Lossless image compression over lossy packet networks

Alan L. Merriam, Albert L. Kellner, Pamela C. Cosman
Electrical and Computer Engineering Dept., University of California, San Diego,
La Jolla, CA 92093-0407 *

1 Introduction

In this paper a reversible (lossless) image compression technique is presented which is robust in an ATM environment to packet loss and to non-sequential packet receipt. In a clinical imaging environment, imagers (for instance, magnetic resonance (MR) imagers) must be interconnected to archival databases and to visualization workstations; ATM (asynchronous transfer mode) networks offer great capabilities for these interconnections. However, increased performance can be obtained if the images are compressed prior to transmission. For medical images, reversible compression schemes have legal advantages, and possibly diagnostic advantages as well, over irreversible methods. While there are many techniques for reversibly compressing images, few if any of these are suitable under packet loss.

In ATM networks, packets from multiple inputs are statistically multiplexed to the output. Packets have a small, but finite, probability of being lost. For most lossless image compression algorithms, loss of a single packet in transmission can preclude useful reconstruction of the image and thus require retransmission of the lost packet or the entire image file. This increases the total image transmission latency and can seriously compromise the service quality required by the end user. Previous work on packetization for compressed images has focused exclusively on lossy compression, and the strategies have generally involved forward error correction, layered coding, and resilient coding (see, for example, [1, 2]). The image compression approaches presented in this paper have been tailored such that they are completely reversible when no packets are lost and they produce a small reconstruction error in the case of lost packets. We use two methods for lossless compression. The first uses linear prediction in the pixel domain to decorrelate the image, followed by Huffman coding of the residuals. The second method uses the multiresolution sequential (S) transform, with Huffman coding of the coefficients. Results are presented for 8-bit grayscale magnetic resonance images.

2 Pixel linear prediction

With pixel-domain linear prediction, an image is typically scanned in raster scan order, and past pixel values are used to predict the current pixel value. The difference is calculated between the actual and predicted pixel values, and this residual is coded with a lossless (entropy) code. The decoder is able to form the same prediction as the encoder. If this bit stream were packetized as it stands, a packet loss would render the decoder unable to match the encoder's prediction after the position of the lost packet, and the process would derail. Derailment can be entirely avoided if we only use pixels within the same packet to form the prediction.

We use three different pixel encoding orders: serpentine raster scans within a strip of width two, four, or six. The goal is to

*This work was funded by the BMDO Focused Research Initiative on *Photonics for Data Fusion Networks*, contract number F49620-95-1-0538.

take advantage of higher spatial correlations between neighboring pixels and increase the order of the predictor. Two of the prediction scan patterns are shown in Figure 1. They proceed from left to right across the width of the image. After the right edge of the image is reached a new scan then begins on the left edge.

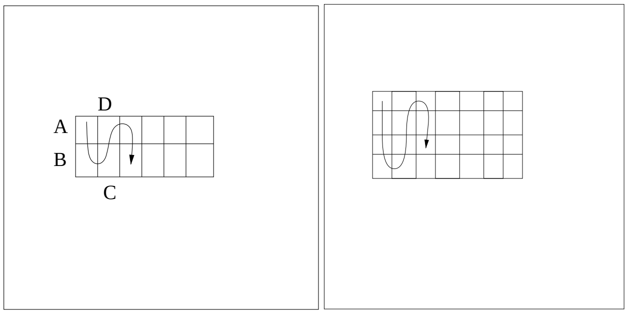

Figure 1: Prediction Scan Patterns Used: (left) serpentine raster within 2-pixel wide strip (2-Scan) and (right) within 4-pixel wide strip (4-Scan)

In the strip of width two, the packet begins with a few bytes that indicate the x and y spatial coordinates of the pixel A, and the intensity value of the pixel A. Then A is used to predict B, and that residual is encoded. Next A and B are used to encode C. For predicting pixel D, we can make use of a 3rd order predictor: A,B,C are available to predict D. Thus, after the very beginning of the packet, the prediction will alternate between a 2nd and 3rd order predictor; the average order of the predictor is 2.5. In the case of the 4-Scan, within the strip of width 4, each column makes use of 4th order prediction twice, and 2nd order and 3rd order prediction once each. The average order of the predictor is 3.25. For the 6-Scan, the average order of the predictor is 3.5. With a space-filling curve such as a Peano scan [5], it is difficult to say exactly what the average order of the predictor is for the approximately 80 pixels that fit in a packet, since it would depend on where in the scan pattern the packet begins. However, for a group of 80 pixels, the Peano scan would allow for prediction orders ranging from 1 to 7 (permitting only the 8 spatially adjacent neighbors for prediction), with an average order between 3.23 and 3.51. So, the 6-scan is essentially equivalent to a Peano scan in this respect.

Training sets consisting of 9–10 MRI brain scan images were used to optimize the prediction coefficients. The images are all of size 256 × 256 with 8 bits per pixel (bpp). The Wiener-Hopf equations which satisfy the principle of orthogonality and thus optimize the prediction coefficients in a minimum mean-squared error sense were employed [3]. The resulting optimized prediction coefficients for the 2-Scan are shown in Table 1. For example, in predicting pixel C, one takes weights of 0.089 for A and 0.914 for B. Pixel B is weighted more heavily since it is a horizontal neighbor, whereas pixel A is only a diagonal neighbor, thus it is farther away by $\sqrt{2}$.

Case	Coeff	Value
1	1A	0.089
	1B	0.914
2	2A	0.721
	2B	-0.522
	2C	0.800
3	3C	0.209
	3D	0.774
4	4C	0.743
	4D	-0.520
	4E	0.824

Table 1: 2-Scan Optimized Prediction Coefficients

The optimized prediction coefficients were then used to generate a residual training sequence from the original training sequence. A Huffman code for this residual training sequence was then generated. It is assumed that the Huffman codebook for this family of medical images is stored in advance by both the encoder and the decoder.

The Huffman encoded image was then placed into ATM packets of 53 bytes. Five bytes are assumed to be used for header information, and 48 bytes are available for the image data. Each packet begins with the row and column position of the first pixel in the packet followed by the actual value of the first pixel. This first pixel becomes the seed value for the prediction sequence within the packet. By including this initial information, each packet becomes self contained (independently reconstructable). Additionally, each packet has equal importance a priori. In the event of lost packets, the missing pixel values are estimated by linearly interpolating from the surrounding reconstructed values.

Depending on where in the scan pattern the new packet begins, the standard prediction cases may not apply to the initial pixels in the packet because the pixels normally used in the standard cases may now be in the end of the previous packet. Repeated first order prediction was used for the initial pixels until a standard case was reached.

When using a variable length code, the codewords do not typically completely fill the entire 48 byte packet. There is wasted space at the end of the packet which gets filled with whatever portion of the next codeword can be fit in. That is, if there are N bits remaining in the packet, and the next pixel to be encoded has a binary Huffman codeword with length $M > N$, then we just fill the packet with the first N of those M bits. Since the code is a prefix code, these N bits cannot be mistaken for an actual codeword.

3 Sequential transform

In a second method, we tried subband coding to decorrelate the image. The sequential (S) transform, which retains integer coefficients and has therefore proven useful for lossless compression, was performed on both the rows and columns of the image. The S transform (similar to the Haar transform) generates a low passed (sum) and high passed (difference) output. Given a sequence of integers $c[n], n = 0, \ldots, N-1$, the low and high passed outputs are given by:

$$l[n] = floor\left(\frac{c[2n] + c[2n+1]}{2}\right) \tag{1}$$

$$h[n] = c[2n] - c[2n+2] \tag{2}$$

for $n = 0, \ldots, N/2 - 1$.

Applying the S transform first to the rows and then to the columns results in the image being divided into four subband images. The variance in the resulting LH, HH and HL subbands is lower than that in the original image. The variance of the high frequency components can be further reduced by predicting the high frequency values based on low frequency components and previously computed high frequency components and then generating the prediction residuals. The prediction is carried out using equations (3)–(5).

$$\Delta l[n] = l[n-1] - l[n], n = 1, 2, \ldots \tag{3}$$

$$\hat{h} = \sum_{i=-1}^{1} \alpha_i \Delta l[n+i] - \beta h[n+1], n = 2, 3, \ldots \tag{4}$$

$$h_d[n] = h[n] - floor(\hat{h}[n] + 1/2) \tag{5}$$

In [6], a 4th order predictor was found empirically to work well with smooth medical images, with prediction coefficients as follows: $\alpha_{-1} = -1/16$, $\alpha_0 = 4/16$, $\alpha_1 = 8/16$, $\beta_1 = 6/16$.

The LL subband is a lower resolution version of the original image formed from mean values. This LL subband can be further decomposed by again applying the S transform with or without the prediction. This process can be repeated to create a multiresolution pyramid structure. When prediction is used in conjunction with the S transform, it is referred to as the S+P transform.

The decomposition was tried with one, two, and three level decompositions. These were tried both with and without prediction. In each case, separate Huffman codes were created for each level of the pyramid. For example, one code was created for the HL_1, HH_1 and LH_1 subbands combined. The LL subband was given its own Huffman code.

Each ATM packet again begins with the row and column position of the starting pixel. Thus the scheme is tolerant of packets being received out of order. The sequence in which the encoded image is packetized consists of trees following the parent/descendant order. For example, a parent pixel from the LH_3 band is followed by the four descendent pixels in the LH_2 band and the 16 descendents in the LH_1 band. Each set of three trees (from LH, HL, and HH subbands) is preceded by a single coefficient from the LL subband. Thereby each packet carries equal weight, on average.

When two-dimensional prediction as described by equations (3)–(5) was used, it was not possible to make the packets self-contained for prediction. Too many coefficients are involved in the prediction; they cannot be encompassed in a packet of only 48 bytes. When prediction was used only in one dimension, and only of 2nd order, the prediction could be self-contained, but less compression was obtained. Using the S transform alone, with no prediction, also allowed the packets to be self-contained.

Regardless of the prediction used, in the event of a lost packet, the missing coefficients in the subbands must be replaced by some values prior to performing the inverse transform. In this work, we replaced missing coefficients in any high frequency subband by zero. In the LL subband, since the values are not zero mean and they have greater correlation between neighoring values than do coefficients in the higher subbands, missing coefficients in the LL band were linearly interpolated from the four surrounding coefficients. In practice, a reconstruction scheme for missing subband coefficients such as that used in [4] could be used.

4 Results and Conclusions

The methods were evaluated using two 256 × 256, 8 bpp, MR brain images. The compression results for the linear prediction cases, both before and after the packetization process, are shown in Table 2. The difference between the rates before and after packetization is caused by the bytes used for the x, y position information for the starting pixel, and by the wasted space at the end of each packet. Also shown is the peak signal-to-noise-ratio (PSNR) resulting after reconstruction from a single lost packet. Each value shown represents the average over 2500 simulations of lost packets. One of the test images, with an example of a lost packet for the 4-Scan case, is shown in Figure 2. The corresponding reconstructed image, shown in Figure 3, is visually indistinguishable from the original. Table 2 suggests the trend that with a wider strip, a higher average predictor order can be used, and this leads to better compression, but also a wider strip results in lower PSNR in the event that a packet is lost, since the interpolation for missing pixels is being performed across a greater distance.

Method	Avg. Pred. Order	bpp before pack	bpp after pack	PSNR after 1 loss
2-Scan	2.5	5.02	5.31	59.6
4-Scan	3.25	4.89	5.21	59.1
6-Scan	3.5	4.85	5.19	56.9

Table 2: Compression Results

The compression achieved by the subband decomposition schemes prior to packetization is shown in Table 3. The column labeled levels indicates the number of levels of subband decomposition. Several trends are apparent. Taking the S transform alone does not provide as much compression as using 1-D prediction in conjunction with the S tranform. Using 1-D prediction does not perform as well as using 2-D prediction. Increasing the number of levels of decomposition results in greater compression, as expected.

The subband methods did not perform as well as the pixel-domain linear prediction methods. Some of the subband methods (the S+P transform involving 2-D prediction, or involving 1-D prediction and three levels of

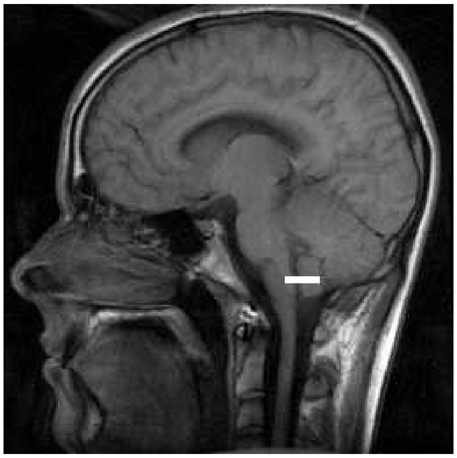

Figure 2: Original test image with location of lost packet for 4-Scan case

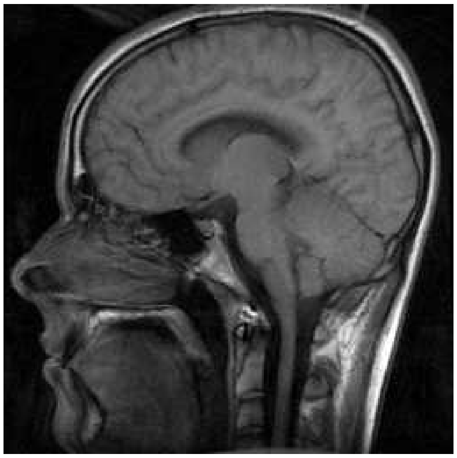

Figure 3: Reconstructed Image - 4-Scan

decomposition) provided greater compression than the 4-Scan, however, the coefficients involved in the prediction could not be fit into 48-byte packets. When the prediction was not self-contained, loss of a single packet could produce errors which propagated over a large region, and which were visually objectionable. The PSNR values were in the range of 30 to 58 dB. The other subband methods (involving no prediction at all, or involving only one or two levels of decomposition) could be packetized in a self-contained way; however, these methods provided no compression advantage over the simpler pixel-domain linear prediction methods.

Method	Levels	Bit rate (bpp)
Sequential	1	5.49
	2	5.20
	3	5.15
S+P 1-D	1	5.28
	2	4.94
	3	4.88
S+P 2-D	1	4.98
	2	4.62
	3	4.55

Table 3: S Transform Compression Results

References

[1] E. Ayanoglu, P. Pancha, and A.R. Reibman. Video transport in wireless atm. In *Proceedings ICIP-95*, volume III, pages 400–3, Washington, D.C., October 1995. IEEE, IEEE Computer Society Press.

[2] G.M. Davis, J.N. Danskin, and X. Song. Joint source and channel coding for internet image transmission. In *Proceedings ICIP-96*, volume I, pages 21–4, Lausanne, Switzerland, Sept. 1996. IEEE, IEEE Computer Society Press.

[3] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, Boston, 1992.

[4] S.S. Hemami and R.M. Gray. Subband-coded image reconstruction for lossy packet networks. *IEEE Trans. Image Processing*, 6(4):523–39, Apr. 1997.

[5] J.A. Provine and R.M. Rangayyan. Lossless compression of Peanoscanned images. *Journal of Electronic Imaging*, 3(2):176–181, Apr. 1994.

[6] A. Said and W.A. Pearlman. An image multiresolution representation for lossless and lossy compression. *IEEE Trans. Image Processing*, 5(9):1303–10, Sep. 1996.