Successive Refinement of Side Information using Adaptive Search Area for Long Duration GOPs in Distributed Video Coding

Abdalbassir Abou-Elailah, Frederic Dufaux, Marco Cagnazzo, and Beatrice Pesquet-Popescu
Signal and Image processing Department
Institut TELECOM - TELECOM Paristech
46 rue Barrault, F - 75634 Paris Cedex 13, FRANCE
Email: {elailah, frederic.dufaux, marco.cagnazzo, beatrice.pesquet-popescu}@telecom-paristech.fr

Joumana Farah
Telecommunications Department, Faculty of Engineering
Holy-Spirit University of Kaslik
P.O. Box 446, Jounieh, Lebanon
Email: joumanafarah@usek.edu.lb

***Abstract*—In distributed video coding, the reference frames are used to generate a side information at the decoder in order to decode the Wyner-Ziv frame. The side information has a strong impact on the coding efficiency of distributed video coding. The estimation of the side information becomes less effective when the temporal distance between the neighboring reference frames increases, as well as, the sequence contains fast motion. In this paper, we propose a new method based on successive refinement of the side information by adapting the search area after decoding the first DCT band. More specifically, different search areas are initially set according to the temporal distance between the neighboring reference frames. Furthermore, the size of the search area is adapted to the current motion after decoding the first DCT band by using partially decoded Wyner-Ziv frame. This adapted search area is used in order to refine the side information after decoding each remaining DCT band. The experimental results show that the proposed technique allow an improvement in rate distortion performance that can reach 0.7 dB for GOP size of 8 compared to the method when a constant search area is used, and a significant gain up to 3.23, with respect to DISCOVER codec. The improvement in final side information up to 5.6 dB compared to constant search area. Moreover, the proposed method by using adaptive search area reduces significantly the time of the decoding process for all test sequences.**

I. INTRODUCTION

Distributed Video Coding (DVC) is an emerging video coding paradigm that consists in exploiting the similarities among successive frames at the decoder side. In DVC, the task of motion estimation and compensation is achieved at the decoder side, unlike conventional video coding standards such as ISO/IEC MPEG and ITU-T H.26x. This new scheme is well-suited for many emerging applications such as wireless video surveillance, multimedia sensor networks, wireless PC cameras, and mobile cameras phones. These applications require a low complexity encoding, while possibly affording a high complexity decoding. Furthermore, the complexity can be flexibly distributed between the encoder and the decoder in DVC.

DVC is based on two major information theoretic results, the Slepian-Wolf and Wyner-Ziv (WZ) theorems. The Slepian-Wolf theorem for lossless compression [1] 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 WZ theorem [2] extends the Slepian-Wolf one to the case of lossy compression of X when Side Information (SI) Y is available at the decoder.

Recently, practical implementations of DVC have been proposed in [3][4] based on these theoretical results. The DISCOVER codec [5][6] is one of the most efficient and popular existing architectures, which is based on transform domain WZ coding. In DISCOVER codec, the images of the sequence are split into two sets of frames, the Key Frames (KFs) and the WZ 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. The WZFs are encoded independently, transformed and quantized. The quantized symbols are fed in a channel code to generate the parity bits. At the decoder, the KFs are first decoded, and then used to generate the SI, which is an estimation of the WZF being decoded. The Motion-Compensated Temporal Interpolation (MCTI) [7] technique is used in DISCOVER codec to estimate this SI. Finally, the parity bits are used to correct the errors in the SI, and decoded WZF is obtained.

The performance gap between DVC and classical inter frame coding has not closed as promised by the theoretical results yet. It is in part due to the quality of the SI, which has a strong impact on the final Rate-Distortion (RD) performance. The quality of the SI is not good when the temporal distance between the neighboring reference frames increases, or the sequence video contains fast motion. Several works have been proposed in order to enhance the SI. High-order motion interpolation has been proposed [8] in order to cope with object motion with non-zero acceleration. An approach proposed by Aaron et al. [9] and by Ascenso et al. [10] consists in sending a hash of the WZF being decoded to enhance the interpolation of the SI. However, these techniques demand some additional data (the hash) to be sent through the channel. Other techniques exist that can avoid this overhead. They are based on the successive refinement of the SI. A solution proposed by J. Ascenso et al. [11] 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 [12], the authors proposed a novel DVC successive refinement approach to improve the motion compensation accuracy and the SI. This approach is based on the N-Queen sub-sampling pattern. The authors in [13] proposed a solution for transform-domain DVC, which refines the SI after the decoding of all DCT bands in order to improve reconstruction. In VISNET 2 project [14], the refinement process of the SI is carried out after decoding all DCT bands, and a deblocking filter is used. In [15][16], solutions are proposed for transform-domain DVC based on the successive refinement of the SI after each decoded DCT band. Moreover, the refinement approach consists of four modules: Suspicious Vector Detection, Refinement, Mode Selection, and Motion Compensation.

In this paper, we propose a new approach in order to improve the SI in transform-domain DVC using adaptive search area. This solution is based on our previous work [16], which consists in progressively improving the SI after each decoded DCT-band. In [16], a constant search area is used to refine the SI after each decoded DCT-band regardless the temporal distance between the neighboring reference frames. This method [16] achieves a significant gain for sequences contain fast motion, as well as, for long duration GOPs. In this paper, variable search areas are initially set according to the temporal distance between the neighboring reference frames. We first start by generating an Initial Side Information (INSI) by using the backward and forward reference frames, similarly to the SI generated in DISCOVER codec. The decoder reconstructs a Partially Decoded Wyner-Ziv (PDWZ) frame by correcting the INSI with the parity bits of the first DCT-band. Then, the PDWZ frame, along with the backward and forward reference frames, is used to adapt the initial search area. Furthermore, the adapted search area is used to refine the INSI. Finally, we correct this refined INSI with the parity bits of the next DCT-band by using the adaptive search area, and we repeat the same procedure to decode all DCT-bands of the current WZF.

This paper is structured as follows. First, the related work is introduced in Section 2. The proposed approach by successive refinement of the SI using adaptive search area is described in Section 3. Experimental results are then shown in Section 4 in order to evaluate and compare the RD performance of the proposed approach. Finally, conclusions and future work are presented in Section 5.

II. Related work

In this section, the related work is represented. First, the DISCOVER codec is briefly represented. The video sequence

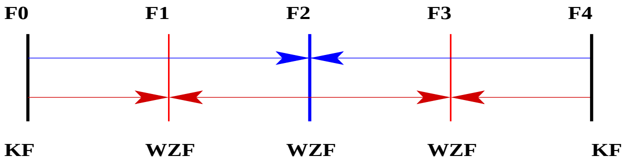

Fig. 1. Interpolation steps for a GOP size 4.

band	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
QI 1	16	8	8	0	0	0	0	0	0	0	0	0	0	0	0	0
QI 2	32	8	8	0	0	0	0	0	0	0	0	0	0	0	0	0
QI 3	32	8	8	4	4	4	0	0	0	0	0	0	0	0	0	0
QI 4	32	16	16	8	8	8	4	4	4	4	0	0	0	0	0	0
QI 5	32	16	16	8	8	8	4	4	4	4	4	4	4	0	0	0
QI 6	64	16	16	8	8	8	8	8	8	8	4	4	4	4	4	0
QI 7	64	32	32	16	16	16	8	8	8	8	4	4	4	4	4	0
QI 8	128	64	64	32	32	32	16	16	16	16	8	8	8	4	4	0

Fig. 2. Various 4×4 quantization matrices corresponding to eight rate-distortion points. For each QI, the number of levels is given for the 16 bands

is divided into WZFs and KFs. The latters are encoded using H.264/AVC Intra coding. Figure 1 shows all necessary interpolations for a GOP size 4. For example, during the interpolation of WZF F2, the forward and backward reference frames are KFs F0 and F4. For the interpolation of F1, the reference frames are the KF F0 and the previously decoded WZF F2. Second, our previous method [16] is presented.

A. DISCOVER codec

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 block-based DCT. The DCT coefficients of the entire WZF are then organized in 16 bands, indicated by b_k with $k \in [1, 16]$, according to their position within the 4×4 blocks. The DC coefficients are placed in the first band $k = 1$, and the others coefficients are grouped in the AC bands $k = 2, 3, ...16$.
  Next, each DCT coefficients band b_k is uniformly quantized with 2^{M_k} levels. Figure 2 shows the number of levels for each band for eight different rates $QI = 1, 2, ...8$. 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 frame interpolation framework is composed of four modules to obtain high quality SI [7]: forward motion estimation between the previous and next reference frames, bi-directional motion estimation to refine the motion vectors, spatial smoothing of motion vectors

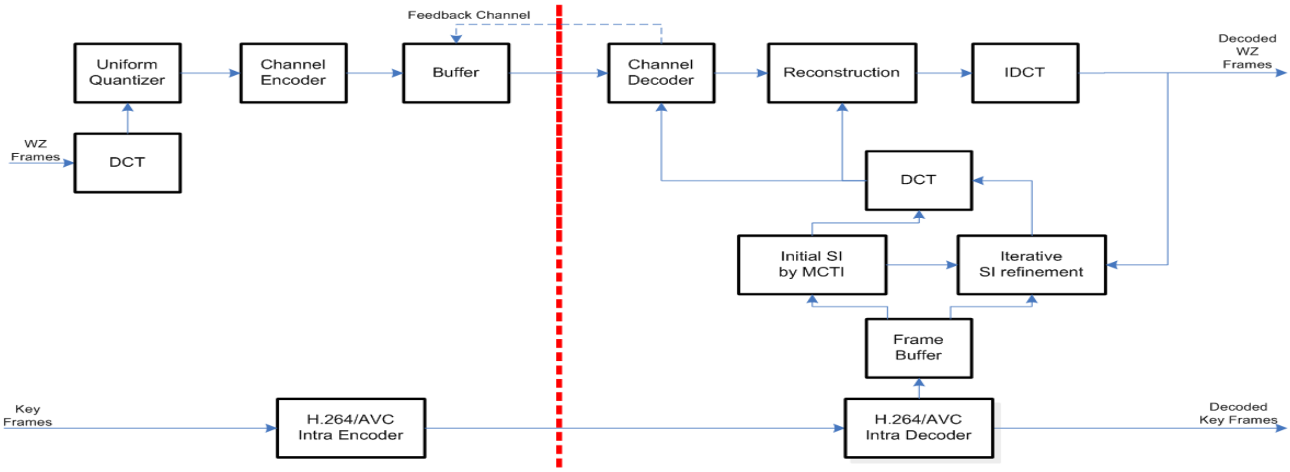

Fig. 3. Overall structure of the proposed DVC codec.

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.
- **Reconstruction and inverse transform -** The reconstruction corresponds to the inverse of the quantization using the SI DCT coefficients and the decoded WZ DCT coefficients. After that, the inverse 4×4 DCT transform is carried out, and the entire frame is restored in the pixel domain.

B. Successive refinement of the SI [16]

The motion vectors estimated by the MCTI technique for certain blocks can be erroneous, especially in sequences containing high motion. For this reason, our algorithm [16] aims at re-estimating suspect vectors after the decoding of each DCT band. 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) is computed between the PDWZ frame and the actual SI as follows:

$$\mathrm{MAD}(\text{actual SI}, \mathrm{PDWZ}(\mathbf{MV})) < T_1. \qquad (1)$$

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 $(8+n) \times (8+n)$ pixels is considered when MAD is computed. If Eq. (1) is not satisfied, the motion vector is identified as a suspicious vector and will be further re-estimated by applying two steps. The first one consists in re-estimating the motion vector within a constant search area of ± 16 pixels in two pixels accuracy. The second step consists in refining the obtained motion vector within a search area of ± 3 pixels in half-pixel accuracy. Otherwise (Eq. (1) 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. It is important to note here that a constant search area of ± 16 pixels is used in this algorithm regardless the temporal distance between the previous and next reference frames.

In this algorithm, a motion-compensated estimate is carried out by selecting the most similar block to the current block from three sources: bi-directional motion-compensated average of the previous and next reference frames (BIMODE) is selected if $|\mathrm{MAD}_n - \mathrm{MAD}_p| < T_2$. Otherwise, the previous reference frame (BACKWARD MODE) is selected if $\mathrm{MAD}_p < \mathrm{MAD}_n$. Otherwise, the next reference frame (FORWARD MODE) is selected. MAD_p and MAD_n are the estimated mean absolute differences between the current block (in PDWZ) and the corresponding blocks in the previous and next reference frames respectively, and T_2 is a threshold.

III. Proposed Method

The block diagram of our proposed codec architecture is depicted in Figure 3. It is based on the DISCOVER codec [5][6]. The INSI is first computed by MCTI with spatial motion smoothing exactly as in DISCOVER codec. The LDPC parity bits of the first band (DC band) are used in order to correct the corresponding DCT coefficients in INSI; the obtained decoded frame is denoted as Partially Decoded Wyner-Ziv (PDWZ) frame. Let d to be the distance between the previous and next reference frames. For example, for a GOP size of 8 (KF_1 WZF_1 WZF_2 WZF_3 WZF_4 WZF_5 WZF_6 WZF_7 KF_2)), d equals 8 for WZF_4, equals 4 for WZF_2 and WZF_6, and equals 2 for WZF_1, WZF_3, WZF_5, and WZF_7. First, we have set an initial search area according to the distance d. A search area of $\mathrm{RA}_4 = \pm 80$ pixels is initially set if d equals 8 (the temporal distance between the current WZF and the previous (or next) reference frame is 4), a search area of $\mathrm{RA}_2 = \pm 56$ pixels is set if d equals 4, and a search area of $\mathrm{RA}_1 = \pm 32$ pixels is

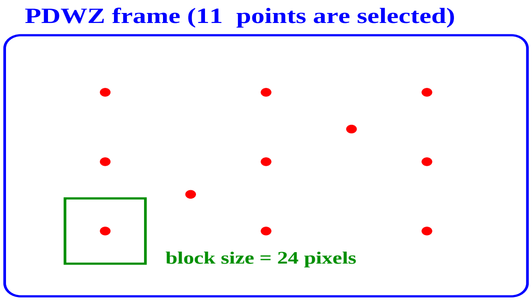

Fig. 4. The selected points in PDWZ frame after decoding the first DCT band.

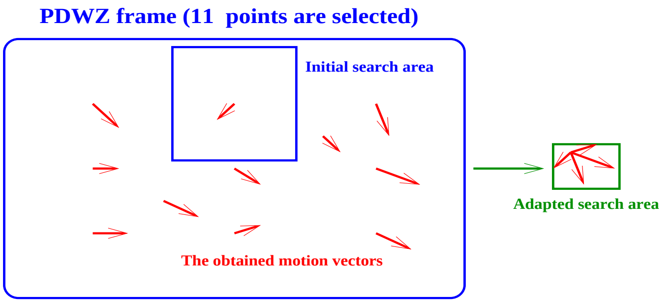

Fig. 5. The obtained motion vectors are used to adapt the search area.

set if d equals 2. Those large search areas are more efficient for high motion in the current GOP of the sequence. However, the computational complexity is significantly increased, in the case of slow motion in the current GOP, without enhancement in the SI, compared to a small search area in this case.

Thus, an adaptive search area algorithm to the current motion is necessary. For this reason, N points are selected in the PDWZ frame. These N points are chosen in a way to recover all parts in the frame. Figure 4 shows $N = 11$ selected points in the PDWZ frame. Those selected points represent the center of a large block of 24 pixels. These blocks are used along the previous and next reference frames in order to adapt the search area. The matching between the PDWZ frame and the previous (and next) reference frame is carried out for those selected blocks in two pixels accuracy. The MAD is penalized (MADp) by the length of the motion vector $\mathbf{MV} = (MV_x, MV_y)$ when the matching is carried out as follows:

$$\text{MADp} = \text{MAD}/(1 + \text{penalty} \times \sqrt{MV_x^2 + MV_y^2}). \quad (2)$$

where penalty is set to 0.008 if $d = 4$, to 0.012 if $d = 2$, and to 0.02 if $d = 1$. This penalty allows to avoid the error of large search area when a selected block is into a homogeneous region.

The initial search area is adapted in the four directions according to the obtained motion vectors. The maximums of the obtained motion vectors in the four directions are selected to adapt the initial search area. Figure 5 shows how to adapt the search area from the obtained motion vectors. This algorithm

	Extended block $(\mathbf{8+n}) \times (\mathbf{8+n})$ - $\mathbf{T_2 = 5}$					
	$T_1 = 3$		$T_1 = 6$		$T_1 = 9$	$T_1 = 12$
	$n = 0$	$n = 4$	$n = 0$	$n = 4$	$n = 4$	$n = 4$
	Stefan sequence					
	Proposed					
Δ_R (%)	-51.65	**-62.67**	-50.29	**-59.89**	-56.45	-53.01
Δ_{PSNR} [dB]	2.86	**3.39**	2.77	**3.23**	3	2.78
Complexity (%)	79	**85**	76	**77**	77	80
	Reference [16]					
Δ_R (%)	-39.45	**-49.46**	-38.89	**-47.75**	-45.22	-42.58
Δ_{PSNR} [dB]	2.17	**2.63**	2.12	**2.54**	2.38	2.22
Complexity (%)	100	**132**	91	**101**	97	96
	Foreman sequence					
	Proposed					
Δ_R (%)	-64.42	**-71.51**	-61.24	**-67.87**	-64.64	-62.35
Δ_{PSNR} [dB]	3.12	**3.44**	2.9	**3.18**	3	2.85
Complexity (%)	69	**75**	71	**74**	76	78
	Reference [16]					
Δ_R (%)	-59.96	**-67.84**	-58.03	**-64.63**	-61.87	-59.58
Δ_{PSNR} [dB]	2.9	**3.27**	2.74	**3.04**	2.87	2.74
Complexity (%)	75	**89**	71	**78**	76	77

TABLE I

RATE-DISTORTION PERFORMANCE GAIN AND DECODING TIME COMPLEXITY FOR *Stefan* AND *Foreman*, TOWARDS DISCOVER CODEC, FOR DIFFERENT VALUES OF T_1 AND n.

is used to adapt the search area between the PDWZ frame and previous (and next) reference frame. The adaptive search area is used to refine the suspected motion vectors along the previous and next reference frames after each decoded DCT band.

Note that the DCT coefficients band b_k is uniformly quantized with 2^{M_k} levels, M_k decreases when k increases (Figure 2). We can see that M_k becomes less effective after the three first DCT bands. For this reason, the SI is refined after each decoded DCT band if $k < 4$, and after two decoded DCT bands otherwise.

IV. Experimental Results

In order to evaluate the performance of the proposed method, we performed extensive simulations, adopting the same test conditions as described in DISCOVER [5][6], *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, and to our previous work [16].

A. Parameter tuning

The parameter T_1 plays an important role in the proposed method, which determines the time of the decoding process and the achieved performance improvement. Moreover, the size of the extended block $(8+n) \times (8+n)$ can improve the performance, but a large extended block may increase the time of the decoding process. For $T_1 = \infty$, this means no block is considered as erroneous, in this case, it is equivalent to DISCOVER codec. In the case of $T_1 = 0$, all blocks are considered as erroneous, and will be refined after each decoded DCT band. In the simulations, we have set $T_2 = 5$ after preliminary tests. The parameter T_2 can improve a little bit the RD performance.

In Table I, the RD performance of the proposed method and the reference [16] is shown for different values of T_1 and n compared to DISCOVER codec, using the Bjontegaard metric

Fig. 6. Visual result comparisons between the SI estimated by MCTI technique (left), the final SI estimated in [16] (center), and the final SI estimated by the proposed method (right) for frame number 95 of Foreman sequence.

Fig. 7. Visual result comparisons between the SI estimated by MCTI technique (left), the final SI estimated in [16] (center), and the final SI estimated by the proposed method (right) for frame number 115 of Stefan sequence.

[17] for a GOP size of 8. In this table, the percentage of the decoding time complexity compared to DISCOVER codec is also shown. The complexity is computed as follows:

$$\text{Complexity (\%)} = 100 \times \frac{\text{Decoding time obtained}}{\text{Decoding time of DISCOVER}} \quad (3)$$

It is clear that for $T_1 = 3$ and $n = 4$, the proposed method and the reference [16] achieve the best RD performance. For Stefan, the proposed method can reduce the time of the decoding process by 15% compared to DISCOVER codec for these values. On the contrary, the time of decoding process is increased by 32% for the reference[16], due to the high motion in this sequence. As we can see, the proposed method reduces the time of the decoding process for all values of T_1, due to well adapted the search area to the motion.

In the case of Foreman sequence, the reference [16] achieve a significant gain compared to DISCOVER codec, and the time of the decoding process is reduced. On the other hand, in the proposed method, the gain becomes more effective and the time of the decoding process is more reduced. In the simulations, we have set $T_1 = 6$ and $n = 4$ due to the high performance and low computational load achieved for these values.

B. SI performance assessment

Figures 6 and 7 show the visual results of the SI estimated by MCTI (DISCOVER), the final SI estimated by [16] after decoding all DCT bands, and the final SI estimated by the proposed method respectively, for frame number 95 of Foreman and frame number 115 of Stafan, for a GOP equals 8. The SI frame obtained by MCTI contain block artifacts. On the contrary, the SI frames obtained by [16] and the proposed method are better.

For Foreman (Figure 6), the proposed method allows an improvement up to 11 dB compared to MCTI technique, and an improvement up to 2.76 dB compared to the final SI estimated in [16]. For the final decoded WZFs of these SI frames, the proposed method achieves a gain up to 2 dB compared to DISCOVER codec, with less requested bits, down from 46.39 Kbits to 36.22 Kbits, and a gain up 0.67 dB compared to [16], with less requested bits, down from 40.30 Kbits to 36.22 Kbits.

For Stefan (Figure 7), the final SI obtained by the proposed method allows a gain up to 8.25 dB compared to MCTI technique, and an improvement up to 5.26 dB compared [16]. For the final decoded WZFs of these SI frames, the proposed method achieves a gain up to 2.41 dB compared to DISCOVER codec, with less requested bits, down from 61.75 Kbits to 47.93 Kbits, and a gain up 2.12 dB compared to [16], with less requested bits, down from 59.75 Kbits to 47.93 Kbits.

	Stefan		Bus	
	Ref. [16]	Proposed	Ref. [16]	Proposed
	GOP size = 2			
Δ_R (%)	-16.73	**-20.58**	-7.54	**-7.85**
Δ_{PSNR} [dB]	1	**1.23**	0.41	**0.42**
	GOP size = 4			
Δ_R (%)	-36.24	**-45.48**	-22.69	**-22.49**
Δ_{PSNR} [dB]	1.98	**2.52**	1.13	**1.14**
	GOP size = 8			
Δ_R (%)	-47.75	**-66.38**	-34.6	**-38.79**
Δ_{PSNR} [dB]	2.54	**3.23**	1.66	**1.87**
	Foreman		**Soccer**	
	GOP size = 2			
Δ_R (%)	-19.85	**-20.23**	-21.41	**-22.66**
Δ_{PSNR} [dB]	1.08	**1.1**	1.06	**1.12**
	GOP size = 4			
Δ_R (%)	-44.57	**-46.63**	-32.69	**-35.07**
Δ_{PSNR} [dB]	2.21	**2.31**	1.62	**1.76**
	GOP size = 8			
Δ_R (%)	-64.63	**-67.87**	-39.25	**-43.23**
Δ_{PSNR} [dB]	3.04	**3.19**	1.89	**2.09**

TABLE II

RATE-DISTORTION PERFORMANCE GAIN FOR *Stefan*, *Bus*, *Foreman*, AND *Soccer* SEQUENCES TOWARDS DISCOVER CODEC, USING BJONTEGAARD METRIC [17].

C. Rate-Distortion performance

The RD performance of the proposed method is shown for the Stefan, Bus, Foreman, and Soccer sequences in Table II, in comparison to the DISCOVER codec, using the Bjontegaard metric [17] for different GOP sizes (2, 4 and 8). The first column represents the performance of the our previous work [16], *i.e.*, a constant search area of ± 16 pixels is used regardless the distance between the reference frames. It is clear that our proposed method achieves a significant gain compared to DISCOVER codec, especially for sequences contain high motion such as Stefan and Foreman sequences.

For Stefan sequence for a GOP size of 8, the approach in [16] can achieve a gain up 2.54 dB with a rate reduction up to 47.75 % compared to DISCOVER codec. Furthermore, the proposed method allows a significant gain up to 3.23 dB with a rate reduction of 66.38 %. For other sequences, the proposed method achieves a little improvement compared to [16], when the time of the deocidng process is reduced even for sequences contain slow motion because the search area is adapted to the current motion in the sequence.

V. CONCLUSION

Successive refinement of the side information using an adaptive search area was proposed in this paper, based on the successive decoding of the DCT bands. The partially decoded frame after decoding the first DCT band is used to adapt the initial search area. The adaptive search area is used in order to progressively refine the SI, along the previous and next reference frames, after each decoded DCT band.

Experimental results showed that our proposed method can achieve a gain in RD performance up to 0.7 dB for a GOP size of 8 compared to [16], and 3.23 dB compared to DISCOVER codec, especially when the video sequence contains high motion. The proposed method allows an improvement up to 5.6 dB in the final SI compared to [16]. Moreover, the time of the decoding process is significantly reduced by using adaptive search area

Future work will be focusing on further improvement of the side information in order to achieve a better RD performance.

REFERENCES

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

[2] 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, Jul. 1976.

[3] 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.

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

[5] 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.

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

[7] C. Brites, J. Ascenso, and F. Pereira, "Improving transform domain Wyner-Ziv video coding performance," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, May 2006, pp. 525–528.

[8] 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)*, Jun. 2010, pp. 2342 – 2345.

[9] 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*, vol. 05, Singapore, Oct. 2004, pp. 3097–3100.

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

[11] 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*, Sep. 2005, pp. 593 – 598.

[12] 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.

[13] 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, p. 15 pages, 2009.

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

[15] A. Abou-Elailah, F. Dufaux, M. Cagnazzo, B. Pesquet-Popescu, and J. Farah, "Successive refinement of motion compensated interpolation for transform-domain distributed video coding," in *19th European Signam Processing Conference (EUSIPCO)*, Barcelona, Spain, 2011, pp. 11 – 15.

[16] 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, Jul. 2011, pp. 42 – 49.

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