MIXED-RESOLUTION DISTRIBUTED VIDEO CODEC WITHOUT MOTION ESTIMATION AT THE ENCODER

B. Macchiavello[1], *E. M. Hung*[2], *R. L. de Queiroz*[2], *D. Mukherjee*[3]

[1]Department of Computer Science, Universidade de Brasilia, Brasilia, DF, Brazil
[2]Electrical Engineering Department, Universidade de Brasilia, Brasilia, DF, Brazil
[3]Hewlett-Packard Labs, Palo Alto, CA, USA
Emails: {bruno,mintsu,queiroz}@image.unb.br, debargha.mukherjee@hpl.hp.com

ABSTRACT

Inspired by recent results showing that Wyner-Ziv coding using a combination of source and channel coding may be more efficient than pure channel coding, we have applied coset codes for the source coding part in the transform domain for Wyner-Ziv coding of video. The framework is a mixed-resolution approach where reduced encoding complexity is achieved by low resolution encoding of non-reference frames and regular encoding of the reference frames. Different from our previous works, no motion estimation is carried at the encoder, neither for the low resolution frames nor the reference frames. The entropy coders of the H.264/AVC codec were tuned to improve their performance for encoding the cosets. An encoding mode with lowest encoding complexity than H.264/AVC intra mode is achieved. Experimental results show a competitive rate-distortion performance especially at low bit rates.

Index Terms— distributed coding, Wyner-Ziv, cosets, zero-motion-vector

1. INTRODUCTION

Distributed source coding (DSC) refers to the compression of two or more correlated sources, that do not communicate with each other during encoding but are jointly decoded. DSC has its roots in the theorem of Slepian and Wolf [1], for lossless compression, and Wyner and Ziv [2], for lossy compression. Recently, practical DSC schemes have received a great deal of attention in efforts such as distributed video coding (DVC) schemes with reversed complexity [3]-[7], improved error resilience [8], efficient multi-view coding for distributed cameras [9], flexible decoding capability [10], and so on.

Most of these applications need a mechanism to encode images or video frames, under the assumption that they would be decoded in the presence of a side-information image/frame available at the decoder but not at the encoder. The prevalent mechanism for doing so in most DVC frameworks is to use bit-plane-wise systematic channel coding in the transform domain, where only the parity/syndrome bits are transmitted. However, it has also been shown in [11] that if sub-optimality of channel coding is explicitly considered, a combination of source and channel coding may be more efficient than pure channel coding. Advantages of using source coding on the least significant symbol planes or cosets in the transform domain were also shown [11]. The higher bit-planes can still use channel coding, but it was shown that even memoryless coset codes (where only source coding of cosets is used) can actually be both efficient and robust when optimal reconstruction is performed. The use of memoryless coset codes along with optimal reconstruction in a DVC framework has already been applied in our previous reversed complexity mixed-resolution architecture [12, 13], that was implemented as an extension mode in both the H.263 and the H.264/AVC codecs. There, the encoding complexity is reduced through lower resolution (LR) encoding of the non-reference frames, followed by Wyner-Ziv (WZ) coding. However, since motion estimation is still needed in the encoder, the complexity reduction is limited.

In this paper, we present an extension to our mixed-resolution DVC codec using H.264/AVC, where all macroblocks are coded using zero-motion-vector mode. This generates a loss in rate-distortion (RD) performance, but the encoding complexity will be lower than H.264 intra coding. Also, both the CAVLC and CABAC entropy coders where tuned to improve their performance for coset coding. Improvements in side-information (SI) generation are also presented. The results show that this new coding mode has very low encoding complexity and a competitive RD performance.

2. CODEC ARCHITECTURE

The encoding architecture is shown if Fig.1, where T stands for transformation and Q for quantization. The raw data is divided into reference frames and non-reference frames. The references frames can be coded as intra frames or using zero-motion-vector encoding, which can be seen as a simple frame subtraction. The non-reference frames are filtered and decimated before zero-motion-vector encoding. The encoding complexity is reduced when compared to regular zero-motion-vector encoding due to LR encod-

Work supported by HP Brasil.

ing. This architecture is a modification of a previous WZ mixed-resolution coding mode that can be found elsewhere [13]. In that previous work, motion estimation is performed in the LR frames and the reference frames. Therefore, this coding mode can achieve lower encoding complexity, with a loss in RD performance. The WZ coder works as follows: if the random variable X denotes a transform coefficient of the Laplacian residue, then, the quantization of X yields $X_q : X_q = \phi(X, QP), QP$ being the quantization step-size. The memoryless cosets C are generated by:

$$C = \begin{cases} (X_q) - M\lfloor X_q/M \rfloor, & (X_q) - M\lfloor X_q/M \rfloor < M/2 \\ (X_q) - M\lfloor X_q/M \rfloor - M, & (X_q) - M\lfloor X_q/M \rfloor \geq M/2 \end{cases} \quad (1)$$

where M is the coset modulus. Note that, if quantization bin x_Q corresponds to interval $[x_l(x_q), x_h(x_q)]$, then the probability of the bin $x_q \in \Psi_{X_q}$, and $c \in \Psi_C$ are given by:

$$p(x_q) = \int_{xh(x_q)}^{xl(x_q)} f_X(x)dx \quad (2)$$

$$\begin{aligned} p(c) &= \sum_{x_q \in \Psi_{X_q}, \psi(X_q, M) = c} p(x_q) \\ &= \sum_{x_q \in \Psi_{X_q}, \psi(X_q, M) = c} \int_{xh(x_q)}^{xl(x_q)} f_X(x)dx. \end{aligned} \quad (3)$$

Since the distribution $p(c)$ is symmetric, the entropy coder that already exists in the conventional video coder can be reused for C. However, an optimized entropy coder can achieve better results. The information that needs to be send in the WZ layer is the coded block pattern (cbp), that indicates which 8×8 blocks have residual content and the coset indices of the residual data. Note that an available choice for a particular coset can be $QP \rightarrow \infty$ or $M = 1$, which means that no coset is created for that coefficient (zero rate coding [13]). Hence, the number of cosets generated in a block is variable. Since the selection of $\{QP, M\}$ is done before entropy coding, instead of reusing the entropy coders of the conventional video codec (H.264/AVC in this case), we optimized the CAVLC and CABAC encoder for coset coding. For the CAVLC coder, the a priori knowledge of the number of cosets in a block, that can be seen as the number of non-zero coefficients, and the information of the modulus M, can be used to generate more adequate tables with smaller codes. For the CABAC, the same information can be used to create new contexts. However, since a great number of contexts already exist in the H.264/AVC standard, the information is used only for context selection.

The basic decoder architecture, a statistical model, an mechanism for estimation of the correlation statistics, and the selection of the encoding parameters $\{QP, M\}$ can be found elsewhere [13].

3. HIGH FREQUENCY ESTIMATION AND SIDE-INFORMATION GENERATION

In the decoder two modes can be selected. The low complexity decoding mode decodes only the base layer, interpolating the non-reference frames. This generates quality fluctuations between frames, but allows for low complexity at both ends without frame loss. The high decoding complexity mode initiates with the high frequency estimation in order to generate the side-information. The main idea is to try to recover the high frequency lost by LR encoding. This is done using the available previously decoded reference frames, which are at regular resolution. This process is referred as semi super-resolution. The super-resolved frame is then used as a noisy version of the original frame for side-information generation. The high complexity decoding mechanism is shown in Fig. 2. The semi super-resolution process presented here represents an improvement of previous works [13, 14].

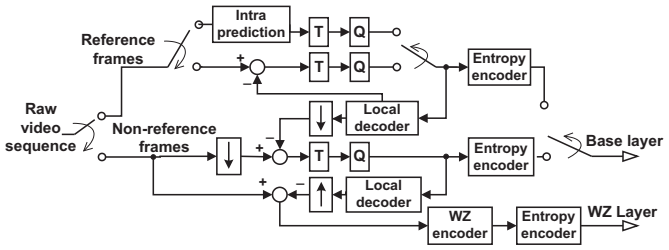

Fig. 1. Encoder Framework

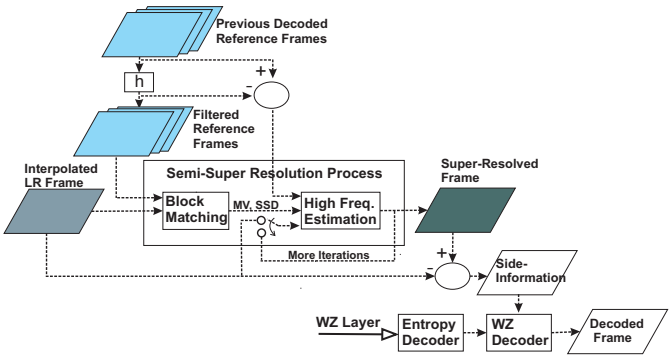

Fig. 2. Semi-super resolution process and SI generation.

In the semi super-resolution process, first, the past and future references frames in the frame store, of the current non-key frame, are low-pass filtered. The low-pass filter is implemented through down-sampling followed by an up-sampling process (using the same decimator and interpolator applied to the non-key frames). Next, a block-matching algorithm is applied as follows. Let a frame be $F = L + H$, where L is the LR (filtered) version of F, while H is the residue, or its high frequency. The block matching algorithm is done using only the LR information, or L, with variable block size and quarter-pixel precision. The blocks are partitioned twice, i. e., if the original block size is 16×16 the block is partitioned down to 4×4 blocks. The search for each block size can be made different. Since smaller partitions will always have lower SSD, a penalty is applied for block parti-

tioning. For example, in order to partition a 16×16 block into four 8×8 blocks the following condition must be true: $P * \sum_{i=1}^{4} SSD_i(8 \times 8) < SSD(16 \times 16)$, where P is the penalty.

Let the corresponding best predictor blocks be denoted as L_p and L_f in the past and future filtered frames respectively, independently of the block size. Then, the corresponding high frequency of the matched block (i.e., H_p and H_f) of the reference frames is added to the block to be super-resolved weighted by the SSD. In other words, we add $(H_p * SSD_f + H_f * SSD_p)/(SSD_f + SSD_p)$ to all blocks. We can also iteratively super-resolve the frames. After the first iteration, the strength of the low-pass filter is reduced (in our implementation, the low-pass filter is eliminated after one iteration). Also, the grid for block matching is offset from iteration to iteration to smooth out the blockiness and to add spatial coherence. It is important to note that after the first iteration we already have a frame with high frequency content. Hence, we replace $L + H$ rather than adding H [13]. However, from iteration to iteration we want less change. So, a threshold is introduced after the first iteration. The block replacement is done only if the SSD is lower than a threshold.

4. RESULTS AND DISCUSSION

In order to analyze the performance of the proposed framework tests were carried out with CIF and 4CIF sequences. We encoded 299 frames of each CIF sequence and 99 frames of each 4CIF sequence. No RD optimization was used, and zero-motion-vector prediction was used in all tests. The coding modes used were $IpPpP...$ and $IbPbP...$, where p and b represents a decimated frame. A bi-directional frame means that forward and backward zero-motion-vector prediction was done and only the best of the two predictors was selected for each marcroblock. The decimation factor used was 2×2. For the semi super-resolution process, the initial block size used 16×16 pixels and was partitioned down to 4×4 pixels. The initial search range for the CIF sequences was set to 16 pixels, as the blocks were partitioned the search range was set to 16 and 4 pixels respectively. For the 4CIF sequences the search ranges used were $32, 16$ and 8.

The gains after tuning the entropy coders were modest but not insignificant. The bit rate reduction in the WZ layer, for 9 different CIF sequences, varies from 1% to 7% for the modified CAVLC, while with CABAC the gains were around 1.5%. In Table 1 it is shown the average number of encoded frames per second using different configurations. For CIF resolution 9 different sequences were used. For 4CIF resolution we used 4 sequences. The proposed framework was implemented in the H.264 KTA software [15], and tests were made on an Intel®Pentium D 915 2.80 GHz Dual Core with Windows®OS. As can be seen the proposed framework reduced significantly the encoding time. Note that zero-motion- vector encoding does not use any of the H.264/AVC intra modes. Therefore, it has the potential to be faster than H.264/AVC intra encoding. In our tests, intra encoding is faster than zero motion-vector encoding due to software implementation. However, using mixed resolution encoding we obtain the fastest encoding configuration.

Table 1. Average encoded frames per second

	CIF		4CIF	
	H.264/AVC	*Proposed*	*H.264/AVC*	*Proposed*
IBPBP	7.88	12.28	2.19	3.54
IPPPP	8.83	12.86	2.72	3.61
IIII	11.60	–	3.21	–

In Figs. 3 (a)-(c) it is presented the RD performance for the Y component of CIF sequences. We tested: (i) the conventional H.264 using zero-motion vector encoding in $IBPBP...$ or $IPPPP...$ modes; (ii) the conventional H.264 with all frames coded as intra; (iii) the base layer of the proposed scheme; and (iv) the high complexity decoding mode of the proposed framework. The sequences shown are Silent, Foreman and Soccer, which have low, mid and high motion content respectively. This can be inferred from the curves, because the gap between intra and zero-motion-vector coding reduces as the sequences presents more motion. As can be seen in the figure, the proposed codec works better for low to mid rates, but it is competitive at all rates and have the lowest encoding complexity of all curves.

In Figs. 3 (d)-(f) similar results are presented for 4CIF sequences. It can be observed that the performance of the codec is also competitive. The results in Fig. 3 show the RD performance only of the luminance component, since this component is more important for subjective evaluation. However, the codec was implemented for all YUV components. For the chrominance component using the $4 : 2 : 0$ format, the mixed resolution framework with the proposed SI generation mechanism and WZ enhancement layer has very good results. As it can be seen in Fig. 4, that shows the same result of Fig. 3 (b) with the PSNR computed for all three components, the RD performance of the proposed framework outperforms regular zero-motion vector and intra coding with lower encoding complexity.

5. CONCLUSION

In this paper, we presented a mixed-resolution DVC codec, without motion estimation at the encoder. The entropy coders of conventional H.264/AVC were tuned for cosets indices. The results have shown that we can achieve a coder with very low encoding complexity and competitive RD performance. The RD performance is similar to regular zero-motion-vector encoding, while the encoding complexity is lower than H.264 intra encoding. However, there is still room to improve the codec. For example, a study can be made in order to determine the best decimation factor for different resolutions.

6. REFERENCES

[1] J. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans on Information Theory*, vol. 19, no. 4, pp. 471–480, Jul 1973.

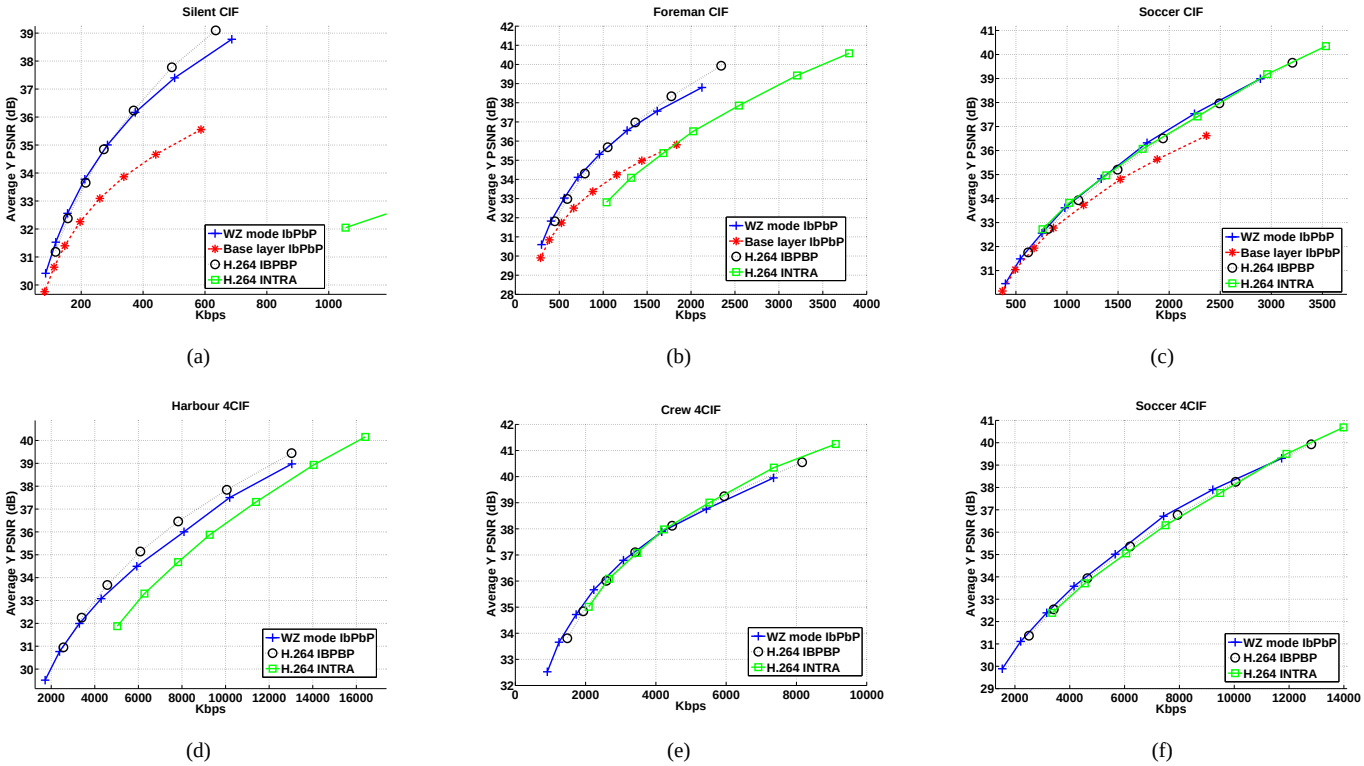

Fig. 3. PSNR results for Y component. All coding modes, except intra coding, are zero-motion-vector coding (a)-(c) CIF sequences, (d)-(f) 4CIF sequences.

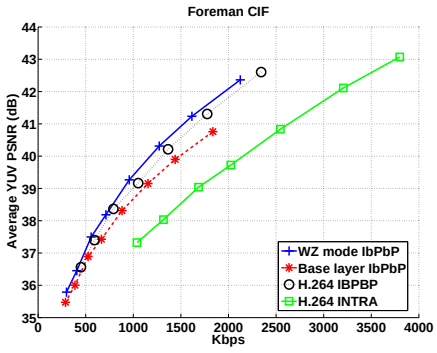

Fig. 4. YUV PSNR results for Foreman CIF sequence

[2] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans on Information Theory*, vol. 2, no. 1, pp. 1–10, Jan 1976.

[3] R. Zhang A. Aaron and B. Girod, "Transform-domain Wyner-Ziv codec for video," *Proc. SPIE Visual Com. and Img. Proc.*, vol. 5308, pp. 520–528, Jan 2004.

[4] S. Rane B. Girod, A.M. Aaron and D. Rebollo-Monedero, "Distributed video coding," *Proc. of the IEEE*, vol. 93, no. 1, pp. 71–83, Jan 2005.

[5] Q. Xu and Z. Xiong, "Layered WynerZiv video coding," *IEEE Trans on Img. Proc.*, vol. 15, no. 12, pp. 3791–3809, Dec 2006.

[6] A. Majumdar M. Tagliasacchi and K. Ramchandram, "A distributed-source-coding based robust spatio-temporal scalable video codec," *Pic. Coding Symposium*, Dec 2004.

[7] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," *Pic. Coding Symposium*, vol. 17, no. 9, pp. 1103–1120, Nov 2007.

[8] B. Girod S. Rane, A. Aaron, "Error-resilient video transmission using multiple embedded Wyner-Ziv decriptions," *Proc. IEEE Int. Conf. on Img. Proc.*, Sept 2005.

[9] F. Wu D. Zhao X. Guo, Y. Lu and W. Gao, "Wyner-Ziv-based multiview video coding," *IEEE Trans. Circ. & Sys. For Video Tech.*, vol. 18, no. 6, pp. 713–724, 2008.

[10] N. M. Cheung and A. Ortega, "Compression algorithms for flexible video decoding," *Proc. of SPIE Visual Com. and Img. Proc.*, Jan 2008.

[11] D. Mukherjee, "Parameter selection for Wyner-Ziv coding of laplacian sources with additive laplacian or gaussian innovation," *IEEE Trans. Signal Processing*, vol. 57, no. 8, pp. 3208–25, Aug 2009.

[12] D. Mukherjee, B. Macchiavello, and R. L. de Queiroz, "A simple reversed-complexity Wyner-Ziv video coding mode based on a spatial reduction framework," *Proc. of SPIE Visual Com. and Img. Proc.*, vol. 6508, pp. 1Y1–1Y12, Jan 2007.

[13] B. Macchiavello, D. Mukherjee, and R. L. de Queiroz, "Iterative side-information generation in a mixed resolution Wyner-Ziv framework," *IEEE Trans. Circ. & Sys. for Video Tech.*, vol. 19, pp. 1409–1423, Oct 2009.

[14] B. Macchiavello, F. Brandi, E. Peixoto, R. L. de Queiroz, and D. Mukherjee, "Side-information generation for temporal and spatial scalable Wyner-Ziv codecs," *EURASIP Journal of Image and Video coding*, vol. 2009, pp. 1–11, 2009.

[15] KTA software and JM H.264/AVC reference software: http://iphome.hhi.de/suehring/tml/.