Fusion of Global and Local Motion Estimation for Distributed Video Coding

Abdalbassir ABOU-ELAILAH, *Student Member, IEEE,* Frederic DUFAUX, *Senior Member, IEEE,*
Joumana FARAH, *Member, IEEE,* Marco CAGNAZZO, *Senior Member, IEEE,*
and Beatrice PESQUET-POPESCU, *Senior Member, IEEE,*

***Abstract*—The quality of side information plays a key role in distributed video coding. In this paper, we propose a new approach that consists in combining global and local motion compensation at the decoder side. The parameters of the global motion are estimated at the encoder using Scale Invariant Feature Transform (SIFT) features. Those estimated parameters are sent to the decoder in order to generate a globally motion compensated side information. Conversely, a locally motion compensated side information is generated at the decoder based on motion-compensated temporal interpolation of neighboring reference frames. Moreover, an improved fusion of global and local side information during the decoding process is achieved using the partially decoded Wyner-Ziv frame and decoded reference frames. The proposed technique improves significantly the quality of the side information, especially for sequences containing high global motion. Experimental results show that, as far as the rate-distortion performance is concerned, the proposed approach can achieve a PSNR improvement of up to 1.9 dB for a GOP size of 2 and up to 4.65 dB for larger GOP sizes, with respect to the reference DISCOVER codec.**

***Index Terms*—Distributed Video Coding, Wyner-Ziv Coding, Side Information Refinement, Global Motion, Local Motion, Rate-Distortion Performance.**

I. Introduction

In video coding standards like ISO/IEC MPEG and ITU-T H.26x, motion estimation and compensation are performed at the encoder in order to achieve high rate-distortion performance, while the decoder can directly use the motion vectors to decode the sequence. This architecture makes the encoder much more complex than the decoder [1]. This asymmetry in complexity is well-suited for applications where the video sequence is encoded once and decoded many times, such as broadcasting or video-on-demand streaming systems. However, some recent applications such as wireless video surveillance, multimedia sensor networks, wireless PC cameras, and mobile cameras phones require a low complexity encoding, while possibly affording a high complexity decoding.

A. ABOU-ELAILAH, F. DUFAUX, M. CAGNAZZO, and B. PESQUET-POPESCU are with the Signal and Image Processing Department, Institut T*él*ecom - TELECOM Paristech, 46 rue Barrault, F - 75634 Paris Cedex 13, FRANCE, e-mail: ({elailah, frederic.dufaux, marco.cagnazzo, beatrice.pesquet }@telecom-paristech.fr).

J. FARAH is with the Department of Telecommunications Engineering, Faculty of Engineering, Holy-Spirit University of Kaslik, P.O. Box 446, Jounieh, Lebanon, e-mail: joumanafarah@usek.edu.lb.

Manuscript received Month X, 2011; revised Month X, 2011.

Distributed Video Coding (DVC) is a new paradigm in video communication, which fits well these scenarios since it enables the exploitation of the similarities among successive frames at the decoder side, making the encoder less complex. Thus, the task of motion estimation and compensation is shifted to the decoder. From information theory, the Slepian-Wolf theorem for lossless compression [2] states that it is possible to encode correlated sources (let us call them X and Y) independently and decode them jointly, while achieving the same rate bounds which can be attained in the case of joint encoding and decoding. The Wyner-Ziv (WZ) theorem [3] extends the Slepian-Wolf one to the case of lossy compression of X when Side Information (SI) Y is available at the decoder.

Based on these theoretical results, practical implementations of DVC have been proposed [4], [5]. The European project DISCOVER [6], [7] came up with one of the most efficient and popular existing architectures. More specifically, it is based on transform domain WZ coding. The images of the sequence are split into two sets of frames, key frames (KFs) and Wyner-Ziv frames (WZFs). The Group of Pictures (GOP) of size n is defined as a set of frames consisting of one KF and $n-1$ WZFs. The KFs are independently encoded and decoded using Intra coding techniques such as H.264/AVC Intra mode or JPEG2000. The WZFs are separately transformed and quantized, and a systematic channel code is applied to the resulting coefficients. Only the parity bits are kept, and sent to the decoder upon request. This can be seen as a Slepian-Wolf coder applied to the quantized transform coefficients. At the decoder, the reconstructed reference frames are used to compute the Side Information (SI), which is an estimation of the WZF being decoded. In order to produce the SI, DISCOVER uses Motion-Compensated Temporal Interpolation (MCTI) [8]. Finally, a channel decoder uses the parity information to correct the SI, thus reconstructing the WZF. Straightforwardly, generating a more accurate SI is very important, since it would result in a reduced amount of parity information requested by the decoder through the return channel. At the same time, the quality of the decoded WZF would be improved during reconstruction.

In this paper, we propose a new method for enhancing the SI in transform-domain DVC. This solution consists in combining global and local SI at the decoder. The global motion parameters are computed at the encoder, while keeping a low encoding complexity. For a given WZF, feature points of the original reference frames and of the original WZF are extracted by carrying out the Scale-Invariant Feature Transform (SIFT) [9]

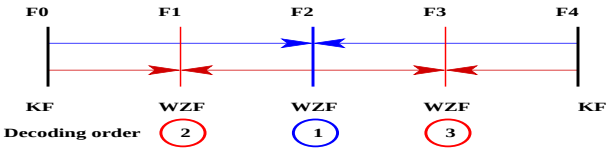


Fig. 1. Interpolation steps for a GOP size 4.

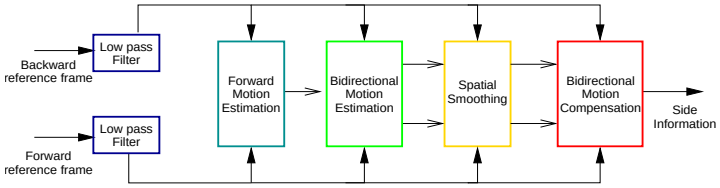


Fig. 2. MCTI technique [8].

algorithm. Then, a matching between these feature points is applied. Next, we need to find the matches which belong to the global motion in the scene. We propose an efficient algorithm which consists in eliminating iteratively the false matches due to local motion, in order to estimate the parameters of a global motion model between the current WZF and the backward or forward reference frame. The parameters of the global model are sent to the decoder in order to generate a SI based on Global Motion Compensation (GMC), and referred to as GMC SI. On the other hand, another SI is estimated using the MCTI technique (local estimation) with spatial motion smoothing, exactly as in DISCOVER codec [6][7]. This SI will be called MCTI SI. Then, a fusion of GMC SI and MCTI SI is performed; it will be referred to as the First Fusion SI (FFSI).

In addition, we also propose to successively improve the fusion of GMC SI and MCTI SI, after the decoding of each DCT band. Starting with the FFSI, the decoder reconstructs a Partially Decoded Wyner-Ziv Frame (PDWZF) by correcting the FFSI with the parity bits of the first DCT band. In this technique, two variations are proposed to enhance the fusion. The first one consists in improving the fusion after decoding the first DCT band, using the decoded DC coefficients of the PDWZF. It is important to note here that this method is very efficient in terms of computational load. The second method consists in improving the FFSI using the PDWZF along with the backward and forward reference frames. This method consists in re-estimating the false motion vectors obtained by the MCTI technique, similarly to [10], after the decoding of each DCT band. Finally, the fusion between GMC SI and MCTI SI is iterated after each improvement of the PDWZF.

This paper is structured as follows. First, the related work is introduced in Section 2. The process that generates the GMC SI frame using the global motion and the fusion technique of MCTI SI and GMC SI frames are depicted in Section 3. Moreover, the improvement of the fusion using two different approaches is illustrated in the same Section 3. Experimental results are shown in Section 4 in order to evaluate and compare the RD performance of the proposed approaches. Finally, conclusions are drawn in Section 5.

II. Related work

A. DISCOVER Architecture

In this section, we briefly present the DISCOVER codec [6], [7]. First, the input video sequence is divided into WZFs and KFs. The latter are encoded using H.264/AVC Intra coding. Figure 1 shows all the necessary interpolations for a GOP of size 4. For example, during the interpolation of WZF F2, the forward and backward reference frames are the KFs F0 and F4. For the interpolation of F1, the reference frames are the KF F0 and the previously decoded WZF F2. This hierarchical interpolation order has been shown to be optimal for a GOP of size 4 [11]. The WZF encoding and decoding procedures are detailed in the following.

- **Wyner-Ziv encoder -** At the encoder side, the WZF is first transformed using a 4×4 integer Discrete Cosine Transform (DCT). The integer DCT coefficients of the whole WZF are then organized into 16 bands. The DC coefficients are placed in the first band, and the other coefficients are grouped in the AC bands.
  Next, each integer DCT coefficient is uniformly quantized. The quantization step depends only on the band. The resulting quantized symbols are then split into bit planes. For a given band, the bits of the same significance are grouped together in order to form the corresponding bit plane, which is then independently encoded using a rate-compatible Low-Density Parity Check Accumulate (LDPCA) code. The parity information is then stored in a buffer and progressively sent (upon request) to the decoder, while the systematic bits are discarded.
- **Generation of side information -**
  In the DISCOVER scheme, the MCTI technique is used to generate the SI [8] at the decoder side. Figure 2 shows the architecture of the MCTI technique. The frame interpolation framework is composed of four modules to obtain high quality SI as follows: Both reference frames are first low-pass filtered in order to improve the motion vector reliability, followed by forward motion estimation between the backward and forward reference frames, bi-directional motion estimation to refine the motion vectors, spatial smoothing of motion vectors in order to achieve higher motion field spatial coherence, and finally bi-directional motion compensation.
- **Wyner-Ziv decoder -** A block-based 4×4 integer DCT is carried out over the generated SI in order to obtain the integer DCT coefficients, which can be seen as a noisy version of the WZF DCT coefficients. Then, the LDPCA decoder corrects the bit errors in the DCT transformed SI, using the parity bits of WZF requested from the encoder through the feedback channel. To decide whether more parity bits are needed for the successful decoding, the convergence is tested by computing the syndrome check error.
- **Reconstruction and inverse transform -** The reconstruction corresponds to the inverse of the quantization using the SI DCT coefficients and the decoded Wyner-Ziv DCT coefficients. Let i be the decoded quantization

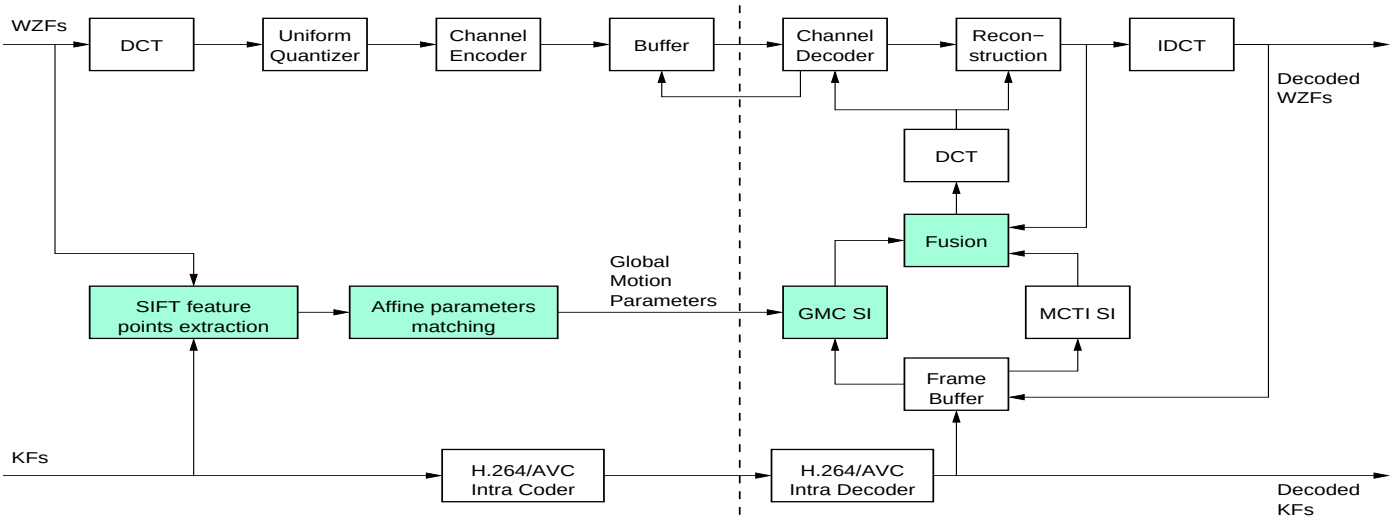


Fig. 3. Overall structure of the proposed DVC codec.

index and y the SI DCT coefficient. The reconstruction step [12] consists in computing the expectation $\hat{x} = \mathbb{E}[x|x \in B_i, y]$, where B_i is the quantization interval corresponding to the index i. After that, the inverse 4×4 integer DCT transform is carried out, and the entire frame is restored in the pixel domain.

B. Improved Side Information Generation

The goal in terms of compression is to achieve a coding efficiency similar to the best available hybrid video coding schemes. However, DVC has not reached the performance level of classical inter frame coding yet. This is in part due to the quality of the SI, which has a strong impact on the final Rate-Distortion (RD) performance.

Several works have been proposed in order to enhance the SI. A solution proposed by Ascenso *et al.* [13] for pixel domain DVC uses a motion compensated refinement of the SI successively after each decoded bit plane, in order to achieve a better reconstruction of the decoded WZF. In [14], a novel DVC successive refinement approach is proposed to improve the motion compensation accuracy and the SI. This approach is based on the N-Queen sub-sampling pattern.

In VISNET II codec [15], the refinement process of the SI is carried out after decoding all DCT bands in order to improve reconstruction [16]. In [10][17], approaches are proposed for transform-domain DVC based on the successive refinement of the SI after each decoded DCT band. In [18], a solution is proposed based on the successive refinement of the SI using adaptive search area for long duration GOPs in transform-domain DVC. High-order motion interpolation has been proposed [19] in order to cope with object motion with non-zero acceleration. In [20], global motion is estimated at the decoder in order to adapt temporal inter-/extrapolation for SI generation.

Commonly, the SI is generated by applying the MCTI technique on consecutive reference frames and already reconstructed WZFs. The quality of the SI is poor in certain regions of the video scene, like in areas of partial occlusions, fast motion, etc. In this case, a hash information may be transmitted to the decoder in order to improve the SI. However, the encoder needs to determine in advance the regions where the interpolation at the decoder would fail. In [21][22], hash information is extracted from the WZF being encoded and sent only for the macroblocks where the sum of squared differences between the previous reference frame and the WZF is greater than a certain threshold.

In [23], the authors proposed a Witsenhausen-Wyner Video Coding (WWVC) that employs forward motion estimation at the encoder and sends the motion vectors to the decoder to generate the SI. This WWVC scheme achieves better performance than H.264/AVC in noisy networks and suffers a limited loss (up to 0.5 dB compared to H.264/AVC) in noiseless channel. The authors in [24] proposed a novel framework that integrates the graph-based segmentation and matching to generate interview SI in Distributed Multiview Video Coding.

In [25][26][27], the authors presented DVC schemes that consist in performing the motion estimation both at the encoder and decoder. In [25], the authors propose a pixel-domain DVC scheme, which consists in combining low complexity bit plane motion estimation at the encoder side, with motion compensated frame interpolation at the decoder side. Improvements are shown for sequences containing fast and complex motion. The authors in [26] present a DVC scheme where the task of motion estimation is performed both at the encoder and decoder. Results have shown that the cooperation of the encoder and decoder can reduce the overall computational complexity while improving coding efficiency. Finally, a DVC scheme proposed by Dufaux *et al.* [27] consists in combining the global and local motion estimations at the encoder. In this scheme, the motion estimation and compensation are performed both at the encoder and decoder.

On the contrary, in this paper, the local motion estimation is only performed in the decoder, while the global motion parameters are estimated in the encoder using a SIFT algorithm. It is important to note that the encoding complexity is kept low. The global parameters are sent to the decoder to estimate

the GMC SI and the combination between the GMC SI and MCTI SI is made at the decoder side.

III. Proposed System

The block diagram of our proposed codec architecture is depicted in Figure 3. It is based on the DISCOVER codec [6][7]. The shaded (green) blocks correspond to the four new modules introduced in this paper: SIFT feature points extraction, affine parameters matching, computation of GMC SI, and fusion of GMC SI and MCTI SI.

At the encoder, global motion parameters are estimated between the current original WZF and the original reference frames. First, SIFT feature points are extracted from the original reference and WZ frames. Second, global motion parameters are derived from matched feature points. The technique is described in Subsection A.

At the decoder, the MCTI SI generation is based on block matching using the decoded reference frames, while the GMC SI is estimated by applying the global parameters on the decoded reference frames. Afterwards, the fusion of the two SI is carried out in order to obtain the FFSI. The fusion step is described in Subsection B.

Two techniques are then proposed to further improve performance. The improvement of the fusion using the decoded DC coefficients is described in Subsection C. Finally, the refinement of the MCTI SI and the fusion during the decoding process is described in Subsection D.

A. Global Motion Estimation and Compensation

Figure 4 shows the block diagram of the proposed GMC technique. At the encoder side, we extract the feature points of the two consecutive original reference frames (forward and backward reference frames) and the feature points of the original WZF. These feature points are extracted by applying the SIFT algorithm [9]. Once the feature points are extracted, we apply the matching between the feature points of the WZF and the backward (and forward) reference frame in order to estimate the global motion parameters.

In this paper, several global motion models are analyzed in order to choose the most suited one for our proposed method. Three parametric models are considered: translational motion model (two parameters), affine motion model (six parameters), and perspective motion model (eight parameters). The perspective motion model is defined as follows:

$$\begin{cases} u_i &= (a_0 + a_2 x_i + a_3 y_i)/(a_6 x_i + a_7 y_i + 1) \\ v_i &= (a_1 + a_4 x_i + a_5 y_i)/(a_6 x_i + a_7 y_i + 1) \end{cases}$$

where $(a_0, a_1, ..., a_7)$ are the motion parameters, $(x_i,\ y_i)$ denotes the pixel location in the WZF, and $(u_i,\ v_i)$ the corresponding position in the backward or forward reference frame. The affine ($a_6 = a_7 = 0$) and the translation ($a_2 = a_5 = 1, a_3 = a_4 = a_6 = a_7 = 0$) models are particular cases of the perspective model.

Afterwards, we carry out an efficient algorithm on these feature matches that estimates the parameters of the model between the WZF and the backward reference frame. This algorithm allows us to remove the false matches, *i.e.*, the matches that exist on individual objects of the scene and correspond to local motion. The motion parameters between the WZF and the forward reference frame are estimated in the same way.

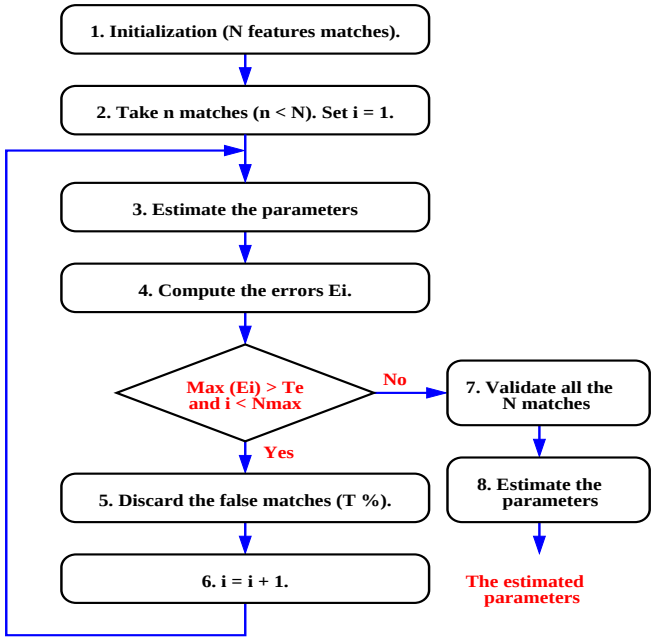


Fig. 5. Flowchart diagram of the proposed global model parameters estimation.

The motion parameters are estimated by minimizing:

$$E = \sum_{i=1}^{N} f(E_i) \tag{1}$$

with

$$f(E_i) = \begin{cases} E_i & \text{if } E_i < T \\ 0 & \text{otherwise} \end{cases}$$

where E_i represents the error of feature match number i, and N represents the number of the feature matches between the two frames. In order to increase the robustness to false feature matches, a threshold T is defined according to a fixed percentage, in order to take into account only the most accurate feature matches. The error of feature match number i is defined as:

$$E_i = \sum_{i=1}^{N} (u_i - r_i)^2 + (v_i - s_i)^2. \tag{2}$$

where

$$\begin{cases} r_i &= a_{0e} + a_{2e} x_i + a_{3e} y_i / (a_{6e} x_i + a_{7e} y_i + 1) \\ s_i &= a_{1e} + a_{4e} x_i + a_{5e} y_i / (a_{6e} x_i + a_{7e} y_i + 1) \end{cases}$$

(r_i, s_i) are the coordinates in the backward or forward reference frame, corresponding to the feature point (x_i, y_i) in the WZF, according to the actual estimated parameters $(a_{0e}, a_{1e}, ..., a_{7e})$.

The flowchart diagram of the proposed algorithm for the estimation of the global model parameters is depicted in Figure 5. The two transforms T_b and T_f are the motion models between the original WZF and the backward and forward original reference frames, respectively. As shown in

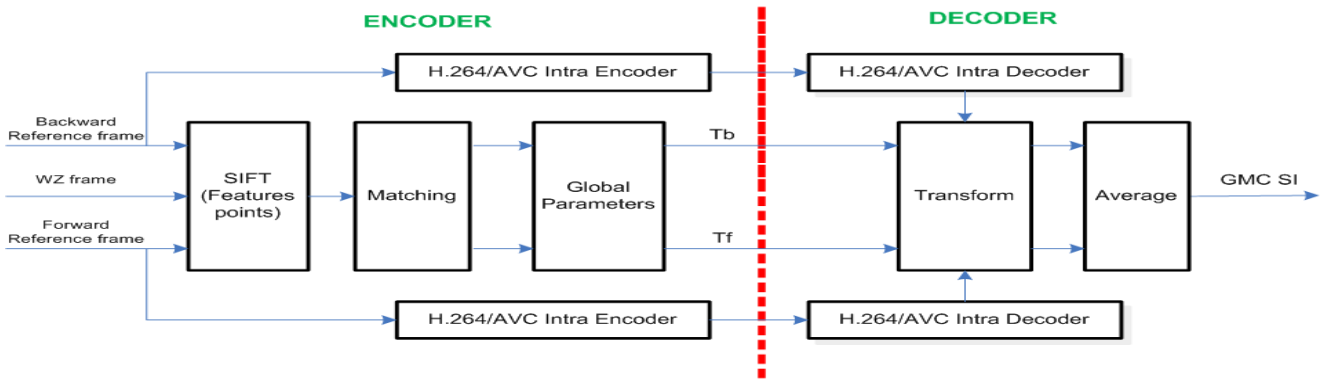


Fig. 4. Block diagram of the proposed GMC technique.

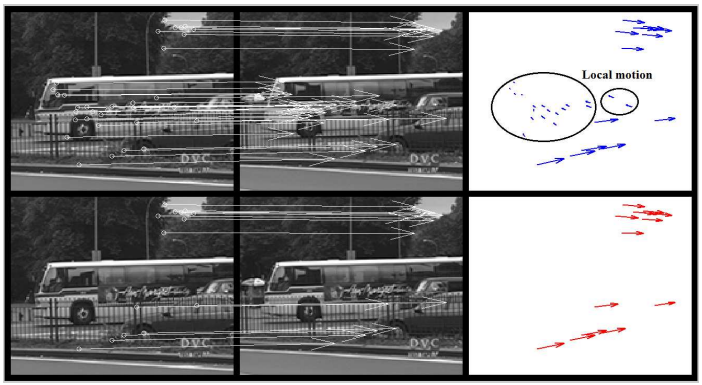


Fig. 6. The obtained feature matches between frames 17 and 21 of Bus sequence before (blue, top) and after (red, bottom) applying the proposed algorithm.

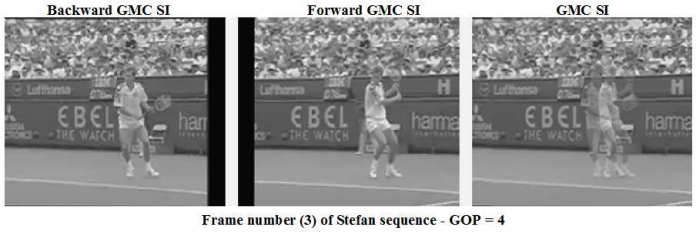


Fig. 7. SI generated by GMC.

the diagram, the algorithm consists of the following steps to estimate the parameters of the two transforms T_b and T_f:

Step 1 - N feature matches are obtained between the original WZ and the reference original (backward and forward) frame. Typically, a large number of matches are found. However, in the unlikely case where no matches are found (e.g. in the case of shot cut), the global motion estimation procedure is stopped and only MCTI SI is used.

Step 2 - Commonly, the moving objects appear in the center of the frame. In order to increase the probability of the feature matches belonging to the global motion compared to the local motion, the proposed algorithm takes the feature points that belong to the top and bottom quarters of the frame (n feature matches are taken, $n < N$). This step allows a quick and accurate convergence of the algorithm.

Step 3 - The parameters of the model T_b, respectively T_f, are estimated by minimizing the Euclidean distance taking the n feature matches, *i.e.*, between the feature points in the WZF and the corresponding feature points in the backward or forward reference frame.

Step 4 - The error of each match E_i (n matches) is computed according to Equation (2). If the maximum error E_{max} ($E_{max} = max(E_i)$) is greater than a threshold T_e, go to **Step 5**. Otherwise, go to **Step 7**.

Steps 5 and 6 - The feature matches which give the largest errors (the top T% of the distribution E_i) are discarded, and the rest of the feature matches are taken for the next iteration ($i = i + 1$).

Step 7 - The feature matches of the entire frame (N feature matches) are fed into the estimated model to identify the valid feature matches. The feature match that gives an error greater than T_e is considered to be as false match (belongs to the local motion) and discarded.

Step 8 - Finally, the algorithm computes once again the parameters of the model T_b, respectively T_f, by taking into account only the valid feature matches (belonging to the global motion) of the entire frame. In this algorithm, at most N_{max} iterations are carried out. In most cases, the algorithm converges rapidly before the N_{max} iterations. We have empirically chosen $N_{max} = 5$ and $T_e = 1$ in our simulations.

Figure 6 shows the feature matches between the frames no. 17 and 21 of Bus sequence. The top frames represent the feature matches (blue) obtained by applying the method in [9]. The bottom frames represent the feature matches (red) obtained by carrying out our algorithm. It is clear that the proposed technique discards all the feature matches corresponding to local motion.

The parameters of these transforms are computed at the encoder. Finally, these estimated parameters (4 in case of a translational model, 12 in case of an affine model or 16 for the perspective model) are sent to the decoder for each WZF.

At the decoder side, the parameters of T_b and T_f are

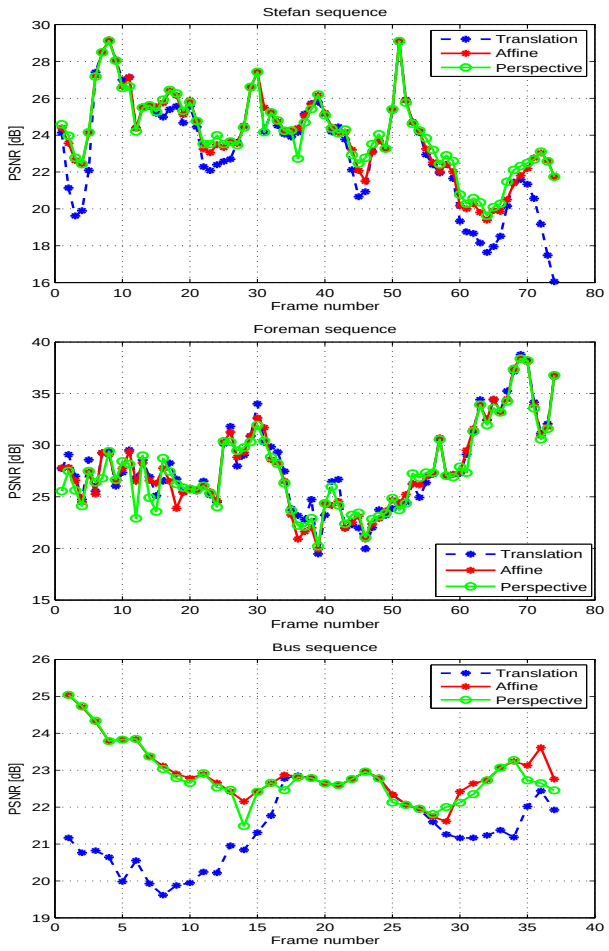


Fig. 8. PSNR of GMC SI for Stefan, Foreman, and Bus sequences, for various global motion models.

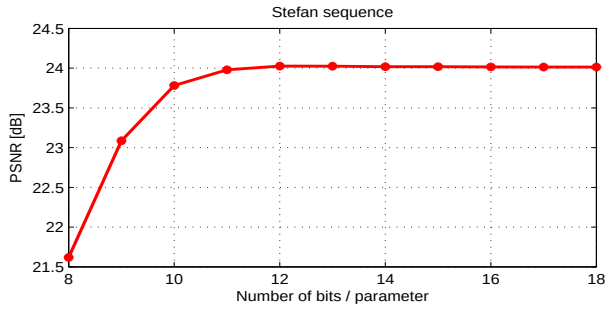


Fig. 9. Average PSNR of the GMC SI frames in terms of number of bits per parameter for Stefan sequence, for a GOP size of 2.

respectively applied to the backward and forward decoded reference frames in order to estimate the GMC SI. Similarly to MCTI SI, the GMC SI is obtained by averaging both backward and forward predictions. Figure 7 shows an example of computation of a GMC SI; the left image represents the backward GMC SI, where T_b is applied to the backward reference frame, and the central image represents the forward GMC SI, where T_f is applied to the forward reference frame. Finally, the average between the pixels of the backward and the forward GMC SI frames is computed to generate the GMC SI, and it is shown on the right. However, when the pixels are black (on the border of the image due to the shift) in the backward GMC SI frame, only the pixels of the forward GMC SI frame are taken for the GMC SI, and vice versa.

The experimental determination of the quality of GMC SI, estimated for various global motion models, is shown in Figure 8 for Stefan, Bus, and Foreman sequences (QCIF, at 15 Hz) for all frames. As it can be seen from the obtained results, the translation model allows a small gain in the Foreman sequence, but it generally fails when the global motion becomes more complicated. On the other hand, the perspective model is less robust in the case of noisy matches. Therefore, the affine model will be adopted for the rest of this paper.

For the affine parameters, in this paper, we encode each parameter on 15 bits as follows: First, a_2 and a_5 represent the scale parameters, a_3 and a_4 represent the shear parameters and the parameters a_0 and a_1 represent the translation vector between the two frames. In a video sequence, the amount of scaling and shearing between successive frames remains typically small, whereas the translation vector may be large. Figure 9 represents the average PSNR of the GMC SI frames in terms of number of bits per parameter for Stefan sequence, for a GOP size of 2. The quality of the GMC SI becomes stationary after 12 bits per parameter.

Specifically, the parameters a_2 and a_5 can be written as $1+s\times f$, where s is the sign and f is a positive floating number ($f<1$). We encode s and f on 1 bit and 14 bits respectively. The parameters a_3 and a_4 can be written as $s\times f$, where s is the sign of the number and f is a positive floating number ($f<1$). We encode s and f on 1 bit and 14 bits respectively. For the translation parameters a_0 and a_1, the maximum translation between two frames is considered to be ± 128 pixels. Thus, these parameters can be written as $s\times(n+f)$, where s, n, and f represent the sign of the number, an integer number ($n<128$) and a positive floating number ($f<1$) respectively. Then, s is encoded on 1 bit, n and f are encoded on 7 bits respectively.

For the case of a video at QCIF resolution and 15 Hz with a GOP size of 2, the supplementary data burden will be only 180 bits (15 bits/parameter) per WZF (1.35 kbps). Thus, the resulting bitrate overhead to transmit the global parameters is negligible.

B. Fusion of MCTI and GMC SI

The current section deals with the fusion between MCTI SI and GMC SI, both generated at the decoder, as described in the previous sections. The block size adopted for the fusion step is 4×4 pixels. Figure 10 shows the combination of the global and local motion estimations. For a given block B in the current SI (Figure 10), the following steps are carried out:

Step 1 - The SAD is computed between the corresponding blocks B_{lb} and B_{lf} in the backward and forward reference frames, these blocks being determined by the MCTI technique.

$$SAD_{MCTI} = |B_{lb} - B_{lf}| \tag{3}$$

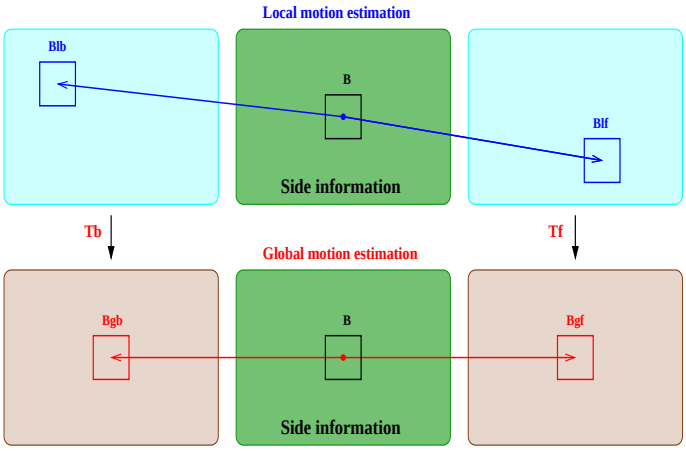


Fig. 10. Fusion of global and local motion estimations.

Step 2 - The global transforms T_b and T_f are applied to the backward and forward reference frames respectively. The corresponding blocks to the current block B are now directly B_{gb} and B_{gf} in the same position of the current block. Then, the SAD between B_{gb} and B_{gf} is computed.

$$SAD_{GMC} = |B_{gb} - B_{gf}| \tag{4}$$

Step 3 - Finally, in order to combine the global and local motion estimations, the corresponding blocks which give the smallest SAD (SAD_{MCTI} or SAD_{GMC}) are taken for the FFSI (from MCTI SI or GMC SI). At the border of the image, if the pixels of the block B_{gb} or B_{gf} are black due to the shift resulting from the application of the global transforms, the average between MCTI SI and GMC SI is computed to generate the fusion of these blocks (in this case, the pixels in the GMC SI is only estimated from B_{gb} if the block B_{gf} is black and vice versa).

The error distribution between the corresponding DCT bands of FFSI and WZFs is necessary for the Slepian-Wolf decoder, in order to correct the errors in the DCT FFSI coefficients. However, the original WZFs are not available at the receiver. Furthermore, an offline process for determining this distribution is not realistic, since it requires either the encoder to recreate the SI or to have the original data available at the decoder. In [28], the correlation noise is estimated online at the decoder, using the residual frame between the backward and forward motion compensated reference frames as a confidence measure for the frame interpolation operation. In this paper, this approach is adopted for the MCTI SI. For GMC SI, the difference between the transformed decoded reference frames (by applying the transforms T_b and T_f) is computed to create the residual frame for the correlation noise. Finally, the correlation noise for FFSI is estimated by combining the two residual frames in the same manner as in Figure 10. In other words, the two residual frames are combined according to the fusion scheme of MCTI SI and GMC SI.

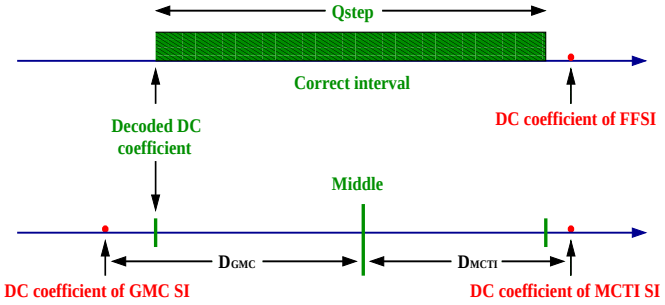


Fig. 11. Improving fusion by using the decoded DC coefficients.

In the following, two different approaches are introduced to improve the fusion of GMC SI and MCTI SI during the decoding process. The first one consists in improving the fusion using the decoded DC coefficients. The second one consists in refining the MCTI SI during the decoding of the DCT bands, and, at the same time, successively improving the fusion between the two SI frames using the PDWZF.

C. Improving the fusion by using decoded DC coefficients

Once the decoded DC coefficients are obtained after decoding the first DCT band, the proposed approach which consists in combining the global and local motion estimations is improved using these decoded DC coefficients (this method will be referred to as DCcoefs). Then, the improved SI is used to decode the remaining DCT coefficients, *i.e.*, the AC coefficients. This improved technique is motivated by the fact that the enhancement of the SI significantly reduces the amount of requested parity bits through the feedback channel, as well as the decoder processing time.

Recall that the WZF is transformed using a 4×4 block-based integer DCT. The DC coefficients are quantized using a quantization step Qstep. In order to improve the fusion, for each block in the current WZF, the decoded DC coefficient is compared to the DC coefficient of the FFSI (Fusion of MCTI SI and GMC SI).

For the current block in the FFSI, let DD_{DC} be the decoded quantization DC coefficient. We refer to the quantization interval which corresponds to DD_{DC} by the term 'correct interval', as shown in Figure 11. Let 'Middle' be the center of the correct interval and FFSI_{DC} the DC coefficient of the FFSI transformed using a 4×4 block-based integer DCT. The FFSI enhancement technique is described by several steps as follows:

Step 1 - If FFSI_{DC} is within the correct interval of the decoded DC coefficient, the fusion for this block can be considered to be accurate, and this block is not changed. Otherwise, go to **Step 2**.

Step 2 - The distance between the DC coefficient of MCTI SI and the Middle is computed (this distance is referred to as D_{MCTI}), as well as the distance between the DC coefficient of GMC SI and the Middle (referred to as D_{GMC}).

Step 3 - The smallest distance between D_{MCTI} and D_{GMC} is chosen to determine the best candidate for

the new SI, except if the difference between these distances is smaller than the half of Qstep. In this case, the average of the two blocks (from MCTI SI and GMC SI) is computed for the new SI.

In summary, this method is described as follows:

if $\mathrm{DD}_{DC} < \mathrm{FFSI}_{DC} < \mathrm{DD}_{DC} + Qstep$
- The fusion for this block is considered to be reliable

otherwise
if $|\mathrm{D}_{MCTI} - \mathrm{D}_{GMC}| < Qstep/2$
- The fusion for this block is considered to be the average of the two blocks (MCTI SI and GMC SI)

otherwise
- The fusion for this block is considered to be the block which is closer to Middle (MCTI SI or GMC SI)

D. Refining MCTI SI and fusion by using the PDWZF

The motion vectors estimated by the MCTI technique for certain blocks can be erroneous, especially in sequences containing high motion. For this reason, we aim at re-estimating suspect vectors by integrating the algorithm that we formerly proposed in [10], due to its high performance. This algorithm is applied after the decoding of each DCT band. Furthermore, the fusion between the global and local motion estimations is carried out after each improvement of the local motion estimation using the PDWZF (this method will be referred to as RefMCTI).

This algorithm consists in re-estimating the vectors suspected of being false. In order to identify these vectors, a threshold T_1 is used. For a given block (8×8 pixels), the Mean of Absolute Differences (MAD) between the PDWZF and the MCTI SI is calculated and compared to T_1 as follows:

$$\mathrm{MAD}(\text{MCTI SI}, \mathrm{PDWZF}(\mathbf{MV})) < T_1, \qquad (5)$$

where $\mathbf{MV} = (MV_x, MV_y)$ is the candidate motion vector. Even though the block size is 8×8 pixels, an extended block of 12×12 pixels is considered when the MAD is computed.

If Eq. (5) is not satisfied, the motion vector is identified as a suspicious vector and will be re-estimated. Otherwise (Eq. (5) is satisfied), the motion vector $\mathbf{MV}$ for this block is only refined twice within a small search area; the first time, after the decoding of the first DCT band and the second time after the decoding of all DCT bands. This step consists in relaxing the symmetric bidirectional motion vectors constrained in MCTI and allows a small refinement of those estimated motion vectors. In the simulations of this paper, we have set $T_1 = 6$ after preliminary tests, in such a way to achieve high performance with a low computational load.

The refinement of MCTI SI is applied during the decoding process by using this algorithm after decoding each DCT band. It starts by a first decoding of the FFSI frame (*i.e.* the SI obtained after the first fusion of MCTI SI and GMC SI) using the parity bits of the first DCT band. The reconstructed PDWZF is then used for refinement, together with the backward and forward reference frames. After each refinement

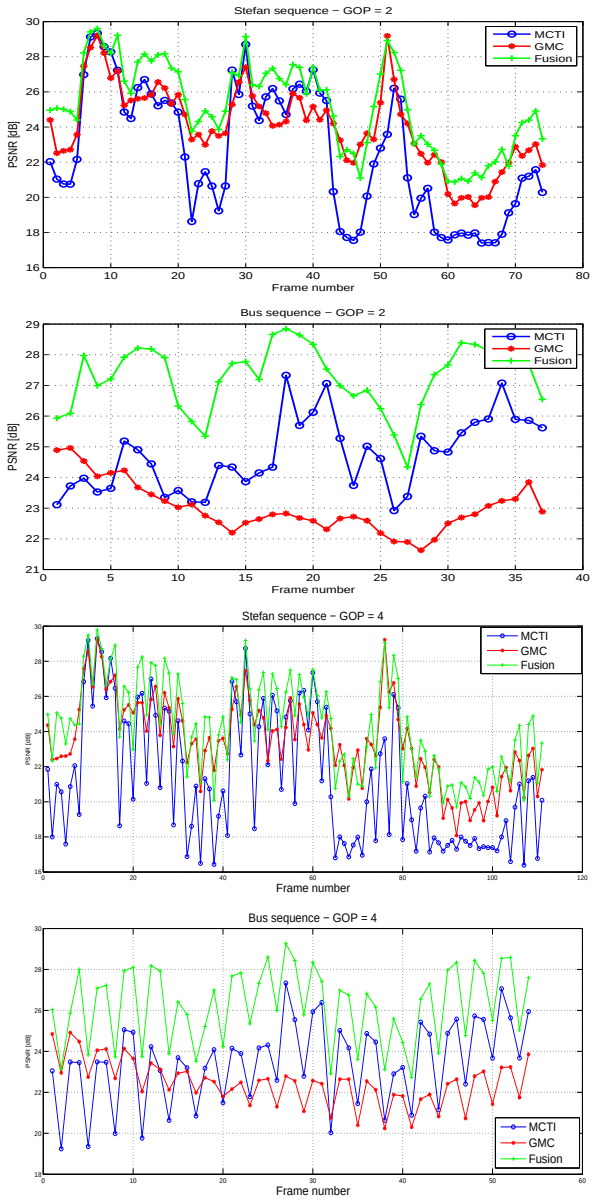


Fig. 12. PSNR of MCTI SI, GMC SI, and the fusion of MCTI SI and GMC SI (FFSI) for Stefan and Bus sequences for a GOP size of 2 and 4.

step, the fusion of MCTI SI and GMC SI is applied using the PDWZF: For each block in the actual SI (4×4 pixels), the SAD between the PDWZF and MCTI SI (or GMC SI) is computed using a window of 8×8 pixels as follows:

$$\begin{aligned}&\mathrm{SAD}(\alpha\mathrm{SI}, \mathrm{PDWZF}) = \textstyle\sum_{i=-4}^{3} \sum_{j=-4}^{3} \\ &|\alpha\mathrm{SI}(i + x_0, j + y_0) - \mathrm{PDWZF}(i + x_0, j + y_0)|\end{aligned} \qquad (6)$$

where αSI is the MCTI SI or GMC SI, and (x_0, y_0) is the coordinate of the center pixel of the current block. The fusion consists in choosing the most similar block in MCTI SI or GMC SI to the current block in PDWZF. In other words, the block which gives the smallest SAD is chosen for the next SI.

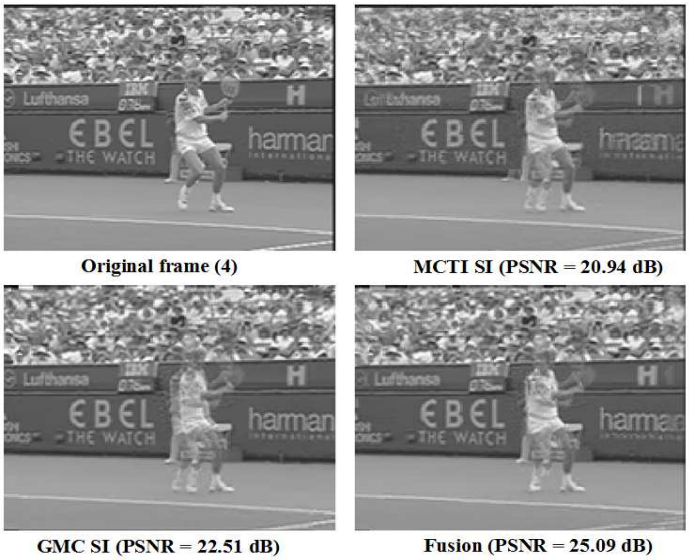


Fig. 13. MCTI SI (top-right) - GMC SI (bottom-left) - Fusion of MCTI SI and GMC SI (bottom-right) - Frame number 4 of Stefan sequence.

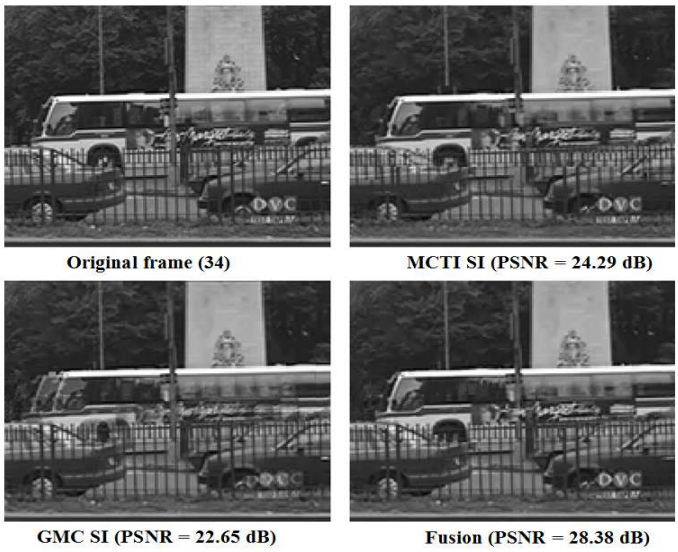


Fig. 14. MCTI SI (top-right) - GMC SI (bottom-left) - Fusion of MCTI SI and GMC SI (bottom-right) - Frame number 31 of Bus sequence.

IV. Experimental Results

In order to evaluate the performance of the proposed methods, we performed extensive simulations, adopting the same test conditions as described in DISCOVER [6], [7], *i.e.* test video sequences are at QCIF spatial resolution and sampled at 15 frames/sec. The obtained results are compared to the DISCOVER codec, the reference results [10], the H.264/AVC Intra (Main profile), H.264/AVC No motion (*i.e.* all motion vectors are zero), and H.264/AVC with Inter prediction and motion estimation in Main profile exploiting temporal redundancy in a IB...IB... structure.

A. SI performance assessment

Figure 12 shows the SI PSNR for Stefan and Bus sequences, for a GOP size of 2 and 4. For Stefan sequence, the quality

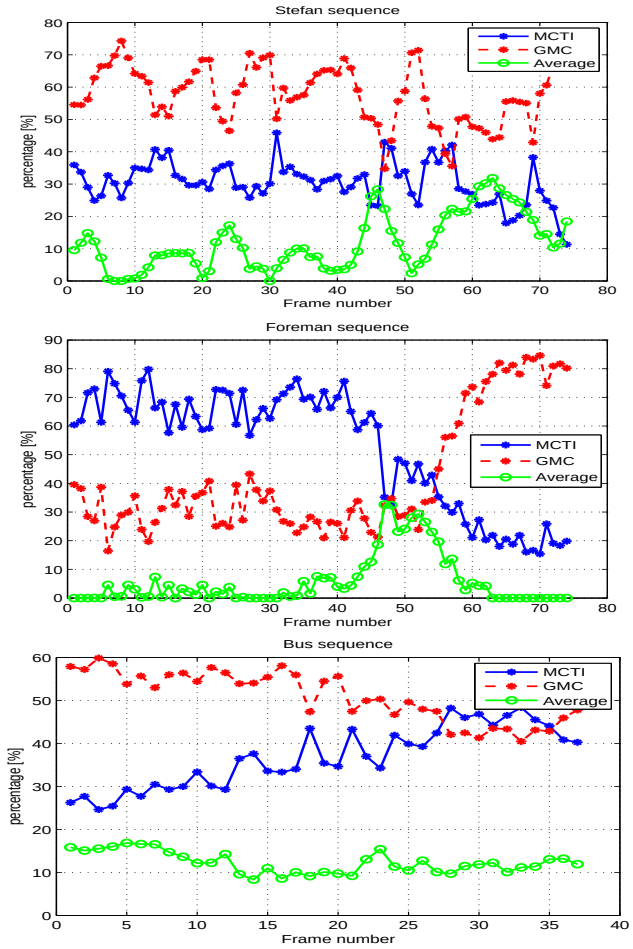


Fig. 15. Percentage of blocks in FFSI from MCTI, GMC, and the average of MCTI and GMC for Stefan, Foreman, and Bus sequences.

of the GMC SI is better than the MCTI SI in most cases. However, for Bus sequence, the MCTI SI is better than the GMC SI most of the time. It is clear that the fusion of global and local motion estimations (FFSI before any refinement) achieves the best quality SI almost for all frames in the two sequences.

Figure 13 shows the visual quality of the SI for Stefan (frame number 4). The SI obtained by DISCOVER codec (MCTI) contains block artifacts (top-right - 20.94 dB). On the contrary, the SI obtained by the GMC technique is free from these artifacts (bottom-left - 22.51 dB). The improvement of the SI obtained with our proposed method (bottom-right - 25.88 dB) by combining the global and local motion estimations is up to 4 dB better compared to MCTI. Figure 14 shows the visual quality of the SI for Bus (frame number 34). In this case, the SI obtained by MCTI technique is better than the SI obtained by the GMC technique. However, the fusion of global and local motion estimations can achieve a gain up to 4 dB compared to MCTI for this frame.

Figure 15 shows the percentage of blocks that are taken from the MCTI SI, the GMC SI, and both the MCTI SI and GMC SI (average) during the fusion of global and local motion estimations (FFSI before any refinement) for Stefan, Foreman,

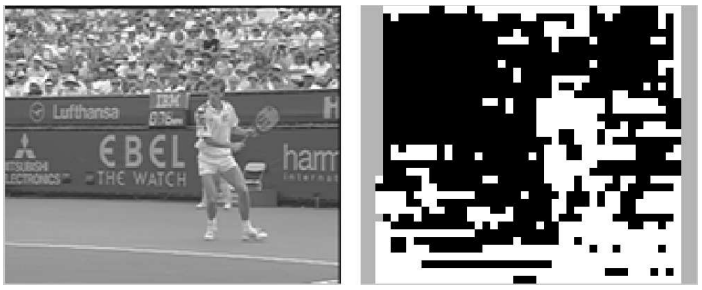


Fig. 16. Frame number 2 of Stefan sequence and the different regions of FFSI. The white region represents the blocks which are taken from MCTI SI, the black region represents the blocks taken from GMC SI, and the gray region represents the blocks taken from both the MCTI SI and GMC SI (average).

and Bus sequences. The average between the MCTI SI and GMC SI is only applied when the block in the GMC SI is taken from one side (from the backward or forward reference frame), e.g. when this block is black in the backward (or forward) GMC SI due to camera motion. The percentage of MCTI SI and GMC SI in the generated FFSI depends on the sequence. It is clear that the percentage of the average between MCTI SI and GMC SI increases with the amount of camera motion in the sequence.

Figure 16 shows the original frame and the regions of the SI which are taken from the MCTI SI (white) and GMC SI (black), for the second frame of Stefan sequence. The gray color represents the blocks where the average between the MCTI SI and GMC SI is computed. It is clear that most of the background blocks are taken from GMC SI (global motion), and that object blocks are taken from the MCTI SI (local motion).

B. Rate-Distortion performance

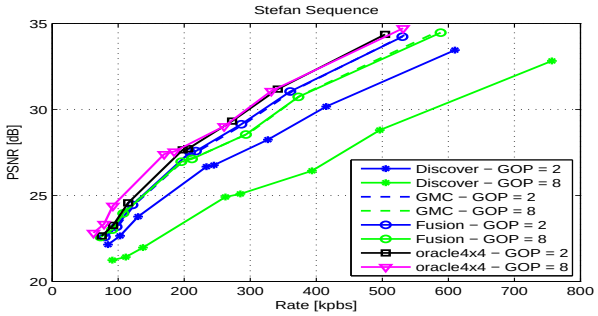
In this section, we show the RD performance for two cases. The first case corresponds to the combination of the global and local motion estimations only once before running the decoding process. The second one consists in improving the fusion during the decoding process using either the decoded DC coefficients or the PDWZF.

1) ***RD performance for the first fusion of global and local motion estimation****:* The RD performance of the proposed method is shown for the Stefan, Bus, Foreman, and Coastguard sequences in Table I, in comparison to the DISCOVER codec, using the Bjontegaard metric [29] for different GOP sizes (2, 4 and 8). The first column represents the performance of the GMC scheme, *i.e.*, the SI is only generated using the global motion estimation, and the second column represents the performance of the proposed method. The last column represents the performance of the Oracle fusion which consists in combining the global and local motion estimations based on the original WZF. The Oracle performance is shown as an upper bound limit in order to assess the efficiency of the proposed fusion method.

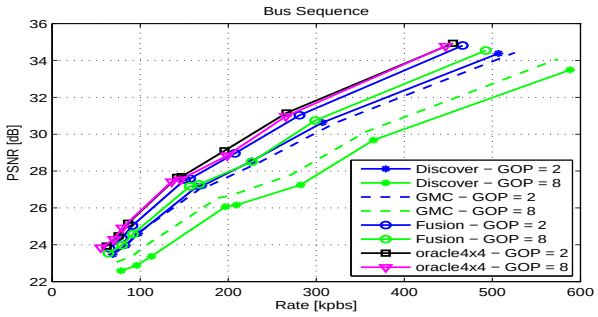
For Stefan sequence (the corresponding curves are shown in Figure 17 for GOP sizes equal to 2 and 8), the proposed method (fusion of global and local motion) can respectively

TABLE I
RATE-DISTORTION PERFORMANCE GAIN FOR *Stefan*, *Bus*, *Foreman*, AND *Coastguard* SEQUENCES TOWARDS DISCOVER CODEC, USING BJONTEGAARD METRIC

	Stefan			**Bus**		
	GMC	Fusion	Oracle	GMC	Fusion	Oracle
	GOP size = 2					
Δ_R (%)	-21.52	**-22.58**	-28.37	4.19	**-14.72**	-21.02
Δ_{PSNR} [dB]	1.47	**1.53**	2.01	-0.2	**0.9**	1.3
	GOP size = 4					
Δ_R (%)	-40.34	**-40.54**	-49.60	-4.96	**-28.69**	-38.82
Δ_{PSNR} [dB]	2.9	**2.87**	3.82	0.27	**1.78**	2.61
	GOP size = 8					
Δ_R (%)	-48.50	**-48.51**	-58.79	-13.65	**-37.15**	-48.81
Δ_{PSNR} [dB]	3.66	**3.61**	4.87	0.74	**2.38**	3.47
	Foreman			**Coastguard**		
	GOP size = 2					
Δ_R (%)	0.11	**-6.91**	-17.85	18.95	**-0.8**	-11.09
Δ_{PSNR} [dB]	-0.01	**0.4**	1.13	-0.89	**0.05**	0.58
	GOP size = 4					
Δ_R (%)	-11.07	**-13.97**	-35.56	20.67	**-9.87**	-29.16
Δ_{PSNR} [dB]	0.62	**0.79**	2.33	-0.86	**0.41**	1.33
	GOP size = 8					
Δ_R (%)	-22.18	**-20.15**	-46.52	6.88	**-22.44**	-45.06
Δ_{PSNR} [dB]	1.24	**1.13**	3.17	-0.35	**0.92**	2.19



(a) RD performance for Stefan sequence with GOP = 2, and 8.



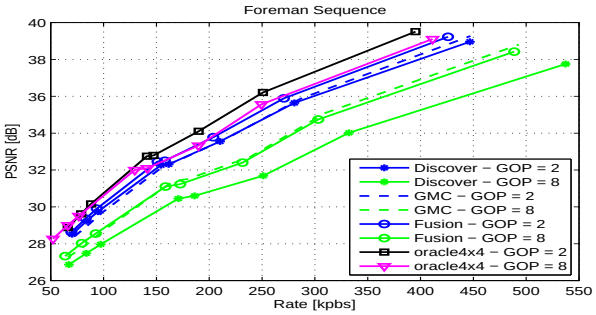
(b) RD performance for Bus sequence with GOP = 2, and 8.

Fig. 17. RD performance comparison - DISCOVER (MCTI), GMC, Proposed (fusion MCTI - GMC), and Oracle for Stefan and Bus sequences.

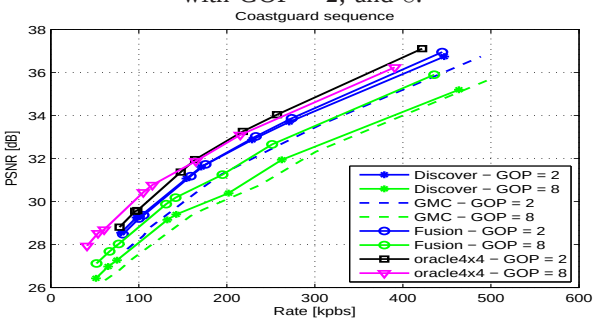
achieve a gain up to 1.53, 2.87, and 3.61 dB with a rate reduction of 22.58, 40.54, and 48.51 %, compared to DISCOVER codec, for GOP sizes of 2, 4, and 8.

For Bus sequence (the curves are shown in Figure 17), the fusion of MCTI and GMC allows respectively a gain up to 0.9, 1.78, and 2.38 dB with a reduction in the rate up to 14.72, 28.69, and 37.15 %, compared to DISCOVER codec for GOP sizes of 2, 4, and 8.

For Foreman and Coastguard sequences (the corresponding curves are shown in Figure 18), the fusion of MCTI and GMC always allows a gain with respect to the DISCOVER codec



(a) RD performance for Foreman sequence with GOP = 2, and 8.



(b) RD performance for Coastguard sequence with GOP = 2, and 8.

Fig. 18. RD performance comparison - DISCOVER (MCTI), GMC, Proposed (fusion MCTI - GMC), and Oracle for Foreman and Coastguard sequences.

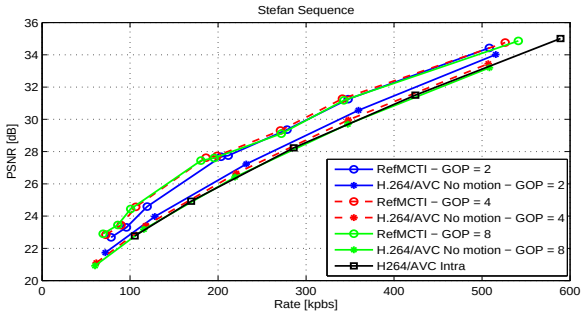
(see Table I). For these sequences, the difference between the proposed fusion and the Oracle fusion is high. Overall, it is clear that the performance of the proposed method is better than both the GMC and MCTI techniques applied separately.

For Soccer sequence, the fusion of MCTI SI and GMC SI (FFSI) does not allow a gain compared to MCTI SI, due to the fact that global motion estimation does not improve the prediction quality.

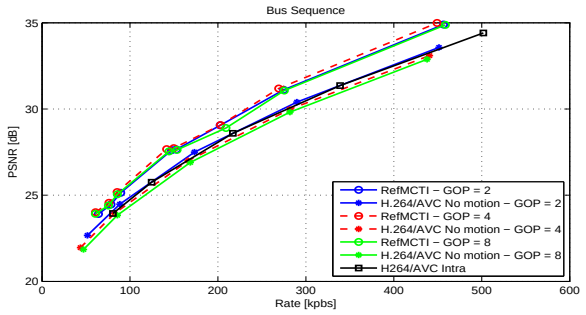
In the next section, we will show that the SI and the proposed fusion can be improved during the decoding process, in such a way to further enhance the performance of the proposed method.

2) ***RD performance for the proposed techniques for fusion improvement:*** In this work, the fusion is improved using the DC coefficients of the PDWZF (DCcoefs) on the one hand. On the other hand, the MCTI SI is refined after decoding each DCT band using the PDWZF and the decoded reference frames [10]. Moreover, the fusion between the MCTI SI and the GMC SI is done after each improvement of the MCTI SI (RefMCTI). The RD performance of the proposed methods is shown in Table II for Stefan, Bus, Foreman, Coastguard sequences, with GOP sizes of 2, 4, and 8.

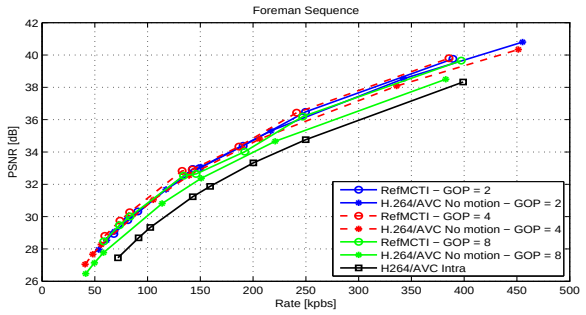
The first proposed method (DCcoefs) can achieve a gain up to 2.26 dB, with a rate reduction up to 39.92 % for Foreman sequence, for a GOP size of 8. On the other side, the first fusion achieves a gain up to 1.13 dB with a rate reduction up to 20.15 %. Thus, the DCcoefs method can improve the fusion by using the DC coefficients of the PDWZF, especially when the gap between the first fusion and the Oracle fusion is high



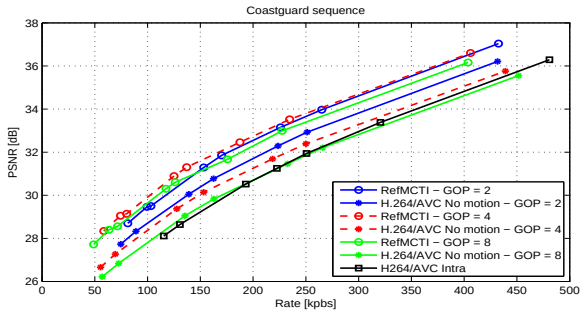
(a) RD performance for Stefan sequence for all GOP sizes.



(b) RD performance for Bus sequence for all GOP sizes.



(c) RD performance for Foreman sequence for all GOP sizes.



(d) RD performance for Coastguard sequence for all GOP sizes.

Fig. 19. RD performance comparison - Proposed (RefMCTI), H.264/AVC Intra, and H.264/AVC No motion for Stefan, Bus, Foreman and Coastguard sequences.

(refer to the results of Foreman and Coastguard sequences in Table I). Moreover, the DCcoefs method is very light in terms of computational load.

The second proposed method (RefMCTI) can achieve a significant gain compared to DISCOVER codec and [10], for all sequences, with different GOP sizes. The gain reaches

TABLE II
RATE-DISTORTION PERFORMANCE GAIN FOR *Stefan*, *Bus*, *Foreman*, AND *Coastguard* SEQUENCES TOWARDS DISCOVER CODEC, USING BJONTEGAARD METRIC

	Stefan						
	Ref. [10]	Fusion	DCcoefs	RefMCTI	H.264 Intra	H.264 No motion	H.264 Inter
	GOP size = 2						
Δ_R (%)	-15.4	-22.58	-23.52	**-27.03**	-10.44	-16.20	**-41.82**
Δ_{PSNR} [dB]	1	1.53	1.61	**1.92**	0.57	1.03	**3.28**
	GOP size = 4						
Δ_R (%)	-30.4	-40.54	-42.61	**-48.06**	-32.62	-33.70	**-62.03**
Δ_{PSNR} [dB]	1.98	2.87	3.08	**3.67**	2.15	2.38	**5.24**
	GOP size = 8						
Δ_R (%)	-37.96	-48.51	-51.23	**-57.26**	-45.36	-43.95	**-66.39**
Δ_{PSNR} [dB]	2.54	3.61	3.93	**4.65**	3.22	3.34	**5.63**
	Bus						
	GOP size = 2						
Δ_R (%)	-7.26	-14.72	-15.49	**-17.93**	-6.19	-8.44	**-37.94**
Δ_{PSNR} [dB]	0.41	0.9	0.96	**1.11**	0.18	0.52	**2.48**
	GOP size = 4						
Δ_R (%)	-20.3	-28.69	-31.11	**-34.72**	-24.95	-23.1	**-57.98**
Δ_{PSNR} [dB]	1.13	1.78	1.97	**2.3**	1.18	1.55	**4.12**
	GOP size = 8						
Δ_R (%)	-29.25	-37.15	-39.81	**-46.75**	-42.04	-38.1	**-60.48**
Δ_{PSNR} [dB]	1.66	2.38	2.63	**3.25**	2.33	2.64	**4.19**
	Foreman						
	GOP size = 2						
Δ_R (%)	-18	-6.91	-12.55	**-21.47**	-1.12	-22.43	**-36.99**
Δ_{PSNR} [dB]	1.08	0.4	0.75	**1.37**	-0.12	1.32	**2.24**
	GOP size = 4						
Δ_R (%)	-35.96	-13.97	-27.23	**-41.48**	-22.31	-37.88	**-61.72**
Δ_{PSNR} [dB]	2.21	0.79	1.63	**2.79**	1.15	2.37	**4.29**
	GOP size = 8						
Δ_R (%)	-47.6	-20.15	-39.92	**-53.20**	-38.35	-47.25	**-72.09**
Δ_{PSNR} [dB]	3.04	1.13	2.26	**3.76**	2.28	3.1	**5.5**
	Coastguard						
	GOP size = 2						
Δ_R (%)	-2.21	-0.8	-4.33	**-6.14**	30.1	9.77	**-17.09**
Δ_{PSNR} [dB]	0.11	0.05	0.22	**0.32**	-1.4	-0.49	**0.89**
	GOP size = 4						
Δ_R (%)	-13.56	-9.87	-16.46	**-20.85**	31.67	9.71	**-39.49**
Δ_{PSNR} [dB]	0.4	0.41	0.7	**0.94**	-0.86	-0.54	**1.88**
	GOP size = 8						
Δ_R (%)	-32.5	-22.44	-30.86	**-37.59**	-11.51	-9.21	**-57.23**
Δ_{PSNR} [dB]	1.07	0.92	1.33	**1.74**	0.45	0.19	**3**

4.65 dB with a rate reduction of 57.26 %, when the method in [10] achieves a gain up to 2.54 dB with a rate reduction of 37.96 % for Stefan sequence, for a GOP size 8. For Foreman sequence, the RefMCTI method can achieve a gain up to 3.76 dB with a rate reduction up to 53.2 %, when the method in [10] achieves a gain up to 3.04 dB with a rate reduction of 47.6 % for a GOP size 8. It can be seen that the RefMCTI method allows an important performance improvement compared to the first fusion of global and local motion estimation, especially for Foreman and Coastguard sequences.

Figure 19 shows the performance of the proposed method (RefMCTI) compared to that of H.264/AVC Intra and H.264/AVC No motion, for Stefan, Bus, Foreman, and Coastguard sequences. The performance of the proposed method is always better than both H.264/AVC Intra and H.264/AVC No motion for all sequences and for all GOP sizes.

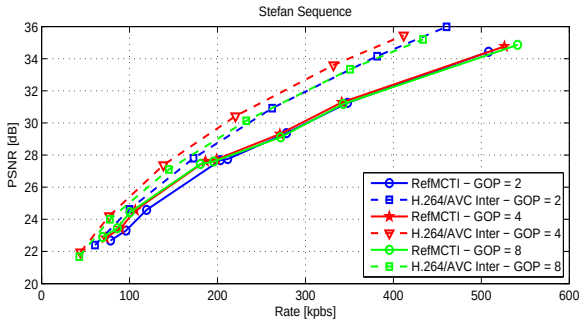
Figure 20 shows the performance of the proposed method (RefMCTI) in comparison to that of H.264/AVC Inter prediction with motion, for Stefan, Bus, Foreman, and Coastguard sequences. The gap between the performance of H.264/AVC Inter prediction with motion and the proposed method is reduced to a large extent, compared to previous techniques.

The performance of our proposed method is significantly better than the performance of [20]. However, it should be noted that [20] uses a pixel-domain DVC. The proposed method in [27] allows a gain up to 1 dB in the higher bitrate range, and up to 0.5 dB in the lower bitrate range for Foreman sequence, for a GOP size of 2. In comparison, our proposed method achieves an average gain of 1.37 dB for this sequence. In [25][26], the RD performance is not shown.

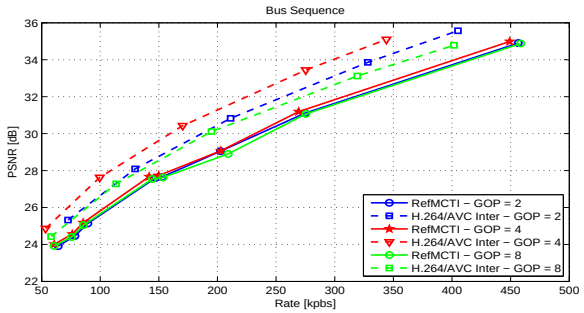
C. Complexity assessment

The complexity of the SIFT algorithm and the matching process increases with the number of feature points and therefore depends on the video content. However, given that original frames are used for global motion estimation, the complexity of the SIFT algorithm is independent of the RD operating point.

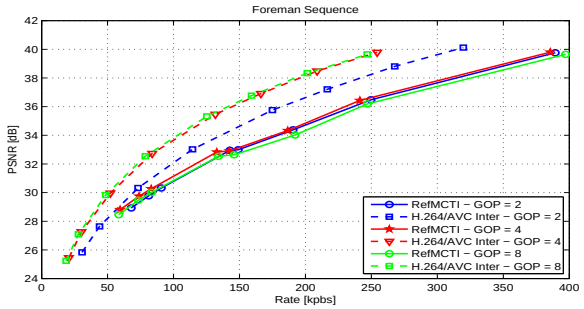
For 60 to 120 feature points, typical for sequences such as Foreman or Coastguard, the encoding complexity is increased by 15 to 30 % compared to the DISCOVER codec. In [30], it is shown that the encoding complexity of WZFs is about $1/6$ the average encoding complexity of H.264/AVC Intra or H.264/AVC No motion. Therefore, despite the complexity increase due to SIFT, the encoding time for the proposed scheme remains lower than the one for H.264/AVC Intra or H.264/AVC No motion.



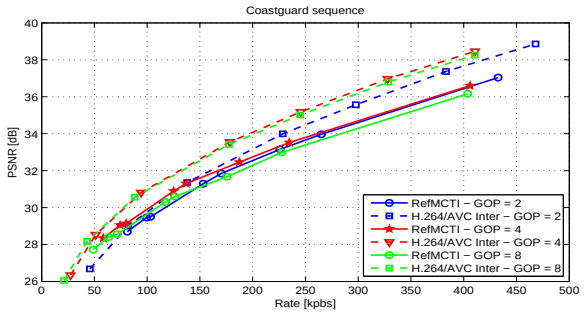
(a) RD performance for Stefan sequence for all GOP sizes.



(b) RD performance for Bus sequence for all GOP sizes.



(c) RD performance for Foreman sequence for all GOP sizes.



(d) RD performance for Coastguard sequence for all GOP sizes.

Fig. 20. RD performance comparison - Proposed (RefMCTI) and H.264/AVC Inter prediction with motion for Stefan, Bus, Foreman, and Coastguard sequences.

With the proposed DVC scheme or DISCOVER, the encoding complexity saving compared to conventional H.264/AVC Intra or No motion coding increases with the GOP size, as fewer KFs are used. However, DISCOVER tends to perform very poorly at a large GOP size, making such operating points less attractive. In contrast, the proposed scheme performs almost equally well at GOP sizes of 2, 4 and 8. Hence, our system makes the use of large GOP sizes more appealing, since it allows for an important reduction in the encoding complexity compared to conventional coding techniques.

In order to further reduce encoding complexity, Speeded Up Robust Features (SURF) [31] could be used instead of SIFT to extract feature points. Indeed, it has been shown that SURF achieves similar performances as SIFT with a greatly reduced complexity. Therefore, SURF could be effectively used at the encoder to extract feature points, allowing for a marginal increase in complexity compared to DISCOVER.

Finally, it should be noted that the execution time of the decoding process is significantly reduced due to the enhancement of the SI, which results in fewer requests through the feedback channel, despite additional processing for global motion compensation.

V. Conclusion

A new technique for the fusion of global and local motion estimations is proposed in this paper. This fusion is performed at the decoder side. Moreover, two methods for further improvement of the fusion during the decoding process are presented in this paper.

Experimental results show that our proposed method can achieve a gain in RD performance up to 1.92 dB for a GOP size of 2 and 4.65 dB for longer GOP sizes, compared to DISCOVER codec, especially when the video sequence contains high global motion. The improvement becomes even more important as the GOP size increases.

With the proposed method, DVC now outperforms H.264/AVC Intra or H.264/AVC No motion in all reported test conditions. Moreover, the performance gap between the proposed DVC scheme and H.264/AVC Inter prediction with motion is significantly reduced.

Future work will be focusing on further improvement of the fusion in order to achieve a better RD performance. We will also investigate the use of SURF in the feature points extraction to reduce encoding complexity.

References

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[2] J.D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. IT-19, pp. 471–480, July 1973.

[3] 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, pp. 1–10, July 1976.

[4] R. Puri and K. Ramchandran, "PRISM: A video coding architecture based on distributed compression principles," *EECS Department, University of California, Berkeley, Tech. Rep. UCB/ERL M03/6*, 2003.

[5] B. Girod, A. Aaron, S. Rane, and D. Rebello-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, pp. 71–83, Jan. 2005.

[6] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M.Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proc. of Picture Coding Symposium*, Lisboa, Portugal, Oct. 2007.

[7] "Discover project," http://www.discoverdvc.org/.

[8] J. Ascenso, C. Brites, and 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, July 2005.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91 – 110, 2004.

[10] A. Abou-Elailah, J. Farah, M. Cagnazzo, B. Pesquet-Popescu, and F. Dufaux, "Improved side information for distributed video coding," in *3nd European Workshop on Visual Information Processing (EUVIP)*, Paris, France, July 2011, pp. 42 – 49.

[11] T. Maugey and B. Pesquet-Popescu, "Side information estimation and new schemes for multiview distributed video coding," *Journal of Visual Communication and Image Representation*, vol. 19, no. 8, pp. 589–599, 2008.

[12] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal reconstruction in wyner-ziv video coding with multiple side information," in *IEEE 9th Workshop on Multimedia Signal Processing (MMSP)*, 2007, pp. 183 – 186.

[13] J. Ascenso, C. Brites, and F. Pereira, "Motion compensated refinement for low complexity pixel based distributed video coding," in *Proceedings of the IEEE international conference on Advanced Video and Signal-Based Surveillance*, Sept. 2005, pp. 593 – 598.

[14] X. Fan, O. Au, N. Cheung, Y. Chen, and J. Zhou, "Successive refinement based Wyner-Ziv video compression," *Signal Processing: Image Communication*, vol. 25, pp. 47–63, Jan. 2010.

[15] J. Ascenso, C. Brites, F. Dufaux, A. Fernando, T. Ebrahimi, F. Pereira, and S. Tubaro, "The VISNET II DVC Codec: Architecture, Tools and Performance," in *Proc. of the 18th European Signal Processing Conference (EUSIPCO)*, 2010.

[16] S. Ye, M. Ouaret, F. Dufaux, and T. Ebrahimi, "Improved side information generation for distributed video coding by exploiting spatial and temporal correlations," *EURASIP Journal on Image and Video Processing*, vol. 2009, pp. 15 pages, 2009.

[17] R. Martins, C. Brites, J. Ascenso, and F. Pereira, "Refining side information for improved transform domain wyner-ziv video coding," *IEEE Transactions on circuits and systems for video technology*, vol. 19, no. 9, pp. 1327 – 1341, Sept. 2009.

[18] A. Abou-Elailah, F. Dufaux, M. Cagnazzo, B. Pesquet-Popescu, and J. Farah, "Successive refinement of side information using adaptive search area for long duration GOPs in distributed video coding," in *19th International Conference on Telecommunications (ICT)*, Jounieh, Lebanon, Apr. 2012.

[19] G. Petrazzuoli, M. Cagnazzo, and B. Pesquet-Popescu, "High order motion interpolation for side information improvement in DVC," in *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, June 2010, pp. 2342 – 2345.

[20] R. Hansel and E. Muller, "Global motion guided adaptive temporal inter-/extrapolation for side information generation in distributed video coding," in *IEEE International Conference on Image Processing*, Brussels, Belgium, Sept. 2011, pp. 2681 – 2684.

[21] A. Aaron, S. Rane, and B. Girod, "Wyner-Ziv video coding with hash-based motion compensation at the receiver," in *Proc. Int. Conf. on Image Processing*, Singapore, Oct. 2004, vol. 05, pp. 3097–3100.

[22] J. Ascenso and F. Pereira, "Adaptive hash-based side information exploitation for efficient Wyner-Ziv video coding," in *Proc. Int. Conf. on Image Processing*, San Antonio, Oct. 2007, vol. 03, pp. 29–32.

[23] M. Guo, Z. Xiong, F. Wu, D. Zhao, X. Ji, and W. Gao, "Witsenhausen-wyner video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, pp. 1049 – 1060, 2011.

[24] H. Xiong, H. Lv, Y. Zhang, L. Song, Z. He, and T. Chen, "Subgraphs matching-based side information generation for distributed multiview video coding," *EURASIP Journal on Advances in Signal Processing*, p. 17 pages, 2009.

[25] T. Clercks, A. Munteanu, J. Cornelis, and P. Schelkens, "Distributed video coding with shared encoder/decoder complexity," in *Proc. IEEE International Conference on Image Processing*, San Antonio, TX, Sept. 2007.

[26] H. Chen and E. Steinbach, "Flexible distribution of computational complexity between the encoder and the decoder in distributed video coding," in *Proc. IEEE International Conference on Multimedia and Expo*, Hannover, Germany, June 2008.

[27] F. Dufaux and T. Ebrahimi, "Encoder and decoder side global and local motion estimation for distributed video coding," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2010, pp. 339 – 344.

[28] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain wyner-ziv video coding," *IEEE Trans. on Circuit ans System for Video Technology*, vol. 18(9), pp. 1177–1190, 2008.

[29] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, Austin, USA, Apr. 2001.

[30] C. Brites, J. Ascenso, J. Pedro, and F. Pereira, "Evaluating a feedback channel based transform domain Wyner-Ziv video codec," *Signal Processing: Image Communication*, vol. 23, no. 4, pp. 269–297, Apr. 2008.

[31] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346 – 359, 2008.