BLOCK-ADAPTIVE WYNER-ZIV CODING FOR TRANSFORM-DOMAIN DISTRIBUTED VIDEO CODING

Wei-Jung Chien and Lina J. Karam
Department of Electrical Engineering
Arizona State University
Tempe, AZ 85287-5706
{Wei-Jung.Chien, Karam}@asu.edu

Glen P. Abousleman
Compression, Communications & Intelligence Lab
General Dynamics C4 Systems
Scottsdale, AZ 85257
Glen.Abousleman@gdc4s.com

ABSTRACT

This paper presents a transform-domain distributed video compression (DVC) system with block-adaptive Wyner-Ziv coding. In the proposed system, the source symbols are reformatted into several blocks. The parity bits are requested only for the blocks with decoding errors. More accurate placement of the parity bits results in an improved system performance that is much closer to the performance of H.263 interframe coding as compared to existing DCT-based DVC schemes. Coding results and comparison with existing DVC schemes and with H.263 are presented to illustrate the performance of the proposed system.

Index Terms— distributed video coding, Wyner-Ziv coding, block based Wyner-Ziv coding

1. INTRODUCTION

Distributed video coding (DVC) is a novel video coding scheme which is designed for many-encoders-to-one-decoder applications, such as wireless sensor networks applications. Unlike the conventional video coding systems, such as MPEG or the ITU-T H.26x, which have complex encoders as compared to their decoders, DVC has a low complexity video encoder which is suitable for low-power wireless sensors and portable devices. To reduce the complexity of the video encoder, the motion estimation that is generally used at the encoder could be removed if intraframe coding can be performed in such a way as to approach the compression performance of the interframe coding.

Among several proposed practical DVC systems [1-5], two popular baseline systems are the turbo-code-based Wyner-Ziv coding system of [1], and PRISM [2]. The turbo-code-based Wyner-Ziv coding system (WZC) [1] is popular because of its simplicity and flexible architecture. It has two different variations, pixel-domain WZC [1, 3, 4] and transform-domain WZC [1, 5]. In both cases, the DVC system performance relies on the accuracy of the side information at the decoder. Moreover, with the same side information, transform-domain WZC exhibits a higher performance than pixel-domain WZC, though the transform-domain scheme has a higher but affordable complexity due to DCT [1]. For pixel-domain WZC, Ascenso et al. proposed a novel motion compensated interpolation [3] to improve the accuracy at the beginning of the decoding stage and further update the side information during the decoding by using motion refinement [4]. In [5], an improved transform-domain WZC is proposed. With different quantization and motion-compensated interpolation schemes for generating more accurate side information at the decoder, the scheme of [5] out-performs the transform-domain WZC of [1]. Therefore, it is chosen as the baseline system in this paper.

All of the aforementioned turbo-code-based DVC systems make use of a rate-compatible-punctured-turbo-codes (RCPT) codec as the kernel of the Wyner-Ziv codec. The input bitstream length to the RCPT codec is usually chosen between 1000 and 2000, based on the desired complexity and performance, and the puncture window size is usually 8, resulting in 16 puncture code rates, which we refer to as rate control levels. Therefore, an increase by one rate control level (from one puncture code rate to the next) corresponds to an increase in the number of parity bits in the order of 12.5% of the input bitstream length. However, if the parity bits are carefully placed, a reduced number of parity bits can be used to successfully decode the codewords. In this paper, a block-adaptive Wyner-Ziv coding scheme is proposed for transform-domain DVC. The system performs rate control on each block by allocating parity bits only to those blocks that are found to be in error and by adaptively increasing the rate control level (transmitting more parity bits) only for blocks which still have decoding errors. This results in a significant reduction in bitrate in addition to a reduction in transmission power at the encoder since less parity bits need to be transmitted.

This paper is organized as follows. Section 2 presents the proposed DVC system, including block-adaptive Wyner Ziv Coding, blockwise formatting for bitplanes, block-adaptive puncturing, and block indices compression. Section 3 provides coding results and comparison with the system [1, 5] and with H.263.

2. PROPOSED DVC SYSTEM WITH BLOCK-ADAPTIVE WYNER-ZIV CODING (BAWZC)

A block diagram of the proposed system is shown in Figure 1. For a video sequence, the odd frames are the Key frames, and the even frames are the Wyner-Ziv frames. The Key frames can be intraframe compressed by using any conventional video codec and intraframe decompressed at the decoder with the same codec. In the proposed DVC scheme, the Wyner-Ziv frames are intraframe compressed by using a block-adaptive Wyner-Ziv codec (BAWZC). However, they are interframe decompressed by the proposed BAWZC jointly with the side information which is generated from the corresponding key frames.

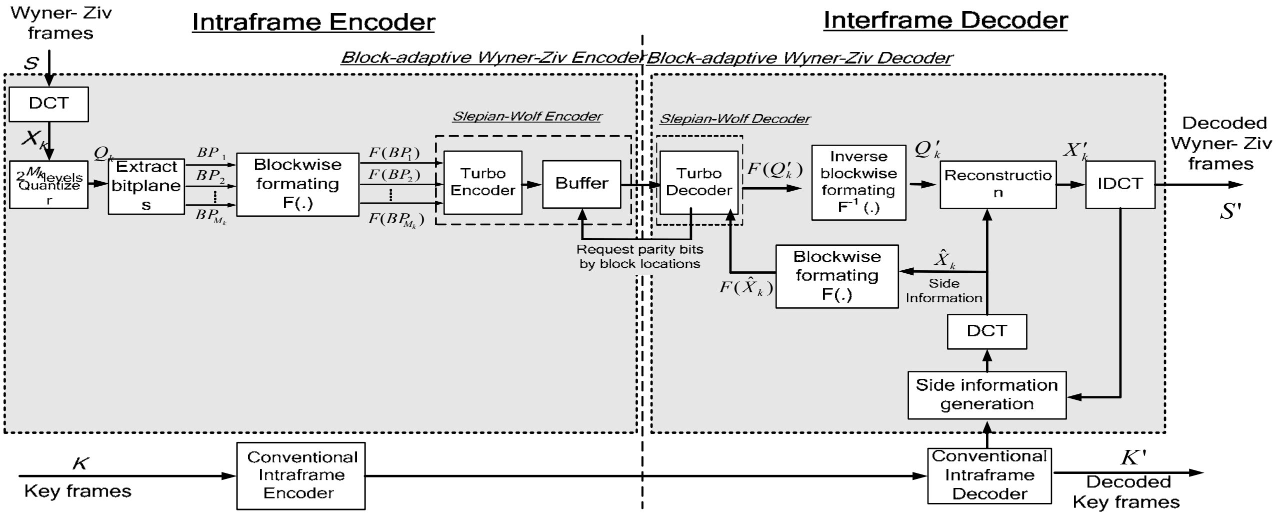


Figure 1. Block diagram of proposed DVC system with block-adaptive Wyner-Ziv Coding.

At the BAWZC encoder, a 4-by-4 DCT is applied to each Wyner-Ziv frame. The DCT coefficients are grouped into subbands. Each subband is then quantized using a uniform scalar quantizer as in [5]. Each quantized subband is then represented as a set of bitplanes according to the number of quantization levels of the corresponding quantizer. While existing WZC schemes [1, 5] use a single puncture rate for each bitplane, the proposed BAWZC system adaptively assigns puncture rates for each bitplane. For this purpose, each bitplane is reformatted into several blocks of size $M \times N$. Note that the existing transform-domain DVC systems [1, 5] correspond to a special case of the proposed system, where the block size is set to be equal to the size of an entire bitplane. A binary sequence is then formed, from each bitplane, by scanning each block rowwise as shown in Fig. 2(b). In contrast, the existing WZC schemes [1, 5] generate a binary sequence by scanning rowwise each bitplane as shown in Fig. 2(a). The resulting binary sequence is then fed into a turbo encoder as a symbol stream. After turbo encoding, all systematic bits are discarded and all parity bits are stored in a buffer. The parity bits are progressively requested by the BAWZC decoder only for decoded blocks with errors as described below.

The decoding process starts with the generation of the side information from two consecutive Key frames. For this purpose, a Three-Dimensional-Recursive-Search (3DRS) motion compensated interpolation scheme is adopted to estimate the side information as in [5]. The DCT coefficients of the generated side information are then computed using a 4×4 DCT. Each DCT subband is in turn decomposed into $M\times N$ blocks. In each subband, the DCT coefficients are rowwise scanned, block by block (Fig. 2(b)). The Turbo decoder starts decoding based on the blockwise-scanned DCT coefficients of the side information. As in [1, 5], perfect error detection is assumed at the decoder. The decoder would then progressively request, from the encoder buffer, parity bits for those blocks that are found to be in error. For this purpose, the indices of the blocks in error are coded and transmitted to the encoder. Existing DVC systems also make use of a similar feedback channel to send which bitplanes are in error to the encoder. The proposed BAWZC system results in a reduced total bit-rate for the Wyner-Ziv frames (which includes the bits needed for the parity bits and for the block indices) as compared to existing transform-domain DVC systems [1, 5]. In addition, since part of the total bit-rate is due to the coded block indices which are transmitted from the decoder to the encoder, the proposed system results in a significant reduction in the encoding bit-rate, which translates to a reduced bit traffic from the encoder to the decoder. This in turn results in a reduced transmission power at the encoder, which is desirable for low-power applications. The decoding process is repeated until there are no decoding errors. After all bitplanes are turbo decoded, an inverse blockwise formatting is applied to retrieve the original format. After reconstruction and IDCT, a decoded Wyner-Ziv frame is obtained.

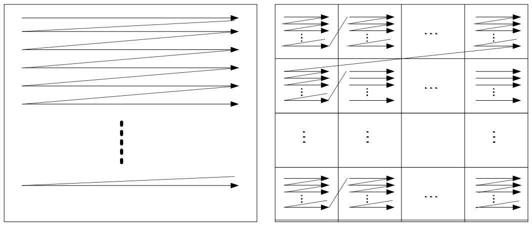
More details about the blockwise formatting, the block-adaptive puncturing, and the coding of the block indices are given below in Sections 2.1, 2.2, and 2.3, respectively.

2.1 Blockwise formatting

In DVC, the parity bits are used for correcting the side information to the source information. The accuracy of the side information affects the bitrate of the system. Common motion estimation problems that cause dramatic erroneous prediction, such as occlusion problem, reduce the accuracy of the side information. However, most of those problems are not uniformly distributed. Instead, they usually happen within some parts of the frame and are localized. A row-by-row scanning (Fig. 2(a)), as used in existing DVC systems [1, 5], to obtain a sequence from each bitplane of a subband of quantized DCT coefficients, spreads those errors in the bitstream. But, the proposed blockwise scanning (Fig. 2 (b)) alleviates this effect by keeping the localized errors close to each other in the generated symbol stream.

2.2 Block-adaptive puncturing

In the existing transform-domain DVC schemes [1, 5], the puncturing is applied uniformly over the whole bitplane.



(a) Row-by-row scanning (b) blockwise scanning

Figure 2. Bitplane scanning patterns

TABLE I. PUNCTURING TABLE

RCPT CODE: Puncturing Period P=8

Puncture Code Rate															
8/9	4/5	8/11	2/3	8/13	4/7	8/15	1/2	8/17	4/9	8/19	2/5	8/21	4/11	8/23	1/3
377	377	377	377	377	377	377	377	377	377	377	377	377	377	377	377
001	021	023	123	127	167	177	377	377	377	377	377	377	377	377	377
000	000	000	000	000	000	000	000	001	021	023	123	127	167	177	377

As discussed in Section 2.1, since the decoding errors are not usually uniformly distributed in the whole bitplane, some of the parity bits generated by uniform puncturing (i.e., a single puncture rate for the entire bitplane) would not help in the error correction and, thus, would be wasted. However, in the proposed BAWZC-based DVC system, the puncturing rate is independently adjusted for each bitplane block. The designed puncturing patterns are shown in Table I. Note that the puncture patterns in Table 1 are in octal form. The puncturing operation adds one parity bit, every eight systematic bits, for each bitplane decoding iteration. Since the puncturing rate is independently adjusted for each block, the interleaver in the turbo codec has to be modified. Instead of interleaving the whole bitplane, the interleaving is applied within each block only, which means the interleaver size is the block size. Also, the same interleaver is applied to all blocks. The decoder sends the block indices of the blocks with decoding errors to the encoder. The encoder incrementally transmits, to the decoder, the parity bits for the blocks in error, according to the puncturing rates at the current decoding iteration and their corresponding puncturing patterns as shown in Table I.

2.3 Block indices compression

The feedback channel plays an important role in turbo-code based DVC systems [1, 5]. The behavior and impact of the feedback channel for the pixel-domain DVC systems of [3,4] are discussed in [6]. The discussion in [6] also applies to existing transform-domain DVC systems [1,5]. In the proposed system, the BAWZC decoder has to perform not only rate control but also inform the WZC encoder which blocks need more parity bits.

After bitplane decoding, the WZC decoder determines which block still contains decoding errors. The block indices are then coded using differential and Huffman coding and are transmitted to the WZC encoder. The resulting compressed bitstream can be easily decoded by the WZC encoder using a look-up table for Huffman coding and simple addition operations for differential coding.

Let L_{CB} and L_{EB} denote the list of correct blocks' indices and the list of erroneous blocks' indices, respectively. In addition, let $N_{CB,I}$ and $N_{EB,I}$ denote, respectively, the size of L_{CB} and L_{EB} at bitplane decoding iteration I at the decoder. The L_{CB} and L_{EB} lists are available at both the WZC encoder and decoder; the decoder keeps these lists synchronized using the following procedure:

- Step 0: Initialization – I = 0; all blocks are in L_{CB}; $N_{CB,0}$=total number of blocks; $N_{EB,0}$=0.
- Step 1: After one bitplane decoding iteration, set $I=I+1$ and determine the indices of blocks in error.
- Step 2: Update the L_{CB} and L_{EB} lists at the decoder, and compute $D_{CB}=N_{CB,I} - N_{CB,I-1}$ and $D_{EB}=N_{EB,I-1} – N_{EB,I-1}$.
- Step 3: If $D_{CB}< D_{EB}$, update the L_{CB} and L_{EB} lists at the WZC encoder by sending to the WZC encoder the indices of the blocks that became correct at the current iteration I, i.e., those blocks that moved form the L_{EB} to the L_{CB}; otherwise, the indices of the blocks that were found to have errors are sent to the WZC encoder and are used in updating the lists at the encoder.

3. EXPERIMENTAL RESULTS

To test the performance of the proposed DVC scheme, the first 100 frames of the QCIF (144 x 176) Foreman and Mother_Daughter sequences at 30 frames per second were coded. In this simulation, we assume that the Key frames (odd frames) are losslessly transmitted to the decoder as in [1,5]. For comparison purposes, we use the same side information generation and reconstruction schemes as in [5]. The finest quantizer set in [5] is chosen for this simulation. Fig. 3 shows the frame-by-frame bit comparison of the proposed DVC system (BAWZC) and the DVC system of [5] (3DRS-DVC) for the Foreman sequence in terms of parity bits and total bits. For the proposed DVC system, the number of total bits is computed as the sum of the the parity bits (sent from encoder to decoder) and the block indices bits (sent from decoder to encoder). Fig. 3 also shows the resulting block indices bits (dotted line). The proposed BAWZC saves up to 70% of parity bits as compared to the system in [5]. When the side information is more accurate, the parity bits are less and more savings can be obtained. Most importantly, as shown in Fig. 3, the parity bits are consistently reduced for every frame as compared to [5]. This reduction in the parity bits corresponds not only to a reduction in the total bit-rate but also to a reduction in transmission power at the encoder, which is important in low-power applications.

Figs. 4 and 5 show the resulting rate-distortion performance of the proposed system for the Wyner-Ziv frames of the Foreman and Mother_Daughter sequences, respectively. For comparison, Figs. 4 and 5 also show the corresponding rate-distortion performance of the transform-domain DVC systems of [1] and [5] and of H.263 interframe coding (B frames). From Figs. 4 and 5, it can be seen that the proposed system (BAZWC) results in a rate-distortion performance that is much closer to H.263 interframe coding than existing DVC systems [1, 5]. In addition, the proposed DVC system results in a 25% to 35% reduction in the average bitrate (which include both parity bits and block indices bits for the proposed system) for the Foreman sequence, and in an average bit-rate reduction of 41% to 51% for the Mother_Daughter sequence, as compared to [1,5].

5. CONCLUSION

This paper presents a block-adaptive Wyner-Ziv coding (BAWZC) scheme for distributed video coding. The proposed system provides a generalization of existing schemes, which can be considered as a special case of the proposed BAWZC framework. Simulation results and comparison with existing state-of-the-art transform-domain DVC systems show that the proposed BAWZC-based DVC system exhibit a superior rate-distortion performance.

6. REFERENCES

[1] A. Aaron, S. Rane, E. Setton and B. Girod, "Transform-domain Wyner-Ziv codec for video", *Proc. Visual Communications and Image Processing*, *VCIP-2004*, San Jose, CA, Jan. 2004.

[2] R. Puri and K. Ramchandran., “PRISM: An Uplink-Friendly Multimedia Coding Paradigm”, in *Proc. of International Conference on Acoustics, Speech, and Signal Processing* (ICASSP), Hong Kong, April 2003.

[3] J. Ascenso, C. Brites, F. Pereira, "Improving Frame Interpolation with Spatial Motion Smoothing for Pixel Domain Distributed Video Coding", in *5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services*, Slovak Republic, July 2005.

[4] J. Ascenso, C. Brites, F. Pereira, "Motion Compensated Refinement for Low Complexity Pixel Based Distributed Video Coding", in *Proc. of International Conference on Advanced Video and Signal Based Surveillance*, Como, Italy, September 2005.

[5] W. Chien, L. J. Karam, G. P. Abousleman, “Distributed video coding with 3D recursive search block matching”, in *Proc. of* IEEE International Symposium on Circuits and Systems (ISCAS), Kos, Greece, June 2006.

[6] C. Brites, J. Ascenso, F. Pereira, "Feedback Channel in Pixel Domain Wyner-Ziv Video Coding: Myths and Realities", in *14th European Conference Signal Processing Conference*, Florence, Italy, September 2006.

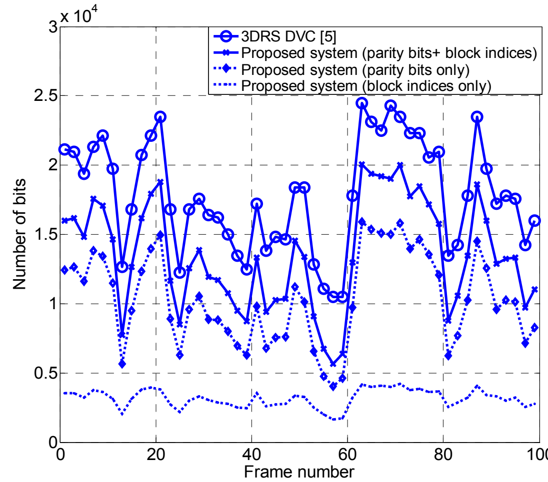


Figure 3. Bit comparison for the Foreman sequence

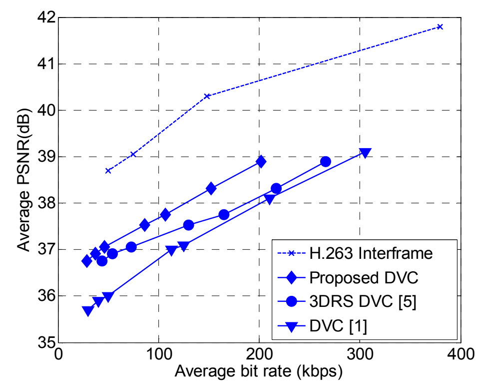


Figure 4. PSNR comparison for the Foreman sequence

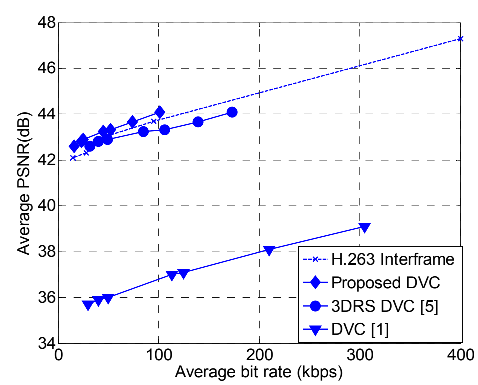


Figure 5. PSNR comparison for the Mother_Daughter sequence