Motion Compensated Refinement for Low Complexity Pixel Based Distributed Video Coding

João Ascenso
Instituto Superior de Engenharia de Lisboa
Instituto de Telecomunicações
joao.ascenso@lx.it.pt

Catarina Brites, Fernando Pereira
Instituto Superior Técnico
Instituto de Telecomunicações
catarina.brites@lx.it.pt, fp@lx.it.pt

Abstract

Distributed video coding (DVC) is a new coding paradigm that enables to exploit video statistics, partially or totally at the decoder. A particular case of DVC, Wyner-Ziv coding, deals with lossy source coding with side information at the decoder and allows a shift of complexity from the encoder to the decoder, theoretically without any penalty in the coding efficiency. The Wyner-Ziv solution here described encodes each video frame independently (intraframe coding), but decodes the same frame conditionally (interframe decoding). At the decoder, motion estimation and compensation tools are responsible to obtain an accurate interpolation of the original frame using previously decoded (temporally adjacent) frames. This paper proposes a novel approach to improve the performance of pixel domain Wyner-Ziv video coding by using a motion compensated refinement of the decoded frame and use it as improved side information. More precisely, upon partial decoding of each frame, the decoder refines its motion trajectories in order to achieve a better reconstruction of the decoded frame.

1. Introduction

Nowadays, the digital video coding solutions available rely on the powerful hybrid block-based motion compensation/DCT transform (MC/DCT) architecture. All the ITU-T VCEG and ISO/IEC MPEG standards follow this approach, mostly targeting applications where the video content is encoded once and decoded multiple times, e.g. broadcasting or video streaming. In such applications, the video codec architecture is primarily driven by the one-to-many model of a single complex encoder and multiple light (cheap) decoders; typically the encoder is 5 to 10 times more complex than the decoder [1]. The complexity burden of the encoder is mainly associated with the motion estimation and compensation tasks, which account for a major share of the coding gain in rate-distortion (RD) performance.

However, this architecture is being challenged by several emerging applications such as wireless video surveillance, multimedia sensor networks, wireless PC cameras and mobile camera phones. These applications have different requirements from those targeted by traditional video delivery systems. For example, in wireless video surveillance systems, low cost encoders are important since there is a high number of encoders and only one or few decoders.

Distributed video coding, a new video coding paradigm, fits well in these scenarios, since it enables to explore the video statistics, partially or totally, at the decoder only, relying on a low encoding complexity. For the wireless video surveillance scenario, if cameras sense partially overlapped geographical areas, DVC allows to explore the correlation between the multiple video sequences just at the decoder. Therefore it is possible to achieve a low encoding complexity, reducing the total cost (in terms of hardware) of the system. In this case, it is essential to have a new coding configuration with low-power consumption and low-complexity encoder devices, possibly at the expense of high-complexity decoders. Simultaneously, also high compression efficiency is needed to reach the best video quality for the available bandwidth, while using the lowest complexity.

Under this new coding paradigm, a challenging problem arises: how to achieve or at least approach the coding efficiency of state-of-the-art hybrid video coding schemes, e.g. the H.264/AVC standard [2]. Information Theory provides two major results that theoretically prove it is possible to achieve this goal: the Slepian-Wolf [3] and the Wyner-Ziv [4] theorems. These two theorems show that separate encoding and joint decoding (with increased complexity at the joint decoder) of two correlated sources is as efficient as joint encoding and decoding. The Slepian-Wolf theorem refers to lossless compression while the Wyner-Ziv theorem refers to lossy compression with side information available at the decoder.

Despite the theory established by the Slepian-Wolf and Wyner-Ziv theorems being well known for a long time (since the 70's), only recently practical video coding schemes have been presented. One of the practical approaches is a turbo-based pixel domain Wyner-Ziv coding scheme [5], where the decoder is responsible to explore all the source statistics, and therefore to achieve compression for the Wyner-Ziv solution. Independently, a

The work presented was developed within VISNET, a European Network of Excellence (http://www.visnet-noe.org).

similar solution was also proposed [6], referred as PRISM, with increased robustness to channel losses and flexible sharing of computational complexity between the encoder and the decoder. In another independent work [7], the Wyner-Ziv approach was proposed to eliminate the mismatch or drift in traditional (predictive) coding schemes when channel errors occur.

The Wyner-Ziv coding architecture presented in this paper is based on the pixel domain Wyner-Ziv coding scheme presented in [5] and was chosen because of its simplicity and low encoding complexity. In this architecture, the video frames are organized into two types: key frames and Wyner-Ziv frames. While the key frames are encoded with a conventional intraframe codec, the frames between them are Wyner-Ziv encoded. At the decoder, the side information is obtained using previously decoded key frames and a motion interpolation scheme, responsible to obtain the most accurate representation of the original frame. The more accurate the side information is, the fewer are the Wyner-Ziv bits required to provide a reliable decoding of the Wyner-Ziv frame.

Several frame interpolation techniques can be employed at the Wyner-Ziv decoder to generate the side information. However, since the RD performance of the Wyner-Ziv (WZ) video coding scheme is highly dependent on the quality of the side information frame, suitable techniques must be used to reliably generate this frame. In a previous work, the authors have proposed a new motion compensated frame interpolation scheme, based on bi-directional motion estimation and spatial smoothing of the motion vectors [8]. The experimental results show solid improvements in RD performance when compared to other similar solutions. In this paper, the main goal is still to get the side information frame as close as possible to the current Wyner-Ziv frame in order to improve the coding efficiency. For this paper, the new idea is to continuously improve, as WZ bits are received, the quality of the decoded frame and use it as refined side information in order to further improve the coding efficiency. To achieve this, upon partial decoding of the frame the decoder refines its motion trajectories and is able to achieve a better reconstruction of the decoded frame. Therefore, the main contributions of this paper are: new matching criteria for the motion refinement process, new interpolation (decoding) modes, an iterative refinement algorithm for the bi-directional motion vectors, and a new motion compensated reconstruction function. The impact of the overall motion refinement scheme is evaluated in the context of a turbo-based pixel domain Wyner-Ziv video codec including the frame interpolation scheme proposed in [8].

This paper is organized as follows. First, in Section 2, the pixel domain Wyner-Ziv video codec architecture and the most important modules are presented. In Section 3, the approach to further refine the decoded frame (and therefore the side information) is described, detailing the three available interpolation modes. In Section 4, several experiments are performed in order to evaluate and compare the coding efficiency of the proposed approach and, finally, in Section 5, the conclusions and future work topics are presented.

2. IST-PDWZ video codec architecture

The IST Pixel Domain Wyner-Ziv (IST-PDWZ) video codec here presented is based on the pixel domain Wyner-Ziv coding architecture proposed in [5]. However, there are major differences in the frame interpolation tools and in the motion refinement scheme which are completely new. This approach offers a pixel domain intraframe encoder and interframe decoder with very low computational encoder complexity. When compared to traditional video coding, the proposed encoding scheme is less complex by several degrees of magnitude. The main IST-PDWZ characteristic is the use of the powerful turbo-based channel codes to compress the source video. Figure 1 illustrates the global architecture of the IST-PDWZ codec.

In this architecture each even frame X_{2i} of the video sequence is called Wyner-Ziv frame and the two adjacent odd frames X_{2i-1} and X_{2i+1} are referred as key frames; for the moment, it is assumed that they are perfectly reconstructed at the decoder. Each pixel in the Wyner-Ziv frame is uniformly quantized with 2^N intervals; from the resulting quantized symbol stream, for the entire image, bitplane extraction is performed and then each bitplane is independently turbo encoded. At the decoder, the frame interpolation module generates the side information, Y_{2i}, which will be used by the turbo decoder and reconstruction modules. The decoder operates in a bitplane by bitplane basis and starts by decoding the most significant bitplane and it only proceeds to the next bitplane after each bitplane is successful turbo decoded (i.e. when most of the errors are corrected). The reconstruction and motion refinement processes are also performed bitplane by bitplane in order to continuously improve the side information. By using the switch S in Figure 1, the side information for the first bitplane is obtained by frame interpolation while for the next bitplanes it is obtained by motion refinement. Finally, the motion refinement module has also the task to improve the quality of the reconstructed image for a certain bitrate, i.e. after decoding an integer number of bitplanes. The refinement is performed with the help of the key frames X_{2i-1} and X_{2i+1} and the motion trajectories obtained by frame interpolation (first iteration) or in the previously decoded bitplane.

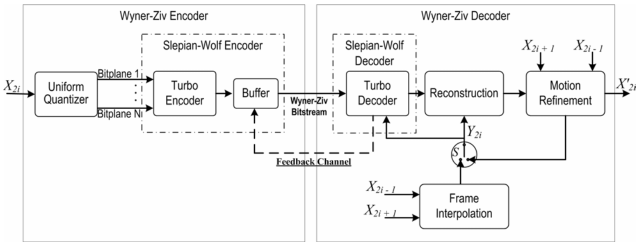

Figure 1. IST-PDWZ video codec architecture.

In the following, the Slepian-Wolf codec, the frame interpolation tools used and the reconstruction module are briefly presented.

2.1. Slepian-Wolf codec

The IST-PDWZ architecture presented in this paper uses a Slepian-Wolf rate compatible punctured turbo (RCPT) coder in order to correct the mismatch (errors) between the side information and the frame to be decoded. As shown in Figure 1, the Slepian-Wolf encoder includes a turbo encoder and a buffer and it produces a sequence of parity bits (redundant bits) associated to each bitplane. In this architecture, two identical recursive encoders of rate ½ are used; this means that for each information bit, two parity bits are produced. The parity bits generated by the turbo encoder are then stored in the buffer, punctured and transmitted upon request by the decoder while the systematic bits are discarded. The puncturing operation allows sending only a fraction of the parity bits and follows a specific puncturing pattern. The feedback channel is necessary to adapt to the changing statistics between the side information and the frame to be decoded, i.e. to the quality (or accuracy) of the frame interpolation or motion refinement process. At the decoder, the iterative MAP (Maximum A Posteriori) turbo decoder employs a Laplacian noise model to aid in the error correction capability of the turbo codes. An ideal error detection capability is also assumed at the decoder, i.e. the decoder is able to measure in a perfect way the current bitplane error rate, Pe.

2.2. Frame interpolation tools

There are several frame interpolation techniques that can be employed at the Wyner-Ziv decoder to generate the side information, Y_{2i}. The choice of the technique used can significantly influence the IST-PDWZ codec RD performance. More accurate side information through frame interpolation means fewer errors (Y_{2i} is more similar to X_{2i}) and therefore the decoder needs to request less parity bits from the encoder and the bitrate is reduced for the same quality. Traditional motion estimation and compensation techniques used at the encoder in conventional video coding are natural candidates to perform frame interpolation. However, best results are obtained if the true motion between the two temporally adjacent key frames is found and only then the motion compensated interpolation is performed. The frame interpolation framework used here is composed of four major tools: forward motion estimation, bi-directional motion estimation (for refinement), spatial smoothing of motion vectors (to eliminate outliers), and bi-directional motion compensation [8].

2.3. Reconstruction module

The side information and the turbo decoded bitplanes are used at the reconstruction module to obtain the first estimate of the decoded frame X'_{2i}. Consider that M bitplanes were successfully decoded; if for each pixel the bitplanes are grouped, it is possible to obtain a quantization bin *q'* for each pixel that tells the decoder where the original pixel value lies (an interval). The quantization bin *q'* is an approximation of the true quantization bin *q* obtained at the encoder before bitplane extraction. The reconstruction function can be mathematically described by the following three equations [5]:

$$X'_{2i}(x,y) = 2^{8-M} q'(x,y),\ q'(x,y) > Q\left(Y_{2i}(x,y)\right) \tag{1}$$

$$X'_{2i}(x,y) = Y_{2i}(x,y),\ q'(x,y) = Q\left(Y_{2i}(x,y)\right) \tag{2}$$

$$X'_{2i}(x,y) = 2^{8-M}\left(q'(x,y)+1\right),\ q'(x,y) < Q\left(Y_{2i}(x,y)\right) \tag{3}$$

where (*x*, *y*) corresponds to the pixel location, *q'*(*x*,*y*) corresponds to the decoded quantization bin and $Q(Y_{2i}(x,y))$ corresponds to the quantized symbol (bin) of the side information frame with the same precision (M bitplanes).

Intuitively, the reconstruction function points out that if the side information $Y_{2i}(x,y)$ is within the decoded quantized bin then the reconstructed value $X'_{2i}(x,y)$ is made equal to the side information value $Y_{2i}(x,y)$, see (1). However if $Y_{2i}(x,y)$ is out of the limits imposed by the decoded bin (side information error), the reconstructed value $X'_{2i}(x,y)$ assumes the lowest intensity value or the highest intensity value within the decoded quantized bin, see (2) and (3). The reconstruction function described by the above equations bounds the error between the Wyner-Ziv frames X_{2i} and the reconstructed frames X'_{2i} to the quantizer coarseness since the reconstructed pixel value is between the boundaries of the decoded quantized bin.

3. Motion compensated refinement

In this paper, the novel idea to be explored is the improvement of the quality of the decoded frame each time a bitplane is successfully decoded and use it as enhanced side information to help in the decoding of future bitplanes. This new approach uses the reconstructed image (see Figure 1) and the adjacent key frames in order to refine the motion vectors and thus to obtain a new and improved version of the decoded and side information frames.

The motion refinement is performed only after the turbo decoder has converged and successfully corrected the majority of errors ($\text{Pe} \leq 10^{-3}$) for a particular bitplane. Then the reconstructed image X'_{2i} is obtained, and the motion refinement process starts. The contribution of the reconstructed image is two-fold. First it can aid in the construction of more accurate side information, especially with fewer errors in the next bitplane to be decoded, which means less parity bits and hence less bitrate for the Wyner-Ziv frames. This operation can be interpreted as an inner-loop in the decoder. Second, the reconstructed image can also be refined with data from adjacent key frames in order to further improve the quality of the decoded image X''_{2i} Previous approaches in the literature don't perform this step, which means that the reconstruction is sub-optimal in the rate-distortion sense. This can be interpreted as a new reconstruction function that is guided by motion estimation to enhance the final quality of the decoded image.

The IST-PDWZ motion compensated refinement works at the block level and refines only the blocks which have pixel values that were corrected by the reconstruction process; see equations (1) and (3). The blocks refined correspond to those with pixels with more accurate values (closer to the original), when compared to the first estimate of the side information (no parity bits received). So, a threshold τ was established that defines which blocks are candidates for refinement (after the decoding of a given bitplane). If a block satisfies the following mathematical criteria, it is refined; otherwise it will not be processed:

$$\sum_{(x,y)\in Block} \left| X'_{2i}(x,y) - Y_{2i}(x,y) \right| \geq \tau \tag{4}$$

Next, the motion refinement algorithm including four tools is described: i) matching criteria function to perform motion estimation; ii) three decoding interpolation modes; iii) iterative technique to jointly optimize the forward and the backward motion vectors in the bi-mode and, finally, iv) motion compensated reconstruction function used to update the decoded frame and consequently the side information. These tools are integrated in a single motion refinement framework, with the following contributions: i) the matching criteria defines how "good" a candidate motion vector is; ii) the interpolation modes define how "good" a reference frame is; iii) the search algorithm defines how to find the "best" motion vectors; and iv) the MC reconstruction function how to obtain an "optimal" reconstruction in the RD sense.

3.1. Matching criteria for reconstructed frame motion estimation

One important issue in the design of the motion refinement algorithm is the matching criteria to be used when motion estimation is performed. Generally, the goal of the motion estimation is to minimize a cost function d – or matching criteria – that measures the prediction error for a given block, i.e. how similar two blocks are. Examples are the popular mean absolute difference (MAD) and the mean squared error (MSE) for the luminance block.

However, here it is proposed to use the reconstructed frame for motion estimation; this frame has some of the bitplanes already corrected but others still with errors. So, a new cost function which gives more importance to blocks in the reference frame which have pixels with equal bins to the reconstructed frame is necessary. Each bin represents the concatenation of all the bitplanes already correctly turbo decoded (almost free of errors). A new cost function based on MAD and in the reconstruction function presented in Section 2.3 is proposed here:

$$MAD(x,y) = X'_{2i}(x,y) - X_R(x+dx, y+dy) \tag{5}$$

$$Q\left(X'_{2i}(x,y)\right) = Q\left(X_R(x+dx, y+dx)\right) \tag{6}$$

$$WMAD(x,y) = \sum_{(x,y)\in Block} \begin{cases} MAD(x,y)/2^M, & \text{if (6) is satisfied} \\ MAD(x,y), & \text{otherwise} \end{cases} \tag{7}$$

where $X'_{2i}(x,y)$ represents the reconstructed frame, $X_R(x,y)$ represents the reference frame and (dx, dy) represents the motion vector. The reference $X_R(x,y)$ can be the past key frame (R=2i-1), the future key frame (R=2i+1) or it can be the average between the two adjacent key frames:

$$X_R(x,y) = (X_{2i-1}(x+dx_b, y+dy_b) + X_{2i+1}(x+dx_f, y+dy_f))/2 \tag{8}$$

with the motion vector (dx_b, dy_b) pointing to the past key frame and the motion vector (dx_f, dy_f) pointing to the future key frame. See the next Section for the interpolation modes permitted and for the mode decision rules. The new criteria is called weighted mean absolute difference (WMAD) and divides the MAD of a given pixel by a fixed weight (2^M) when the pixel in the reconstructed frame $X'_{2i}(x,y)$ belongs to the same bin of the collocated pixels in the reference frame $X_R(x+dx,y+dy)$. The idea is to penalize blocks in the reference frame which have pixels with different bin values of the reconstructed frame, since the bin of the reconstructed frame is almost free of errors, i.e. equal to the original frame. Of course, the bin is very coarse if only one bitplane is turbo decoded (two possible values) but the precision is higher if more bitplanes have been requested and thus increased gains in quality are possible.

3.2. Interpolation modes for refinement

The first time motion refinement is performed, after receiving the most significant bitplane, the motion vectors used to initialize the refinement algorithm are those resulting from the frame interpolation process. Each time a bitplane is turbo decoded and a new reconstructed image is obtained, the motion refinement starts in order to update the motion vectors and to obtain an improved decoded frame. There are three block interpolation modes (Figure 2) that can be chosen by the decoder motion refinement tool:

- **Bw-mode (backward):** the block in the reconstructed frame is interpolated using only one block from the previous decoded key frame X_{2i-1}.
- **Fw-mode (forward):** the block in the reconstructed frame is interpolated using only one block from the future decoded key frame X_{2i+1}.
- **Bi-mode (bi-directional):** the block in the reconstructed frame is interpolated using the average of one block in the future key frame and another block in the past decoded key frame, at arbitrary positions.

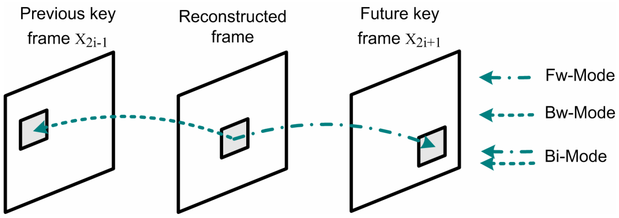

Figure 2. Available interpolation modes at the decoder.

The mode selection is decided exclusively by the decoder which contrasts to previous approaches, e.g. in [6] the encoder performs the mode decision. The three interpolation modes are similar to the modes available in the encoder for traditional video coding schemes (e.g. H.263+, H.264), but the mode decision is performed each time a bitplane is turbo decoded and a new reconstructed image is obtained. The mode decision is performed accordingly to the matching criteria WMAD defined in the previous section. For the backward and forward modes, full search for the best block, i.e. the minimum WMAD, in the corresponding reference frame is performed. For the bi-mode, joint estimation of the forward and backward motion vectors is performed to search for the minimum WMAD when both references are used. In the next section, the search algorithm is presented. Hence, three values that correspond to the WMAD of the best block(s) for the three modes are obtained and then the mode with the minimum WMAD value is chosen. Since the motion refinement is carried out each time a bitplane is turbo decoded, the decoder is allowed to choose any mode at any time, with only one restriction: if the coding mode chosen for a block is Fw-mode or Bw-mode, in future bitplanes the coding mode for this block cannot be changed, i.e. mode decision is not performed. This restriction allows for considerable savings in computational complexity without sacrificing RD performance (the search is restricted to just one mode).

3.3. Searching strategy for the bi-mode

An iterative motion refinement algorithm based on [9] is used to calculate the forward and backward motion vectors for the bi-mode interpolation. It needs as initialization the motion vectors obtained by the frame interpolation scheme or in the previous motion refinement runs and refines the motion vectors iteratively until a locally optimal solution is found.

It should be noted that the forward and backward motion vectors calculated by the frame interpolation module are symmetric (assuming constant motion), since only the key frames are available before the first bitplane is decoded. Since after that more accurate versions of the decoded image are successively available, forward and backward motion vectors are not necessarily symmetric.

So, for each block in the decoded image, a similar block is searched in temporally adjacent key frames and a pair of motion vectors that satisfies the matching criteria described above is obtained. In this approach, an iterative technique jointly optimizes both motion vectors. The core of this algorithm is to keep one motion vector fixed and then to find a better match, according to the matching criteria WMAD, for the remaining motion vector. This operation is repeated with the other motion vector fixed. Consider the backward motion vector $v_b = (dx_b, dy_b)$ and the forward motion vector $v_f = (dx_f, dy_f)$. The search range is defined as S; B'_{2i} represents one block in the reconstructed frame X'_{2i}, B_{2i-1} represents one block in X_{2i-1} and B_{2i+1} represents one block in $X_{2i+1.}$ Next, the algorithm is described in detail:

1. Fix the motion vector v_b and update v_f according to:

$$\min_{S} \; WMAD\left(B'_{2i}, \frac{\left(B_{2i-1}\left(v_b\right) + B_{2i+1}\left(v_f\right)\right)}{2} \right) \tag{9}$$

2. Fix motion vector v_f and update v_b according to (9).
3. Repeat steps 1 and 2 until v_f and v_b remain unchanged.

Finally, the motion vectors are updated with the result of this algorithm; they will be the starting point when motion refinement will be performed for the next turbo decoded bitplane.

3.4. Motion compensated reconstruction

Now that the motion vector(s) and the corresponding modes were obtained, a new version of the decoded frame is constructed by simply using motion compensation as defined in standard video coding schemes. However, since with this process some of the pixels lie outside the decoded bin (almost free of errors), this frame is not the best solution as a new decoded frame. In order to overcome this problem it is proposed here to only refine the pixels in the decoded frame if the bin of the refinement pixel agrees with the decoded bin. This procedure is called here motion compensated reconstruction since it corresponds to a selective reconstruction of the decoded frame using the computed motion vectors and modes only when improvements are obtained. This process can also be interpreted as an improvement of the reconstruction process as defined in equation (2). Finally, the side information is updated with the newly decoded frame inside the decoding bin, closing the inner-loop at the decoder, and so improving its quality for the decoding of the remaining bitplanes.

4. Experimental results

In order to evaluate the RD performance of the proposed improvements to the IST-PDWZ codec, three configurations are considered: i) no refinement: only frame interpolation without motion refinement as in [8]; ii) Partial MR: only with motion compensated reconstruction of the decoded frame, i.e. without updating the side information for the decoding of remaining bitplanes; and iii) Full MR: full update of the side information and motion compensated reconstruction of the decoded frame. The test conditions used are:

- **Frame interpolation**: 8×8 block size, ±8 pixels for the search range of the full block motion estimation, and ±2 pixels for the search range of the bi-directional motion estimation (used for refinement) [9].
- **Motion refinement:** Block size remains unchanged; the threshold τ to determinate if the block is refined or not is an average difference of 0.15 per pixel and the search range is ±4 pixels.

These parameters were obtained after performing extensive experiments and are the best fit for QCIF sequences. The results for the first 100 frames of the Foreman and Coastguard QCIF sequences are shown in

Figures 3 and 4. In both figures, it is only included the rate and the distortion for the luminance of the Wyner-Ziv frames as typically used for these cases; the Wyner-Ziv frame rate is 15 frames per second. Different quantization levels were applied, $2^M \in \{2,4,8,16\}$, to obtain the four rate-distortion points. The results are compared against the H.263+ intraframe coding and the H.263+ interframe coding with a I-B-I-B structure. In the later case, it is only taken into account the rate and PSNR of the B frames. For the Foreman sequence it is also included as benchmarking the RD performance of the best available pixel domain Wyner-Ziv codec from the literature [10] in the same test conditions.

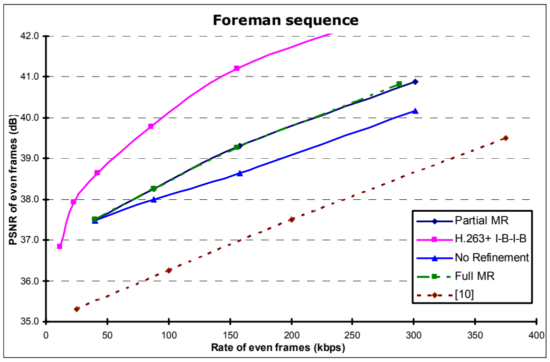

Figure 3. RD performance for the Foreman sequence.

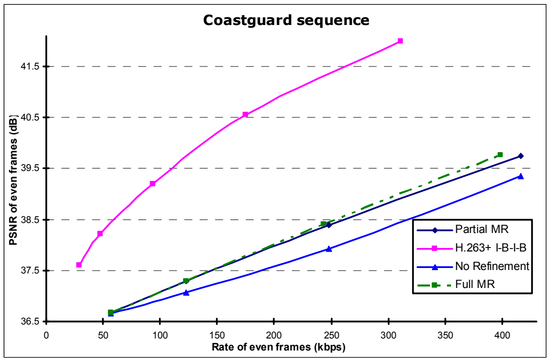

Figure 4. RD performance for the Coastguard sequence.

First, it is important to notice that the IST-PDWZ codec provides significantly better results than the best results from the literature [10], with coding improvements up to 2 dB for the Foreman sequence. It is observed from the results improvements up to 0.7 dB for Foreman just by using the partial MR configuration (compared to the "no refinement" configuration). The results for Coastguard are slightly below with improvements up to 0.4 dB. Notice that for the partial MR configuration only improvements in quality occur, since the rate remains the same as for the "no refinement" configuration; the coarse reconstructed image is improved with data from the key frames (already available) to build the final decoded image, no additional rate is needed. In the full MR configuration, the refinement loop is closed, which forces the side information to be continuously enhanced (i.e. with fewer errors) and therefore fewer parity bits should be requested from the encoder (less bitrate). In practice only minor improvements occur; for Foreman up to 15 kbps (in a total of 300 kbps) are saved for 40.7 dB decoded quality. The main reason is that the Laplacian distribution used by the turbo decoder to model the correlation between the side information and the original frame is not updated as the accuracy of the side information improves. In our experiments the parameter that characterizes the Laplacian distribution remains constant for all the decoding process which is clearly not the best solution.

5. Conclusions and future work

In this paper, a novel motion compensated refinement scheme for the IST-PDWZ codec is presented. The results obtained show that the proposed techniques improve the rate-distortion performance of the IST-PDWZ codec up to 0.7 dB without sacrificing the encoder complexity. As future work, it is planned to further enhance the RD performance of the codec with algorithms that estimate online the error distribution between the side information and the original frames. This should bring more improvements in the rate-distortion sense, especially when the side information is enhanced (the inner loop of the decoder).

6. References

[1] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockammer and T. Wedi, "Video Coding with H.264/AVC: Tools, Performance, and Complexity", IEEE Circuits and Systems, Vol. 4, No. 1, 2004.

[2] ISO/IEC International Standard 14496-10:2003, "Information Technology – Coding of Audio-visual Objects – Part 10: Advanced Video Coding".

[3] J. Slepian and J. Wolf, "Noiseless Coding of Correlated Information Sources", IEEE Trans. on Information Theory, Vol. 19, No. 4, July 1973.

[4] A. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder", IEEE Transactions on Information Theory, Vol. 22, No. 1, January 1976.

[5] A. Aaron, R. Zhang and B. Girod, "Wyner-Ziv Coding for Motion Video", Asilomar Conference on Signals, Systems and Computers, Pacific Grove, USA, November 2002.

[6] R. Puri and K. Ramchandran, "PRISM: A New Robust Video Coding Architecture Based on Distributed Compression Principles", 40th Allerton Conference on Communication, Control and Computing, Allerton, USA, October 2002.

[7] A. Sehgal, A. Jagmohan and N. Ahuja, "Wyner-Ziv Coding of Video: Applications to Error Resilience," IEEE Transactions on Multimedia, Vol. 6, No. 2, April 2004.

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

[9] S.-W. Wu and A. Gersho, "Joint Estimation of Forward and Backward Motion Vectors for Interpolative Prediction of Video", IEEE Transactions on Image Processing, Vol. 3, No. 5, September 1994.

[10] A. Aaron, S. Rane, E. Setton and B. Girod, "Transform-Domain Wyner-Ziv Codec for Video", VCIP, San Jose, USA, January 2004.