Multiresolution Lossless Image Compression *

DRAFT

I. Kontoyiannis † J. Turek V. Castelli
J.T. Robinson

IBM T.J. Watson Research Center,
Box 704, Yorktown Heights, NY10598.
{yiannis, jjt, vittorio, robnson}@watson.ibm.com

December 1995

Abstract – In this paper we describe a new compression algorithm for the efficient transmission, storage and analysis of large sets of complex spatial data, such as satellite images and remotely sensed data for earth science studies. The main goal of our current research is to develop image processing algorithms that operate on data in the compressed domain, thus providing significant increase in speed (and, possibly, in accuracy) over their classical counterparts; one such operator, progressive classification, is described in [1]. The contingent application area is content-based search of large databases of satellite images. The algorithm we describe here, called Multiresolution Compression for Image Analysis (MCIA), provides a representation of the image that is well suited for our goals. The decoding is of low computational complexity, allowing the image processing operators to access selected portions of the data with negligible computational overhead. Our representation of the image can be used to provide both lossless and lossy reconstructed versions of the original data, that at the same time is roughly of the same size as a representation using only lossless compression with any of the standard lossless image compression algorithms (typically within ±5% of the lossless-mode performance of the JPEG standard [3]). MCIA is well-suited for hierarchical storage systems: content-based search can be performed on the lossily coded image, that has high compression ratio and can be stored on fast (and smaller) disks, while the residuals can be stored on large and slower tertiary storage.

*This work was funded in part by grant no. NASA/CAN NCC5-101.
†On leave from the Information Systems Laboratory, Stanford University, Stanford, CA94305-4055, U.S.A.

1 Description of the algorithm

The structure of our algorithm is shown schematically in figure 1. It consists of a lossy component and a lossless component; the lossy component uses transform (subband) coding followed by quantization of the transform coefficients, and then lossless coding of the quantized values. For the lossless component we simply reconstruct the lossy image, and we save the difference between that and the original image. The resulting *residual* image is also losslessly encoded.

For transform coding we use the Discrete Wavelet Transform (DWT); this step provides us with a multiresolution representation of the original image. (For more on Wavelets see the article [2] and the references therein.) Notice that the wavelet transform is, in principle, a perfectly reversible operation, so no loss of information occurs here. Taking the DWT of the image to, say, k levels, results in a matrix of wavelet coefficients, divided into $(3k+1)$ *wavelet subbands*. We quantize this coefficient matrix, using uniform, scalar quantization, with a different number of bits-per-pixel (or, rather, bits-per-coefficient) for each subband. This is the only "lossy" step in the algorithm; it is in the quantization where actual loss of information occurs. Finally we losslessly encode each subband independently, using predictive coding (DPCM), followed by a fixed-model, two pass, arithmetic code. This produces a representation of the data that can be used to reconstruct a lossy version of the original data. In fact, one can use one or more of the quantized wavelet subbands to increase the resolution of the reconstructed image, depending on the needs of specific applications. Notice that each subband is quantized to a number of bits-per-pixel (bpp) which is a tunable parameter in a somewhat similar manner to the Q-matrix used in the sequential DCT mode of the JPEG algorithm: Depending on the size-reduction needed, or the resolution required for a specific application, one can select which subbands are going to be finely quantized and which ones are not.

The residual is computed by inverting the DWT of the quantized wavelet coefficients, and calculating the difference between that and the original image; the residual is then losslessly encoded, using DPCM followed by arithmetic coding.

As we mentioned above, the DWT is a reversible operation, but in order to recover the original data one would in principle need to store the exact values of the wavelet coefficients. Although it is impossible in practice to store infinite precision reals, we were able to derive the following lower bound on the precision required to allow perfect reconstruction of the data from the quantized coefficients.

Lemma: *Consider an image with a dynamic range of n_0 bpp, and suppose we take the DWT of the image down L levels. If the maximum tolerable error at level 0 (original image) is ϵ_L, then we can quantize the DWT coefficients corresponding to any one of the level-L subbands to n_L bpp:*

$$n_L = n_0 + \left\lceil 2L \log_2 \left(\frac{\mathcal{G}\mathcal{G}'}{2} \right) \right\rceil - \lceil \log_2(\epsilon_L) \rceil,$$

where $\mathcal{G}$ denotes the sum of the absolute values of the coefficients of the analysis lowpass filter, and similarly $\mathcal{G}'$ for the corresponding synthesis lowpass filter.

Since the original data will always have finite precision, the above formula gives a lower bound on the number of bpp required for the storage of the DWT coefficients that will still allow for *perfect* reconstruction. Assuming, for example, that we start with an 8bpp, integer-valued image, the maximum tolerable error (per pixel) in the reconstructed image is 0.5. Therefore,

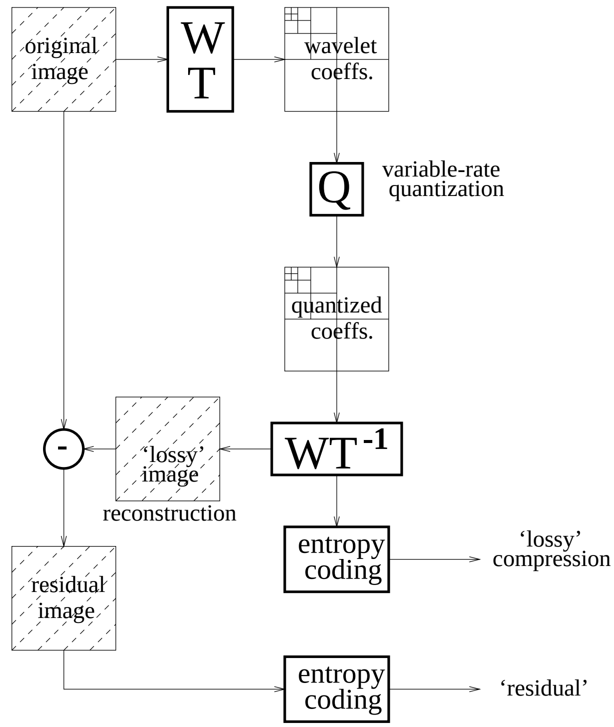

Figure 1: Overall structure of the compression algorithm.

if we take the DWT to $k = 2$ levels, we can distribute the overall error among the 7 wavelet subbands by choosing values for ϵ_L that sum up to 0.5. If, for example, we let $\epsilon_1 = 1/8$ for all the level-one subbands and $\epsilon_2 = 1/32$ for the remaining subbands, the Lemma tells us that we can quantize the level-one subbands to 13bpp, the level-two subbands to 17bpp, and still get perfect reconstruction of the original image.

An important feature of the MCIA algorithm is that, with the exception of the quantization and the entropy coding steps, all other operations involved are linear. Thus, if the entropy coding is done on separate blocks of the image (and the residual) independently, we can extract a selected portion of transform without having to read and decode the entire image. In particular, we can reconstruct a lossy version at any level of resolution, simply by decoding the blocks that contain wavelet coefficients corresponding to the required portion, and inverting the wavelet transform for these coefficients only. This gives a significant speedup during the decoding process since we do not need to process the whole image, and allows image processing operations to be efficiently applied to reduced resolution image constructs. This last consideration is relevant since our interest in source coding is also motivated by the need of efficiently applying signal processing operators to images stored in compressed format, for content-based retrieval of information from multimedia databases.

Another advantage is the low computational complexity of the algorithm. The cost of decoding the data to the maximum resolution level (that is, reproducing the original image exactly) is linear in the number of pixels in the image. Finally, the algorithm is amenable to efficient implementations on modern superscalar computer architectures, such as the IBM RS/6000 family.

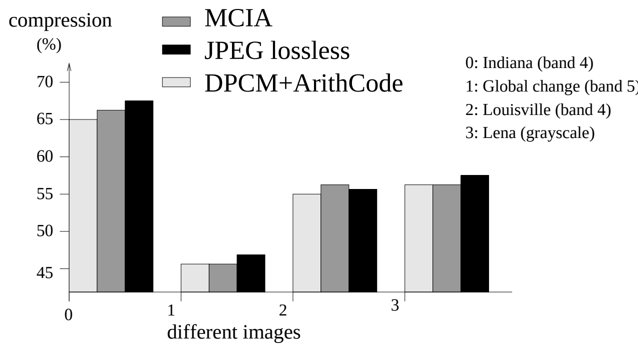

Figure 2: Comparison of MCIA with the lossless mode of the JPEG standard, and DPCM with arithmetic coding.

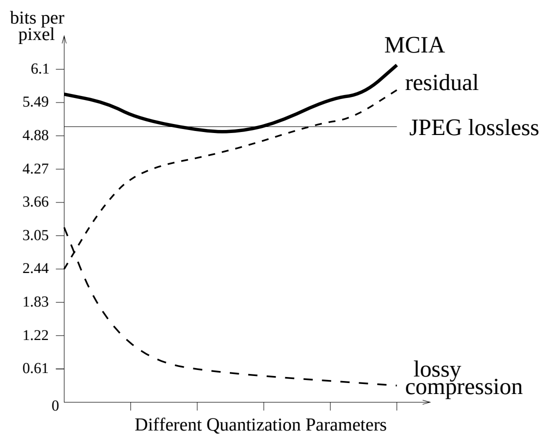

Figure 3: Compression results using the MCIA algorithm with different quantization parameters. The MCIA curve is the sum of the "lossy compression" and the "residual" curves.

2 Performance

We have performed extensive experiments using satellite images of the earth's surface, as well as natural images. We are currently investigating how to select the quantization parameters in an automated fashion. As with JPEG, the compression ratios achieved as well as the visual quality of the lossy images depend heavily on the choice of the quantization parameters, and coefficients corresponding to higher frequencies can be quantized more coarsely than low-frequency coefficients without significantly affecting the visual quality of the reconstructed image.

The graph in figure 3 shows average compression ratios (in bpp) for several test images, as the quantization parameters vary.

Clearly, uniformly finer quantization across the transformation results in smaller "residual" images and larger "lossy" representations. Typical parameters in our experiments were: dynamic range of original images: 8 bpp; $k = 2$ to 5 levels in the wavelet decomposition; quantization values: 1 to 8 bpp; wavelet filters: most of the experiments were done using Daubechies biorthogonal wavelets of order 3,3, but the compression results were rather insensitive to the choice of different wavelet filters. This is not surprising, since the images used in the experiments (some of which are described below) are rather heterogeneous in nature and often very rich in detail.

The chart in figure 2 illustrates the performance of the MCIA algorithm compared with the lossless JPEG standard algorithm, and with DPCM followed by arithmetic coding Notice that the DPCM-with-arithmetic-coding algorithm is an extreme point in the MCIA compression curve, corresponding to the case where we quantize all the DWT coefficients to zero bpp, and we entropy encode the residual. The images shown in the chart are the commonly used test image "Lena" (512×512 pixels, 8 bpp) and three satellite images. Image 0 is a 512×512 Thematic Mapper (TM) image of the Indiana-Kentucky border at 8 bpp (spectral band 4); image 1 is a 800×1000 Multi-Spectral Scanner (MSS) image of Lake Tahoe, California (8 bpp, band 5); and image 2 is a 512×512 TM image of Louisville, Kentucky (8 bpp, band 4).

Acknowledgements

We would like to thank Chung-Sheng Li and Chris Elvidge for their useful comments on the paper.

References

[1] Vittorio Castelli, Chung-Sheng Li, John J. Turek and Ioannis Kontoyiannis, "Progressive Classification in the Compressed Domain for Large EOS Satellite Databases", to appear in *Proc. of 1996 IEEE International Conference on Acoustics, Speech and Signal Processing.*

[2] Olivier Rioul and Martin Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Magazine*, vol. 8, n. 4, Oct. 1991, pp. 14-38.

[3] William B. Pennebaker and Joan L. Mitchell, *JPEG: Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, 1993.