Journal of Engineering and Development, Vol. 10, No. 4, December (2006) ISSN 1813-7822

A protection of Transmitted MRI over Channel with Packet Losses

Dr. Raghad. Z. Al-Macdici
Computer & Software Eng. Dept., College of Eng.
Al-Mustansiriya University, Baghdad, Iraq

Asst. Lect. Mohammed Hatim Khdair
Computer & Software Eng. Dept., College of Eng.
Al-Mustansiriya University, Baghdad, Iraq

Abstract

This paper presents a framework in which images can be transmitted over channels with high packet loss rates with the addition of unequal amounts of forward error correction (FEC). It develops an algorithm that can optimize the allocation of FEC to the image data so as to maximize the expected image quality.

By using both the importance of the output of the image coder and an estimate of the probability of losing packets, it considers distortion-rate tradeoffs in its assignments to provide graceful degradation when packets are lost.

The algorithm is also modular in that it can use any compression scheme that produces a progressive bit-stream.

الخلاصـــــــة

هذا البحث يقدم عمل فيه ترسل الصور الرقمية خلال قنوات اتصال والتي تكون فيها نسبة فقد أجزاء من الصورة الرقمية عالية جدا. تم في هذا البحث استخدام إلية أمامية لتصحيح الأخطاء (FEC) وتم أيضا تطوير خوارزمية قادرة على تحديد الخطأ بمعلومات الصورة مما نتج عنة تحسن في جودة الصورة المستلمة.

استخدام كلا من مرمز للصور والية لتخمين احتمالية فقد جزء من الصور (packet) أدى إلى تقليل نسبة الخطأ والحصول على اضمحلال قليل في الصورة عندما يفقد أجزاء منها.

الخوارزمية المقدمة متكيفة بحيث أنها تستطيع أن تستخدم أي خوارزمية ضغط تودي إلى تحسين معدل نقل المعلومات.

1. Introduction

When an ambulance arrives at the scene of a medical emergency, the paramedic team is responsible for assessing the situation, stabilizing the victims, and then transporting them to a medical facility for treatment. The ambulance will usually radio ahead to the medical facility and inform them of the patient's condition before arrival. This information includes a description of the accident site; the patient's vital statistics, such as heart rate, pulmonary rate, blood pressure; and the paramedics' assessment of injuries through visual inspection and tactile probing.

Recently, a group of people has been trying to give the medical facility more information about the patient before he arrives. Giovas, et. al. [1] equipped an ambulance with a notebook computer and connected it to an electrocardiogram (ECG) recorder.

By transmitting the ECG signal over a cellular telephone, they estimate that acute myocardial infarction could be diagnosed 25 minutes earlier. That extra time could afford the patient a better chance of survival.

It would be helpful if a similar approach could be used with a portable ultrasound device to send images to the destination medical facility. With a near real-time feed from the ultrasound device, that facility could monitor the patient as he is transported and evaluate the patient's internal injuries to speed triage, taking the patient directly to the medical unit most suited for dealing with a particular type of injury.

Many elements of such a system are currently in development. Battelle's Pacific Northwest National Laboratory has created a prototype portable ultrasound device, the 85 pound MUSTPAC-1. It consists of a Silicon Graphics Indy computer with at panel display, a modified Hitachi two-dimensional ultrasound scanner, and specialized software [2]. Though originally developed for battlefield use, the MUSTPAC-1 could also be installed in an ambulance.

Another important element of such a system is the wireless communication channel between the ambulance and the medical facility. Although a cellular telephone is adequate for transmitting ECG traces, a faster channel is desirable for images. For example, Metricom, Incis deploying to the general public their Ricochet2 wireless access technology in large metropolitan areas over the next few months. Ricochet2 provides a 128 kbps connection from a mobile unit to a corporate LAN using a standard Internet Protocol packet interface.

Unfortunately, wireless communications are often plagued by bursty errors, which cause packets to be discarded. Those discarded packets cause loss of data and most likely decoding failure if the lost data are not replaced. When each packet is assigned a unique sequence number, it is known which packets are received and which have been discarded, so an attempt can be made to recover from the error burst. The sequence number allows the receiver to sort the packets according to their transmission order: any gaps in the sequence are known to be lost packets and out-of-order packets can be positioned correctly. The receiver takes whatever action it deems best to decode the data.

Because we would like the medical facility to receive near real-time updates of the ultrasound images, the automatic retransmission request mechanism of TCP/IP can-not be used. With TCP/IP, not only are packet losses treated as a sign of congestion, which is not an issue in this scenario, but the receiver requests that the sender try transmitting a particular packet again. We instead need some way of adding redundancy to the transmitted images. That redundancy can then be used by the receiver to display a possibly-degraded image without having to wait for any retransmissions. Note that we do of course want that image to be degraded as little as possible, even when a large fraction of transmitted packets are lost.

This paper focuses on the issue of adding controlled redundancy to a progressive image coder so that the overall expected image quality can be maximized when an estimation of packet loss probabilities exists. The integration of this algorithm into a complete system that includes a portable ultrasound device and a wireless communication device is left as potential future work.

Section 2 covers background material on a wavelet-based image compression algorithm that uses Modified Set Partitioning in Hierarchical Trees (MSPIHT) to encode significance information about transform coefficients. Section 2 also discusses previous work on protecting the EZW coder from packet losses. Section 3 introduces a framework that relies on Forward Error Correction (FEC) to recover from lost packets, and then uses that framework to develop an algorithm that can efficiently determine a near-optimal assignment of FEC to the image data. Section 4 presents the results of this algorithm on test images, while Section 5 concludes.

2. Background

Because packets in a wireless channel are discarded at random, there is no way to specify an intrinsic priority level for a particular packet. However, the transmitted data vary in importance: some bytes of an image coder's output will be extremely important to the received image's quality, while other bytes will have much less effect on image quality. An efficient coding strategy needs to quantify the importance of different chunks of data and protect important data more than it protects less-important data. A compression algorithm that provides a progressive ordering of the data will determine which data are most important to the signal quality. In this section a progressive image compression algorithm is described, and previous work on protecting it from bit errors and channel losses is presented.

2-1 Set Partitioning in Hierarchical Trees

An example of a progressive image compression algorithm is the EZW coder [3]. This algorithm is an extension of Shapiro's Embedded Zero tree Wavelet algorithm [4]. Embedded means that the encoder can stop encoding at any desired target rate. Similarly, the decoder can stop decoding at any point result in the image that would have been produced at the rate of the truncated bit stream. In addition, the algorithms allow for progressive transmission [5], which means that coarse approximations of an image can be reconstructed quickly from beginning parts of the bit stream. They also5 require no training and are of low computational complexity.

In addition, the effects of wavelet-based compression on the diagnostic quality of images has been well-studied in recently published literature [6,7,8,9]. Given the apparent acceptance of wavelet-based compression by the medical community, the EZW coder should give excellent compression results while providing medical professionals with documented quality measures.

2-2 Joint Source/Channel Coding Using EZW

Joint source/channel coding is an area that has attracted a significant amount of research effort. Despite the fact that Shannon's separation theorem [10] states that for a noisy channel, the source and channel coders can be independently designed and cascaded with the same results as given by a joint source/channel coder, complexity considerations have led numerous researchers develop joint source/channel coding techniques [11,12]. To date, a majority of this effort has been for fixed rate codes because they do not suffer from the synchronization problems that occur with variable rate codes. (Notable exceptions that have considered joint source/channel coding schemes for variable rate codes include work on reversible variable length codes that can be decoded in both directions [13]. However, these codes can still have problems with synchronization.)

EZW has the advantage of yielding impressive compression ratios for still images, but images compressed with EZW are vulnerable to data loss. Furthermore, because EZW produces an embedded or progressive bit stream, meaning that the later bits in the bit stream refine earlier bits, the earlier bits are needed for the later bits to even be useful. However, EZW's excellent compression performance is leading researchers to consider the problem of transmitting images compressed with EZW over lossy channels and networks.

2-3 Forward Error Correction for Packet Erasure Channels

Priority Encoding Transmission (PET) [14] is an algorithm that assigns FEC, according to priorities specified by the user, to message fragments (also specified by the user) sent over lossy packet networks. Each of these fragments is protected from packet erasures by added FEC. Priorities can range from high, which means that the message fragment can be recovered if relatively few packets are received by the decoder, to low, meaning that most or all packets need to be received to recover the message fragment. This recovery is possible by treating the message fragment as the coefficients of a polynomial in a Galois field and evaluating it at a number of additional points, thus creating redundant data [15]. The receiver can recover the message fragment by interpolation from any subset of the transmitted packets, so long as it receives a fraction of packets at least as large as the priority of the message fragment.

In the PET algorithm, each message fragment is assigned a fixed position within each packet. For **Fig.(1)**, the first fragment M1 and its FEC F1 consist of the first L1 bytes of each packet, the second fragment M2 and its FEC F2 consist of bytes from (L1+1) to (L2) of each packet, and M3 and F3 consist of the remaining bytes of each packet. PET determines the value of Li for each fragment and the total number of packets N, making the assumption that the number of fragments is much smaller than the number of bytes in each packet, and constrained by the user-specified priorities.

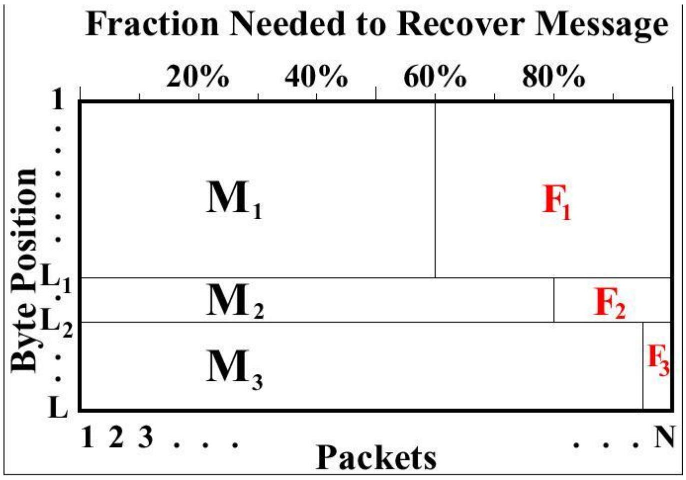


Figure (1) In Leicher's application of PET to MPEG [16], he applied 60% priority to message fragment M1 (the I frames), 80% priority to M2 (the P frames), and 95% priority to M3 (the B frames). Each message and its associated FEC use the same range of bytes in every packet

The PET algorithm does not specify how to choose the priorities to assign to the various message fragments: this assignment is left to the user. It defines priorities as the fraction of transmitted packets that must be received to decode the message; thus a high priority is represented by a low percentage. Leicher [16] applied PET to video compressed with MPEG and transmitted over packet loss channels. He used a simple three-class system in which M1 was the intraframe (I) frames and had priority 60%, M2 was the forward-only predicted (P) frames and had priority 80%, and M3 was the forward-backward predicted (B) frames and had priority 95%. Thus, he can recover the I frames from 60% of the packets, the I and P frames from 80% of the packets, and all the data from 95% of the packets. This is diagrammed in **Fig.(1)**.

In related work, Davis, Danskin, and Song [17] presented fast lossy Internet image transmission (FLIIT) which is a joint source/channel coding algorithm that, like PET, assigns different levels of FEC to different types of data, but it considers distortion-rate tradeoffs in its assignments. They begin with a 5-level discrete wavelet transform, create an embedded bit stream by quantizing each sub-band's coefficients in bit planes, apply entropy coding, and pack the bit stream from each sub-band into 64-byte blocks.

To do bit allocation, they determine the reduction in distortion due to each block, similar to work in [18].

Then compare the greatest decrease in distortion from those blocks with the addition of a block of FEC data to the already-allocated blocks and allocate the block of data or block of FEC that decreases the expected distortion the most. They only consider three simple cases of assigning FEC to a block: no protection, protection that consists of one FEC block shared among a group of blocks, and replication of the block. They find that, as expected, it is advantageous to apply more FEC to the coarse/low frequency wavelet scales and to the most significant bit planes of the quantization. The FLIIT algorithm is one of the first pieces of work to explicitly consider distortion-rate tradeoffs in making FEC assignments for lossy packet networks. However, it is limited by the coarse assignment of only three cases and the reliance on the compression algorithm they have selected (for example, EZW can yield a PSNR that is over 1 dB higher than their algorithm).

In the next section, we discuss our algorithm for adding FEC to images com-pressed with EZW that are transmitted over lossy packet networks. It considers distortion-rate tradeoffs in its assignments and uses a powerful error correction code that is derived from Reed Solomon codes.

2-4 Channel Loss Model

To determine the FEC vector $\overline{f}$, we need an estimate of the channel loss model ρ_n that a message is likely to encounter. In keeping with a modular design philosophy, our approach assumes that channel loss behavior can be modeled by an estimator that outputs a PMF which indicates the likelihood that a particular number of packets are lost. That estimate will be affected by a number of factors. In the case of the mobile ultrasound machine, the estimator could account for the type of surrounding terrain, any inclement weather, and interference from other transmitters or it could use only the previous history of losses. Indeed, one might imagine a simple estimator that uses a sliding window of recent packet losses to estimate future channel conditions.

3. An Algorithm for Assigning Forward Error Correction

While the algorithms in [19,20,21,22] yield good results for memory-less fading channels and for lossy packet networks, there are additional ways to transmit compressed images over a channel that discards packets such that image quality gracefully degrades with increasing packet loss-even at high loss rates. Specifically, we will protect images with unequal amounts of FEC in a manner similar to the PET scheme, but we will consider the effect on image quality of each data byte when assigning protection. This section introduces a framework that relies on FEC to recover from lost packets, and then uses that framework to develop an algorithm that can efficiently determine a near-optimal assignment of FEC to the image data.

In our approach to assigning unequal amounts of FEC to progressive data, we remove PET's restriction that the number of message fragments is much less than the number of bytes in each packet. Instead, we use a number of message fragments equal to the number of available bytes in each packet and have our algorithm dynamically choose the length and content of each message fragment. We add FEC to each message fragment to protect against packet loss such that the fragment and the FEC form a stream. The message is divided into L streams such that each stream has one byte of each of N packets.

In **Fig.(2)**, each of the L = 7 rows is a stream and each of the N = 6 columns is a packet. For a given stream i, for i = 1;2; : : :; L, containing both data bytes and FEC bytes, as long as the number of lost packets is less than or equal to the number of FEC bytes, the entire stream can be decoded [1]. **Figure (2)** shows one possible way to send a message of 32 bytes of data (numbers 1-32) and ten bytes of FEC (F). Notice that in the Figure, more bytes of FEC are applied to the earlier parts of the message and fewer are used for the later parts of the message. For EZW, which produces an embedded bit stream, the earlier parts of the message should have the highest priority because they are most important to the overall quality of their production.

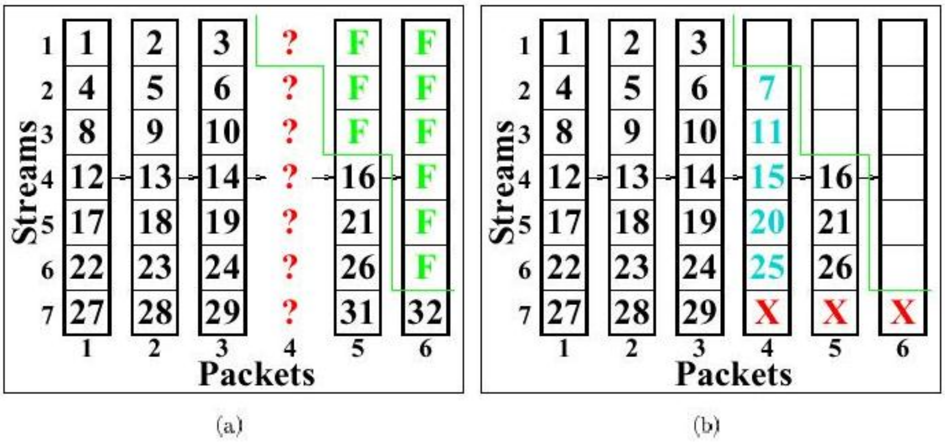


Figure (2) Each of the rows is a stream and each of the columns is a packet. A stream contains one byte from each packet. Here, a message of 32 bytes of data (numbers 1-32) and ten bytes of FEC (F) is divided into seven streams and sent in six packets

Figure (3) shows the case where one packet out of six is lost, and five are received correctly. In this case, the first six streams can be recovered since they contain five or fewer data bytes. The last stream cannot be decoded since it contains six bytes of data and no FEC. We point out that bytes 27-29 from the seventh stream are useful since they were received correctly, but for an embedded bit stream, bytes 31 and 32 are not useful without byte 30. Similarly, if two packets are lost, bytes 1-11 are guaranteed to be recovered and bytes 12-15 may or may not be recovered. In messages of practical length, however, those few extra bytes have only a small effect on image quality. Analogous to progressive transmission [23], even if severe packet loss occurred, we could recover a lower fidelity version of the image from the earlier streams that are decoded correctly. Each additional stream that is successfully decoded improves the quality of the received message, as long as all previous streams are correctly decoded.

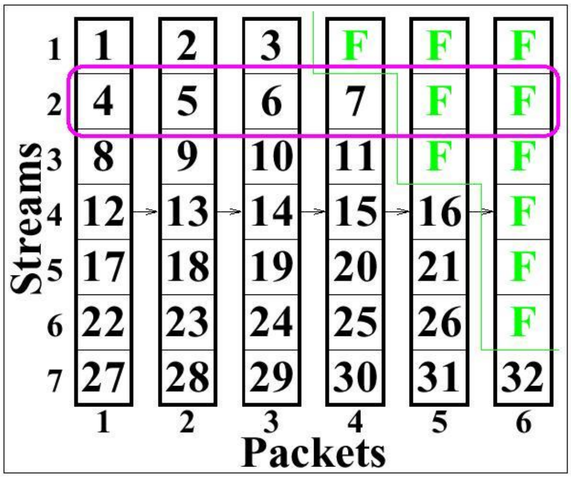


Figure (3) Demonstration of how much data can be recovered when one of six packets is lost. Here, stream one is unaffected by the loss, streams two through five use FEC to recover from the loss, and in stream seven, only the bytes up to the lost packet are useful to the decoder

Then finding the globally optimal assignment of FEC data to each stream is not computationally prohibitive for a useful amount of data, nor can a dynamic programming approach be used because the solution to one sub-problem affects the solution of another. We therefore developed a hill-climbing algorithm that makes limited assumptions about the data, but is computationally tractable, we constrain $f_i \geq f_{i+1}$. This constraint is a direct consequence of using progressive compression, because the importance of each byte of data is generally decreasing: more important data appears earlier in the sequence, while less important data appears later in the sequence.

Additionally, we assume that a single byte missing from the progressive bit stream causes all later bytes to become useless. That is, the image coder makes extensive use of context when compressing the images, both in its zero tree coding and in the arithmetic entropy coder. Even if the bytes of data that follow the missing byte are received correctly by the decoder, without context, any improvement in image quality is negligible. Finally, the PMF should be reasonably well-behaved: the more it deviates from a unimodal function, the larger the search distance must be to capture the irregularities.

We initialize each stream to contain only data bytes, such that $m_i = N$ and $f_i = 0; i = 1,2,\ldots\ldots, L$. In each iteration, our algorithm examines a number of possible assignments equal to $2QL$, where Q is the search distance (maximum number of FEC bytes that can be added or subtracted to a stream in one iteration) and L is the number of streams. We determine $G(\bar{f})$ after adding or subtracting 1 to Q bytes of FEC data to each stream see

Fig.(4), while satisfying our constraint $f_i \geq f_{i+1}$. We choose the $\overline{f}$ corresponding to the highest $G(\bar{f})$, update the allocation of FEC data to all affected streams, and repeat the search until none of the cases examined improves the expected PSNR. This algorithm will find a local maximum that we believe is quite close to the global maximum and, in some cases, may be identical to the global maximum.

Note that for every byte of FEC data that we add to a stream, one byte of data move the last data byte of this stream to the next stream, and so on. This causes a cascade of data bytes to move down the streams until the last data byte from stream L is discarded. This part of the algorithm makes use of our assumption that the compressed sequence is progressive, because the data byte that we discard is among the least important in the embedded bit stream. In the next section, we present the results of applying our algorithm to real images and compare those results to ones previously published.

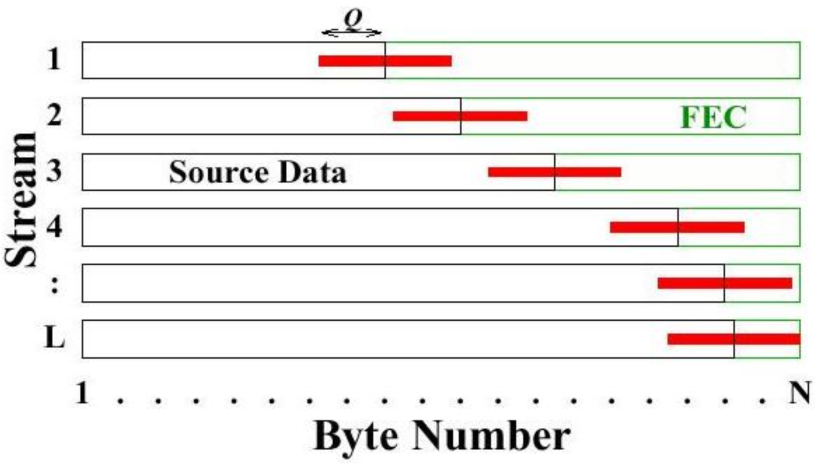


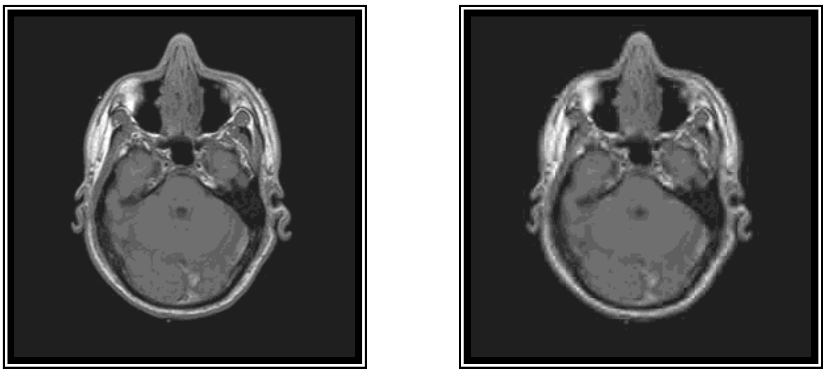
Figure (4) At each iteration of the optimization algorithm, Q bytes of data can be added or subtracted to any of the L streams

4. Results

In this section, the algorithm developed in the previous section is applied to a variety of test images. Because the work in this paper draws upon previous publications in the areas of image compression, joint source/channel coding, and networking, the first test image is chosen to be the standard within those fields, the "MRI" image. After placing the algorithm within that context, results are given for an ultrasound image of a jugular vein thrombosis. Finally, to show that the algorithm also works well on other kinds medical images, results for brain MRI are presented.

4-1 MRI Test Image

Finally, to show that the FEC assignment algorithm works well on MRI images, it was applied to a 256×256 grayscale magnetic resonance image of a brain compressed with EZW see **Fig.(5-a)**. In this case, the image was transmitted in 174 47-byte payloads over a channel with an exponential mean loss rate of 10%. The total bit rate was 0.917 bits per pixel for the combination of data and FEC bytes, see **Fig.(5-b)**.



(a) (b)

Figure (5) The 256 X 256 MRI test image
(a) The original. (b) Compressed at 0.917 bpp with EZW

In **Fig.(6)**, we show the results of using our algorithm for both UEP and for EEP. Under better channel conditions (erasure rates of up to 25.5%), UEP yields a PSNR of 35.98 dB, which is 1.06 dB higher than the 364 dB result of EEP. As before, UEP degrades gracefully, whereas EEP would give very poor image quality if the experienced loss rate were above 32%. **Figure (7)** shows the graceful degradation of the image protected with UEP and transmitted over a lossy packet network with loss rates of 10%, 20%, 30%, and 40%. Notice that the image quality remains high at a 30% loss rate and the image is still recognizable as brain slice at the 40% loss rate.

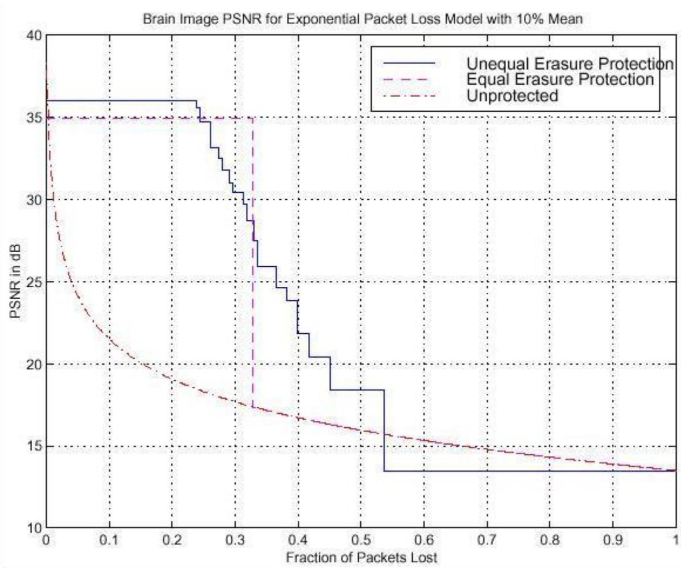


Figure (6) Comparison of PSNR of MRI test image vs. fraction of packets lost for unequal erasure protection, equal erasure protection, and no loss protection. The channel loss model is an exponential with a mean of 10%

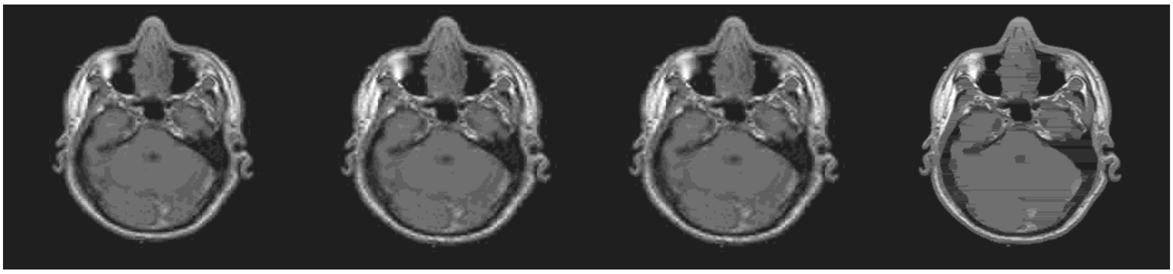


Figure (7) Channel with an exponential loss model that averages 10%. Loss rates from left to right: 10%, 20%, 30%, and 40%

5. Conclusions

We have presented a framework that allows recovery of image data, even when the transmission medium is affected by large packet loss rates. We then developed an algorithm that can optimize the allocation of FEC within that framework. By using both the importance of the output of the EZW coder and an estimate of the probability of losing packets, it considers distortion-rate tradeoffs in its assignments to provide graceful degradation when packets are lost. Those assignments result in guarantees about image quality for a particular assignment: the image quality is fixed. For a given fraction of lost packets, regardless of which specific packets are lost. The algorithm is also modular in that it can use any compression scheme that produces a progressive bit stream and any channel estimator that can provide a probability distribution function of packet losses. We imagine three distinct directions for potential future work.

Finally, improvements might be made to the UEP assignment algorithm itself. Careful optimization of the source code would decrease its execution time, as would implementation on a programmable multimedia processor. It might also be possible to design an algorithm that uses successive approximation to refine the allocation of FEC in a progressive manner, an enhancement that would allow the algorithm to terminate before completion and still find a reasonably good allocation. Successive approximation could be useful for situations in which time constraints require an extremely fast algorithm.

6. References

1. P., Giovas, D., Papadoyannis, D., Thomakos, G., Papazachos, M., Rallidis, D., Soulis, C., Stamatopoulos, S., Mavrogeni, and N., Katsilambros, ***"Transmission of Electrocardiograms from A Moving Ambulance"***, Journal of Telemedicine and Telecare, 4(1):5{7, 1998.

2. J., Coleman, A., Savchenko, A., Goettsch, K., Wang, P., Bono, R., Littlefield, and M., Macedonia, ***"Teleinvivo: A Collaborative Volume Visualization Application"***, In Medicine Meets Virtual Reality, pages.

3. Amir Said, and 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, 6(3):243{250, June 1996.

4. J. M., Shapiro, ***"Embedded Image Coding Using Zero Trees of Wavelet Coefficients"***, IEEE Transactions on Signal Processing, 41(12):3445{3462, December 1993.

5. K-H Tzou, ***"Progressive Image Transmission: A Review and Comparison of Techniques"***, Optical Engineering, 26(7):581{589, July 1987.

6. B. J., Erickson, A., Manduca, P., Palisson, K., Persons, F., Earnest, V., Savcenko, and N. J., Hangiandreou, ***"Wavelet Compression of Medical Images"***, Radiology, 206(3):599{607, March 1998.

7. K., Persons, P., Palisson, A., Manduca, B. J., Erickson, and V., Savcenko, ***"An Analytical Look at the Effects of Compression on Medical Images"***, Journal of Digital Imaging, 10(3 Suppl 1):60{66, Aug 1997.

8. D. F., Schomer, A. A., Elekes, J. D., Hazle, J. C., Huffman, S. K., Thompson, C. K., Chui, and W. A., Murphy, ***"Introduction to Wavelet-Based Compression of Medical Images"***, Radiographics, 18(2):469{481, Mar{Apr 1998.

9. B., Zhao, L. H., Schwarz, and P. K., Kijewski, ***"Effects of Lossy Compression on Lesion Detection: Predictions of the Nonprewhitening Matched Filter"***, Medical Physiology, 25(9):1621{1624, September 1998.

10. C. E., Shannon, ***"A Mathematical Theory of Communication"***, Bell Systems Technical Journal, 27:379{423, 1948.

11. N., Farvardin, ***"A Study of Vector Quantization for Noisy Channels"***, IEEE Transactions on Information Theory, 36(4):799{809, July 1990.

12. Ren-Yuh Wang, Eve A. Riskin, and Richard Ladner, ***"Codebook Organization to Enhance Maximum a Priori Detection of Progressive Transmission of Vector Quantized Images Over Noisy Channels"***, IEEE Transactions on Image Processing, 5(1):37{48, January 1996.

13. J., Wen, and J. D., Villasenor, ***"Reversible Variable Length Codes for E_Cient and Robust Image and Video Coding"***, In Proceedings Data Compression Conference, pages.

14. Andres Albanese, Johannes Blomer, Je Edmonds, Michael Luby, and Madhu Sudan, ***"Priority Encoding Transmission"***, IEEE Transactions on Information Theory, 42(6):1737{1744, November 1996.

15. L., Rizzo, ***"E_Ective Erasure Codes For Reliable Computer Communication Protocols"***, ACM Computer Communication Review, 27(2):24{36, April 1997.

16. Christian Leicher, ***"Hierarchical Encoding of Mpeg Sequences Using Priority Encoding Transmission (PET)"***, Technical Report TR-94-058, ICSI, November 1994.

17. Geoffrey M. Davis, John M. Danskin, and Xiyong Song, ***"Joint Source and Channel Coding For Internet Image Transmission"***, In Proceedings of ICIP, Volume 1, Pages 21{24, September 1996.

18. Y., Shoham, and A., Gersho, ***"Efficient Bit Allocation for an Arbitrary Set of Quantizers"***, IEEE Transactions on Acoustics Speech and Signal Processing, 36(9):1445 {1453, September 1988.

19. P. G., Sherwood, and K., Zeger, ***"Error Protection for Progressive Image Transmission Over Memoryless and Fading Channels"***, IEEE Transactions on Communications, 1998. Accepted for Publication, 471{480, March-April 1998.

20. Jon K. Rogers, and Pamela C. Cosman, ***"Robust Wavelet Zero Tree Image Compression with Fixed-Length Packetization"***, In Proceedings Data Compression Conference, Pages 418{427, March-April 1998.

21. Pamela C. Cosman, Jon K. Rogers, P. Greg Sherwood, and Kenneth Zeger, ***"Combined Forward Error Control and Packetized Zero Tree Wavelet Encoding for Transmission of Images Over Varying Channels"***, IEEE Transactions on Image Processing, 1998. Submitted for Publication.

22. A., Gersho, and R. M., Gray, ***"Vector Quantization and Signal Compression"***, Kluwer Academic Publishers, Boston, 1992.

23. Amir Said, and 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, 6(3):243{250, June 1996.