DWT image compression using Contextual bitplane coding of Wavelet coefficients

Khanh Nguyen-Phi Hans Weinrichter

Vienna University of Technology
Gusshausstraße 25/389, Wien A-1040 Austria
Email: knguyen@nt.tuwien.ac.at, jweinrichter@email.tuwien.ac.at

Abstract

Image coding using Discrete Wavelet Transform can attain very good results compared to traditional methods. In fact, most of the best known results for image coding have been related to DWT. A simple scheme to code the DWT coefficients is the Embedded zero-tree wavelet algorithm[1] by Shapiro. Embedded zero-tree wavelet can exploit both the similarity among subbands and the sparseness of the subbands. In this paper, we propose a new simple image coder based on DWT. The DWT coefficients are coded in bitplanes. Each bi-level pixel is coded in a context formed by its parental and surrounding pixels. In this way, both the similarity and the sparseness among subbands are exploited. We show the experimental results, both in terms of distortion measurement and visual comparison, and compare them to well-known methods.
Keywords: *Wavelet, Image compression, Bitplane coding, Zero-tree coding*

1 INTRODUCTION

In a typical DWT image encoder, there are three main operations: first a DWT is applied to the image, then the DWT coefficients are quantized, and the quantized coefficients are entropy coded. The decoder will inverse these three steps. For entropy coding, the popular solution is using arithmetic codes or fix-to-variable length codes (such as Huffman code) combined with run-length codes. The coefficients can be quantized using scalar quantization or vector quantization. Recently, there is a lot of work in appying vector quantization in DWT image coders but it is still a high computational method. Scalar quantization leads to worse performance in principle, but it is much simpler. Thus most of fast implementations for DWT image coding utilize scalar quantization. In some coders, the quantization and entropy coding process can be combined. A well-known method in this class is the Embedded zero-tree wavelet by Shapiro[1]. In complexity, zero-tree method falls between vector quantization and scalar quantization. But in performance, it attains very good results, even better than most coders using vector quantization. The success of Zero-tree coder is based on its ability to exploit both the sparseness of the subbands and the similarities among subbands. With this realization, other methods can also attain good performance if they take into account these two characteristics of the subbands.
We have suggested such a method in [3]. In this method, the DWT coefficients are coded in bitplanes. If the bitplanes are coded in the order from the Most-significant-bit to the Least-significant-bit (which is always the case) we get an embedded bitstream. In each bitplane, each pixel is coded in its context, which is formed by its parental pixels. A binary arithmetic coder (the QM-Coder) is used. It is shown that this simple coder attains good performance, even better than Zero-tree method. In this paper, we suggest some improvements to this method, and compare their results.

2 THE PROPOSED METHOD FOR DWT COEFFICIENTS CODING

First we describe briefly the method in [3]. In this method, the DWT coefficients are coded in bitplanes, from the MSB to the LSB. The binary entropy coder is QM-Coder. In each bitplane, the coding order is from the lowest frequency subband to the highest frequency subband. For each DWT coefficent, we call its first non-zero value the First significant bit (FSB), call all bits preceding FSB the Zero bits (ZBs), represent its sign information by the Sign bit (SB) and call all bits following the FSB the Raw bits (RBs). For example, if a DWT coefficient has a value of -0000000110101, then its FSB is the first 1 from left to right (bitplane 5), its ZBs are 0000000, its SB is 1 (negative) and its RBs are 10101. When the bitplanes are coded from the MSB to the LSB, for each DWT coefficient the coding order will be: at first all the ZBs will be coded until a non-zero value is detected. This is the FSB. The SB is coded right after the FSB. Then the RBs are coded. For signalling which coefficient has already been coded to the FSB, we form an additional bitplane. The pixels in this bitplane are 0 or 1 if the corresponding DWT coefficients are coded before or after the FSB. This additional bitplane is called Significant bitplane (SBP). The SBs from all DWT coefficients are expected to have the same statistics. The RBs are also expected to have the same statistics. That means we can use a single context for the SBs and another single context for RBs. In practice, it is found that we can also use one fixed context for both SBs and RBs. This context has a probability of 0.5:0.5 for symbol 0 and 1. Thus for SBs and RBs we can leave them uncoded if possible, or code them with state 0 in QM-Coder. Now the problem of efficient coding of DWT coefficient values becomes the problem of efficient coding of ZBs. In [3], each ZB in the bitplanes is coded according to its context form from its parental pixels. Consider an example in Fig.1, which shows only 3 steps of decomposition. In this example, the order of coding the subbands will be: the lowest frequency subband (top-left, contains A) is coded first with order-0. After that, the subbands contain pixels B,E,F,C,G,H,D,I,J are coded respectively. In each subband, the pixels are coded from top to bottom, from left to right.

Our new experiment with the method shows that it can be improved in the following ways:

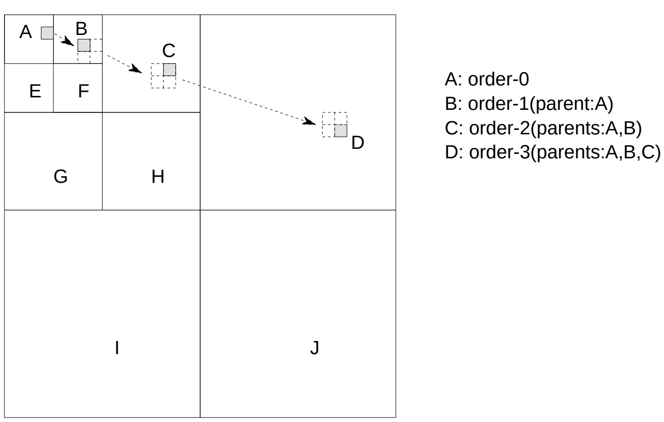

Figure 1. Example of coding a bitplane

- the pixels forming the contexts are taken from the SBP only.
- it's not neccessary to span the context over three subbands, that means we use only direct parental pixels, and not higher.
- the surrounding pixels can also help to form the contexts.

In our improved version, the pixels to form the context are: one parental pixel and four pixels at positions North,South,West,and NorthEast to the current pixels. Note that all these pixels are taken from the SBP. As an example, in Fig.2, the pixels forming the context for coding D are: P (parent of D in SBP), N,W,S,and NE (which are neighbors of D in SBP).

Because the bitplanes are coded sequentially, the statistical information collected from higher bitplanes, which are expressed in the positions of the contexts in the FSM of QM-Coder, is used to code the lower bitplanes. Inherit from its sequential bitplane coding, the proposed method has the feature of progressive transmission. As an embedded image coder, the coding process can stop at any desired bitrate or any desired quality level.

3 EXPERIMENTAL RESULTS

We have implemented the proposed algorithm in software. A set of standard test images is used to evaluate the performance of the new algorithm. We compare this method (PM=proposed method) with our old method (OM=old method) in [3]. We also compare them with JPEG's. For our JPEG comparison, we use software from The Independent JPEG Group (IJPEG) (with the -optimize option switched on). We show here the experimental results with the well-known test image Lena (512x512x8). Antonini's biorthogonal wavelet with symmetrical perfect reconstruction filters of order 9/7 [2] is used. All of our experimental results are based on a 5-level pyramid of the DWT. We also use 'reflection' technique for the bounded values. In Table 1, the PSNR are shown against various compression ratios (CR). The proposed method is superior to JPEG, especially at low bitrates. We don't show the results with Zero-tree methods here, because there might be some slight differences between our implementation and Shapiro's, but from our evaluation as well as from the results of Zero-tree methods reported in the literature, we can see that the proposed method clearly outperforms Zero-tree.

For a visual comparison, we show Lena at two bitrates after being compressed by the proposed algorithm. Again, IJPEG is used for comparison. Note that IJPEG couldn't

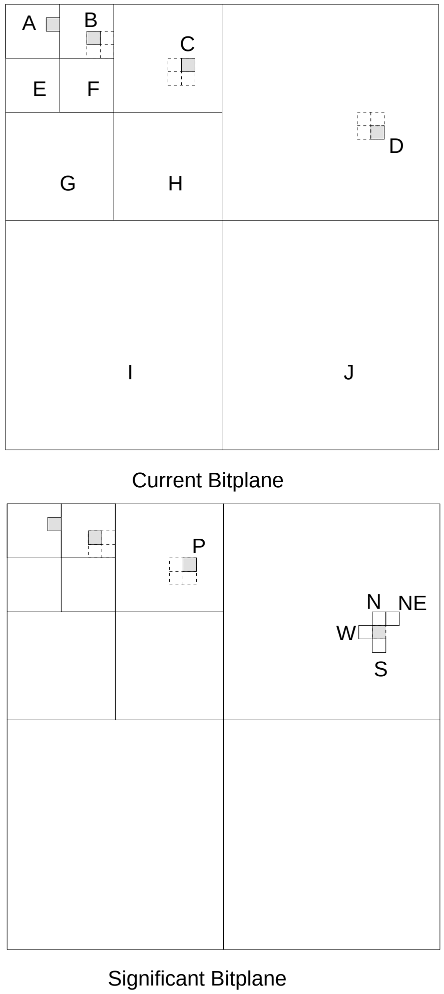

Figure 2. Improved version of example 1

reach the desired bitrates (which correspond to compression ratios of 50 and 100) so we use the nearest available bitrates. For comparison, the proposed coder could easily reach to the same bitrates as IJPEG's.

4 CONCLUSION

Image coding using DWT are among the best methods for image compression. Many methods in DWT image compression are based on bitplane coding. Combining the traditional methods for bi-level image compression and bitplane coding of DWT coefficients, we have applied and improved a contextual coding model for coding the DWT coefficients bitplanes. The obtained results are superior to JPEG's results and competitive with best methods using DWT.

CR	PSNR(JPEG)	PSNR(OM)	PSNR(PM)
10	36.99	38.77	39.11
20	33.81	35.76	36.03
40	30.40	32.66	32.94
60	28.24	30.74	31.20
80	26.47	29.72	30.00
100	24.77	29.03	29.33
130	21.92	27.86	28.22

Table 1. Compression comparison of proposed method

5 ACKNOWLEDGEMENT

The first author has been supported by "Der Österreichischer Akademischer Austauschdienst" (ÖAD).

References

[1] JM.Shapiro 'Embedded Image coding using Zero-trees of Wavelet coefficients', IEEE Trans. Signal Processing 12/1993, p.3445-3462

[2] M.Antonini,M.Barlaud,P.Mathieu,I.Daubechies 'Image coding using Wavelet transform' IEEE Trans. Image Processing 2/1992 p.205-220

[3] K.Nguyen-Phi and H.Weinrichter 'Image compression using bitplane coding of wavelet coefficients' Electronics Letters 9/96 Vol. 32 No.19 p. 1773-1775

JPEG: CR=48

Proposed method: CR=48

JPEG: CR=103

Proposed method: CR=103

Figure 3. Visual comparison of JPEG and proposed method