# IMAGE INTERPOLATION WITH EDGE-PRESERVING DIFFERENTIAL MOTION REFINEMENT

*Marco Cagnazzo, Wided Miled, Thomas Maugey, Béatrice Pesquet-Popescu*

Télécom ParisTech, TSI departement, 46 rue Barrault 75634 Paris (FRANCE)
E-mail: {cagnazzo,miled,maugey,pesquet}@telecom-paristech.fr

## ABSTRACT

Motion estimation (ME) methods based on differential techniques provide useful information for video analysis, and moreover it is relatively easy to embed into them regularity constraints enforcing for example, contour preservation. On the other hand, these techniques are rarely employed for video compression since, though accurate, the dense motion vector field (MVF) they produce requires too much coding resource and computational effort. However, this kind of algorithm could be useful in the framework of distributed video coding (DVC), where the motion vector are computed at the decoder side, so that no bit-rate is needed to transmit them. Moreover usually the decoder has enough computational power to face with the increased complexity of differential ME.

In this paper we introduce a new image interpolation algorithm to be used in the context of DVC. This algorithm combines a popular DVC technique with differential ME. We adapt a pel-recursive differential ME algorithm to the DVC context; moreover we insert a regularity constraint which allows more consistent MVFs. The experimental results are encouraging: the quality of interpolated images is improved of up to 1.1 dB w.r.t. to state-of-the-art techniques. These results prove to be consistent when we use different GOP sizes.

***Index Terms***— Distributed video coding, dense motion vector field, image interpolation, differential motion estimation

## 1. INTRODUCTION

Motion estimation (ME) is a very useful tool both in video analysis and compression, even though in the two contexts different characteristics are required. In the first case one usually wants as much accurate a motion vector field (MVF) as possible, ending up with dense, high-precision vectors. For these applications, the optical flow approach [1] has proved to be quite effective, and has generated a large class of ME algorithm, namely gradient techniques, along with the derived class of pel-recursive (PR) techniques. However, the dense MVF produced by gradient and PR methods is unsuited for the classical video coding paradigm, since it requires huge coding resources to be encoded. In this case one rather tries to optimize a rate-distortion compromise: less accurate but much less costly (in terms of bit-rate) MVFs, as the ones produced by block matching (BM) algorithms are by far preferred for video compression. For this reason, when the new paradigm of distributed video coding (DVC) has emerged, it seemed natural to use BM motion estimation and compensation, even though the peculiarity of DVC could call for alternative approaches.

DVC deals with the compression of multiple outputs of correlated sources which do not communicate with each other. The sensors send their compressed outputs to a central point for joint decoding. One of the most interesting characteristics of this new paradigm is that separated encoding is theoretically just as efficient as joint encoding, both for lossless [2] and lossy coding [3]. Unfortunately the arguments at the basis of the distributed coding theory are asymptotic and non-constructive, so many efforts have been devoted to the search of techniques able to achieve these theoretical limits.

In this work we consider one of the most popular approaches for DVC, Stanford [4], the main other being PRISM [5]. In the Stanford paradigm, the input sequence is split into two subsets, the key frames (KF) and the Wyner-Ziv frames (WZF). KFs and WZFs alternate within each group of pictures (GOP) so that the two subsets can be seen as two correlated sources. The number of WZFs can be fixed or adaptively chosen in order to optimize performances [6, 7]. The KFs are coded with a still image technique (*e.g.* H.264 in INTRA mode), and used at the decoder to generate an estimation of the WZF, called side information (SI). This process is sometimes called image interpolation, and is performed via motion estimation and compensation: it is just here that we propose to use differential ME methods. No matter how the SI has been obtained, the idea behind DVC is to consider it as a noisy version of the original WZF: this means that the encoder just needs to send the parity bits (computed by efficient channel coding methods, as LDPC or turbo coding) of the WZF, or possibly, of its quantized transform coefficients, and the decoder uses them to correct the SI. The decoder can set the rate of parity bits via a feedback channel.

Since in DVC the ME is performed only at the decoder side, the main disadvantages of differential methods (coding cost and computational complexity) are circumvented, and on the contrary, SI interpolation can benefit from the increased accuracy of a dense MVF without having to pay the cost of an exceedingly high coding rate.

Differential ME for DVC has been introduced in [8], where a PR algorithm, proposed by Cafforio and Rocca [9] was adapted to come on the top of the popular DISCOVER [10] image interpolation technique, ending up with a non negligible performance improvement. Encouraged by these results, we have developed a new motion interpolation scheme that is presented in this paper. Here we use the Cafforio-Rocca (CR) algorithm upstream in the processing chain. Moreover we modify it in order to enforce a regularity constraint (the Nagel-Eckelmann constraint [11]) on vectors. This allows for more consistent motion vector fields. The experimental results are encouraging: the quality of interpolated images (called side information in the context of DVC) is improved of up to 1.1 dB w.r.t. to state-of-the-art techniques. These results prove to be consistent when we use different GOP sizes, and moreover we show that these improvements in SI reflect into gains in the global coding scheme. Moreover, even though in this paper we focus on DVC, we observe that other applications can take advantage from improved image interpolation, such us frame-rate up-conversion.

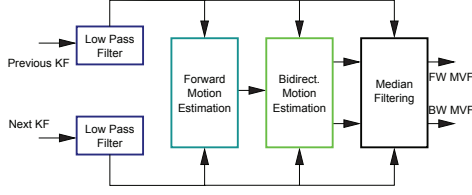The rest of the paper is organized as follows: the reference DVC

**Fig. 1**. Discover image interpolation



**Fig. 2**. Proposed image interpolation scheme

scheme, along with the proposed method is briefly introduced in Section 2; then, in Section 3 we recall the original CR algorithm and we describe the modifications needed to use it in the DVC scheme. Experimental results are shown in Section 4, while Section 5 draws conclusions and outlines the future work.

## 2. REFERENCE SCHEME

In this paper we consider the DISCOVER [10] scheme, which is an established reference for DVC [12, 8], and we propose a variation of the motion interpolation part, based on a PR ME technique. The DISCOVER scheme provides that KFs are coded using the Intra mode of H.264 [13], with an assigned quantization step. The quantized DCT coefficients of WZFs are turbo-encoded but only parity bits are sent to the turbo decoder, where they are used to correct the side information.

The SI (*i.e.* the estimation of the current WZF $I_k$) is generated at the decoder side, using the image interpolation scheme depicted in Fig. 1. The inputs are the adjacent KFs, let them be $I_{k-1}$ and $I_{k+1}$ (but farther frames can be used for larger GOP sizes). The KFs undergo spatial low-pass filtering, and then a forward BM ME is performed between them. This mono-directional MVF is split and used to estimate the backward MVF $\mathbf{v}_B$ between $I_{k+1}$ and $I_k$, and the forward MVF $\mathbf{v}_F$ between $I_{k-1}$ and $I_k$. Finally a weighted median filter is then applied on $\mathbf{v}_B$ and $\mathbf{v}_F$ in order to eliminate the outliers and to get a smooth solution. The WZF estimation is the average between $I_{k+1}$ compensated by $\mathbf{v}_B$ and $I_{k-1}$ compensated by $\mathbf{v}_F$. This technique can adapt to larger GOP sizes. For example, for a GOP size equal to 4, we just use three times the same method: first, we interpolate $I_k$ from KFs $I_{k-2}$ and $I_{k+2}$; then we use it again to interpolate $I_{k-1}$ from $I_{k-2}$ and $I_k$ and $I_{k+1}$ from $I_k$ and $I_{k+2}$. The optimality of this approach has been shown in [14].

In our previous paper on differential ME in DVC [8], it was proposed to use the Cafforio-Rocca algorithm [9] to improve MVFs $\mathbf{v}_B$ and $\mathbf{v}_F$. In this paper we propose to use the CR algorithm in advance, that is, we use it to improve the forward MVF produced by the mono-directional ME. This requires to adapt the CR algorithm to the DVC paradigm; moreover we show how it is possible to modify it in order to integrate a stronger regularity constrain. The resulting image interpolation scheme is shown in Fig. 2. We note that, given the regularization effect of the CR algorithm, the final median filtering is no longer needed.

## 3. THE MODIFIED CAFFORIO-ROCCA ALGORITHM

In this section, we first recall the principles of the CR algorithm (CRA), and then we give the details of the modifications needed to profitably use it within the scheme of Fig. 2. When using the CRA, the image pixels are scanned in a fixed order (*e.g.* in raster order), and for each position a new motion vector (MV) is computed. The algorithm is pel-recursive, in the sense that previously computed
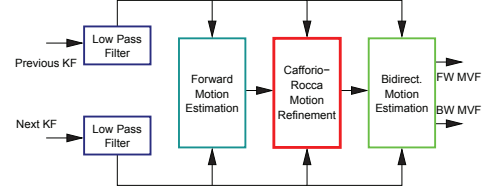
MVs can be used for the initialization of the current one. The original CRA consists in applying, for each pixel $\mathbf{p}$ of the image, three steps, until the estimated MV $\mathbf{v}(\mathbf{p})$ is obtained.

**Initialization.** Some *a priori* information is used as initialization value, $\mathbf{v}^{(1)}(\mathbf{p})$. Often the vector computed for the previous position is used for initialization.

**Validation.** The motion-compensated error $A = |I_{k+1}(\mathbf{p}) - I_k(\mathbf{p} + \mathbf{v}^{(1)})|$ is compared to the non-compensated error, incremented by a positive quantity $\gamma$: $B = |I_{k+1}(\mathbf{p}) - I_k(\mathbf{p})| + \gamma$. If $A \leq B$ the initialization vector is validated and kept for the next step: $\mathbf{v}^{(2)} = \mathbf{v}^{(1)}$. Otherwise the null vector is used: $\mathbf{v}^{(2)} = \mathbf{0}$. The validation steps allows and to prevent algorithm divergence and to get rid of outliers, which can occur for example when the initialization vector belongs to a different object w.r.t. the current position. Of course, it can happen that the non-compensated error is smaller than the compensated error even if the current vector is not an outlier: the threshold value $\gamma$ allows to control the number of validated vectors which are reset to zero.

**Refinement.** The last step consists in refining the validated vector $\mathbf{v}^{(2)}$ by adding to it a correction $\delta\mathbf{v}$. This correction is obtained by minimizing the energy of the prediction error, under a constraint on the norm of the correction vector. The Lagrangian cost function is then:

$$J(\delta\mathbf{v}) = [I_k(\mathbf{p}) - I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)} + \delta\mathbf{v})]^2 + \lambda\|\delta\mathbf{v}\|^2 \quad (1)$$

Using a first order expansion of $I_{k-1}$, it turns out that the value of $\delta\mathbf{v}$ minimizing $J$ is:

$$\delta\mathbf{v}(\mathbf{p}) = \frac{-\epsilon\varphi}{\lambda + \|\varphi\|^2} \quad (2)$$

where $\epsilon = I_k(\mathbf{p}) - I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)})$ is the prediction error associated to the MV $\mathbf{v}^{(2)}$, and $\varphi = \nabla I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)})$ is the spatial gradient of the motion-compensated reference image.

Now we describe the CRA modifications needed in the context of DVC image interpolation. The three steps are modified and moreover we use a different scanning order, based on the blocks used in the forward motion estimation: the blocks are scanned in a raster scan order, and the same is done for the pels within each block.

Our version of the CRA takes as an input $\mathbf{v}^{\text{FWD}}$ the MVF produced by the forward ME, see Fig. 2. These vectors are used in the initialization step: if $\mathbf{p}$ is the first position (*i.e.* top and leftmost) in the block, the vector $\mathbf{v}^{(1)}(\mathbf{p})$ is initialized with $\mathbf{v}^{\text{FWD}}(\mathbf{p})$. Otherwise, we use a weighted average of the left, up, and up-right neighboring vectors, with different weights if the neighbors are in the same block or not.

As far as the validation step is concerned, we not only compute the motion-compensated error associated to $\mathbf{v}^{(1)}(\mathbf{p})$ and the non-compensated error, but also the motion compensated error associated

to $\mathbf{v}^{\text{FWD}}(\mathbf{p})$, and we choose the vector with the least absolute error. As in the original algorithm, the non-compensated error is increased by a threshold $\gamma$ in order to reduce the reset frequency.

The new validation step allows to reintroduce the $\mathbf{v}^{\text{FWD}}(\mathbf{p})$ as validated vector while scanning the current block. This is useful, since, independently from the scanning order, it can happen that, within the same block, we pass several time from one object to another. At the first object boundary crossing, the MV is likely reset by the validation pass, then the pel-recursive nature of the CRA allows to reconstruct the MV of the new object by accumulating the corrections from one pel to the other. However, if during the scanning we come back to the first object, with the original CRA we can only reset to zero the MV; with this modification, we can benefit of a fast recover of the first object MV. Note that we proposed a bidirectional version of this modified validation step in [8].

In the last step, we refine the validated MV $\mathbf{v}^{(2)}(\mathbf{p})$ by adding a correction $\delta\mathbf{v}$. Like in the original algorithm, the correction should minimize the prediction error, under the constraint of a regularization condition. In the original algorithm it is possible to find a closed form of the optimal solution when the regularization is simply a constraint on the correction norm. Here we want to use a stronger constraint. Namely, we consider the diffusion matrix $\mathbf{D}(\nabla I)$:

$$\mathbf{D}(\nabla I) = \frac{1}{|\nabla I|^2 + 2\sigma^2} \left[ \left( \begin{matrix} \frac{\partial I}{\partial y} \\ -\frac{\partial I}{\partial x} \end{matrix} \right) \left( \begin{matrix} \frac{\partial I}{\partial y} \\ -\frac{\partial I}{\partial x} \end{matrix} \right)^T + \sigma^2 \mathbf{I}_2 \right]$$

When the regularization constraint takes into account the diffusion matrix, one is able to inhibit blurring of MVF across object boundaries [11, 15]. This kind of constraint is well known in the literature about optical flow motion estimation and is called Nagel-Enkelmann constraint [11]. We propose therefore the following cost function:

$$J(\delta\mathbf{v}) = [I_{k+1}(\mathbf{p}) - I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)} + \delta\mathbf{v})]^2 + \lambda \delta\mathbf{v}^T \mathbf{D} \delta\mathbf{v} \quad (3)$$

where we used the shorthand notation $\mathbf{D} = \mathbf{D}(\nabla I_{k-1})$. We notice that, in the homogeneous regions where $\sigma^2 \gg |\nabla I_{k-1}|^2$, the cost function becomes equivalent to the one used in the original algorithm, see Eq. (1).

Here we show that even with the new cost function, a closed form of the optimal vector refinement exists, and we give it at the end of this section. Like in the original algorithm, the first step is a first order expansion of the cost function:

$$J \approx \left[ I_{k+1}(\mathbf{p}) - I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)}) - \nabla I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)})^T \delta\mathbf{v} \right]^2 +$$
$$- \lambda \delta\mathbf{v}^T \mathbf{D} \delta\mathbf{v} = \left( \epsilon + \boldsymbol{\varphi}^T \delta\mathbf{v} \right)^2 + \lambda \delta\mathbf{v}^T \mathbf{D} \delta\mathbf{v}$$

where we defined the motion compensation error $\epsilon = I_{k+1}(\mathbf{p}) - I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)}(\mathbf{p}))$ and the compensated gradient $\boldsymbol{\varphi} = \nabla I_{k-1}(\mathbf{p} + \mathbf{v}^{(2)}(\mathbf{p}))$. Then we look for the refinement $\delta\mathbf{v}^*$ which minimizes the function cost: we set to zero the partial derivatives of $J$.

$$\mathbf{0} = \frac{\partial J}{\partial \delta\mathbf{v}}(\delta\mathbf{v}^*) = 2(\epsilon \boldsymbol{\varphi}^T \delta\mathbf{v}^*)\boldsymbol{\varphi} + 2\lambda \mathbf{D}\delta\mathbf{v}^* \quad (4)$$
$$= 2\left( \boldsymbol{\varphi}\boldsymbol{\varphi}^T + \lambda\mathbf{D} \right) \delta\mathbf{v}^* + 2\epsilon\boldsymbol{\varphi}.$$

Note that the derivative of $\delta\mathbf{v}^T \mathbf{D} \delta\mathbf{v}$ has been computed in Eq. (4) using the symmetry of $\mathbf{D}$. The last equation is equivalent to:

$$\delta\mathbf{v}^* = -\left( \boldsymbol{\varphi}\boldsymbol{\varphi}^T + \lambda\mathbf{D} \right)^{-1} \epsilon\boldsymbol{\varphi}$$

| $\lambda$ | 500 | 1000 | 2000 | 3000 | 5000 |
|-----------|------|------|------|------|------|
| PSNR [dB] | 30.31 | 30.46 | 30.57 | 30.52 | 30.50 |

**Table 1**. Impact of $\lambda$ parameter on side information quality. Average over test sequences, QP=31.
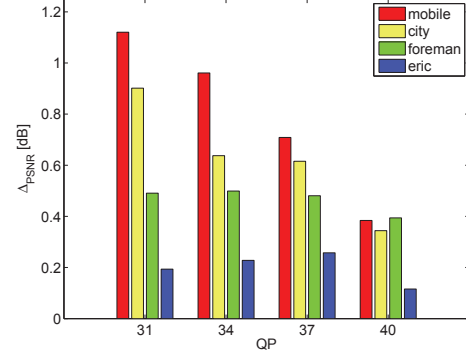


**Fig. 3**. SI PSNR improvement [dB] between reference and proposed method.

Using the matrix inversion lemma, we find the optimal refinement:

$$\delta\mathbf{v}^* = \frac{-\epsilon\mathbf{D}^{-1}\boldsymbol{\varphi}}{\lambda + \boldsymbol{\varphi}^T\mathbf{D}^{-1}\boldsymbol{\varphi}} \quad (5)$$

It is interesting to observe the similitude between the final formula and the original one in Eq. (2). Actually, Eq. (5) reduces to Eq. (2) in homogeneous regions or for very high values of the parameter $\sigma$.

## 4. EXPERIMENTAL RESULTS

In the experimental test we have used a set of 4 test sequences, characterized by different motion content: *city*, *eric*, *foreman*, and *mobile*. First, we have performed some experiments in order to tune the parameters $\lambda$, $\gamma$, and $\sigma$ of the proposed algorithm. We look for the parameter values maximizing the PSNR between the reconstructed and the original WZF. For the sake of brevity, here we show some results only for $\lambda$, see Tab. 1. We report the average PSNR of reconstructed WZF for different values of the parameter, averaged over the test sequences, and with KFs encoded at QP=31. Similar results were obtained for other quantization steps. We conclude that the best value for the parameter $\lambda$ is 2000, and in the following we consider only this value. Likewise, we have experimentally determined that the best values for the other parameters are $\gamma = 20$ and $\sigma = 50$. These values proved to be quite robust for several test sequences, and will be used in the following. Moreover we found that the order of parameters selection has a very small impact on their optimal values.

In order to evaluate the effectiveness of the proposed technique, we first compared the SI produced by our algorithm with the one produced by DISCOVER using out set of four input sequences. The parameter considered for the comparison was the PSNR between the original WZF and its estimation produced by each of the techniques.

The results of the first tests are summarized in Fig. 3. We note that for each sequence and for each KF's quantization step, our algorithm produces a SI more similar to the original WZF (in the sense of the PSNR). However the gain can be quite different according to the sequence. We obtain higher gain when there is high, regular motion

| GOP size | QP values | | | |
|---|---|---|---|---|
| | 31 | 34 | 37 | 40 |
| 2 | 0.68 | 0.58 | 0.52 | 0.31 |
| 4 | 0.38 | 0.33 | 0.28 | 0.22 |
| 8 | 0.23 | 0.23 | 0.22 | 0.18 |

**Table 2**. SI PSNR improvement [dB] of proposed method over reference for different GOP sizes, average over the test set.

| QP | 31 | 34 | 37 | 40 |
|---|---|---|---|---|
| $\Delta$ PSNR | 0.43 | 0.35 | 0.33 | 0.18 |

**Table 3**. SI PSNR improvement [dB] of proposed method over [8], average over the test set.

like in *mobile* and *city* (up to more than 1.1 dB). When the motion is less regular we have a bit smaller but still significant gain (up to about 0.5 dB for *foreman*). Finally, some gains are still obtained for the "head and shoulder" sequence *eric*, around 0.2 dB. We observe as well that the gain is generally smaller for severely quantized KFs: this is reasonable since low quality KFs provide a less reliable gradient information, which is at the basis of the proposed method.

These first experiments were conducted for a GOP size of 2, *i.e.* KFs are adjacent to WZFs. We repeated the same experiment for larger GOPs, and we found that the proposed technique is still better than the reference, even though the gap becomes smaller. The results of these tests are reported in Tab. 2. Even in the less favorable case the proposed method is almost 0.2dB better than the reference.

We performed a last set of experiments on the SI information quality, comparing the technique proposed in [8] and the new one. The results for GOP size equal to 2 are shown in Tab. 3. Similar results are obtained for other GOP sizes. The new method consistently outperforms the previous one, suggesting that the earlier the CRA is introduced into the processing chain, the better the results are.

The results presented so far confirm that the proposed technique produces better image interpolation than the references. However, the target of the method is to improve the global performance of the DVC scheme, so we computed the global RD performance of the scheme described in Section 2 for the sequences of the test set. The results were compared with those of the reference DISCOVER coder using the Bjontegard metric [16] (recommended by the VCEG for comparing video coding methods) at four operational points corresponding to QP $\in \{31, 34, 37, 40\}$. We observed an average rate reduction of 5.9% and an average PSNR improvement of 0.32 dB for the sequences of the test set. These results validate the proposed method, and, along with those shown in [8], confirm the intuition that differential ME methods can be profitably employed in the framework of DVC.

## 5. CONCLUSION AND FUTURE WORK

The idea of using differential methods for ME in the context of DVC has recently been introduced [8], showing promising results. In this paper we continue our study about the adaption of gradient-based ME methods into a popular DVC coding scheme as DISCOVER. The proposed ME technique allows for regular MVFs and the results in image interpolation are encouraging: w.r.t. the state of the art, we have observed gains up to 1.1dB for a complex motion sequence as *mobile*. However the proposed technique has better results than the reference for all the test sequences. The technique proposed here improves the results of [8] as well.

These results allow us concluding that differential ME techniques find a natural application in DVC, where the problem of an overwhelming coding cost for a dense MVF is completely circumvented by the fact that the motion vectors have not to be encoded, since they are computed at the decoder side. Future work will focus on further application of differential ME techniques and dense MVFs for DVC. In particular, on one hand we want to complete the analysis of the application of the CRA to the DVC paradigm, on the other we are also looking for other differential technique which could profitably be used in this context.

## 6. REFERENCES

[1] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

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

[3] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the receiver," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–11, Jan. 1976.

[4] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Asilomar Conference on Signals and Systems*, Pacific Grove, California, Nov. 2002.

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

[6] J. Ascenso, C. Brites, and F. Pereira, "Content adaptive Wyner-Ziv video coding driven by motion activity," in *Proceed. of IEEE Intern. Conf. Image Proc.*, Atlanta, GA (USA), July 2006, pp. 605–608.

[7] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, "Feedback channel suppression in distributed video coding with adaptive rate allocation and quantization for multi-user applications," *EURASIP J. Wireless Comm. Networking*, vol. Vol. 2008, 2008, Article ID 427247, 13 pages.

[8] M. Cagnazzo, T. Maugey, and B. Pesquet-Popescu, "A differential motion estimation method for image interpolation in distributed video coding," in *Proceed. of IEEE Intern. Conf. Acoust., Speech and Sign. Proc.*, Taipei, Taiwan, Apr. 2009.

[9] C. Cafforio and F. Rocca, "The differential method for image scene analysis," in *Image Sequence Processing and Dynamic Scene Analysis*, T. S. Huang, Ed. 1983, pp. 104–124, Springer-Verlag.

[10] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding: Basics, problems and recent advances," *IEEE Signal Processing Mag.*, pp. 67–76, Sept. 2007.

[11] H. Nagel and W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 5, pp. 565–593, Sept. 1986.

[12] T. Maugey, T. Andre, B. Pesquet-Popescu, and J. Farah, "Analysis of error propagation due to frame losses in a distributed video coding system," in *Proceed. of Europ. Sign. Proc. Conf.*, Lausanne, Switzerland, Aug. 2008.

[13] *Advanced video coding for generic audiovisual services*, ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), Version 1: May 2003, Version 8: Consented in July 2007.

[14] T. Maugey and B. Pesquet-Popescu, "Side information estimation and new schemes for multiview distributed video coding," *Elsevier J. Vis. Comm. and Image Repres.*, vol. 19, no. 8, pp. 589–599, Dec. 2008.

[15] L. Alvarez, J. Weickert, and J. Sánchez, "Reliable estimation of dense optical flow fields with large displacements," *Intern. J. Comput. Vision*, vol. 39, no. 1, pp. 41–56, Aug. 2000.

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