A METHOD FOR CODING THE IMAGE SUBBANDS

Ovidiu COSMA

Abstract This article presents a new numeric code that reduces the truncation error, and its application to the progressive image coding algorithms.

Keywords: image compression, numeric codes

Introduction

The progressive image coding algorithms usually process the transform coefficients in several passes in which a bit from each of the coefficients is inserted in the output stream [4], [5], [7]. All the bits processed at the same pass form a bit plane. The existing algorithms are focused on the efficient specification of the bit planes. The decoder completes the bits in the planes that it has no information about with zeroes. Thus the coefficients are truncated at the bit that corresponds to the last of the bit planes, and the approximation improves with each new pass.

This article presents a new numeric code that reduces the truncation error of the transform coefficients and implicitly improves the quality of the reconstructed images.

1. The MTE code

The first variant of this code is named MTE (*Minimizing the Truncation Error*) because its aim is to reduce the truncation error at each of the passes. The proposed MTE code is a positional system for representing arithmetic values, which uses the binary digits 0 and 1. The weight of each of the digits depends on their position in the representation. The null digits have their usual meaning, and the other ones can have the meaning $1 \cdot 2^i$ or $-1 \cdot 2^i$, where *i* indicates the position of the bits. The first non-zero bit will always be positive, and the signs of the following ones will alternate (the second's sign will be *minus*, the third's *plus*, and so on.)

The following algorithm can be used to generate the MTE code of the positive arithmetic value *x*. The result will be written in variable *c*.

MTE coding algorithm

$n = \lceil log_2 x \rceil$
$c = 0$
while($x > \varepsilon$ **and** $n \geq$ the position of the least significant bit of c)
    if ($2^n - x < x$) **then**
        $x = 2^n - x$
        set the bit at position n of c
    $n = n - 1$

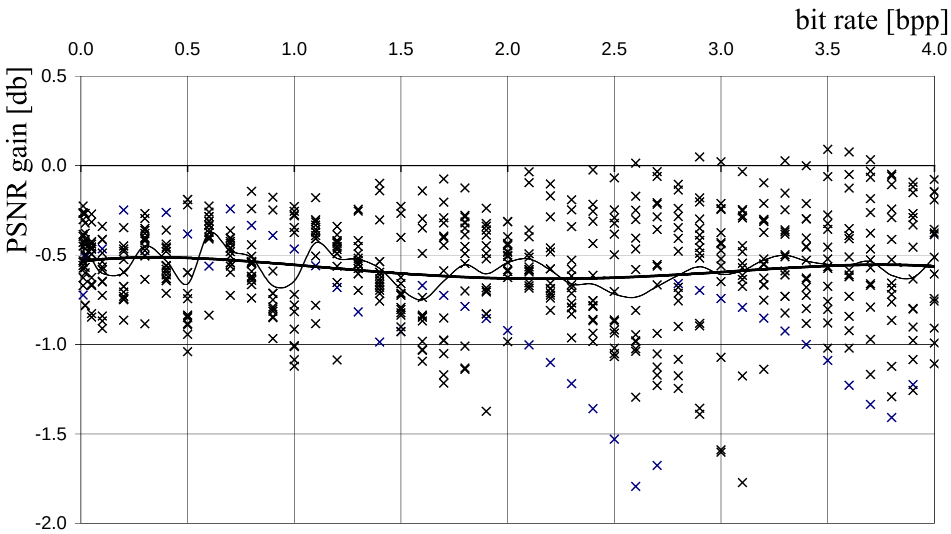
$$e_{i,j}^{k} = c_{i,j} - \widetilde{c}_{i,j}^{k} = c_{i,j} \in [p/2, \quad p) \tag{1}$$

In the case of the MTE coding, if $c_{i,j} \in (p/2, \quad p]$, then $\widetilde{c}_{i,j}^{MTE,k} = p$, and the truncation error $e_{i,j}^{MTE,k}$ is:

$$e_{i,j}^{MTE,k} = c_{i,j} - \widetilde{c}_{i,j}^{MTE,k} = c_{i,j} - p \in (-p/2, \quad 0] \tag{2}$$

Relations (1) and (2) indicate that $|e_{i,j}^{MTE,k}| < |e_{i,j}^{k}|$, therefore the truncation error reduces in the case of the MTE code.

For establishing the performance of the MTE code, a set of 25 test images were coded with a compression scheme based on the DWT [1],[3], MTE and SPIHT algorithm [4]. The PSNR [6] gain given the original SPIHT algorithm was computed for the reconstructed images. The results are presented in figure 1. The narrow line represents the average and the thick one represents a polynomial regression curve.


bit rate [bpp]
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0
PSNR gain [db]
0.5 0.0 -0.5 -1.0 -1.5 -2.0

n_M

3 The MTEA code

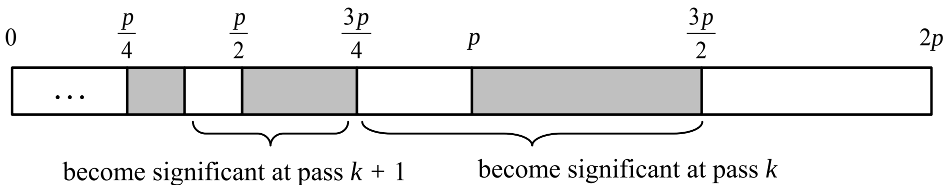
The MTEA (*Minimizing the Truncation Error – Arithmetic*) code is a combination between the MTE code and the binary arithmetic representation of coefficients, which eliminates the disadvantage of the MTE code.

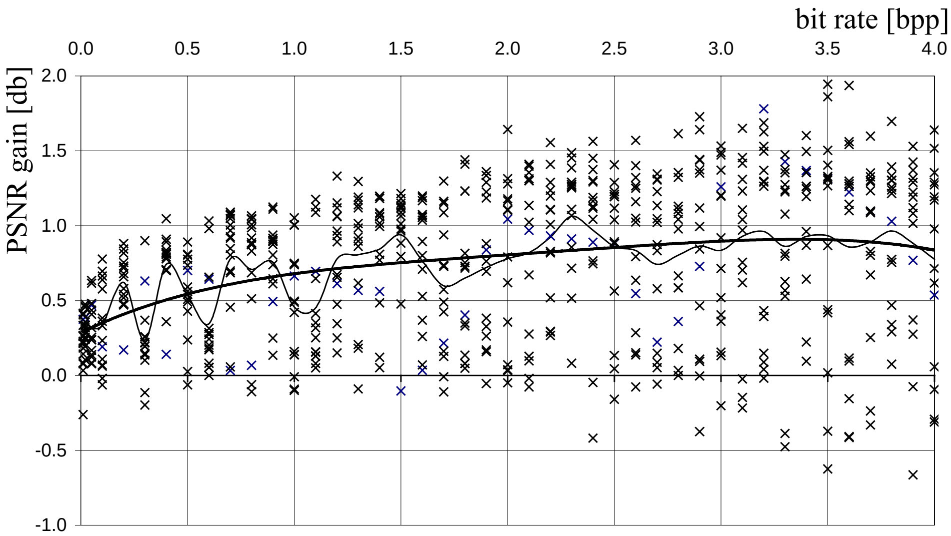
In the case of the MTE code, at pass number *k* all the coefficients in the interval $(p/2, \quad p]$ became significant, where $p = 2^{n-k+1}$ and *n* represents the position of the first non-zero bit of the largest coefficient. Only the coefficients situated in the second half of this interval are efficiently coded, because those in the interval $(p/2, \quad 3p/4)$ are to be corrected at the next pass, thus in two passes is obtained the same precision that would be obtained in the case of the arithmetic representation in a single pass. Because at each pass a series of bits have to be inserted in the output stream to indicate the decomposition of the trees that contain significant coefficients in zerotrees, for the coefficients in the first half of the interval $(p/2, \quad p]$, the arithmetic representation is better.

In the case of the MTEA code, the coefficients in the interval $[3p/4, \quad p)$ are MTE coded, and for the ones situated in the interval $[p/2, \quad 3p/4)$, the arithmetic representation is used. Because they are differently represented, the coefficients in the two halves of the interval $[p/2, \quad p)$ do not become significant at the same pass. Those situated in the interval $[3p/4, \quad p)$ become significant at pass *k*, at which the threshold reaches the value *p*, and the others become significant at the next pass. Therefore at pass *k* become significant the coefficients in the interval $[p, \quad 3p/2)$ for which the arithmetic representation will be used, and the coefficients in the interval $[3p/4, \quad p)$ that will be MTE coded.

The intervals of the coefficients that become significant at the passes *k* and *k*+1 are presented in figure 2. The intervals corresponding to the MTE coded coefficients are white and the other ones are dark.

Because there are two different variants of representing the values, an extra bit should be inserted in the data stream for each coefficient, to indicate the actual one. This solution would compromise the compression ratio. Luckily there is another variant that does not require an extra bit. Because of the way the intervals were selected, the second bit for each of the coefficients is always 0 and can be used to indicate the coding variant. For example 0 can indicate the arithmetic representation, and 1 the MTE coding.


0
$\frac{p}{4}$
$\frac{p}{2}$
$\frac{3p}{4}$
p
$\frac{3p}{2}$
$2p$
...
become significant at pass $k + 1$
become significant at pass k


bit rate [bpp]
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
PSNR gain [db]
2.0
1.5
1.0
0.5
0.0
-0.5
-1.0

1. Ovidiu Cosma, *Wavelet Transform application in image compression*, Buletinul Ştiinţific al Universităţii din Baia Mare seria B, 2000
2. Ovidiu Cosma, *Contributions to the Coding of Image Subbands*, thesis, Politehnica University Bucharest, 2003
3. Leena-Maija Reissell, *Multiresolution and Wavelets*, SIGGRAPH'95 Course Notes
4. Amir Said, William A. Pearlman, *A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees*, IEEE Transactions on Circuits and Systems for Video Technology, vol.6, 1996
5. Jerome M. Shapiro, *Embedded Image Coding Using Zerotrees of Wavelet Coefficients*, IEEE Transactions on Signal Processing, vol. 41 no. 12, 12.1993
6. Wim Sweldens, *Wavelets, Signal Compression and Image Processing*, SIGGRAPH'95 Course Notes
7. Gregory K. Wallace, *The JPEG Still Picture Compression Standard*, Communications of the ACM, Volume 34, Issue 4, 04.1991

Received: 01.10.2004

Department of Mathematics and Computer Science
Faculty of Sciences, North University of Baia Mare
E-mail: cosma@alphanet.ro