoders.

When the second mechanism is used, the encoder can elect to discard coding passes in order to control the rate. The encoder knows the contribution that each coding pass makes to rate, and can also calculate the distortion reduction associated with each coding pass. Using this information, the encoder can then include the coding passes in order of decreasing distortion reduction, until the desired rate is achieved.

4. Experimental Results

In this section, we briefly demonstrate the compression performance of JPEG2000. We compare JPEG2000 with the coder SPIHT. The comparison with the coder SPIHT is very interesting since this method uses, as JPEG2000, the wavelet transform. The compression efficiency of these algorithms is the comparison of the compression ratio CR (bpp) and the visual quality of the image PSNR (dB).

Table (1) present the results PSNR for each test image and each compression system (JPEG2000, SPIHT), JPEG2000 and SPIHT encoder for 5 level decomposition with 9/7 filter, JPEG and SPIHT used the arithmetic coding.

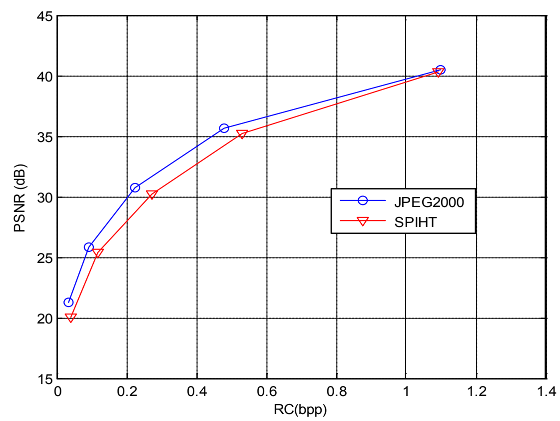
Table.1. Comparison of the PSNR between JPEG2000 and SPIHT; for "Lena" and "Goldhill".

Image / Codeur	Lena		Goldhill	
	RC (bpp)	**PSNR (dB)**	**RC (bpp)**	**PSNR (dB)**
JPEG 2000	0.0324	21.2948	0.0090	19.8118
	0.0935	25.8765	0.0561	23.0016
	0.2236	30.7456	0.2223	27.5055
	0.4790	35.6696	0.5631	31.9329
	1.0968	40.4446	1.1729	36.9310
SPIHT	0.0396	20.0794	0.0363	19.9559
	0.1166	25.4088	0.1596	24.5335
	0.2741	30.2508	0.4236	29.2589
	0.5315	35.2138	0.9169	34.3403
	1.0935	40.2956	1.2647	37.1046

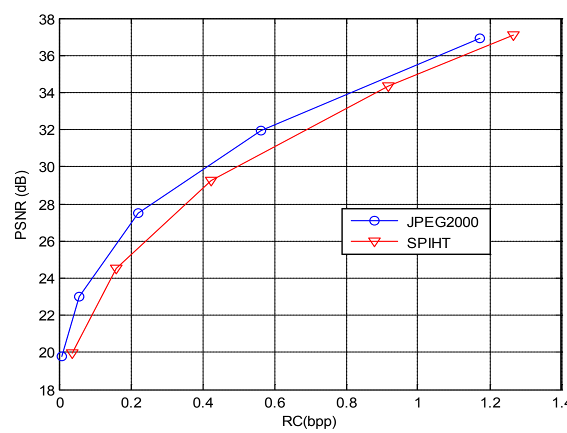
From the results obtained, we can see that the visual quality of images by JPEG2000 is better than SPIHT, especially for low bit-rates as shown in Figure 7. However, the performance of JPEG2000 and SPIHT are very similar compression ratios greater than 1 bpp. The running time of JPEG2000 and SPIHT are very close (2 algorithms that use the wavelet transform).

4.1. Characteristics of JPEG2000

The new JPEG2000 standard offers new features (ROI, progressivity in quality or resolution) to cope with current needs in the field of imaging, it uses the EBCOT coding highly scalable. In this work, we presented the two characteristics, region of interest (ROI) and progressive in quality or resolution.



(a)



(b)

Fig.7. PSNR results for the lossy compression of different compression standards : (a). Lena, (b). Goldhill

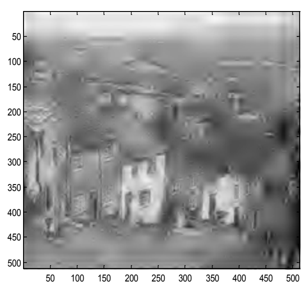
- ✓ **Region of interest ROI:**

The functionality of ROI is important in applications where certain parts of the image are more important than others [5][8]. The code of the ROI in Part 1 of the standard is based on a technique called "MAXSHIF [3]. The figure 8 shows two images with and without ROI, respectively. One clearly sees the difference in visual quality in the region of interest for the image "Goldhill".

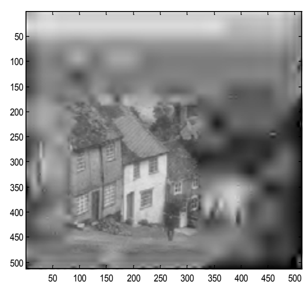
- ✓ **Progressivity in quality or resolution**

JPEG2000 compression system is based on the progressivity. The principal advantage of the compression Progressive is the desired rate or resolution. The two types of progressivity are very important for Internet applications, for example.

Progressivity in quality (SNR), Figure 9 shows the reconstructed images for different layers.
Progressivity in resolution, Figure 10 shows the reconstructed images for different resolutions.

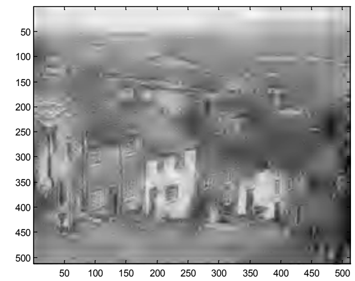


without ROI



with ROI

Fig.8 : Images reconstructed « Goldhill » without ROI et with ROI.



PSNR=23.0016 db , RC=0.0561 bpp

PSNR= 27.5055 db , RC=0.2223 bpp

PSNR= 38.2602 db , RC=1.3641 bpp

Fig.9 : Images reconstructed « Goldhill » for different layers.

5. Conclusion

In this paper, we study the coding part EBCOT of the compression standard JPEG2000 still images and a comparison with SPIHT views qualitative and quantitative (image quality PSNR and compression ratio CR). The results show that JPEG2000 is very similar to SPIHT in terms of quality, because both methods use the wavelet transform, but the major difference between the two standards lies in the way of entropy coded coefficients provided by the wavelet transform . SPIHT uses the correlation between sub-bands to improve coding efficiency, which has the effect of reducing the flexibility of the bitstream and increases the propagation of errors. JPEG2000 encodes the subbands separately and divided into relatively small blocks, the code-blocks. The fact of encoding each code-block independently has several advantages, among which we mention the random access to an area of the image, turning capabilities, greater protection against errors and effective monitoring of the flow . In future work we propose to study other features of the JPEG2000 standard, such as Error resilience.

6. References

[1] T. Acharya, P.S. Tsai, "JPEG2000 Standard for Image Compression Concepts, Algorithms and VLSI Architectures", *John Wiley & Sons*, Hoboken, New Jersey, 2004.

[2] C. Christopoulos, A. Skodras, T. Ebrahimi, "The JPEG2000 Still Image Coding System : An Overview", IEEE *Transactions on Consumer Electronics*, Vol. 46, No. 4, pp.1103-1127, november 2000.

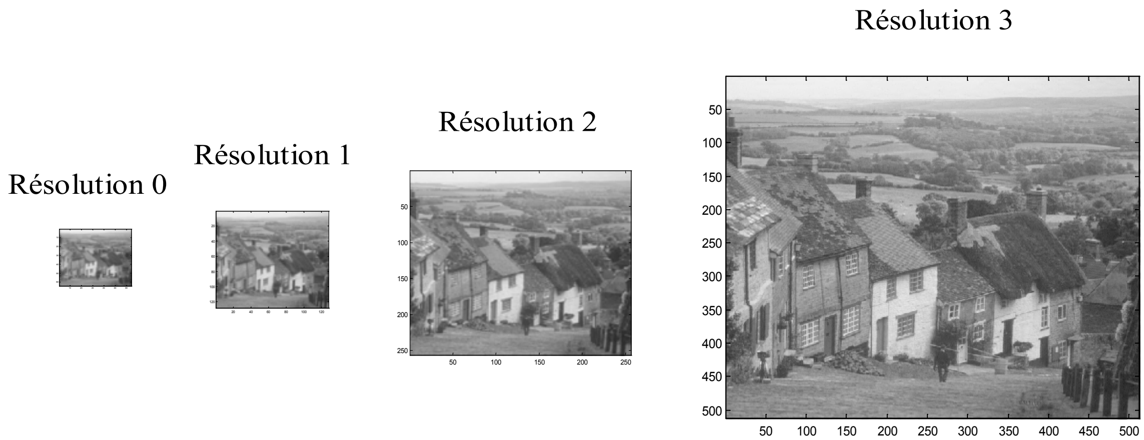


Fig.10 : Image reconstructed « Goldhill » for different resolutions.

[3] C. Christopoulos, J. Askelöf, M. Larsson, "Efficient methods for encoding regions of interest in the upcoming JPEG2000 still image coding standard", *Published in IEEE Signal Processing Letters*, Vol. 7, No. 9, pp. 247-249, September 2000.

[4] ISO/IEC 15444-1, “Information Technology - JPEG2000 Image Coding System-Part 1: Core Coding System”, 2000.

[5] M. Rabbani, R Joshi**,** “An overview of the JPEG2000 still image compression standard”, *Signal Processing : Image Communication*, Vol. 17, pp. 3-48, 2002.

[6] A. Said, W. A. Peralman, "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 6, No. 3, pp. 243-250, June 1996.

[7] J.M. Shapiro, Embedded Image Coding Using Zerotrees of Wavelet Coefficients, *IEEE Trans. on Signal Processing*, Vol. 41, No. 12, pp. 3445-3462, December 1993.

[8] A. Skodras, C. Christopoulos, T. Ebrahimi, “The JPEG2000 Still Image Compression Standard”, *IEEE Signal Processing Magazine*, pp.36-58, September 2001.

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

[10] D. Taubman, E. Ordentlich, M. Weinberger, G. Seroussi, I. Ueno, and F. Ono, “Embedded block coding in JPEG 2000 ”, *in Proc. IEEE Int. Conf. Image Processing*, Vancouver, Canada, Sept. 2000, vol. II, pp. 33-36.

[11] S. Medouakh , Z.E . Baarir, “Compression d’images fixes par la norme JPEG2000*” JIG’2007-3èmés Journées Internationales sur l’Informatique Graphique* , pp 10-15, Constantine, Algérie, oct. 2007.

[12] S. Medouakh, Z.E. Baarir, “ JPEG2000 : La nouvelle norme de la compression d'images” *2nd International Conference on Electronics & Electrical Engineering ICEEE'08*, pp 667-671, Laghouat, Algeria, April 2008.

[13] S. Medouakh, Z.E. Baarir, “ L'utilisation des filtres 9/7 et 5/3 dans la norme J PEG2000 ” *6èmés Séminaire Nationales en Informatique* Biskra, Mai 2008, pp241-245.

[14] S. Medouakh, Z.E. Baarir, "Study of the Standard JPEG2000 in Image Compression", *International Journal of Computer Applications (0975 – 8887)*Volume 18– No.1, pqge 27-33, March 2011