L-INFINITY PROGRESSIVE IMAGE COMPRESSION

Armando J. Pinho and António J. R. Neves

Signal Processing Lab, DETI / IEETA
University of Aveiro, 3810-193 Aveiro, Portugal
ap@det.ua.pt — an@ieeta.pt

ABSTRACT

This paper presents a lossless image coding approach that produces an embedded bit-stream optimized for L_∞-constrained decoding. The decoder is implementable using only integer arithmetic and is able to deduce from the bit-stream the L_∞ error that affects the reconstructed image at an arbitrary point of decoding. The lossless coding performance is compared with JPEG-LS and JPEG2000. Operational rate-distortion curves, in the L_∞ sense, are presented and compared with JPEG2000.

***Index Terms*—** L-infinity image coding, progressive transmission, finite-context models, binary trees.

1. INTRODUCTION

Near-lossless or L_∞-constrained image coding has been capturing the attention of the researchers for more than a decade (for some early works see, for example, [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]). This interest is motivated, on one hand, by the need for effective image compression algorithms and, on the other hand, by restrictions regarding the amount and nature of the distortion introduced by the compression techniques. In fact, whereas for some types of imagery the main concern is the visual impact of the distortion, some others require a precise control of the error on a pixel basis. Among these latter image classes are, for example, those related with medical imaging and remote sensing.

Frequently, the alternative to the general purpose lossy image compression methods, which can only guarantee a given L_2 error, is to use lossless techniques. However, the lossless image coding methods can only provide modest compression rates, motivating the development of intermediate techniques, commonly known as near-lossless or L_∞-constrained. In this case, the reconstructed image must obey a given L_∞-error (also known as the maximum absolute difference, MAXAD, error). The importance of this paradigm was recognized by the ISO/IEC standardizing committee and reflected in the current standard for the lossless compression of continuous-tone images, JPEG-LS [12, 13], which provides a near-lossless mode. Other techniques proposed in the context of this call, such as CALIC [14], have been further improved in order to provide a better near-lossless performance [15].

More recently, a new functionality started to be investigated in the context of L_∞-constrained image compression, namely, progressive coding. In fact, although this characteristic is not a new one in the field of image coding (the "old" JPEG standard has this capability, as well as the newer JPEG2000), it was only addressed for the first time in association to near-lossless image coding by Avcibaş *et al.* [16]. The proposed approach relies on a predictive-based method that successively refines the probability density function (pdf) used to estimate each pixel and by restricting the region of support of the pdf to fixed size intervals, which have to be predefined before encoding [16, 17]. Almost simultaneously, Alecu *et al.* proposed a wavelet-based scheme that allows full L_∞ scalability [18, 19, 20]. This algorithm was compared with JPEG2000 in terms of L_∞ rate-distortion, showing better results [20].

In a recent work, we proposed an intensity domain method for lossy-to-lossless image compression, based on binary tree decomposition and finite-context modeling [21]. The experimental results obtained with this technique showed its competitiveness in relation to the current standard that is able to offer similar lossy-to-lossless capability, i.e., JPEG2000. In this paper, we propose a number of changes that enable the encoder to produce a L_∞-constrained embedded bit-stream. Decisions made by the encoder that need to be exactly reproduced by the decoder require only the use of integer operations, a characteristic that might be of paramount importance in avoiding potential incompatibilities when a bit-stream is decoded in a computational platform that differs from the one that was used for encoding. Moreover, the decoder is able to deduce from the already decoded bits what is the L_∞ error affecting the reconstructed image, allowing to stop decoding whenever a desired error is attained.

2. THE CODING METHOD

The compression technique proposed in this paper is based on a hierarchical organization of the intensity levels of the image. This organization is obtained by means of a binary tree. Each node of the binary tree, n, represents a certain subset, $\mathcal{S}^n$, of the intensities of the image. The root node is associated to the complete set of image intensities, $\mathcal{I} = \{I_1, I_2, \ldots, I_N\}$. Therefore, $\mathcal{S}^n \subset \mathcal{I}$ and $\mathcal{S}^1 \equiv \mathcal{I}$. Each node possesses a representative intensity, I^n, given by

$$I^n = \left\lfloor \frac{I^n_m + I^n_M}{2} \right\rfloor, \tag{1}$$

where I^n_m and I^n_M are, respectively, the smallest and largest intensities in $\mathcal{S}^n$, and where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. Note that computing the value of I^n according to (1) leads to the smallest possible L_∞ reconstruction error when the intensities associated to node n (those in $\mathcal{S}^n$) are all substituted by I^n. This error is given by

$$\epsilon^n_\infty = I^n_M - I^n.$$

Figure 1 shows an example of a small binary tree of the same kind as those produced by the image coding method presented in this paper. In this example, the image is composed of five intensities, $\{10, 22, 50, 95, 130\}$. The construction of the tree begins

This work was supported in part by the Fundação para a Ciência e a Tecnologia (FCT).

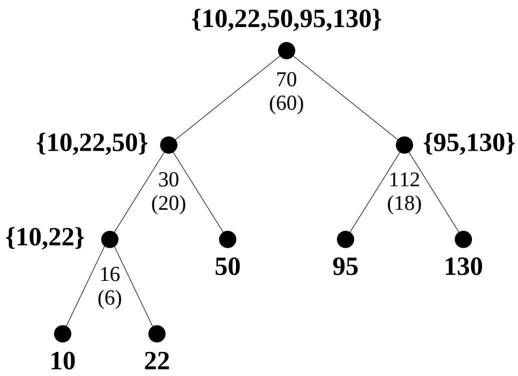

Fig. 1. Example of a small binary tree used to illustrate how the proposed algorithm works.

with the association of this information to the root node and with the calculation of I^1 according to (1). In the example of Fig. 1, $I^1 = (10 + 130)/2 = 70$ and $\epsilon_\infty^1 = 130 - 70 = 60$.

The next step requires expanding the root node into two children nodes and, therefore, splitting $\mathcal{S}^1$ into two subsets. This is done simply by putting all intensities $I \in \mathcal{S}^1$ that are smaller or equal to I^1 into the left node and those that are larger into the right node. Generically, the set $\mathcal{S}^n$ ($|\mathcal{S}^n| > 1$, where $|\cdot|$ denotes the cardinality of the set) associated with node n is partitioned into two subsets, $\mathcal{S}_l^n$ and $\mathcal{S}_l^n$, such that

$$\mathcal{S}_l^n = \{I \in \mathcal{S}^n : I \leq I^n\}$$

and

$$\mathcal{S}_r^n = \{I \in \mathcal{S}^n : I > I^n\}.$$

This procedure is repeated until expanding all nodes, i.e., until having a tree with N leaves (remember that N is the number of image intensities). The decision regarding which node to expand is related with the goal of always having the smallest possible L_∞ reconstruction error. Therefore, we have to choose the node associated with the largest L_∞ error. In case of having several nodes with the same error, one is arbitrarily chosen, although requiring that the decoder picks the same node.

At the decoder side, a similar binary tree needs to be constructed. Therefore, the decoder needs to know, right from the beginning, all intensity values of the image. This can be efficiently encoded using the following strategy. First, the maximum intensity value, I_N, is sent. Then, a string of I_N bits is transmitted, such that if the n^{th} bit of the string is one then the intensity $n - 1$ exists in the image and if is zero it does not exist. This information is sufficient for the decoder to completely mimic the encoder regarding the construction of the tree.

When a node is expanded and two new nodes are created, it is necessary to communicate to the decoder which pixels will have the intensity changed to I_l^n and which need to change to I_r^n, respectively the representative intensities of the left and right newly created nodes. Since the location of the pixels having intensity I^n is known by the decoder and since only those pixels will change, then it is enough to encode a binary mask where, for example, a zero indicates that the pixel needs to change its intensity to I_l^n and a one indicates a change to I_r^n. This mask is an arbitrarily shaped region, defined by those pixels having intensity equal to I^n, and is encoded using arithmetic coding based on variable size finite-context models [22, 23, 24].

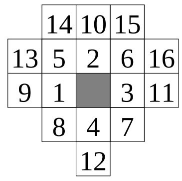

Fig. 2. Context template used in this work. The use of non-causal pixels is possible, because context information can be obtained from the previous version of the reconstructed image.

The contexts are constructed based on the template shown in Fig. 2, where the context pixels are numbered according to their distance to the encoding pixels. A particular context is represented using a sequence of bits,

$$b_1 b_2 \dots b_k \tag{2}$$

where

$$b_i = \begin{cases} 0, & \text{if} \quad |I(i) - I_l^n| \leq |I(i) - I_r^n| \\ 1, & \text{otherwise} \end{cases}$$

and where $I(i)$ denotes the intensity of the pixel in the current reconstructed image corresponding to position i of the context template.

Due to the fact that, generally, as encoding proceeds, the size of the sample used for learning the finite-context model decreases and in order to alleviate the context dilution problem, the value of k in (2) is progressively reduced. This aspect was pointed out in [25] and further addressed in [26], for the case of the compression of color-quantized images. According to the model proposed in [26], the size of the context is chosen using a relation that depends on the number of nodes already expanded, n, and on the number of pixels of the image, M:

$$k(n) = \lceil \alpha(M) - \log_2(n + 1) \rceil ,$$

where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x and where

$$\alpha(M) = c_1 \log_2 M + c_2,$$

i.e.,

$$k(n) = \left\lceil \log_2 \frac{M^{c_1} 2^{c_2}}{n + 1} \right\rceil .$$

The function $\alpha(\cdot)$ was adjusted using training data from a number of images, resulting in $c_1 = 0.671$ and $c_2 = -0.859$ (for further details, see [26]). In this work, we adopted the same function for context size adaptation. Note that this operation is only performed by the encoder. Because it involves non-integer arithmetic, the decoder receives, for each node that is expanded, the value of k.

3. EXPERIMENTAL RESULTS

Figure 3 shows operational rate-distortion curves for the 512×512 "Lena" and "finger" images, both for JPEG2000 [27, 28] and for the proposed L_∞ progressive compression algorithm. It is worth noting that, in the case of the proposed method, the decoder is able to deduce the L_∞ error that affects the reconstructed image at an arbitrary point of decoding. Unfortunately, a JPEG2000 decoder cannot offer this useful capability.

Table 1 presents lossless compression results, in bits per pixel, obtained using the proposed method, JPEG-LS [12, 13] and JPEG2000

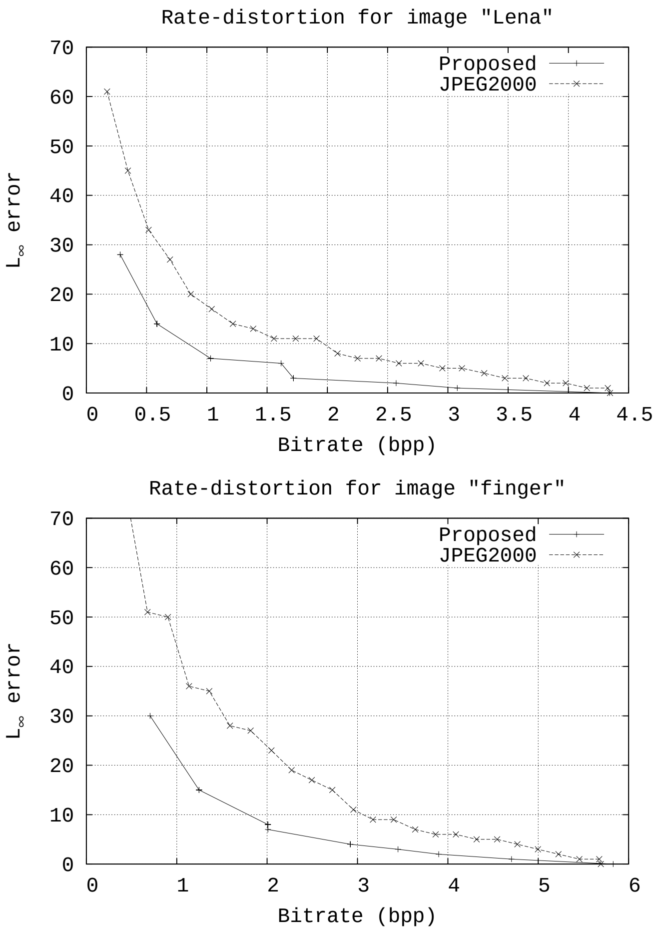

Fig. 3. Operational rate-distortion curves of the “Lena” and “finger” images for the proposed method and for JPEG2000.

[27, 28], using the graylevel version of the 768×512 Kodak images[1]. JPEG-LS results are given just for reference, since this technique does not produce progressive bit-streams and, therefore, cannot be compared directly with the proposed method. Nevertheless, it can be seen that, on average, and for the image test set used, the proposed algorithm was only 0.2% worse than JPEG-LS. Regarding JPEG2000, the lossless compression advantage goes to the proposed method (3.1%), whereas the L_∞ rate-distortion advantage is illustrated by the curves of Fig. 3.

JPEG2000 lossless compression was obtained using version 5.1 of the JJ2000 codec with default parameters for lossless compression[2]. JPEG-LS coding was obtained using version 2.2 of the SPMG JPEG-LS codec with default parameters[3].

[1]Available from `ftp://ftp.ieeta.pt/~ap/images/kodak/768x512/gray`.

[2]`http://jj2000.epfl.ch`.

[3]The original web-site of this codec, `http://spmg.ece.ubc.ca`, is currently unavailable. However, it can be obtained from `ftp://www.ieeta.pt/~ap/codecs/jpeg_ls_v2.2.tar.gz`.

Table 1. Lossless compression results, in bits per pixel, obtained with the proposed method and with the JPEG-LS and JPEG2000 image coding standards.

Image	JPEG-LS	JPEG2000	Proposed
01	5.265	5.467	5.259
02	3.982	4.199	3.905
03	3.452	3.572	3.417
04	4.154	4.210	4.176
05	5.165	5.337	5.176
06	4.561	4.702	4.617
07	3.603	3.781	3.685
08	5.282	5.559	5.288
09	3.957	4.042	3.935
10	3.972	4.123	3.987
11	4.373	4.580	4.414
12	3.798	3.942	3.803
13	5.955	6.142	5.925
14	4.898	5.070	4.892
15	3.863	3.975	3.797
16	4.051	4.198	4.119
17	4.110	4.238	4.095
18	5.096	5.195	5.052
19	4.499	4.577	4.507
20	3.137	3.342	3.133
21	4.510	4.651	4.580
22	4.542	4.647	4.639
23	3.486	3.539	3.510
Average	**4.336**	**4.483**	**4.344**

4. CONCLUSION

In this paper, we presented an image coding method that produces an embedded bit-stream optimized for L_∞-constrained decoding. As in [20], the most recent work addressing this research topic, a comparison was made with JPEG2000. This comparison showed that the proposed method clearly outperforms JPEG2000, both in terms of operational L_∞ rate-distortion and lossless compression.

5. REFERENCES

[1] K. Chen and T. V. Ramabadran, “Near-lossless compression of medical images through entropy-coded DPCM,” *IEEE Trans. on Medical Imaging*, vol. 13, no. 3, pp. 538–548, Sept. 1994.

[2] L. Karray, O. Rioul, and P. Duhamel, “L^∞-coding of images: a confidence interval criterion,” in *Proc. of the 1st IEEE Int. Conf. on Image Processing, ICIP-94*, Austin, TX, Nov. 1994, vol. II, pp. 888–892.

[3] L. Ke and M. W. Marcellin, “Near-lossless image compression: minimum-entropy, constrained-error DPCM,” in *Proc. of the 2nd IEEE Int. Conf. on Image Processing, ICIP-95*, Washington, DC, Oct. 1995, vol. I, pp. 298–301.

[4] M. Das, D. L. Neuhoff, and C. L. Lin, “Near-lossless compression of medical images,” in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, ICASSP-95*, Detroit, MI, May 1995, vol. 4, pp. 2347–2350.

[5] N. Moayeri, "A near-lossless trellis-searched predictive image compression system," in *Proc. of the 3rd IEEE Int. Conf. on Image Processing, ICIP-96*, Lausanne, Switzerland, Sept. 1996, vol. I, pp. 93–96.

[6] H. Hartenstein, R. Herz, and D. Saupe, "A comparative study of L_∞-distortion limited image compression algorithms," in *Proc. of the Picture Coding Symposium, PCS-97*, Berlin, Germany, Sept. 1997, pp. 51–55.

[7] L. Karray, P. Duhamel, and O. Rioul, "Image coding with an L^∞ norm and confidence interval criteria," *IEEE Trans. on Image Processing*, vol. 7, no. 5, pp. 621–631, May 1998.

[8] L. Ke and M. W. Marcellin, "Near-lossless image compression: minimum-entropy, constrained-error DPCM," *IEEE Trans. on Image Processing*, vol. 7, no. 2, pp. 225–228, Feb. 1998.

[9] H. Hartenstein and X. Wu, "Analysis of trellis quantization for near-lossless image coding," in *Proc. of the Data Compression Conf., DCC-98*, Snowbird, Utah, Mar. 1998, p. 551.

[10] R. Ansari, N. Memon, and E. Ceran, "Near-lossless image compression techniques," *Journal of Electronic Imaging*, vol. 7, no. 3, pp. 486–494, July 1998.

[11] B. Meyer and P. Tischer, "Extending TMW for near lossless compression of grayscale images," in *Proc. of the Data Compression Conf., DCC-98*, Snowbird, Utah, Mar. 1998, pp. 458–470.

[12] ISO/IEC, *Information technology - Lossless and near-lossless compression of continuous-tone still images*, ISO/IEC 14495–1 and ITU Recommendation T.87, 1999.

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

[14] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. on Communications*, vol. 45, no. 4, pp. 437–444, Apr. 1997.

[15] X. Wu and P. Bao, "L_∞ constrained high-fidelity image compression via adaptive context modeling," *IEEE Trans. on Image Processing*, vol. 9, no. 4, pp. 536–542, Apr. 2000.

[16] İ. Avcibaş, N. Memon, B. Sankur, and K. Sayood, "A progressive lossless/near-lossless image compression algorithm," *IEEE Signal Processing Letters*, vol. 9, no. 10, pp. 312–314, Oct. 2002.

[17] İ. Avcibaş, N. Memon, B. Sankur, and K. Sayood, "A successively refinable lossless image-coding algorithm," *IEEE Trans. on Communications*, vol. 53, no. 3, pp. 445–452, Mar. 2005.

[18] A. Alecu, A. Munteanu, P. Schelkens, J. Cornelis, and S. Dewitte, "Wavelet-based L-infinite scalable coding," *Electronics Letters*, vol. 38, no. 22, pp. 1338–1340, Oct. 2002.

[19] A. Alecu, A. Munteanu, P. Schelkens, J. Cornelis, and S. Dewitte, "Wavelet-based fixed and embedded L-infinite-constrained image coding," *Journal of Electronic Imaging*, vol. 12, no. 3, pp. 522–538, July 2003.

[20] A. Alecu, A. Munteanu, J. Cornelis, and P. Schelkens, "Wavelet-based scalable L-infinity-oriented compression," *IEEE Trans. on Image Processing*, vol. 15, no. 9, pp. 2499–2512, Sept. 2006.

[21] A. J. Pinho and A. J. R. Neves, "Lossy-to-lossless compression of images based on binary tree decomposition," in *Proc. of the IEEE Int. Conf. on Image Processing, ICIP-2006*, Atlanta, GA, Oct. 2006, pp. 2257–2260.

[22] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text compression*, Prentice Hall, 1990.

[23] D. Salomon, *Data compression - The complete reference*, Springer, 2nd edition, 2000.

[24] K. Sayood, *Introduction to data compression*, Morgan Kaufmann, 2nd edition, 2000.

[25] X. Chen, S. Kwong, and J.-F. Feng, "A new compression scheme for color-quantized images," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 12, no. 10, pp. 904–908, Oct. 2002.

[26] A. J. Pinho and A. J. R. Neves, "A context adaptation model for the compression of images with a reduced number of colors," in *Proc. of the IEEE Int. Conf. on Image Processing, ICIP-2005*, Genova, Italy, Sept. 2005, vol. 2, pp. 738–741.

[27] ISO/IEC, *Information technology - JPEG 2000 image coding system*, ISO/IEC International Standard 15444–1, ITU-T Recommendation T.800, 2000.

[28] D. S. Taubman and M. W. Marcellin, *JPEG 2000: image compression fundamentals, standards and practice*, Kluwer Academic Publishers, 2002.