# A Progressive Lossless/Near-Lossless Image Compression Algorithm

İsmail Avcıbaş, *Member, IEEE*, Nasir Memon, *Member, IEEE*, Bülent Sankur, *Senior Member, IEEE*, and Khalid Sayood

***Abstract*—A novel image compression technique is presented that incorporates progressive transmission and near-lossless compression in a single framework. Experimental performance of the proposed coder proves to be competitive with the state-of-the-art compression schemes.**

***Index Terms*—Near-lossless compression, probability mass estimation, successive context refinement.**

## I. Introduction

NEAR-LOSSLESS compression techniques provide the guarantee that no pixel difference between the original and the compressed image is above a given value [1, 6], while lossless schemes result in reconstructed data that exactly matches the original. Reversible integer wavelets are known to allow for an integrated lossless and lossy compression scheme, though their performance remains inferior to the state-of-the-art techniques such as context-based adaptive lossless image coding (CALIC) [2]. Furthermore, no precise bounds can be set on the extent of distortion at the intermediate stages of the decoding. Approaches to furnish near-lossless and progressive character to these methods have resulted, so far, in inferior compression as compared to near-lossless compression in conjunction with predictive coding [1].

The compression algorithm that we present incorporates the above two desirable characteristics, namely progressive reconstruction and near-losslessness. We produce a bit stream that results in progressive reconstruction of the image. In addition, we guarantee near-lossless reconstruction with respect to a given bound at each layer. We discuss the successive refinement and probability density estimation schemes we use in Section II. The proposed compression method is summarized in Section III, along with experimental results.

## II. Near-Lossless Successive Refinement

Lossless compression schemes exploit context information, which is represented by the conditional probability density function (pdf) of a pixel given its neighborhood. Once this pdf is obtained, it can be used as a guide in determining the value of a pixel. In the sequel, let $X_{i-1}, X_{i-2}, \ldots, X_{i-N}$ denote random variables representing the causal neighbors of the current pixel $X_i$ (the white circles in Fig. 1), and let $x_{i-1}, x_{i-2}, \ldots, x_{i-N}$ indicate their realizations. To identify the value of the current pixel within $\delta$ counts of its actual value, we divide the support of the current pixel's pdf, say [0, 255], into $2\delta+1$-long intervals and sort these intervals according to their probabilities. Then, we start trying to locate the interval in which the current pixel lies, starting from the interval with highest probability.

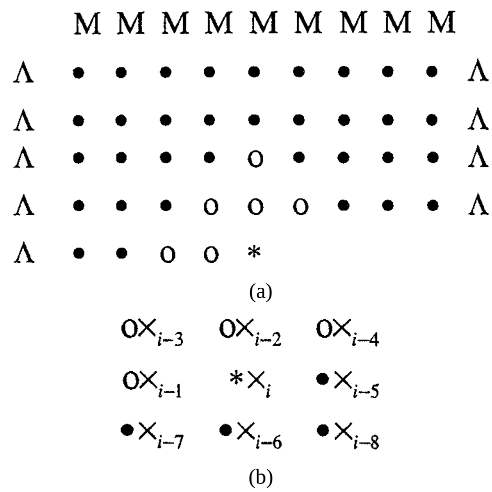

Fig. 1. (a) Context pixels used in the covariance estimation of the current pixel $*$ and denoted by $\bullet$ and 0. The number of context pixels is $K = 40$, while the neighborhood pixels count $N = 6$. (b) Causal (0) and noncausal ($\bullet$) neighbors of the current pixel ($*$) used for probability mass update in the second and higher passes.

Let $p_i(x)$ denote the pdf of the $i$th pixel at gray value $x$. Let the probability mass of the $m$th interval, of length $2\delta + 1$ and denoted by the upper and lower limits $[L_m, R_m]$, be $p_m = \sum_{x\in(L_m, R_m]} p_i(x)$. If the pixel gray value is found in this $m$th interval $x \in (L_m, R_m]$, the entropy coder is fed with bit 1, otherwise with bit 0. After every success (the pixel value is in that interval), we set the pdf to zero in the complement of the interval, i.e., $p_i(x) = 0,\, x \in [L_m, R_m]^c$. On the other hand, after a failure the probability mass in the failing interval is set to zero, i.e., $p_i(x) = 0,\, x \in [L_m, R_m]$, the pdf is renormalized to mass of 1, and we proceed to look at the next highest probability interval. At the end of the pass, the maximum error in the reconstructed image $||e||_\infty$ is $\delta$, since the midpoint of the interval is selected as the reconstructed pixel value. The procedure we have described above is general and effectively gives us the framework for a successively refinable near-lossless compression scheme. The question arises: how do we get estimates of the pdf? We address this question in Sections II-A and II-B.

Manuscript received June 13, 2001; revised April 22, 2002. This work was supported by TUBITAK BDP, Boğaziçi Research Fund Project 01A201 and by National Science Foundation Award INT 9996097. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ricardo L. de Queiroz.

The authors are with the Department of Computer Science, Polytechnic University, Brooklyn, NY 11201 USA (e-mail: memon@poly.edu).

Digital Object Identifier 10.1109/LSP.2002.804129.

### A. PDF Estimation in the First Pass: The Gaussian Model

We assume that for natural images, at a coarse level and in a small neighborhood, context statistics are Gaussian and stationary. Thus, we fit a multivariate Gaussian density function and use linear prediction to predict the $X_i$ pixel value given its causal neighborhood of size $N, X_{i-1}, X_{i-2}, \ldots, X_{i-N}$

$$X_i = \sum_{j=1}^{N} \beta_j(x_{i-j}) + \nu_i \tag{1}$$

where $\{\beta_j\}_{j=1}^N$ are real-valued linear prediction coefficients of the process, and $\{\nu_i\}$ is an i.i.d. sequence of random variables having a Gaussian density with zero mean and variance $\sigma^2$. The minimum-variance linear unbiased estimator $\boldsymbol{\beta} = [\beta_1 \cdots \beta_N]$ can be obtained by standard techniques [3], and the density of $X_i$ conditioned on its causal neighbors is given by

$$p(x_i|x_{i-1}, x_{i-2}, \ldots, x_{i-N}) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\left(\frac{1}{2\sigma^2}\right)\left[x_i - \sum_{j=1}^{N} \beta_j(x_{i-j})\right]^2\right). \tag{2}$$

### B. PDF Updates in Refinement Passes

In a multipass scheme, where pdfs are updated in each pass, we drop the Gaussian assumption in later passes and use both causal and noncausal neighborhood densities. Obviously, the causal ones ( having been just updated) are more precise, while the noncausal ones (yet to be updated) are less precise, but are still useful. Our pdf update is based on the $L_2$-norm estimate of the current pixel's pdf using the pdfs of the causal and noncausal neighborhoods. Let $p_i^*(x)$ denote the probability mass function to be estimated given the causal and noncausal distributions $\{p_{i-j}\}_{j=1}^N$. Minimizing the average difference of $\sum_j (p_i^*(x) - p_{i-j}(x))^2$ subject to the constraint $\int p_i^*(x) = 1$ and using Lagrange multipliers, we then have the updated pdf estimate $p_i^*(x) = (1/T)(\sum_{j=1}^N p_{i-j}(x))$ where $T$ is the normalizing constant. Implicit in this sum is an interval-censored addition of the pdfs involved, i.e., if the neighboring pdf (say $p_{i-j}(x)$) does not overlap with the interval $(X_L^i, X_R^i]$ of the current pdf $p_i(x)$, which is being estimated, then it has no contribution on the estimate. In other words, if $p_i(x)$ has an empty bin, it will veto any contribution to that bin from its neighbors $p_{i-j}(x), j = 1, \ldots, N$. Otherwise, $p_i(x)$ is updated from the corresponding bins of the neighboring $p_{i-j}(x), j = 1, \ldots, N$. Notice that this method of summing neighboring densities gives more importance to the more precise pdfs in the causal neighborhoods. In fact, the causal pdfs will be concentrated in narrower supports after the latest pass, while the noncausal neighborhoods have more dispersed pdfs over a larger support, since their updates are yet to occur.

## III. The Compression Method

In the multipass procedure, we refine the interval of the pixel value with increasing precision, i.e., decreasing uncertainty intervals $\lambda_1, \ldots, \lambda_f$. We first estimate the Gaussian pdf as in (2), not based on the exact pixel values (which would not be available at the decoder site), but on their less precise values resulting from quantization with step size $\lambda_1$ in the first pass. Thus, we regress on the "quantized" version of the context pixels $X_{i-1}, X_{i-2}, \ldots, X_{i-N}$ in (1). In other words, we substitute for each pixel $X_j \leftarrow \text{Midpoint}_{\lambda\_\text{interval}}\lfloor X_j \rfloor$ the midpoint of the $\lambda$-interval within which it is found. When the correct interval is determined in the first "Gaussian" pass, the pixel's maximum error is $\delta$, since $\lambda_1 = 2\delta + 1$. Though the initial Gaussian pdf estimate will be noisier as a consequence of midpoint-quantization, the "missed" information will be recovered later using pdf updates as in Section II-B. But the pdf updates necessitate that we pass over the image more than once; hence, we call it the multipass mode of compression. Thus, after all the pixels have been scanned in the first pass, we start the second pass to refine the supports of the gray levels, to size $\lambda_2$ and successively all the way to $\lambda_f$, using the minimum-norm pdf update rule described above.

TABLE I
COMPARISON OF THE THREE NEAR-LOSSLESS COMPRESSION METHODS. THE SCORES RESULT FROM THE COMPUTED AVERAGES FOR THE ENSEMBLE OF EIGHT IMAGES (BARBARA, BOAT, FINGER, GOLDHILL, LENA, MANDRILL, PEPPERS, ZELDA). NOTE THAT THE FIRST ROW IN THE TABLE, $\delta = 0$, CORRESPONDS TO THE PROGRESSIVE MODE WHERE THE MULTIPASS ALGORITHM IS ARRANGED IN THREE STEPS, SO THAT INITIALLY BINS OF SIZE 8 ARE CONSIDERED, FOLLOWED BY A BIN SIZE OF 4, AND FINALLY OF SIZE 1, RESULTING IN LOSSLESS COMPRESSION. THE REMAINING ROWS CORRESPOND TO SINGLE-PASS MODE

| | SPIHT | | | CALIC | | | Proposed | | |
|---|---|---|---|---|---|---|---|---|---|
| Precision | bpp | PSNR | $\|e\|_\infty$ | Bpp | PSNR | $\|e\|_\infty$ | bpp | PSNR | $\|e\|_\infty$ |
| $\delta=0$ | 4.75 | $\infty$ | 0 | 4.63 | $\infty$ | 0 | 4.56 | $\infty$ | 0 |
| $\delta=1$ | 3.03 | 47.75 | 4.62 | 3.09 | 49.90 | 1 | 3.03 | 49.89 | 1 |
| $\delta=3$ | 1.94 | 42.07 | 10.13 | 2.02 | 42.21 | 3 | 1.94 | 42.23 | 3 |
| $\delta=7$ | 1.12 | 37.59 | 18.38 | 1.22 | 36.10 | 7 | 1.12 | 36.16 | 7 |

The decoder reverses the operations in the encoder. Based on the estimated pdf and decoded sequences of successes and failures, the pdf support for the current pixel is found. Within the support of the estimated pdf, the intervals are sorted with respect to their probability mass, and the correct interval is found using decoded failure and success events. Whenever the decoded event does not report a success for the $m$th interval, this interval is discarded, with the refining continued until the correct interval is reached. The reconstructed value of the current pixel is taken as the midvalue of the interval reporting success.

To assess the performance of our algorithm, we compare it with algorithms for CALIC [2] and set partitioning in hierarchical trees (SPIHT) [4]. For the simulation tests, we used a context size of $K = 40$ and the prediction neighborhood size as $N = 6$ in Fig. 1. The multipass algorithm was arranged in three steps with initial bin size of 8, followed by bin size of 4, and finally 1. Compression results are given in Table I where we list bit rate, PSNR, and maximum error magnitude $||e||_\infty$ for the parameters $\delta = 0$ (lossless), $\delta = 1$, $\delta = 3$, and $\delta = 7$. The results indicate that the proposed coder has better lossless performance than SPIHT and CALIC. It also has better rate distortion performance than SPIHT for bounds $\delta = 1, \delta = 3$ in PSNR, and $||e||_\infty$ criteria. As $\delta$ increases to 7, SPIHT starts to catch up with the proposed coder in PSNR, but at the same time its $||e||_\infty$ score is two to three times larger than that of our algorithm or that of CALIC, resulting in objectionable quantization errors. The rate distortion performance vis-à-vis CALIC improves as the bound gets higher.

## REFERENCES

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

[2] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun.*, vol. 45, pp. 437–444, Apr. 1997.

[3] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[4] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–249, June 1996.

[5] L. Karray, P. Duhamel, and O. Rioul, "Image coding with an $L^\infty$ norm and confidence interval criteria," *IEEE Trans. Image Processing*, vol. 7, pp. 621–631, May 1998.