Multiple Description Distributed Video Coding Using Redundant Slices and Lossy Syndromes

Simone Milani and Giancarlo Calvagno

***Abstract*—During the last years, video coding designers have proposed robust coding approaches that combine Multiple Description Coding (MDC) schemes with Distributed Video Coding (DVC) principles. In this way, it is possible to obtain a better error resilience since the distortion drifting through the sequence is significantly mitigated by the DVC coding unit. The paper presents a Multiple Description Distributed Video Coder (MDDVC) that codes the input video signal generating a set of "lossy" syndromes for each pixel block and creates different descriptions multiplexing primary and redundant video packets. Experimental results show that at high loss probabilities the proposed solution improves the results of the original MDC approach.**

***Index Terms*—Multiple Description Coding, redundant slices, Distributed Video Coding, H.264/AVC, robust video transmission.**

I. Introduction

The provisioning of digital video services over wireless networks requires the implementation of appropriate strategies to deal with the presence of delays, packet losses and bandwidth limitations. Among the robust video coding architectures proposed in technical literature, Multiple Description Coding (MDC) schemes [1] have proved to be significantly more efficient with respect to traditional single description coding (SDC) architectures in contrasting the visual distortion caused by time-varying channel conditions.

The possibility of characterizing a single video source via multiple correlated data streams (*descriptions*) permits reconstructing the transmitted video sequence corrupted by packet losses since the missing information is estimated exploiting the correlation existing between the different streams.

During the last years, video coding designers have proposed alternative solutions that rely on the information-theoretic concept of Distributed Source Coding (named Distributed Video Coding or DVC schemes) [2]. DVC approaches permit an independent coding of the data at the transmitter, and a robust predictive decoding at the receiver, which can be performed correctly provided that at least a sufficiently-correlated predictor is available at the decoder [3].

Recent works have been focusing on combining these two techniques in Multiple Description Distributed Video Coders (MDDVC) since coding the prediction residual using DVC permits reducing the drifting problem in traditional MDC [4]–[7]. This paper proposes a novel MDDVC scheme that creates two descriptions including redundant slices in the video stream and adopts a hybrid DVC unit that processes the video signal both in the spatial and in the temporal domain. More precisely, the set of syndromes is generated in the pixel domain and compressed in the transform domain. In this way, the distributed source coding operations are separated from signal compression. Processing the signal in the pixel domain permits handling the temporal and spatial correlation more efficiently, while transform coding permits reducing the size of the coded bit stream. Experimental results will show that the proposed approach permits improving the PSNR value of the reconstructed sequence up to 1 dB with respect to the original MDC scheme.

In the following, Section II provides an overview of the implemented codec, while Section III describes the Distributed Source Coding (DSC) scheme that has been adopted to characterize the prediction residual signal. Experimental results in Section IV show how the proposed MDDVC approach improves the quality of the coded sequences at high loss rates. Conclusions are drawn in Section V.

II. The proposed MDC architecture

The MDDVC schemes proposed in literature have been derived replacing the traditional temporally-predictive video coder of traditional MDC approaches with a Wyner-Ziv video coding scheme (see [4] as an example). Similarly, the proposed MDDVC solution has been designed starting from the MDC scheme presented in [8], which has been developed using the concepts of *primary* and *redundant* slices defined within the H.264 standard [9]. Primary slices are used to code the primary picture and are associated to a normative decoding procedure. Redundant slices present an alternative representation of the image which is employed whenever the primary signal can not be correctly decoded. In our approach (like in [8]), redundant slices are characterized by the adoption of a higher Quantization Parameter (QP) with respect to the primary ones.

Figures 1 and 2 show the block diagrams of the employed encoder and decoder, which have been developed using the building blocks of the H.264/AVC coder. The adopted scheme is derived from the scheme of the codec in [8] made exception for the fact that the traditional residual coding unit is replaced by a syndrome coding unit (see Fig. 1), which is based on the principle of Wyner-Ziv coding. Note that in the proposed MDDVC approach, side information is estimated by the encoder via motion-search, and therefore, the computational

S. Milani and G. Calvagno are with the Dept. of Information Engineering, University of Padova, via Gradenigo 6/B, 35131 Padova - Italy, phone: + 39 049 827 7641, fax: +39 049 827 7699, e-mail: {simone.milani,calvagno}@dei.unipd.it

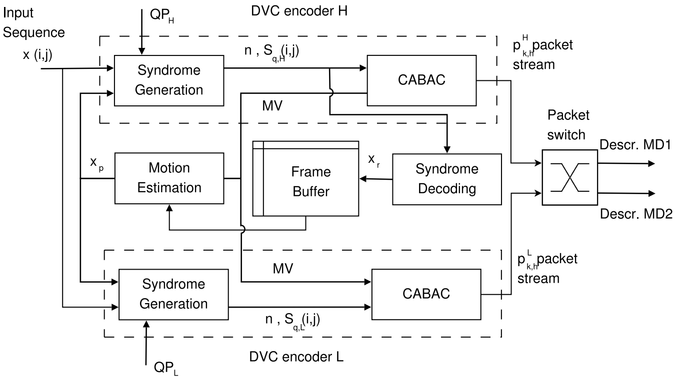


Fig. 1. Block diagram of the proposed encoder.

complexity is close to that of the original MDC scheme (see Section IV). Nevertheless, the adopted DVC paradigm permits mitigating the channel distortion produced by packet losses on the reconstructed video sequence.

The input video signal is partitioned into macroblocks (MBs) of 16×16 pixels, which are temporally predicted from the reference pictures stored in the frame buffer. Then, each 4×4 block within the input MB is coded using a Distributed Source Coding paradigm (described in Section III) permitting the decoder to reconstruct the video sequence at the quality level related to a given quantization parameter QP_H. As a result, the MDDVC coder generates the primary RTP packet $p^H_{k,h}$, which codes the kth rows of macroblocks (one slice) within the hth frame. Then, the proposed approach generates a redundant packet $p^L_{k,h}$, which codes the input MB with a quantization parameter $QP_L > QP_H$ and permits reconstructing the transmitted sequences at a lower quality level. Note that the predictor block is the same since the motion estimation is performed using references from the same frame buffer. Further details will be given in Section III. The generated $p^L_{k,h}$ and $p^H_{k,h}$ packets are then multiplexed in two descriptions that are sent to the receiver via two separate channels. More precisely, one description (MD1) includes odd high-quality packets $p^H_{k,2m+1}$ and even low-quality packets $p^L_{k,2m}$ (with $m = 0, \dots, 8$ for CIF sequences), while the other description (MD2) is made of even high-quality packets $p^H_{k,2m}$ and odd low-quality packets $p^L_{k,2m+1}$ (see [8] for more details).

At the decoder, the transmitted video signal is reconstructed using $p^H_{k,h}$ packet whenever it is available. In case $p^H_{k,h}$ is missing, its corresponding low-quality counterpart $p^L_{k,h}$ is used decoding the input sequence at a lower quality. In case both $p^H_{k,h}$ and $p^L_{k,h}$ are missing, the lost signal is estimated using an error concealment algorithm (in our case, copying the corresponding macroblocks from the previous frame). In the following we will describe how syndromes are generated.

III. Syndrome generation and coding

In the technical literature DVC approaches can be divided into schemes that process the video signal in the pixel domain [10] and schemes that operate in the transform domain [2], [11]. In our work, we adopted a hybrid pixel-transform domain DVC scheme that generates syndromes in the pixel domain and compresses them using a transform coding paradigm. The

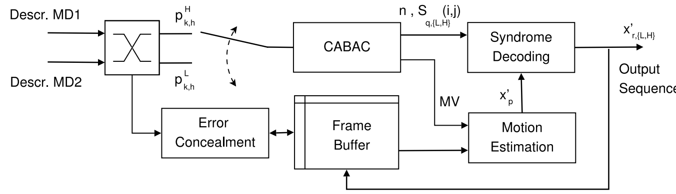


Fig. 2. Block diagram of decoder.

current section will describe how the DVC encoder has been implemented in the proposed MDDVC architecture.

Given the current 4×4 block $\mathbf{x}$ of pixels and its predictor $\mathbf{x}_p$, for each pixel $x(i,j)$ of block $\mathbf{x}$ at position (i,j), $i,j = 0, \dots, 3$, we compute the number of bits $n(i,j)$ as

$$n(i,j) = \begin{cases} \lfloor \log_2 |d(i,j)| \rfloor + 2 & \text{if } |d(i,j)| > \delta \\ 0 & \text{otherwise,} \end{cases} \qquad (1)$$

where $d(i,j) = x(i,j) - x_p(i,j)$ and δ is a threshold value depending on the Quantization Parameter (QP) chosen for the current block (in our implementation, we have set $\delta = \Delta/12$ where Δ is the quantization step associated to the current QP). Then, the coding unit computes the maximum value

$$n = \max_{i,j=0,\dots,3} n(i,j) \qquad (2)$$

within the current block and, in case n is higher than 0, it generates a block of syndromes $s(i,j)$ via the following equation

$$s(i,j) = x(i,j) \& (2^n - 1) \qquad (3)$$

where the symbol & denotes a bitwise AND operator. In this way, the n least significant bits of each pixel are selected to generate the block $\mathbf{s}$. The value $s(i,j)$ is called "syndrome" (as in [11]), while the parameter n is the number of syndrome bits. The block $\mathbf{s}$ is then transformed via the 4×4 H.264/AVC integer DCT into the block $\mathbf{S}$. At this stage, the block $\mathbf{S}$ is quantized using the quantization parameter QP_H at the DVC encoder H (see the block diagram in Fig. 1) and generating the block of coefficients $\mathbf{S}_{q,H}$, while the DVC encoder L employs the quantization parameter $QP_L > QP_H$ creating the coefficients $\mathbf{S}_{q,L}$. The coding operations that transform $\mathbf{s}$ into $\mathbf{S}_q$ can be associated to the `Intra4x4` coding mode of H.264/AVC standard [9] with no spatial prediction.

The blocks $\mathbf{S}_{q,H}$ and $\mathbf{S}_{q,L}$ are then coded in the packets $p^H(k,h)$ and $p^L(k,h)$ respectively, together with the Motion Vectors. The block $\mathbf{S}_{q,H}$ is then dequantized, and inversely-transformed into the block $\mathbf{s}_{r,H}$. The syndrome $s_{r,H}(i,j)$ (as well as its counterpart $s_{r,L}(i,j)$ which can be obtained dequantizing and inversely-transforming the coefficients $\mathbf{S}_{q,L}$) is a lossy version of the original syndrome $s(i,j)$ such that $s_{r,H}(i,j) = s(i,j) + e_H(i,j)$, where $e_H(i,j)$ is the distortion introduced by the quantization in the transform domain. Each lossy[1] syndrome $s_{r,H}(i,j)$ identifies a different quantizer $Q_{s_{r,H}(i,j)}$ with quantization step 2^n and offset $s_{r,H}(i,j)$ such that the reconstruction levels for the adopted quantizer can be written as $s_{r,H}(i,j) + k\ 2^n$, where $k \in \mathbb{Z}$ is the associated index of the quantized value. Given the

[1]Here the term *lossy* in relation to the syndromes refers to the lossy coding scheme adopted to characterize them.

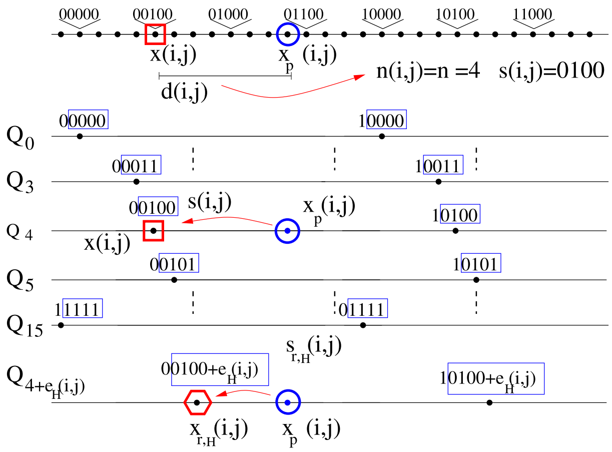


Fig. 3. Example of syndrome generation and decoding. Note that using the original syndrome $s(i,j)$ leads to a lossless reconstruction of $x(i,j)$. The proposed scheme transforms and quantizes $s(i,j)$, and as a consequence, the final quantizer is $Q_{4+e_H(i,j)}$ in place of Q_4.

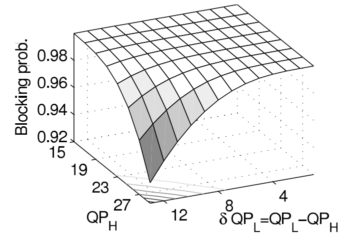


Fig. 4. Probability of blocking the error drifting for the sequence `foreman` at different quantization settings. The blocking probability is reported as a function of QP_H and $\delta QP_L = QP_L - QP_H$.

predictor block $\mathbf{x}_p$, it is possible to reconstruct the coded pixel $x_{r,H}(i,j) = x(i,j) + e_H(i,j)$ by quantizing $x_p(i,j)$ using the quantizer characteristics associated to $s_{r,H}(i,j)$. As an example, Figure 3 shows the syndrome generation and reconstruction processes for the pixel $x(i,j)$. According to the distance $|d(i,j)| = 7$ between $x(i,j)$ and $x_p(i,j)$, the syndrome $s(i,j) = 0100$ (associated to Q_4) is found from $n = \lfloor \log_2 |d(i,j)| \rfloor + 2 = 4$ least significant bits of $x(i,j)$. After coding $s(i,j)$ into the syndrome $s_{r,H}(i,j)$ (associated to $Q_{4+e_H(i,j)}$), it is possible to reconstruct a distorted version of $x(i,j)$ by quantizing $x_p(i,j)$ with $Q_{4+e_H(i,j)}$.

The syndrome generation process is based on a nested scalar quantization (NSQ) and differs from other NSQ-based approaches (like [11]) for the facts that it codes syndromes with a lossy technique and operates in the pixel domain. In this way, it is possible to separate the channel coding operations (related to the estimation of n) and the compression of the signal (related to the quantization) permitting a better optimization of the DVC coder. Moreover, the adoption of a NSQ-based approach in syndrome generation permits reducing significantly the computational complexity at the decoder with respect to approaches based on Turbo/LDPC codes, which require longer block lengths (see [11] for more details).

Whenever the packet $p_{k,h}^H$ is not available, it is possible to decode the corresponding slice using the syndromes $s_{r,L}(i,j)$ from the packet $p_{k,h}^L$ included in the other description provided that it has been correctly received. The syndrome $s_{r,L}(i,j) = s(i,j) + e_L(i,j)$ permits reconstructing the coded video sequence with a quantization error $e_L(i,j) > e_H(i,j)$, but the adoption of the DVC coding paradigm permits mitigating the distortion drifting to the other frames. Note that a correct decoding is possible even using a different predictor $x'_p(i,j) \neq x_p(i,j)$ provided that the correlation between $\mathbf{x}$ and $\mathbf{x}'_p$ is the same or higher (i.e. the difference $d'(i,j) = x(i,j) - x'_p(i,j)$ in eq. (1) leads to a value $n'(i,j) \leq n(i,j)$). In the MDDVC decoder a different predictor is found in case the previous frames have been corrupted by some packet loss, and as a consequence, the reconstructed frames are affected by a certain amount of distortion depending either on the adoption of $QP_L > QP_H$ in the second DVC coder or on the error concealment algorithm. In the first case, the predictor pixel $x_p(i,j)$ is replaced by its corrupted version $x'_p(i,j) = x_p(i,j) - e_{p,H}(i,j) + e_{p,L}(i,j)$ (where $e_{p,H}(i,j)$ and $e_{p,L}(i,j)$ are the quantization errors for the predictor pixel $x_p(i,j)$ obtained with QP_H and QP_L respectively). Assuming that the difference $d(i,j)$ between $x(i,j)$ and $x_p(i,j)$ and the difference $e_t(i,j) = e_{p,L}(i,j) - e_{p,H}(i,j)$ can be approximated by Gaussian variables with zero mean and variances σ_d^2 and σ_t^2 respectively,[2] the probability of blocking the distortion drifting (i.e. decoding correctly the current pixel $x_r(i,j)$) is given by

$$P[|d'| < 2^{n-1}] = P[|d + e_t| < 2^{n-1}] \simeq erf\left(\frac{2^{n-1}}{(\sigma_t + \sigma_d)\sqrt{2}}\right). \quad (4)$$

The variance of $e_t(i,j)$ can be computed in the transform domain (see [8]) and can be approximated as

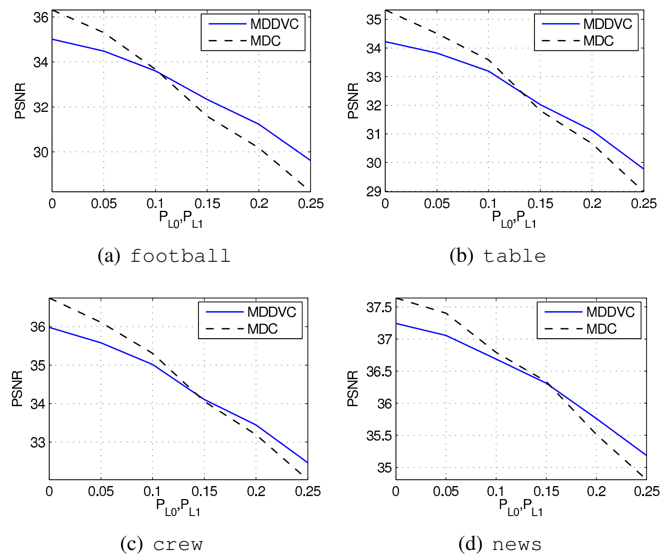
$$\sigma_t^2 = \frac{\Delta_L^2}{12}\left[1 - \left(\frac{\Delta_H}{\Delta_L}\right)^2\right] \quad (5)$$

where Δ_H and Δ_L are the quantization steps corresponding to the quantization parameters QP_H and QP_L respectively. Replacing the parameters in eq. (5) and (4) with values derived from the characteristics of the coded sequence and the quantization setting suggested in [8] shows that the proposed scheme permits reducing the error drifting probability to about 1% (see Fig. 4). More details will be given in the following section.

IV. Experimental results

The proposed DSC coder has been tested simulating the transmission of different video sequences in a scenario where the two descriptions are transmitted over two independent lossy channels C_1 (associated to MD1) and C_2 (associated to MD2) simulated via a Gilbert two-state model with burst length $L_B = 4$. The loss probabilities P_{L1} and $P_{L2} = P_{L1}$ of C_1 and C_2 vary within the range $[0.05, 0.25]$ with steps of 0.05. In our tests we coded different CIF sequences at different QPs with GOP structure IPPP, slices of 22 macroblocks, and

[2]More precisely, the distortion $e_t(i,j)$ in the transform domain can be modelled by a uniform random variable as described in [8]. After the inverse transformation, the quantization error in the pixel domain is shaped by the coefficients of the inverse transform, and as a matter of fact, we approximate the statistical model of the quantization error in the pixel domain with a Gaussian variable.



(a) `football` (b) `table`

(c) `crew` (d) `news`

Fig. 5. Experimental results for MDDVC and MDC algorithms with $QP_H = 28$ and $QP_L = 26$. The graphs report the average values of average PSNR for the reconstructed sequences vs. the packet loss probability P_{Ld}, $d = 1, 2$.

CABAC entropy coding. The adopted rate-distortion optimization strategy is the one defined within the JVT for the H.264/AVC coder, and the relation between the quantization parameters is $QP_L = QP_H + 6$, which permits obtaining a rate increment around 30% with respect to a single description approach.

The plots in Figure 5 report the PSNR values averaged over 10 realizations for each channel C_d given its loss probability P_{Ld}, $d = 1, 2$. In each plot we compare the proposed MDDVC algorithm with the traditional MDC scheme (adopted in [8]). It is possible to notice that for most of the sequences the MDDVC scheme improves the quality of the reconstructed sequences whenever the loss probability becomes significant. As an example, Fig. 5(a) shows that for $P_{Ld} > 0.1$, $d = 1, 2$, the PSNR value for the MDC approach is always lower with respect to the proposed coder (the difference is approximately 1 dB for $P_{Ld} = 0.2$). Note that at low loss percentages the performance of the MDDVC coder is either the same or worse than the performance of the MDC approach. This fact is partly due to the adopted optimization algorithms, whose parameters are optimized and tuned for the H.264/AVC coder, and partly due to the loss in compression gain which is proper to the DVC approach. However, MDC approaches proves to be competitive with other SDC technique at high loss rates, and therefore, the improvement offered by the MDDVC approach motivates its adoption.

The same behavior can be noticed for other sequences too (see Fig. 5(b) and 5(c)), even if the crossing point between the MDC and the MDDVC plots depends on the characteristics of the video sequences. As a matter of fact, for the sequence `news` (see Fig. 5(d)) the high correlation and the low amount of motion allow the error concealment procedure to perform quite well, and as a matter of fact, the PSNR values of the traditional MDC scheme are always quite close to those of the MDDVC scheme. Therefore, an adaptive algorithm that is able to switch from the MDC setting to the MDDVC setting depending on the characteristics of the video sequence and on the estimated channel loss probability proves to be an interesting topic for future investigations.

TABLE I
ENCODING AND DECODING TIME INCREMENT FOR MDDVC

Sequence	**Encoding Time Increment (%)**	**Decoding Time Increment (%)**
`football`	$+0.73$	$+0.51$
`crew`	$+0.57$	$+0.37$

As for the complexity of encoding and decoding, Table I reports the relative increment of computational load with respect to the traditional MDC scheme in [8] (measured via the total encoding and decoding times). No significant increases can be noticed since the adoption of motion search at the encoder equalizes the amount of required calculation for the MDDVC scheme with respect to its MDC predecessor. Despite this, the proposed solution permits improving the robustness of the video transmission at high loss rates thanks to its effectiveness in blocking the distortion drifting.

V. CONCLUSION

The paper proposed an MDDVC scheme that employs a DVC coder based on lossy syndromes and generates two descriptions multiplexing primary and redundant packets defined in the H.264/AVC standard. The proposed scheme outperforms its traditional MDC counterpart (up to 1 dB in PSNR) at high loss rates thanks to the effectiveness of DVC in blocking the distortion drifting through the sequence. Future work will be devoted to extend the designed approach switching adaptively between the MDC and the MDDVC approaches according to the channels conditions and the characteristics of the sequence.

REFERENCES

[1] V. K. Goyal, “Multiple Description Coding: Compression Meets The Network,” *IEEE Signal Process. Mag.*, vol. 8, no. 5, pp. 74–93, Sep. 2001.

[2] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, “Distributed Video Coding,” *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[3] A. Sehgal, A. Jagmohan, and N. Ahuja, “Wyner-ziv coding of video: an error-resilient compression framework,” *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 249 – 258, Apr. 2004.

[4] J. Wang, X. Wu, S. Yu, and J. Sun, “Multiple Descriptions in the Wyner-Ziv Setting,” in *Proc. of IEEE ISIT 2006*, Seattle, WA, USA, Jun. 9 – 14, 2006, pp. 1584 – 1588.

[5] O. Crave, C. Guillemot, B. Pesquet-Popescu, and C. Tillier, “Multiple Description Source Coding with Side Information,” in *Proc. of EUSIPCO 2008*, Lausanne, Switzerland, Aug. 25 – 29 2008.

[6] Y. Fan, J. Wang, J. Sun, P. Wang, and S. Yu, “A Novel Multiple Description Video Codec Based on Slepian-Wolf Coding,” in *Proc. of DCC 2008*, Snowbird, UT, USA, Mar. 25 – 27, 2003, p. 515.

[7] A. Wang, Y. Zhao, and H. Bai, “Robust multiple description distributed video coding using optimized zero-padding,” *Sci China Ser F-Inf Sci*, vol. 52, no. 2, pp. 206 – 214, Feb. 2009.

[8] T. Tillo, M. Grangetto, and G. Olmo, “Redundant slice optimal allocation for H.264 multiple description coding,” *IEEE Trans. on CSVT*, vol. 18, no. 1, pp. 59 – 70, Jan. 2008.

[9] T. Wiegand, “Version 3 of H.264/AVC,” in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6), 12^{th} Meeting*, Redmond, WA, USA, Jul. 17 – 23, 2004.

[10] C. Brites, J. Ascenso, and F. Pereira, “Modeling Correlation Noise Statistics at Decoder for Pixel Based Wyner-Ziv Video Coding,” in *Proc. of PCS 2006*, Beijing, China, Apr. 24 – 26, 2006.

[11] R. Puri, A. Majumdar, and K. Ramchandran, “PRISM: A Video Coding Paradigm With Motion Estimation at the Decoder,” *IEEE Trans. Image Process.*, vol. 16, no. 10, pp. 2436 – 2448, Oct. 2007.