ITERATIVE JOINT SOURCE-CHANNEL DECODING USING TURBO CODES FOR MPEG-4 VIDEO TRANSMISSION

Xiao Feng Ma *, *William E. Lynch*

Concordia University
Department of Electrical and Computer Engineering
Montreal, Quebec, Canada H3G 1M8

ABSTRACT

This paper presents a novel iterative joint source-channel decoding scheme for MPEG-4 video transmission over noisy channels. The proposed scheme, on one hand, utilizes the channel soft outputs generated by the turbo decoder to assist video decompression. On the other hand, the syntactic/semantic information from the video decompressor is used to modify the extrinsic information so as to improve the error correction capability of the turbo decoder. With the proposed video packet mixer, the scheme can correct most turbo coding blocks with a large number of errors. Simulation results show significant improvement in terms of PSNR, reconstructed video quality, as well as BER over turbo decoding only.

1. INTRODUCTION

Compression is always used on video signals when they are transmitted due to bandwidth considerations. Compressed video is fragile in that a few channel errors may result in severe quality degradation because of error propagation, synchronization loss, etc. To alleviate this, channel coding is usually applied to protect the compressed video. Since their introduction by Berrou et. al. [1] in 1993 turbo codes have been shown to give very good error correction performance. In addition,compressed video still contains residual redundancy which can be used to combat channel noise [2, 3, 4, 5, 6, 7]. In this paper, an iterative joint source-channel decoding using turbo codes for MPEG-4 video transmission is investigated.

Examples of recent joint source-channel coding using turbo codes with compressed video/images include [5, 6, 7]. Both [5] and [6] dealt with non-standard still image transmission, which is not the subject of this paper. [7] discussed joint decoding schemes for both still image and MPEG-1 video. Their work on MPEG-1 video makes use of the high degree of predictability of the MPEG-1 start codes. Other than these start codes it does not appear to use other syntactic/semantic information of the compression scheme. Our previous work [2, 3, 4] developed a measure that combined the likelihood of a slice from the channel decoder with the syntax/semantic and smoothness information. In these, a simple (not turbo code) error protection scheme was employed. In the work presented here a joint source-channel decoding scheme is proposed that uses turbo codes as the error protection scheme and MPEG-4 as the video compressor. Information from syntactic/semantic errors from all parts of the video stream below the Video Packet (VP) layer, as opposed to just the start codes, are fed back to the turbo decoder from the video decompressor. Note that a VP is analogous to a slice in the MPEG-4 terminology, which is the lowest syntax layer beginning with a Resynchronization Marker (RM). A VP consists of header bits, such as the RM, the absolute macroblock number of its first macroblock and the quantization parameter, and combined motion and DCT data.

This paper is organized as follows: Section 2 describes the proposed scheme, Simulation results are given in Section 3 and Section 4 concludes the paper.

2. THE PROPOSED SCHEME

An MPEG-4 bitstream is structured into several layers [8] corresponding to different objects, space locations and time instants. Here, it is assumed that bits above the VP layer and the header bits in the VP layer are error-free and that bits after the header bits are turbo encoded and suffer from channel noise. We consider an AWGN channel and BPSK modulation. The proposed scheme, Iterative Joint Source-Channel Decoding (IJSCD) is shown in Figure 1.

The turbo code used in this paper is the one from [9]. The two recursive systematic convolutional encoders have the same generator matrix [31 27] in octal form and the constraint length is 5. The coded bits are punctured so that the code rate is $\frac{1}{2}$ and a random interleaver of length 1000 is employed.

For a maximum *a posteriori* (MAP) decoder, the soft

* This research was supported by the Natural Sciences and Engineering Research Council of Canada.

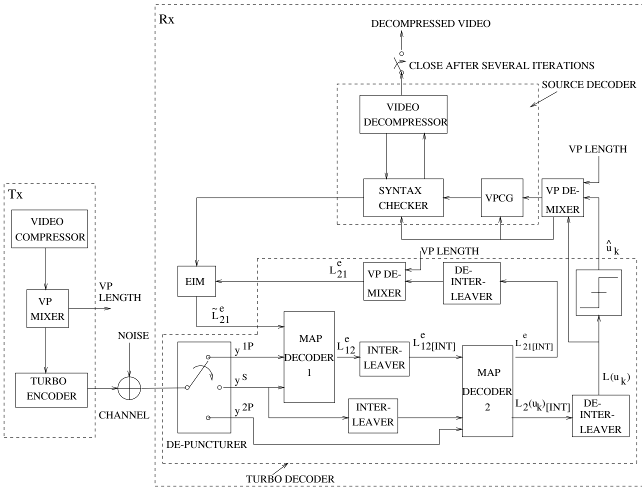


Fig. 1. Block diagram of IJSCD.

decision on each transmitted bit u_k (k is the time index) is given by [10]

$$L(u_k) \triangleq \log\left(\frac{P(u_k = +1|\mathbf{y})}{P(u_k = -1|\mathbf{y})}\right) \tag{1}$$

where **y** is the received noisy bit sequence. Incorporating the convolutional code's trellis, Equation 1 may be written as [10]

$$L(u_k) = L_c y_k^s + L_{12}^e\,(u_k) + L_{21}^e\,(u_k) \tag{2}$$

for MAP decoder 2. In Equation 2, $L_c = \frac{2}{\sigma^2}$ and σ^2 is the channel noise variance. y_k^s is the received value for each information bit. $L_{21}^e\,(u_k)$ represents the extrinsic information generated by MAP decoder 2 and is to be used as *a priori* information by MAP decoder 1. $L_{12}^e\,(u_k)$ is the *a priori* information generated by MAP decoder 1. The reliability information about each decoded information bit is the absolute value of $L(u_k)$. Generally, $|L(u_k)|$ is large for correct bits and small for error bits [7].

The Video Packet Candidate Generator (VPCG) in Figure 1 makes use of the available soft output $L(u_k)$ of each bit from the turbo decoder to generate VP candidates for the decompressor. Since an error bit generally has small $|L(u_k)|$, we choose n_F bits with the smallest $|L(u_k)|$ in a VP, as Flip Bit Candidates (FBC). By flipping some or all of these n_F bits, VP candidates are generated. Each bit's *a posteriori* Probability (APP) can be easily calculated according to Equation 1 once $L(u_k)$ is known. The APP of each VP candidate is the product of its constituent bit APPs (here the bit APPs are assumed to be independent). The VP candidates are sorted in descending APP order and sent to the Syntax Checker.

The Syntax Checker in Figure 1 determines whether or not there is an MPEG-4 syntactic/semantic error in each VP candidate. Since the VP candidates are presented in descending APP order the Syntax Checker stops once it finds the first VP candidate with no syntax errors. (If no VP candidates pass syntax checking, the one with the highest APP is chosen as the best VP candidate.) In this way, we combine the available source and channel information.

The Extrinsic Information Modifier (EIM) in Figure 1 feeds back syntax information to the turbo decoder. Our feedback scheme is to modify $L_{21}^e(u_k)$ as follows:

$$\tilde{L}_{21}^e(u_k) = L_{21}^e(u_k) + \hat{L}(u_k) \tag{3}$$

Based on the best VP candidate information from the Syntax Checker, the modification value $\hat{L}(u_k)$ is experimentally decided according to six combinations of three factors: (1) whether or not the bits are FBCs, (2) whether or not the decisions made by the source decoder and the turbo decoder are agree with each other and (3) whether the value of $L(u_k)$ is positive or negative. Table 1 lists $\hat{L}(u_k)$ in all these cases. Note that, if no VP candidate was syntactic/semantic error free then the extrinisic information is not modified and the EIM is skipped.

Table 1. $\hat{L}(u_k)$ in different cases

Turbo decoder decision	Source decoder decision for FBC		Source decoder decision for non - FBC	
	1	0	1	0
1	+3.0	-0.5	+0.5	N/A
0	+0.5	-3.0	N/A	-0.5

Table 2. Turbo coding blocks of a large number of errors at channel SNR 1.5 and 1.6 dB after 15 iterations

Channel SNR	Turbo coding block NO.	TD	IJSCD
1.5	1246	62	0
	1375	83	105
	2163	114	127
	4122	39	0
	4651	51	64
	5342	115	101
	5850	91	0
	6730	86	0
	7569	15	0
	7729	17	0
	9857	53	0
1.6	3790	124	0
	9330	106	0
	9897	124	0

In turbo decoding simulations, we find that there are some turbo coding blocks which may have a large number of errors even after 15 iterations. This phenomenon is also noted by other researchers [11] as one of the convergence patterns of turbo decoding. In this case, the Syntax Checker will often be presented no VP candidates which are syntactic/semantic error free unless a large number of VP candidates are examined. To alleviate this, the VP mixer is employed to distribute the bits from one VP into several different turbo coding blocks. In the case of a large number of error bits in one turbo coding block, each VP only receives a few. In our scheme, each turbo coding block contains at most M bits from the same VP. The VP de-mixer recovers the bits, as well as their corresponding $L_{21}^{e}(u_k)$ and $L(u_k)$, of a turbo coding block into their original VP's. To do this each VP's length is sent and is assumed to be error free. In Table 2, we list the turbo coding blocks with a large number of error bits when only turbo decoding is employed. For comparison, we also list the number of errors for the same turbo coding blocks when the VP mixer/de-mixer is employed. Table 2 clearly shows that VP mixing effectively corrects most of these blocks.

3. SIMULATION RESULTS

The performance of the proposed scheme IJSCD is evaluated objectively and subjectively. Peak Signal-to-Noise Ratio (PSNR), Bit Error Rate (BER) and bits in error are the objective measures. For performance comparison, simulation results of two different schemes: Turbo Decoding (TD) and Decoupled IJSCD (DIJSCD) are shown. DIJSCD runs TD, and after TD is complete the output is examined VP by VP and VP candidates are selected without syntactic/semantic errors if possible. Thus DIJSCD runs only one time what IJSCD runs at each iteration of the TD.

Table 3. Performance comparison when number of bits (M) from the same packet varies at channel SNR 1.6 dB

M	Scheme	delay (frame)	Bits in error	PSNR (Y)
100	TD	0	396.2	34.238
	DIJSCD		392.6	34.376
	IJSCD		423.6	34.384
50	TD	1	302.2	32.218
	DIJSCD		292.2	32.614
	IJSCD		226.2	32.996
20	TD	5	398.4	32.378
	DIJSCD		361.2	33.744
	IJSCD		98.4	35.501

In the simulations, the 6-second video "Table-Tennis" is used, which has a frame size 352×240 and is interlaced with 30 frames/second. It is encoded using MPEG-4 at bit rate 2 Mb/s. In the MPEG-4 encoding, each row in a frame has 4 video packets. The group of video object plane layer is employed in the compression and it has 15 frames with 1 I-frame, 4 P-frames and 10 B-frames. All simulations are run 5 times and the final results shown in Table 3 and Figure 4 are the simple average of the 5 experiments after 15 iterations.

The performance of the proposed scheme is evaluated with various M at channel SNR 1.6 dB. Table 3 presents the results, and clearly shows the performance improvement when M decreases, as expected. The observed frame delay caused by VP mixing is also measured and listed in Table 3. The frame delay is not significant considering B-frames are used.

The complexity on the VPCG and VP mixing/de-mixing can be neglected compared to turbo decoding. The complexity of decompressing VP candidates is difficult to measure accurately. In terms of running time, the IJSCD is about 1.5 times that required by TD.

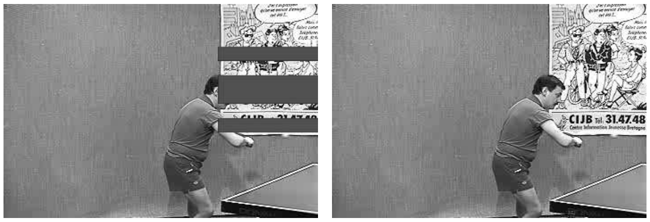
The objective and subjective performance of the proposed scheme at different channel SNR's is investigated when different numbers of FBCs (n_F) are used. In these simulations, M is chosen as 20. Figure 4 shows that IJSCD improves the performance, in terms of PSNR and BER, as compared with DIJSCD and TD. Also, more improvement is available as the number of FBCs are increased. At channel SNR 1.5 dB, IJSCD reaches 35 dB in luminance PSNR, which can be considered as close to perfectly recovered, while TD is only at 24 dB. The coding gain can be over 0.1 dB in channel SNR at output BER 10^{-5}. For subjective performance evaluation, Figure 2 and 3 present two specific frames after decoded by IJSCD and TD. In Figure 2, the lost paddle and defective table edge are corrected by the proposed scheme. In Figure 3, the proposed scheme removed all of the black stripes which are generally caused by serious syntax violations.

(a) TD (b) IJSCD

Fig. 2. Frame 65 from decompressed video "Table-Tennis" at channel SNR 1.6 dB, FBC 4.

4. CONCLUSION

This paper presents a new joint source-channel decoding scheme using turbo codes, which iteratively makes use of the available residual information from the source decoder and soft output of each bit from the turbo decoder. The simulation results show that the proposed scheme is superior to turbo decoding, in terms of PSNR, BER and the recovered video quality. Moreover, the proposed scheme can effectively correct the turbo coding blocks with a large number of error bits.



(a) TD (b) IJSCD

Fig. 3. Frame 166 from decompressed video "Table-Tennis" at channel SNR 1.6 dB, FBC 4.

5. ACKNOWLEDGEMENT

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

6. REFERENCES

[1] Claude Berrou, Alain Glavieux, and Punya Thitimajshima, "Near shannon limit error-correcting coding and decoding: turbo codes," *Proc. 1993 Inter. Conf. Commun.*, pp. 1064–1070, 1993.

[2] William E. Lynch, Vasilios Papadakis, Rajesh Krishnamurthy, and Tho Le-Ngoc, "Syntax based error concealment," *Signal Processing: Image Communication*, vol. 16, no. 9, pp. 827–835, June 2001.

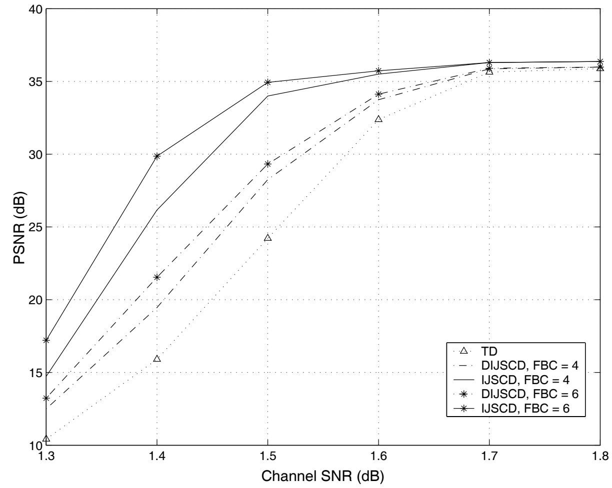
[3] William E. Lynch and Tho Le-Ngoc, "Syntax and discontinuity based error concealment for compressed video in a packet enviroment," *ITCC-01*, pp. 258–262, 2001.

[4] Yan Mei, Tho Le-Ngoc, and William E. Lynch, "A combined multiple candidate likelihood decoding and error concealment scheme for compressed video transmission over noisy channels," *Signal Processing: Image Communication*, vol. 18, no. 10, pp. 971–980, October 2003.

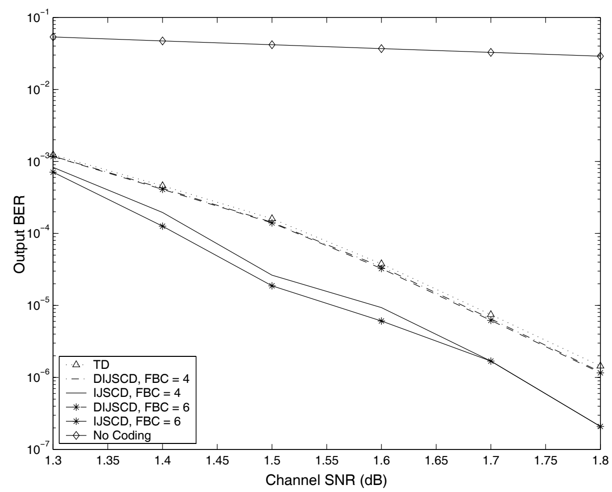
[5] Wen Xu, Joachim Hagenauer, and Jürgen Hollmann, "Joint source-channel decoding using the residue redundancy in compressed images," *Data Compression Conference*, pp. 142–148, 1996.

[6] Jörg Kliewer and Norbert Görtz, "Iterative source-channel decoding for robust image transmission," *Acoustic, Speech, and Signal Processing, 2002 IEEE conference*, vol. 3, pp. 2173–2176, 2002.

[7] Zhishi Peng, Yih-Fang Huang, and Daniel J. Costello, "Turbo codes for image transmission- a joint channel and source decoding approach," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 868–879, June 2000.



(a) Luminance PSNR



(b) Bit error rate

Fig. 4. Performance evaluation at different channel SNR and number of FBC in terms of a) Luminace PSNR, b) BER.

[8] ISO/IEC 14496-2, "Information technology - generic coding of audio - visual objects," 1998.

[9] William E. Ryan, "A turbo code tutorial," *http://www.ece.arizona.edu/ ryan/*.

[10] Joachim Hagenauer, Elke Offer, and Lutz Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, March 1996.

[11] Oscar Y. Takeshita, Oliver M. Collins, Peter C. Massey, and Jr. Daniel J. Costello, "On the frame-error rate of concatenated turbo codes," *IEEE transactions on communications*, vol. 49, no. 4, pp. 602–608, April 2001.