

ERROR RESILIENT VIRTUAL SPIHT FOR IMAGE TRANSMISSION OVER NOISY CHANNELS

Ekram Khan and Mohammed Ghanbari
Audio & video Networking Research Lab
Dept. of Electronic Systems Engineering
University of Essex, Colchester, UK
Emails: {ekhan, [ghan](mailto:ghan@essex.ac.uk)}@essex.ac.uk

ABSTRACT

In this paper, we present an efficient unequal error protection (UEP) scheme for Virtual SPIHT (VSPIHT) coded bitstream. It divides bitstream into two substreams namely critical and noncritical bits, which have different error resilient property. Different BCH codes are used to protect all critical bits and only 30% of the noncritical bits. The remaining noncritical bits are transmitted without any protection. The simulation results demonstrate that proposed three tiers UEP gives 0.2-0.5 dB improvements over equal error protection for binary symmetric channels (BSC).

1. INTRODUCTION

The transmission of images over noisy channels is fundamentally important and is still a challenging problem for many types of channels. Since the introduction of embedded zero-tree wavelet (EZW) concept [1], its many improved derivatives have been developed. The SPIHT [2] is the simplest and most efficient variation of EZW. Recently we have proposed an improved version of SPIHT, called Virtual SPIHT (VSPIHT) [3], suitable for very low bit rate image/video coding. All these techniques have excellent rate distortion characteristics in noise free environments. However in the presence of noise they are extremely sensitive to bit errors. Even a single bit error may lead to loss of synchronisation between the encoder and the decoder beyond erroneous bit. As a result the quality of the reconstructed image degrades substantially unless proper error correcting methods are used.

Recently, in the literature many error resilient methods utilising forward error correction (FEC) are proposed for EZW and SPIHT bit streams. Sherwood and Zeger [4] used rate compatible punctured convolution / cyclic redundancy check (RCPC/CRC) to protect each fixed size packets equally, irrespective of the relative importance of individual bits. In [5], modified SPIHT bit streams are separated based on their error sensitivity, and three tiers UEP is applied. Fixed length coding was used for certain parts of the SPIHT bit stream, and unequal error protection using RCPC channel code was used for sorting pass (significant map only), while refinement pass

was not protected at all. Since, modified SPIHT has an inferior error free performance compare to SPIHT, the overall performance of this method remains limited. More recently, Alatan, Zhao and Akansu [6] have proposed UEP by dividing SPIHT bit streams into three classes, namely location bit class (LBC-, LBC+) and value class bits (VCB), having decreasing error immunity. The VCB is protected with a light RCPC code while LBC are protected with strong RCPC/CRC. The LBC- is protected more strongly compared to LBC+.

The objective here is not to protect all the bits, but to accept few errors without loss of synchronisation. In this paper we propose an UEP of VSPIHT/SPIHT coded bit streams using BCH code. The source bit stream is divided into two classes: critical (CB) and noncritical (NCB) bits. In each class, the importance of bits reduces along the bit stream. The error in the critical bits causes global distortion while the noncritical bits, if erroneous, results in distortion only locally. All the critical bits are protected heavily prior to transmission, while only fraction of noncritical bits are protected while remaining noncritical bits are transmitted without any protection. An optimal partitioning of the non-critical bits is also suggested.

The organisation of the paper is as follows. Section 2 describes brief overview of Virtual SPIHT. The proposed UEP scheme is presented in Section 3. Section 4 includes simulation results and concluding remarks are given in section 5.

2. VIRTUAL SPIHT (VSPIHT)

The VSPIHT is an efficient variant of SPIHT algorithm, which combines virtually generated zero trees on the top of the SPIHT's zero trees in order to combine many zero trees into longer zero trees. It is highly efficient, particularly at the early passes of the encoding. The algorithm works as follows:

After n levels of dyadic wavelet decomposition, the LL band coefficients are separated and their addresses are used to initialise a list of insignificant pixels (LIP). A new LL band with zero value coefficients is created to complete the wavelet frame. This new LL band is further virtually decomposed by v levels (although there is no

Pass		1	2	3	4	5	6	7	8	9	10	11	12
SPIHT	Sorting	476	531	885	1976	5490	11725	23419	43949	89545	208059	256050	182113
	Refinement	0	28	104	265	570	1427	3372	7528	15620	31905	72498	137853
VSPIHT	Sorting	296	406	853	1993	5494	11729	23419	43949	89545	208059	256050	182113
	Refinement	0	28	104	265	570	1427	3372	7528	15620	31905	72498	137853

Table: 1. Comparison in terms of number of bits generated in each pass for LENA (512x512) image

actual decomposition), resulting in pyramidal sub-bands of zero value coefficients containing virtual zero trees with roots in the virtual LL band. The list of insignificant set (LIS) is initialised with the address of all coefficients in the virtual LL band except those, which don't have any descendants. The list of significant pixels (LSP) is initially an empty set. Like SPIHT, VSPIHT also consist of two main stages, sorting and refinement.

During the sorting pass, the algorithm first traverses through the LIP, testing the magnitude of its elements against the current threshold and representing their significance by 0 and 1. Whenever a coefficient is found significant, its sign bit is coded and it is moved to LSP. The algorithm then examines LIS and performs the magnitude check on all coefficients of the set. If a particular set is found to be significant it is then partitioned into subsets (children and descendants) and tested for significance, otherwise a single bit is appended to the bit stream to indicate insignificant set (or zero tree). After each sorting pass, algorithm outputs refinement bits at the current level of bit significance of those coefficients that are found significant in the earlier passes. This process continues by decreasing the current threshold by a factor of two until the desired bit rate is achieved.

The VSPIHT differs with SPIHT in the following ways. Firstly, in VSPIHT there are two LL bands, real and virtual. The real LL band is used during LIP testing phase and refinement pass, while the virtual one is for LIS testing phase. Secondly, in SPIHT trees consists of n generation of coefficients with roots in the real LL band, but in VSPIHT it consists of $(n + v)$ generations of the coefficients with roots in the virtual LL band. The two algorithms mainly differ in LIS testing phase of sorting pass. The VSPIHT is designed to reduce the number of zero trees, which are otherwise in large numbers particularly in the early passes. It reduces the number of bits generated during the sorting pass but no change in the refinement pass. Table 1 compares pass wise break-up of bits of SPIHT and VSPIHT for Lena (512x512) image with $n=5$ and $v=2$. The advantage of VSPIHT is obvious from the table particularly in the early passes, and hence more efficient than SPIHT.

3. ERROR RESILIENT VSPIHT

We modify the bit stream such that the resulting bit stream is better protected using any FEC. Except for a negligible

header and some bit position changes, the resulting bit stream has almost the same rate distortion characteristic as original for error free application.

3.1 Bit Classification

Both SPIHT and VSPIHT encode images using bit planes, with two passes for each bit plane as explained in section 2. However, bits generated have different degree of vulnerability to errors. The effect of error in some bits is more severe damaging the image globally by disturbing synchronisation between the encoder and the decoder. The other bits have less severe effect in the presence of error and their effects are limited locally without affecting synchronisation between the encoder and the decoder. Let us denote these classes of bits as critical bits (CB) and noncritical bits (NCB) respectively. In SPIHT/VSPIHT context, CB consists of bits generated during sorting pass excluding the sign bits. A single bit error in CB causes catastrophic failure of the decoding process beyond that point. The NCB are sign bits of sorting pass and all bits of refinement pass. Any error in NCB doesn't propagate along the bit stream as long as CB is error free.

In our proposed scheme, the source bit streams are partitioned into two substreams NCB and CB, while maintaining the relative positions of bits within each substream. For most of the images, on an average NCB comprises 25% of the total bit budget in the range of 0.1-1 bpp (for VSPIHT this percentage is slightly higher than in SPIHT). NCB can be transmitted before CB so that receiver can partially maintain progressiveness after buffering NCB portion of the bit stream. The maximum delay depends on the size of NCB. To inform the decoder about switching from NCB to CB, the length of NCB substream (total 17 bits) is padded as a header in the beginning. Since CB has more severe error effect than NCB, they need to be protected heavily for worst case in order to prevent loss of synchronisation. Our simulation results show that error in NCB generated at higher passes has negligible effect on the reconstructed image, they can be left unprotected, leaving only NCB of early passes need to be protected. Thus NCB can further be partitioned, as protected NCB and unprotected NCB. The optimum partitioning of NCB can be obtained through simulation. The substream partitioning and transmission order is shown in Fig. 1.

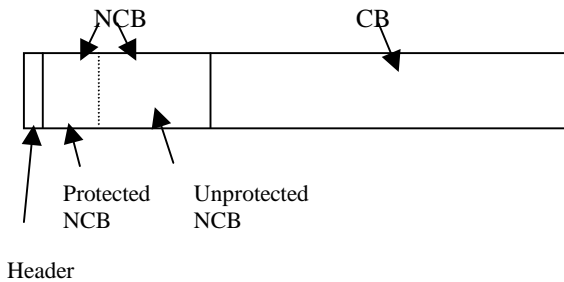


Fig.1: Bit stream partitioning for UEP

3.2 Unequal Error Protection (UEP)

Using the above demultiplexing of substreams, randomly located bits of different resilient in the original bit stream are grouped together. However they have no immunity against the error without FEC. But partitioning of the substreams makes their protection simple, as they can be protected differently. In this work, a simple block based channel code like BCH is used to protect these substreams. The critical bits are protected with a stronger BCH code than the NCB. However only first few NCB are protected with relatively lower BCH code, allowing the remaining NCB unprotected. For these NCB, rather than using overhead to protect against errors, picture quality can be improved further, if these bits are used to code image at higher quality. Here, it should be noted that our partition mechanism for UEP is different from that in [6], in which critical bits are divided into two parts (LBC- and LBC+) and all critical and non-critical bits are protected with different FEC. However, in our case only fraction of non-critical bits need to be protected. Thus for any given FEC, our partition mechanism gives better performance than that in [6], due to saving of parity bits in unprotected NCB.

3.3 VSPIHT versus SPIHT with UEP

As mentioned earlier, the proposed UEP scheme can equally be applied to SPIHT and VSPIHT. However size of NCB substream determines overall performance of the scheme. The larger is the size of NCB substream, more gain can be achieved with the UEP scheme. As VSPIHT generates less critical bits than SPIHT in the early passes, and that difference is carried out until the end of any given bit rate. At any bit rate, VSPIHT generates less CB, hence more NCB than that in SPIHT. Hence, the proposed UEP scheme is better suited for VSPIHT than SPIHT.

4. SIMULATION RESULTS

The performance of the proposed UEP scheme with SPIHT and VSPIHT is evaluated on grayscale image 'Lena' (512x512) coded at bit rates 0.1, 0.2, 0.25, 0.5 and 1.0 bpp. All simulations are carried out for binary

symmetric channels (BSC) with bit error rates (BER) 10^{-2} and 10^{-3} . The experimental results are the average of MSE, (represented in terms of PSNR) of 60 independent simulations.

In order to select optimal code for protection, we have considered families of 31,63,127 length BCH codes. First an equal error protection (EEP) is used to original SPIHT/VSPIHT bit streams. Through 60 independent simulations, the optimal BCH codes found for channels with $BER=10^{-3}$ and 10^{-2} are (127,106,7) and (63,36,11) respectively. The results for EEP with these codes are shown in table 3. These BCH codes are appropriate to protect CB substreams. However protection of NCB should be slightly weaker than CB. The BCH codes used for each substream at each BER are listed in Table 3. These codes are selected from a permissible set, such that the overall performance is best for each channel. Further, to achieve an optimal partitioning for protected and unprotected NCB, an experiment is carried out by protecting 0-100% NCB with increment of 5% for 20 independent channel conditions at $BER=10^{-2}$ at 0.1, 0.5 and 1.0 bpp. The average PSNR for each percentage of protected NCB is plotted in Fig. 2. It can be interpreted that only 25-30% of NCB need to be protected, with optimum at 30%. Thus in all subsequent experiments of UEP, only 30% of the NCB and all CB are protected with BCH codes listed in table 2.

BER	UEP		EEP
	NCB	CB	
10^{-3}	(127,113,5)	(127,106,7)	(127,106,7)
10^{-2}	(63,39,9)	(63,36,11)	(63,36,11)

Table 2. Optimal BCH codes at different BER

The simulation results of the proposed scheme with SPIHT & VSPIHT at five bit rates in the range of 0.1-1.0 bpp are illustrated in table 3. The results of VSPIHT are slightly better than that of SPIHT at lower bit rates. Further UEP gives the gain of approximately 0.2-0.5 dB over optimal EEP under the same channel conditions. The results of VSPIHT at $BER=10^{-3}$ and 10^{-2} are also plotted and shown in Fig. 3(a) and (b) respectively. It can be seen that the gain of UEP is more at higher bit rates, due to more number of unprotected NCB transmitted. Also by analysing plots of figure 2, it can be interpreted that the proposed 3 tiers protection mechanism is approximately 0.15-0.4 dB better than 2 tiers protection. The proposed scheme outperforms the results of [5] by approximately 0.4 dB and is comparable to that in [4], although both uses more complex RCPC channel codes. However, our results are inferior to that in [6] by 0.2 dB only, besides the fact that we have used simple channel codes. However our results can further be improved by using more advance and complex channel codes like Reed-Solomon, convolution or RCPC codes.

bpp	SPIHT		VSPIHT	
	EEP	UEP	EEP	UEP
0.1	28.82	29.01	28.88	29.10
0.2	31.68	31.87	31.72	31.90
0.25	32.83	32.98	32.85	33.00
0.5	35.93	36.08	35.94	36.10
1.0	38.87	39.07	38.88	39.08

(a)

bpp	SPIHT		VSPIHT	
	EEP	UEP	EEP	UEP
0.1	27.33	27.51	27.38	27.67
0.2	30.22	30.47	30.25	30.51
0.25	30.92	31.41	30.94	31.42
0.5	33.96	34.55	33.98	34.59
1.0	37.09	37.59	37.10	37.65

(b)

Table 3. Coding results for 'Lena' image (a) BER=10⁻³ and (b) BER=10⁻²

5. CONCLUSIONS

In this paper we have proposed an UEP scheme for SPIHT/VSPIHT coded image bit streams with simple BCH codes. The extensive simulation results demonstrate the advantage over EEP for noisy channels. This improvement is at the cost of slightly increased complexity. Furthermore, the proposed technique is better suited for VSPIHT than SPIHT.

6. REFERENCES

- [1] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients", IEEE Trans. Signal Processing, vol. 41, pp. 3445-3462, Dec. 1993.
- [2] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees", IEEE Trans. Circuit & Systems for Video Technology, vol. 6, pp. 243-250, June 1996.
- [3] E. Khan and M. Ghanbari, "Very low bit rate video coding using virtual SPIHT", IEE Electronics Letters, vol. 37, No.1, Jan 2001.
- [4] P. G. Sherwood and K. Zeger, "Progressive image coding on noisy channels", IEEE Signal Processing Letters, vol. 4, pp. 189-191, July 1997.
- [5] H. Man, F. Kossentini and M. J. T. Smith, "A family of efficient and channel error resilient wavelet/subband image coders", IEEE Trans. Circuit & Systems for Video Technology, vol. 9, pp. 95-108, Feb. 1999.
- [6] A. A. Alatan, M. Zhao and A. N. Akansu, "Unequal error protection of SPIHT encoded image bit streams", IEEE Journal on Selected Area in Communications, vol. 18, pp. 814-818, June 2000.

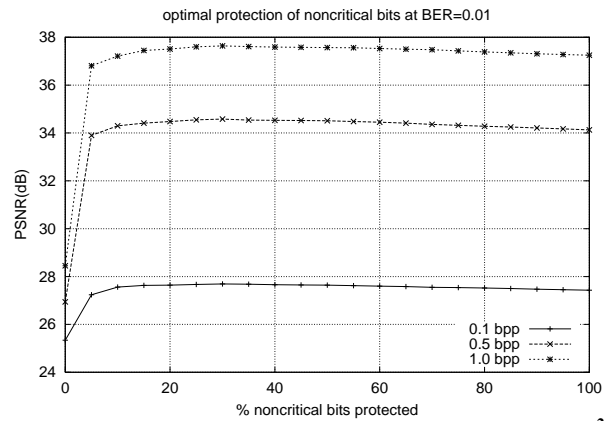
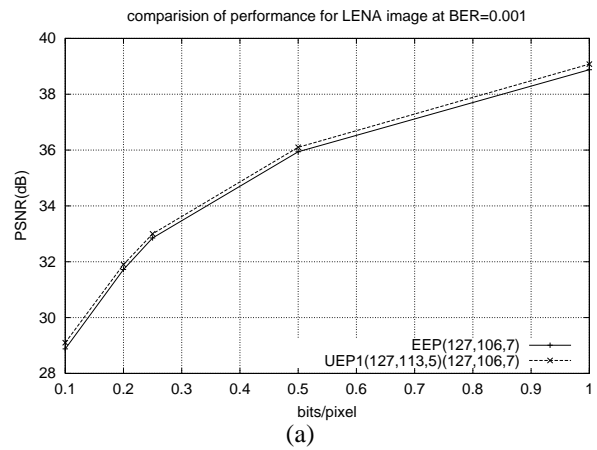
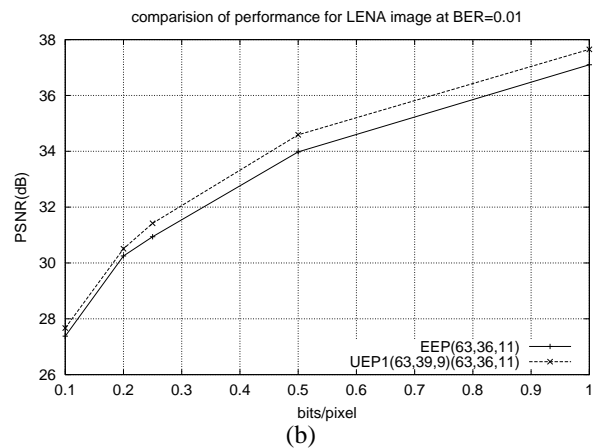


Fig. 2. The average PSNR vs. %NCB protected at BER=10⁻²



(a)



(b)

Fig. 3: Results of VSPIHT for Lena Image at BER (a) 10⁻³ and (b) 10⁻²