A NEW UNEQUAL ERROR PROTECTION SCHEME FOR SPIHT IN PROGRESSIVE IMAGE TRANSMISSION

Shuo Sheng, Salim Manji, Narayan Mandayam

WINLAB, Department of ECE, Rutgers University
94 Brett Road, Piscataway, NJ, 08854, USA

ABSTRACT

In the output of the state-of-art embedded wavelet image encoder SPHIT, three kinds of bits, namely the significance information bits, sign bits and refinement bits are interleaved, making unequal error protection(UEP) ineffective. In this work, we observed a beautiful property of SPIHT. The synchronization in decoding is of no influence if we split the original embedded bit stream into three in transmission. Based on this observation, we proposed an new unequal error protection scheme for SPIHT, which requires the least modification of the SPIHT algorithm. Experimental results showed the superiority of this UEP scheme to equal error protection(EEP) scheme.

1. INTRODUCTION

With the explosion in multimedia applications involving image and video communication, a lot of research has been focused on error protection of progressive image transmission[2][3][4][5][6]. recently. Most systems in the work were based on a class of wavelet zero-tree image encoders because they provide embedded bit streams. Among them, the implementation based on set partitioning in hierarchical trees(SPIHT)[1] is a competitive candidate due to its fastness and high performance.

However, due to its high complexity, SPIHT is extremely vulnerable to channel errors. A single bit error in the source bit stream could result in a complete decoding failure due to the loss of synchronization between the encoder and decoder caused by that bit error. So researchers have been looking for methods to improve the error resilience of SPIHT. Packetization is the main technique used so far. The idea is to break the embedded source bit stream into packets and apply channel codes to the packets separately so that error propagation is limited to one packet. In [2], Sherwood and Zeger proposed a concatenated channel coding scheme in which an inner RCPC code and an outer CRC code are concatenated to protect each packet. This method requires no modification to the SPIHT coder. In addition, unequal error protection can be performed by varying channel code rates according to the order of the bits appeared in the bit stream since ouptput bits are ordered in their contribution to the decoded distortion in SPIHT. Other work involves modifying the SPIHT to accommodate error resilience. In [6], Rogers and Cosman proposed to reordering the SPIHT bit stream in such a way that complete trees of wavelet coefficients were contained within packets. Thus each packet is independently decodable, lost packets do not lead to loss of synchronization.

In this paper, we proposed an unequal error protection scheme from a different point of view. It separates the original source bit stream into three bit streams by their class. Providing unequal protection to the three bit streams makes the SPIHT decoding process more robust. In addition, it requires the least modification of SPIHT because it is based on the inherent beauty of SPIHT.

2. UNEQUAL PROTECTION OF SPIHT

2.1. Observation of SPIHT

In order to improve the error resilience of SPIHT, we need to first investigate why SPIHT is so vulnerable to errors. Taking a careful study of the SPIHT, we know that the essence of SPIHT is based on three concepts[1]: 1) partial ordering of the transformed image elements by magnitude, with transmission of order by a subset partitioning algorithm that is duplicated at the decoder, 2) ordered bit plane transmission of refinement bits, and 3)exploitation of the self-similarity of the image wavelet transform across different scales. As a result, the output bit stream of SPIHT encoder is composed of three kinds of bits, namely the significance information bits, the sign bits and the refinement(magnitude) bits. Among them, the significance information bits are the most important because each of them specifies a tree of coefficients and thus defines the encoding and decoding path. An error occurred in the significance bit will cause the decoder to lose synchronization with encoder and hence the complete decoding failure of subsequent process. On the other hand, the sign bits and refinement bits decide the sign and magnitude of the wavelet coefficients. Although an error of any of them will cause a distortion in the decoded image, it will not lead to loss of synchronization. This difference in influence on decoding implies a scheme of unequal channel error protection is suitable for SPIHT. We should give more protection to the significance information bits than to the sign and refinement bits. However, in the output bit stream of SPIHT encoder, all the three kinds of bits are interleaved. Which kind of bits occur in which location is dynamically decided by the encoder. This makes the UEP difficult in this case.

To achieve the desired UEP for the three kinds of bits, we must explicitly separate them so that we can apply different rates of channel codes. This may seem impossible at the first sight. However, our study shows that it is not only possible, but also easy to implement with few modification to SPIHT. The justification lies in the observation that the synchronization between encoder and decoder is of no influence if we split the encoded source bit stream into three in transmission.

We know that the SPIHT decoder duplicates the encoding process. In doing this, the executing path of encoding algorithm is recovered by the decoder, provided that each significance information bit is decoded correctly. Although the three kinds of bits are interleaved in the encoded source bit stream, their class information(whether they are significance bit, sign bit or refinement bit)are recovered by the decoder. If the decoder decode a significance information bit correctly, it will know then what kind of bits it is in need for the next bit. So, if we output the encoded source bit stream into three buffers by their class, then the decoder can still decode correctly with the three separated bit streams because as soon as it decodes a significant information bit from the significance buffer, it will know which buffer it should go next to fetch the following bit. This beautiful property of SPIHT enables us to separate the three kinds of bits and apply different rates in channel coding in transmission.

2.2. Proposed scheme

Based on our observation in previous subsection, we propose the following transmission and channel coding scheme. First, little modification is made to the SPIHT. In the encoding algorithm, we replace the one *output* function with three *output* function corresponding to the three kinds of bits. Thus we get three bit streams instead of one bit stream. We name them the significance buffer, sign buffer and refinement buffer. In the decoding algorithm, we simply replace the one *input* function with three *input* function directed to the three buffers respectively. In order to simplify the encoding/decoding process, we skip the arithmetic encoding/decoding in the original SPIHT algorithm because only a 0.3-0.6dB loss in PSNR is observed[1].

Next, in channel coding, we take a fixed portion of each of the three buffers, apply different rate of channel codes(we use block codes here)to them and concatenate them to form one packet to transmit. We apply lower rate of channel code to the significance part and higher rate to sign part and refinement part. Details will be given in the next section.

At the receiver, channel decoding is performed first. Because the length of each part in a packet has been predefined when we form the packet, the channel decoder can decode them easily. After each part is decoded, it place the bits into their corresponding buffers, supplying for the SPIHT decoder. At last, a reconstructed image is obtained through decoding by the modified SPIHT decoder.

3. EXPERIMENTAL RESULTS

In this section we give the detail of our experiment system and channel code rates used. The system model is shown in Figure 1. The image used is the "lena" image which is 512x512 pixels in size and each pixel is represented by 256 level grayscale. Following the SPIHT encoder, BCH channel codes are applied to add channel bits to the bit stream before transmission. The channel model is chosen as a BSC memoryless channel with crossover probability p_e. To show the effectiveness of UEP, two transmission schemes are presented for comparison, equal error protection(EEP) and unequal error protection(UEP). We have different ways of packetizing the bit stream for the EEP and UEP. This is illustrated in Figure 2. In equal error protection, the output bit stream of SPIHT is broken into packets of equal length and the same rate of channel code is applied for all the bits in transmission. Within each packet, the three kinds of bits are mixed as they are in the original SPIHT encoded bit stream. In unequal error protection scheme, each packet is composed of three parts, the significance part, the sign part and the refinement(magnitude) part. Each part are filled with bits coming from the three bit buffers generated from the modified SPIHT encoder. Due to their different importance, significant part is applied with

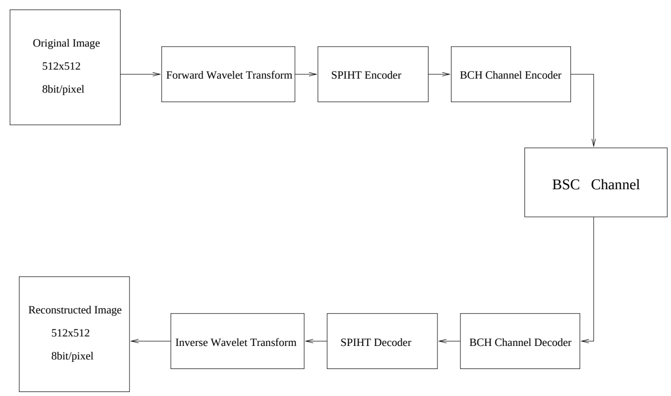

Figure 1: Image Transmission System Model

Bit rate(bpp)	2.0	1.0	0.5	0.25	0.2	0.125	0.1
Sig bits	4.32	4.33	4.48	4.78	4.92	4.93	5.14
Mag bits	1.78	1.71	1.68	1.61	1.54	1.395	0.91
Sgn bits	1	1	1	1	1	1	1

Table 1: Ratio of three kinds of bits in different source coding rate

the lowest rate of channel codes, the magnitude part is applied with the highest rate of channel codes. An interesting

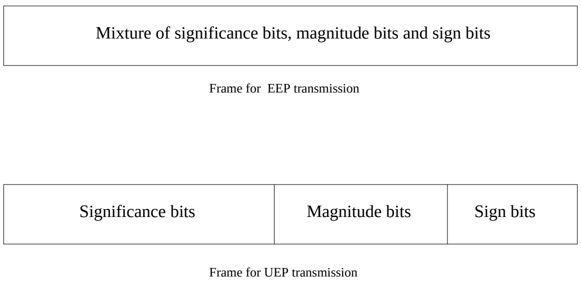

Figure 2: Frame forming of EEP and UEP

question arising for the UEP is how to decide the ratio of the amount of each kind of bits in one packet? The basic criterion is to try to keep as the same ratio as in the original SPIHT bit stream. Unfortunately, this ratio is time varying in the original SPIHT bit stream. In Table 1 we list the ratio computed from SPIHT compressed bit stream of the lena image under different source coding rate. We can see from Table 1 that although the ratio varies from rate to rate, the range is limited. So we choose a ratio that falls in the middle of the range.

Another factor in deciding the ratio is the fairness of comparison. To compare the result of the EEP and UEP scheme, we need to let the overall channel rates be the same. We choose the BCH code rate as follows

- For EEP, we use $(39, 63)$ BCH block codes for each block, and transmit 22 blocks together in one packet.
- For UEP, we use $(36, 63)$ BCH codes for the significance part, which comprises of 15 blocks, $(39, 63)$ BCH codes for the sign part, which comprises of 3 blocks, and $(51, 63)$ BCH codes for the refinement part, which comprises of 4 blocks.

Thus the over all channel rate can be computed for the two schemes. For the EEP, the overall channel rate is

$$R_{EEP} = \frac{39}{63} = 0.6190 \tag{1}$$

For the UEP, the overall channel rate is

$$R_{UEP} = \frac{36 \times 15 + 39 \times 3 + 51 \times 4}{63 \times (15 + 3 + 4)} = 0.6212 \tag{2}$$

the ratio among the three bits in the UEP packets is

$$Sig : Mag : Sign = 36\times 15 : 51\times 4 : 39\times 3 = 4.6 : 1.7 : 1 \tag{3}$$

We can see that it agrees with table 1. Although the overall channel rate of the UEP is a little bit higher than the EEP, we will see later that it achieves a better decoded image for the same channel.

The results of simulation are shown in Figure 3. We fix the source rate to be $0.123bpp$, transmit and receive the lena image 100 times with EEP and UEP scheme across a BSC channel with crossover error probability p_e, calculate the average decoded PSNR. p_e was varied to study the performance of both scheme under different channel condition. From Figure 3, we can see that the UEP scheme obtain an

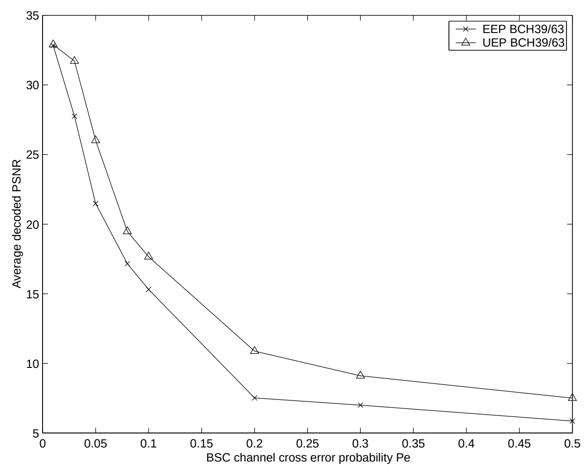

Figure 3: Average decoded PSNR for EEP and UEP at same channel rate and source rate

extra $3 - 5dB$ in PSNR than the EEP scheme, no matter the p_e is high or low.

4. DISCUSSION AND CONCLUSION

Our notion of UEP is different from the UEP proposed in [2], where the early transmitted packets were given more protection than the later transmitted packets. This is in consistent with SPIHT in that the SPIHT orders the coefficients by magnitude and outputs the most significant bits first. However, as the authors declared, the decoder must stop at the first detected but uncorrected error because the SPIHT decoder does not know whether it is a significance bit or not. In our work, we care more about if the decoding process can be continued when an error occurs and is not corrected. In other words, our goal is to maximize the throughput. If the error is a sign or refinement bit error, the decoder can continue decoding without correcting it. This error resilience feature can be easily implemented in our scheme since we have explicitly separated the three kinds of bits. The distortion caused by ignoring the bit error could be reduced by some post processing technique such as interpolation or filtering, which is called ”error concealment”. In addition, in our scheme, more channel bits are saved to give more protection to the significance bits, making the SPIHT more robust. Unlike the modified SPIHT proposed in [6], our scheme makes the least modification to the original SPIHT algorithm, maintaining the same complexity and high performance.

The major weakness of our scheme lies in using fixed proportion of the three kinds of bits in transmission. As we can see from Table 1, the difference in ratio of the three kinds of bits at different source rates can not be neglected. It is difficult and impossible to choose a ratio that is suitable for all target rate. This might lead to some problem. For example, if we transmit more significance bits in a packet, we could have stalls in SPIHT decoding because corresponding sign and refinement bits have not yet arrived. If we transmit more sign and refinement bits in a packet, they could be wasteful because the necessary significant bits have not been ready. One way to alleviate this effect is further compress the significance part by arithmetic coding. Then the deviation of the ratios in Table 1 will be reduced.

In this paper we proposed a new UEP protection for the SPIHT embedded image coder. Unequal error protection for SPIHT can be explored from the view in both Sherwood and Zegers paper[2] and our paper. We believe a better approach can be developed when combining the two together.

5. REFERENCES

[1] A.Said, W.A.Pearlman, ”A New Fast and Efficient Image Codec Based on Set Partitioning in Hierachical Trees”, IEEE Trans. Circuit Syst. Video Tech., Vol 6, pp.243-50, June 1996.

[2] P.G.Sherwood, K.Zeger, ”Progressive Image Coding for Noisy Channels”, IEEE Signal Proc. Letters, vol.4, n7, p189-191, 1997.

[3] P.G.Sherwood, K.Zeger, ”Error Protection for Progressive Image Transmission Over Memoryless and Fading Channels”, IEEE Trans. Comm., Vol.46, pp.1555-59, Dec. 1998.

[4] B.Kim, Z.Xiong, W.A.Pearlman and Y.Kim, ”Progressive Video Coding for Noisy Channels”, ICIP 98.

[5] Te-Chung Yang, C-C.Jay Kuo, ”Error correction for wireless image communication with a rate-distortion model”, Conference Record of the Asilomar Conference on Signals, Systems & Computers. v 2 1998. p963-966

[6] Jon Rogers and Pamela Cosman, ”Robust wavelet zerotree image compression with fixed-length packetization”, Proceedings of DCC 98, p418-427.