Hierarchical Motion Estimation for Side Information Creation in Wyner-Ziv Video Coding

João Ascenso
Instituto Superior de Engenharia de Lisboa – Instituto de Telecomunicações
Rua Conselheiro Emídio Navarro, 1,
1950-062 Lisboa, Portugal
+351 21 8418463
joao.ascenso@lx.it.pt

Fernando Pereira
Instituto Superior Técnico – Instituto de Telecomunicações
Av. Rovisco Pais,
1049-001 Lisboa, Portugal
+351 21 8418460
fp@lx.it.pt

ABSTRACT
Recently, several video coding solutions based on the distributed source coding paradigm have appeared in the literature. Among them, Wyner-Ziv video coding schemes enable to achieve a flexible distribution of the computational complexity between the encoder and decoder, promising to fulfill requirements of emerging applications such as visual sensor networks and wireless surveillance. To achieve a performance comparable to the predictive video coding solutions, it is necessary to increase the quality of the side information, this means the estimation of the original frame created at the decoder. In this paper, a hierarchical motion estimation (HME) technique using different scales and increasingly smaller block sizes is proposed to generate a more reliable estimation of the motion field. The HME technique is integrated in a well known motion compensated frame interpolation framework responsible for the creation of the side information in a Wyner-Ziv video decoder. The proposed technique enables to achieve improvements in the rate-distortion (RD) performance up to 7 dB when compared to H.263+ Intra and 3 dB when compared to H.264/AVC Intra.

Keywords
Wyner-Ziv video coding, side information, motion interpolation

1. INTRODUCTION
Nowadays, emerging applications such as wireless video cameras, multiview image acquisition and visual sensor networks, have different requirements than those targeted by the traditional video delivery systems. For some applications, a more flexible allocation of the global video codec complexity is necessary, e.g. low encoding complexity and battery constrained operation, which suits well the many-to-one scenario where many encoders communicate with a small number of decoders, e.g. in video surveillance networks. Also, for many applications, error robustness is a strong requirement; therefore, the video codec must gracefully handle the packet losses or bit errors, typical in many networks, and mitigate the effects of error propagation.

To fulfill the flexible complexity requirement, it is essential to have a coding configuration where the balance between the encoder and decoder complexity can be controlled, e.g. by performing more or less of the time consuming motion estimation task at the decoder. To achieve an improved robustness to channel errors, it is necessary to abandon the prediction framework used in hybrid video coding schemes, where to decode the current frame without errors, several (depending on the GOP size) previous frames need to have been received correctly. In this case, to avoid the loss of synchronization or "drift" between encoder and decoder, it is preferable to encode each frame independently (Intra) and decode each frame conditionally (Inter). Finally, all these requirements should be achieved without loss in terms of coding efficiency, i.e. with a performance close to the best available hybrid video coding schemes, in this case the H.264/AVC standard.

The techniques to obtain a reliable and accurate motion field at the decoder influence significantly the rate-distortion (RD) performance of the Wyner-Ziv (WZ) video codec in the same way as efficient motion estimation/compensation tools bring compression advances in the block based hybrid video coding. In WZ video coding, one possible solution to create the side information frame Y_i is by motion compensated frame interpolation; in this case, the motion trajectories between two (past and future) reference frames, X'_b and X'_f, are obtained by motion estimation and the motion vectors are used to interpolate the Y_i frame in between.

In order to improve the quality of Y_i, the solutions available in the literature make assumptions or restrictions about the motion pattern of the video sequence. In [1], a frame interpolation framework is presented where linear motion between the reference frames X'_b and X'_f is assumed; the techniques bidirectional motion refinement and weighted median filters are used to increase the quality of the side information frame. In [2], a modified matching criteria with a regularization term and an adaptive search range is proposed, targeting cases where high motion occurs and/or long GOP sizes are selected. In [4], a 3D recursive block matching approach attempts to find the true motion field using neighboring candidate motion vectors both spatially and temporally.

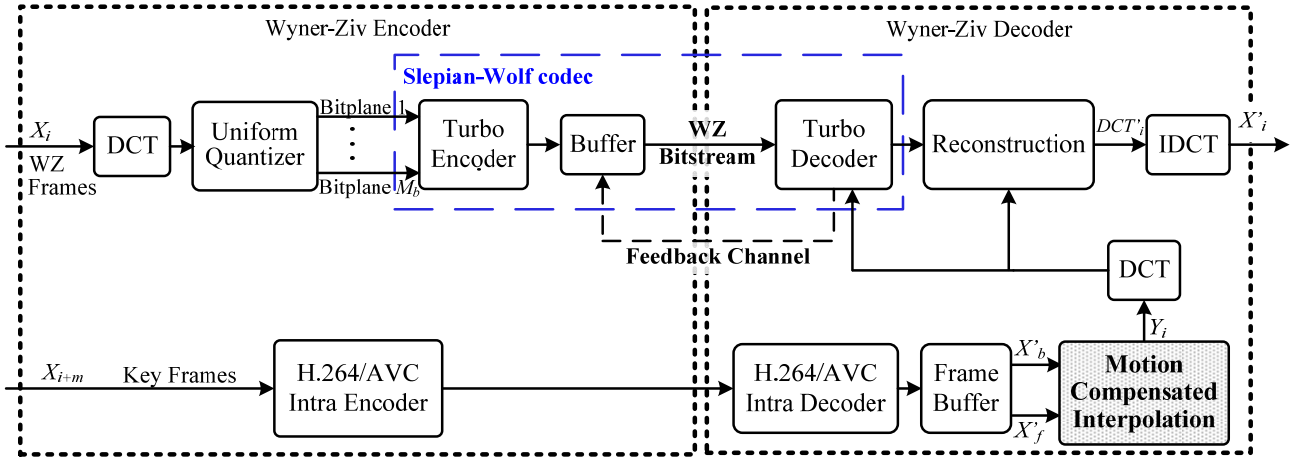

Figure 1. Wyner-Ziv video codec architecture.

All these techniques have one characteristic in common, they attempt to find a spatial coherent motion field, where discontinuities only occur at object boundaries, and the motion field expresses the true or physical motion of the sequence.

In this paper, a new hierarchical motion estimation (HME) technique is proposed; this tool is integrated in a powerful frame interpolation framework which is part of a feedback channel based Wyner-Ziv video codec. In the proposed HME technique, three scales are defined with block sizes 16×16, 8×8 and 4×4; at each scale, the motion field is represented by an affine model which is used to accurately estimate the motion between different scales. The novel frame interpolation framework allows improving the side information quality and thus the final RD performance.

This paper is organized as follows. First, in Section 2, the adopted Wyner-Ziv video codec architecture is presented. In Section 3, the HME technique is proposed and described in detail. In Section 4, the novel frame interpolation solution is described while Section 5 evaluates the RD performance of the improved WZ video codec. The conclusions and some future work topics are presented in Section 6.

2. WYNER-ZIV CODEC ARCHITECTURE

Figure 1 illustrates the Wyner-Ziv coding solution used in this paper which follows the coding architecture proposed in [1, 2, 3] by the same authors. Regarding the previous architecture, only the motion compensated interpolation (MCI) module will be changed in order to accommodate the hierarchical motion estimation technique proposed in this paper.

The coding process implies dividing the video frames into key frames and Wyner-Ziv (WZ) frames. The key frames are the first frames sent from the encoder to the decoder for each GOP and are encoded using the H.264/AVC Intra mode [5]. For each WZ frame, side information is obtained by MCI at the decoder, while the encoder has the task to send parity bits which the decoder uses to improve the decoded quality.

The WZ frame is coded by applying a H.264/AVC 4×4 block-based discrete cosine transform (DCT). Then, the DCT coefficients of the entire frame X_i are grouped together, according to the position occupied by each DCT coefficient within the 4×4 blocks, forming the so-called DCT coefficients bands. Each band is uniformly quantized and bitplanes are created and sent to the turbo encoder. The number of bitplanes generated for each band defines the amount of errors that will be corrected at the decoder, and thus the final decoded quality. The turbo encoder starts with the most significant bitplane and generates the corresponding parity bits which are stored in the buffer and transmitted in small amounts upon decoder request via the feedback channel, which is available in many applications.

At the decoder, the MCI module generates the side information Y_i, an estimate of the X_i frame, based on two references, one temporally in the past and another in the future. The Laplacian distribution is used as the correlation model between Y_i and X_i for each DCT coefficient [3]. The iterative turbo decoder uses the received parity bits and the side information to generate the decoded (with error probability $P_e = 10^{-3}$) quantized symbol stream. The side information is also used in the reconstruction module, together with the decoded quantized symbol stream, to help in the X_i reconstruction task [3].

Finally, the key frames and WZ coded frames are mixed again to generate the decoded video sequence with a quality defined by the encoding quantization process and a bitrate largely defined by the side information quality.

3. HIERARCHICAL MOTION ESTIMATION

In this paper, a HME technique with multiple scales is proposed; this technique combines two tools in a very efficient way: i) adaptive search range; and ii) affine motion model.

The hierarchical coarse-to-fine search approach tracks fast motion and reduces sensitivity to quantization noise; the latter one is very common in the reference (coded) frames at low bitrates. The procedure starts with motion estimation based on large block sizes where a coarse motion field is calculated; then the motion trajectories are refined by using a smaller block size; this process can be repeated until a enough detailed motion field is obtained. The minimum block is a 1×1 region which corresponds to a pixel and, in this particular case, a dense motion field is obtained. This

paper proposes to combine the solution above with an available adaptive search range approach [2] which restricts the motion trajectories by including information from its neighbors, in order to capture the true motion field and correct discontinuities. The combined solution performs better than a fixed search range, since it restricts the motion vector trajectory to be close to the neighboring motion vectors while still leaving room to increase its accuracy and precision.

In the HME algorithm, another important issue is the relationship between the motion vectors at a coarse grain scale with the motion vectors used as the starting point for the motion estimation (ME) at the finer scales. One possible solution is to perform a simple copy of the motion vectors from the coarser scale *k-1* to the motion vectors at the finer scale *k*, i.e. each four $L_k \times L_k$ blocks (per $L_{k-1} \times L_{k-1}$ block) will have at its center the motion vector calculated at the coarser *k-1* scale. However, a more reliable estimation can be obtained if the local motion between adjacent blocks, at the coarser *k-1* scale, is calculated and used to obtain the motion vectors at the finer scale *k*. The motion model chosen to describe the true motion is an affine motion model with a triangular region of support since it can represent well the local motion of each frame and has a low complexity. However, since the motion of each triangle is independent, it is quite possible to estimate more complex motion content globally appearing in the scene, e.g. zooms and rotations. The algorithm proposed here combines both tools and consists in the following steps:

1. Set the block size to $L_1 \times L_1$ (L_k represents block size at scale *k*) and for each block:

   1.1 Set the search range in the backward and forward reference frames based on the motion vectors of neighboring blocks according to:

$$\begin{aligned} x_U + L_k \le d_x \le x_B - L_k \\ y_L + L_k \le d_y \le y_R - L_k \end{aligned} \quad (1)$$

   Figure 2 shows the positions x_U, x_B, y_L, y_R in frame X'_f and illustrates the search range selected based on the neighboring motion vectors; the corresponding positions in frame X'_b can be calculated assuming linear motion [1].

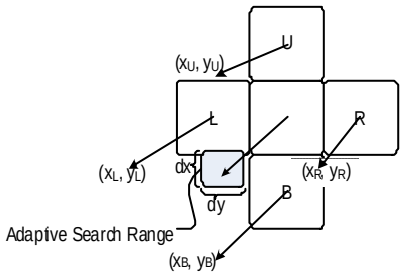

Figure 2. Search range adaptation.

   1.2. Obtain the best motion vector $(v_{i,x}, v_{i,y})$ for each i^{th} block $L_k \times L_k$ by minimizing the mean absolute difference (MAD) matching criteria.

2. Set $L_k = L_{k-1}/2$ and, for each block, compute the motion vector at the block center by using an affine motion model constructed with the motion vectors calculated at the coarser $L_{k-1} \times L_{k-1}$ scale. Figure 3 illustrates the triangularization method used; to estimate a motion vector for a position at a finer scale, e.g. v_4, a triangle with the three closest motion vectors (v_1, v_2, v_3) is constructed. Following this, the affine motion of each triangle is estimated and a motion model with the corresponding parameters is obtained; this model will be used to predict the motion vector at the finer scale, i.e. the v_4 value.

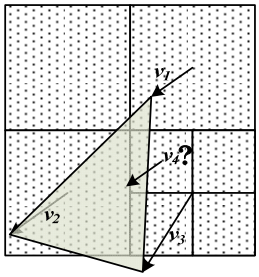

Figure 3. Triangularization method applied to construct a local affine motion model.

3. Repeat steps 1 and 2, for each block, until $L_k = L_{min}$.

For sequences at QCIF and CIF resolutions, as used in the experiments performed later, it was found experimentally that $L_1 = 16$ and $L_{min} = 4$ are good solutions.

The triangularization method proposed here consists in finding the motion of point (x', y') in the current frame to the corresponding point (x, y) in the reference frame, i.e. an unknown motion vector, considering that the motion vectors $(v_{i,x}, v_{i,y})$ are known, where *i* is the block index at the coarser level. Assume that the affine motion model describes well the motion of each point:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

where $\{a_1, \ldots, a_6\}$ are the six affine motion parameters; assuming further that the three vertices of the triangle are under the same affine motion, it is possible to calculate the corresponding parameters by using (2) to construct a six parameter affine model from the three motion vectors at the coarser scale *k-1* (see step 2 above) and obtain the following system of equations:

$$x = Ha \equiv \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix} = \begin{bmatrix} x'_1 & y'_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x'_1 & y'_1 & 1 \\ x'_2 & y'_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x'_2 & y'_2 & 1 \\ x'_3 & y'_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x'_3 & y'_3 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} \quad (3)$$

where (x'_i, y'_i) and (x_i, y_i) correspond to points in the current and reference frames from which the motion vectors $(v_{i,x}, v_{i,y})$ are made. Solving (3) for *a*, it is obtained:

$$a = H^{-1}x \quad (4)$$

Therefore, to obtain a motion vector for block *j* at a finer scale, step 2 as described above corresponds to the following process:

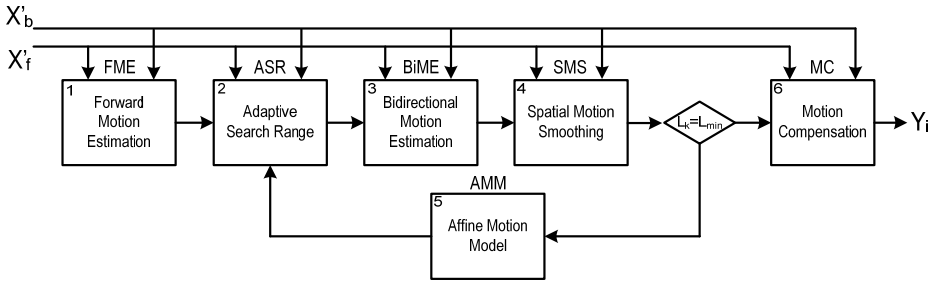

Figure 4. Architecture of the motion compensated interpolation framework.

1. Matrix H_j is constructed (as defined in (3)) using the three neighboring motion vectors at the coarser scale, with the corresponding points in both current and reference frames.

2. j^{th} block affine parameters $\{a_1, \ldots, a_6\}$ are calculated using (4).

3. The motion vector at the finer scale can, finally, be calculated with (3).

However, to guarantee the coherence of the motion vector field (i.e. piecewise smoothness), it is necessary to limit the application of this method when the motion is badly behaved or when it is not well described by an affine model. In this case, a simple metric based on the area of each triangle is used, but more complex techniques based on the characteristics (e.g. variance) of the residual can also help.

4. MOTION COMPENSATED INTERPOLATION FRAMEWORK

The HME technique proposed in the previous section allows to improve the quality of the side information, increasing the correlation between the side information and the frame to be encoded (this means Y_i will be more similar to X_i). This has a positive impact in terms of RD performance, since the decoder will request fewer parity bits from the encoder to achieve a certain target quality. To take full advantage of the HME technique, it is proposed to include it in the block-based MCI framework proposed in [2]; the overall structure is shown in Figure 4. This framework corresponds now to the MCI module of the Wyner-Ziv video codec shown in Figure 1.

The MCI framework generates the side information Y_i, an estimate of the X_i frame, based on two references, one temporally in the past (X'_b) and another in the future (X'_f).

In the Forward Motion Estimation (FME) module, the reference frames are first low pass filtered to improve the reliability of the motion vectors. Then a block matching algorithm is used to estimate the motion between the X'_b and X'_f frames. A full-search motion estimation with modified matching criteria [2] is performed which includes a regularized term favoring motion vectors closer to the origin.

The next algorithm, Adaptive Search Range (ASR), calculates the search range in an adaptive way as described in Section 3. However, to apply this technique it is necessary to guarantee that each non-overlapping Y_i block has a motion vector; thus, for each Y_i block the motion vector that intersects the Y_i block closer to its center is selected from all the motion vectors calculated in the previous step.

In the Bidirectional Motion Estimation (BiME) module, the motion vectors obtained in the previous step are refined with a bidirectional motion estimation algorithm which assumes a linear trajectory [1] between reference frames. Thus, the motion vectors calculated by FME and AMM are refined according to the search range calculated by ASR (in most cases confined to a small displacement).

The calculated motion vectors are then filtered out with the spatial motion smoothing algorithm [1] in the Spatial Motion Smoothing (SMS) module. In the case the motion field is not detailed enough, i.e. $L_k \neq L_{min}$, the motion vectors at the finer scale are calculated using the affine motion model of Section 3 (equations (2) and (4)). Then, the algorithms ASR/BiME/SMS (which correspond to modules 2 to 4 defined in Figure 4) are repeated. Three iterations with $L_k = \{16, 8, 4\}$ are performed and $L_{min} = 4$. Finally, when $L_k = L_{min}$, the final motion field is obtained and the side information can be constructed by motion compensation.

In all the techniques proposed, the motion vector precision is always integer-pel; it is expected that fractional-pel precision (especially half-pel) could further increase the RD performance, although not significantly [6].

5. EXPERIMENTAL RESULTS

The HME technique was evaluated under the following test conditions: all frames of the *Coastguard*, *Hall Monitor* test sequences with QCIF@15Hz and GOP length of 2 (corresponds to the most common conditions used in the literature). For the MCI, ±32 pixels are used for the search range of the forward motion estimation; both references are first low pass filtered with a 3×3 size mean filter. The key frames are always encoded with H.264/AVC Intra in the Main profile [5] since this is one of the best performing Intra coding schemes; the H.264/AVC reference software JM9.6 with rate-distortion optimization on and all Intra modes enabled has been used. The quantization parameters (QP) for the key frames and the quantization matrices (QI) for the WZ frames define the quality of the key frames and the WZ frames, respectively [3]. Each combination of QI and QP is understood as a RD point; the QIs have been chosen in such a way that the average quality (PSNR) of the WZ frames is similar to the quality of the key frames. All rate and distortion results refer only to the luminance.

To evaluate the coding efficiency of the proposed HME technique, several experiments have been performed. First, the

motion compensated interpolation framework is evaluated as a standalone tool, i.e. when it is not integrated in the Wyner-Ziv video decoder. This experiment intends to assess the quality of the side information, which has a major influence on the overall RD performance of a Wyner-Ziv video codec.

Table 1 shows the average PSNR results (in dB) for the side information (motion compensated interpolated frame) computed over the whole sequence, for all the HME scales and all the algorithms of the MCI framework. In this experiment, it was also included results for the *Foreman* and *Soccer* sequences under the same test conditions (QCIF@15Hz). The contribution of each module/algorithm identified in Figure 4 is evaluated in a cumulative way. The evaluated algorithms must improve the quality of the motion field, and therefore ASR is merged with BiME, since the adaptive search range must be used in combination with a motion estimation technique to increase effectiveness. As it can be noticed from Table 1, one of the most effective tools in the MCI framework is the HME algorithm, e.g. compare step 4 with 16×16 block sizes with step 4 with 4×4 blocks, since it allows refining the motion field with smaller block sizes, which is essential to obtain a good interpolation.

Table 1. Side information quality (in dB) for each step in the MCI framework (RD Point Q_8)

Sequences (QCIF@15Hz)	L_k = 16 (16×16)			L_k = 8 (8×8)		L_k = 4 (4×4)	
	1	2-3	4	2-3	4	2-3	4
Coastguard	30.52	30.82	30.49	31.08	31.09	31.40	31.59
Hall Monitor	35.49	35.49	35.26	35.76	35.43	35.84	35.83
Foreman	28.50	28.61	28.77	28.99	28.98	29.09	29.14
Soccer	21.40	21.53	22.02	22.15	22.25	22.16	22.26

Next, the Wyner-Ziv video codec described in Section 2 with the MCI framework proposed in Section 4 is compared against the conventional H.264/AVC Intra and H.263+ Intra low complexity encoders; notice that the Wyner-Ziv video codec has an encoding complexity which is lower than the one for these standard solutions [7]. The Intra encoders chosen do not exploit temporal correlation; however, H.264/AVC Intra exploits quite efficiently the spatial correlation with several 4×4 and 8×8 Intra modes, a feature that is missing in the proposed WZ video codec. The Wyner-Ziv video codec is also compared with H.264/AVC zero-motion (IBI GOP structure), which has a lower encoding complexity compared to the full H.264/AVC Inter codec since it uses the collocated blocks in the previous and/or future reference frames for prediction in a DPCM type mode. The Intra mode is also used in B-frames since the Intra/Inter mode decision is activated.

Figure 5 and Figure 6 show the RD performance for the *Coastguard* and *Hall Monitor* sequences. There are significant gains for all RD points in both sequences with average gains up to 7 dB (for the *Hall Monitor* sequence) when compared to H.263+ Intra; for H.264/AVC Intra, the performance gains are between 0.5 dB (high bitrates for *Coastguard* sequence) to 3 dB (low bitrates for *Hall Monitor*). Therefore, it is possible to conclude that the adopted WZ codec and the proposed side information extraction framework can effectively exploit the temporal correlation in an efficient way. When compared to the H.264/AVC zero-motion codec, the proposed Wyner-Ziv video codec remains competitive for the *Coastguard* sequence where some gains (up to 1 dB) are observed; however, some RD loss is still observed for the *Hall Monitor* sequence, especially for higher bitrates. This may be partly explained by the H.264/AVC SKIP macroblock mode where no information other than the mode is coded thus bringing additional benefits when coding this low motion sequence with H.264/AVC zero-motion. For the sequences *Foreman* and *Soccer*, not shown here due to space limitations, the RD results are always above H.263+ Intra, under the same test conditions. However, for the *Soccer* sequence, a RD loss is observed when compared to H.264/AVC Intra, because the complex and erratic motion content leads to worse side information.

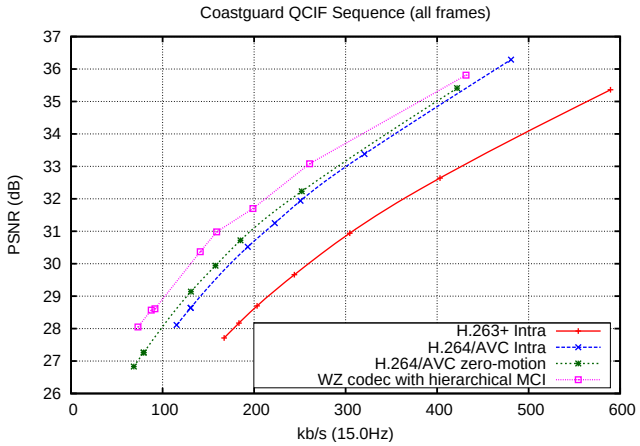

Figure 5. RD Performance for the *Coastguard* QCIF sequence.

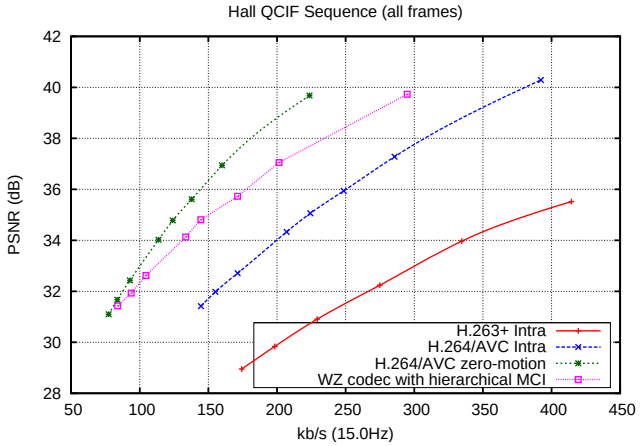

Figure 5. RD Performance for the *Hall Monitor* QCIF sequence.

6. FINAL REMARKS

In this paper, a HME technique is proposed which makes use of local affine motion models to represent increasingly more detailed motion fields. At each scale of the HME technique, corresponding to a certain block size, affine motion parameters are calculated; after, a reliable estimate of the motion field at a finer scale is produced. This technique was integrated in the context of an advanced side information creation framework and its RD performance was evaluated in the context of a transform domain turbo based video codec. The improved solution reaches promising results when compared to conventional "low encoding

complexity" standard coding schemes, such as H.264/AVC Intra and H.264/AVC zero-motion. As future work, the development of a block based Intra mode and adequate Intra/WZ mode decision algorithms for the WZ frames could lead to some coding efficiency improvements. This technique can provide an alternative estimation for the regions where the motion compensated interpolation fails more often, i.e. covered/uncovered regions, complex motion trajectories, illumination changes or severe quantization noise.

7. ACKNOWLEDGMENTS

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

8. REFERENCES

[1] Ascenso, J., Brites, C., and Pereira, F. 2005. 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* (Smolenice, Slovak Republic, June 29 – July 02, 2005).

[2] Ascenso, J., Brites, C., and Pereira, F. 2006. Content Adaptive Wyner-Ziv Video Coding Driven by Motion Activity. *IEEE International Conference on Image Processing* (Atlanta, USA, October 08 – 11, 2006).

[3] Brites, C., Ascenso, J., and Pereira, F. 2006. Improving Transform Domain Wyner-Ziv Video Coding Performance. *IEEE International Conf. on Acoustics, Speech and Signal Processing* (Toulouse, France, May 14 – 19, 2006).

[4] Chien, W.-J., Karam, L. J., and Abousleman, G. P. 2006. Distributed video coding with 3D recursive search block matching. *IEEE International Symposium on Circuits and Systems* (Island of Kos, Greece, May 21 – 24, 2006).

[5] Information Technology, Coding of Audio-visual objects, Part 10: Advanced Video Coding, ISO/IEC Std. 14496-10, 2003.

[6] Klomp, S., Vatis, Y., and Ostermann, J. 2006. Side Information Interpolation with Sub-pel Motion Compensation for Wyner-Ziv decoder. *International Conference on Signal Processing and Multimedia Applications* (Setúbal, Portugal, August 7 – 10, 2006).

[7] "The DISCOVER codec evaluation", http://amalia.img.lx.it.pt/~discover/home.html.

Filename:	ICUIMC'08_final_final.doc
Directory:	\\fernao\ICUIMC08
Template:	C:\Users\jascenso\AppData\Roaming\Microsoft\Templates\Normal.dotm
Title:	Proceedings Template - WORD
Subject:	
Author:	End User Computing Services
Keywords:	
Comments:	Edited by G. Murray on Aug. 23rd. 2007 for 'ACM Reference Format' / updated reference examples.
Creation Date:	1/15/2008 8:03:00 PM
Change Number:	17
Last Saved On:	1/17/2008 4:17:00 PM
Last Saved By:	João Miguel Duarte Ascenso
Total Editing Time:	24 Minutes
Last Printed On:	1/17/2008 10:39:00 PM
As of Last Complete Printing	
Number of Pages:	6
Number of Words:	3,814 (approx.)
Number of Characters:	21,743 (approx.)