Signal Processing: *Image Communication* 23 (2008) 391–403

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Rate allocation for robust video streaming based on distributed video coding ☆

R. Bernardini [a], M. Naccari [b,*], R. Rinaldo [a], M. Tagliasacchi [b], S. Tubaro [b], P. Zontone [a]

[a] *Università degli Studi di Udine, Italy*
[b] *Dipartimento di Elettrocnica e Informazione, Politecnico di Milano, via Ponzio 34/5, 20133 Milano, Italy*

ARTICLE INFO

Article history:
Received 4 April 2008
Accepted 4 April 2008

Keywords:
Error resilience
Distributed video coding
Rate allocation
Channel induced distortion estimation

ABSTRACT

This paper proposes an error resilient coding scheme that employs distributed video coding tools. A bitstream, produced by any standard motion-compensated predictive codec (MPEG-x, H.26x), is sent over an error-prone channel. In addition, a Wyner–Ziv encoded auxiliary bitstream is sent as redundant information to serve as a forward error correction code. At the decoder side, error concealed reconstructed frames are used as side information by the Wyner–Ziv decoder, and the corrected frame is used as a reference by future frames, thus reducing drift. We explicitly target the problem of rate allocation at the encoder side, by estimating the channel induced distortion in the transform domain. Rate adaptivity is achieved at the frame, subband and bitplane granularity. Experimental results conducted over a simulated error-prone channel reveal that the proposed scheme has comparable or better performance than a scheme where forward error correction codes are used. Moreover the proposed solution shows good performance when compared to a scheme that uses the intra-macroblock refresh procedure.

1. Introduction

The transmission of coded bitstreams through noisy channels is a very challenging task. The error control in video coding is further complicated by the predictive nature of modern video coders [34]. If the encoder and the decoder are out of synchronization, e.g., because of packet losses, current errors will propagate along the time, corrupting the future correctly received data.

In a video streaming scenario, encoded bitstreams are sent to multiple receivers over a packet switched network; when the network does not provide any QoS, the encoded data is protected by adding some redundancy bits to ensure correct decoding even if some packets are lost during transmission. This method is known as forward error protection (FEP) and exploits techniques provided by research in the channel coding field.

In this paper, we propose a FEP coding scheme that employs an auxiliary redundant stream encoded according to a Wyner–Ziv (WZ) video coding approach, to protect a primary stream encoded with any motion-compensated predictive (MCP) codec. We work in the transform domain, by protecting the most significant bitplanes of the discrete cosine transform (DCT) coefficients. We use LDPC codes to compute the syndrome bits of the auxiliary stream. In order to allocate the appropriate number of syndrome bits, we introduce a modified version of the ROPE algorithm (recursive optimal per-pixel estimate of end-to-end distortion) [42], that works in the DCT domain. The proposed EDDD algorithm (expected distortion of decoded DCT coefficients) provides an estimate of the channel induced distortion for

☆ As far as the authors from the Politecnico di Milano is concerned, the work here presented has been developed within VISNET II, a European Network of Excellence (http://www.visnet-noe.org), funded under the European Commission IST FP6 programme. This work was presented in part in Refs. [7,2].

* Corresponding author. Tel.: +39 0313327619.
E-mail address: naccari@elet.polimi.it (M. Naccari).

0923-5965/$ - see front matter © 2008 Published by Elsevier B.V.
doi:10.1016/j.image.2008.04.004

each frame and DCT subband. This information is then used to determine the number of syndrome bits to be produced by the WZ encoder. At the receiver, the primary stream is decoded and motion-compensated error concealment is applied. The concealed reconstructed frame is used as side information by the WZ decoder, which performs LDPC decoding based on the received syndrome bits. We also show how prior information that can be obtained at the decoder, based on the observed error pattern, can be used to efficiently help LDPC decoding.

In order to assess the performance of the proposed scheme in comparison with well established error protection schemes, we compare the proposed solution with one where FEC codes are used. Moreover, we consider the use of the intra-macroblock refresh procedure provided as a non-normative tool in the standard H.264/AVC [10]. Experimental results show that the proposed scheme has comparable or better performance, especially at high packet loss probability, than a scheme using FEC codes. One possible advantage of the proposed solution, is that it naturally allows for rate adaptivity and unequal error protection (UEP) achieved at the frame, DCT band and bitplane granularity. In addition, the proposed scheme outperforms the use of an intra-macroblock refresh procedure. The latter requires to be applied either at encoding time, or to transcode a pre-encoded bitstream to perform mode switching. Conversely, in our scheme, we consider a pre-encoded sequence and simply add WZ bits for protection, maintaining the original bitstream unaltered. Also, the proposed scheme is independent from the actual codec adopted for the primary stream, provided that the ROPE algorithm is properly adjusted to work with the desired codec [39].

The remainder of the paper is organized as follows: in Section 2 we thoroughly review the state-of-the-art in error resilient video coding based on distributed video coding principles. Section 3 introduces the system architecture proposed in this paper. Section 4 illustrates the channel induced distortion estimation algorithm adopted in our scheme. Sections 5 and 6 consider, respectively, WZ encoding and decoding, including the rate allocation algorithm adopted at the encoder. Section 7 presents some experimental results obtained with real video sequences and compares the proposed scheme with two other schemes: the first one uses FEC codes, while the second one performs error resilient transcoding by means of intra-macroblock refresh. Finally, Section 8 concludes the paper.

2. Related work

Most of the literature on distributed video coding has addressed the problem of light encoding complexity, by shifting the computationally intensive task of motion estimation from the encoder to the decoder. Nevertheless, distributed video coding principles have been extensively applied in the field of robust video transmission over unreliable channels. One of the earliest examples is given by the PRISM coding framework [22], which simultaneously achieves light encoding complexity and robustness to channel losses. In PRISM, each block is encoded without the deterministic knowledge of its motion-compensated predictor, that is made available at the decoder side only. Therefore, the encoder determines the number of cosets to be encoded based on an estimate of the statistical correlation between the block to be encoded and its motion-compensated predictor [15,21]. If the predictor obtained at the decoder is within the noise margin for the number of encoded cosets, the block is successfully decoded. The underlying idea is that, by adjusting the number of cosets based on the expected correlation channel, decoding is successfully achieved even if the motion compensated predictor is noisy, e.g., due to packet losses affecting the reference frame. When video transmission suffers from packet losses, a significant gain is shown with respect to H.263 + protected with forward error correction (FEC) techniques and/or intra-macroblock refresh [21,20]. This is primarily due to the lack of the prediction loop that characterizes conventional MCP codecs. Therefore, the PRISM coding framework allows to mitigate the effect of drift, which arises when the encoder and the decoder are out of sync, e.g., because of packet losses. These results were extended to a fully scalable video coding scheme in [17,30,31], which is shown to be robust to losses that affect both the enhancement and the base layer. This is due to the fact that the correlation channel that characterizes the dependency between different scalability layers is captured at the encoder in a statistical, rather than deterministic, way. In [20,16] it is shown that there is an intrinsic trade-off between complexity and robustness. In other words, by shifting part of the computational burden from the encoder to the decoder, it is possible to increase the robustness of the system. Experimental results consider both the "uplink" (light encoding complexity) and "downlink" (light decoding complexity) versions of the PRISM codec, showing that the former is highly robust, while the latter matches the coding efficiency of H.263+ in a noiseless scenario. Also based on the PRISM framework, robust multi-view video coding has been recently proposed in [40,41].

Despite PRISM, most of the distributed video coding schemes that focus on error resilience try to increase the robustness of standard encoded video by adding redundant information encoded according to distributed video coding principles. One of the first works along this direction is presented in [28], where auxiliary data is encoded only for some frames, denoted as "peg" frames, in order to stop drift propagation at the decoder. The idea is to achieve the robustness of intra-refresh frames, without the rate overhead due to intra-frame coding. This is accomplished by periodically transmitting a small amount of additional information, termed *coset information*, to the decoder. The coset information is capable of correcting errors even if the encoder does not have a precise knowledge of which packets will be lost. In [28] LDPC codes are applied to bitplanes of DCT coefficients to compute parity bits that are used as coset information. In [38], a layered WZ video coding framework similar to fine

granularity scalability (FGS) coding is proposed, in the sense that it treats the standard coded video as the base layer and generates an embedded bitstream as the enhancement layer. However, the key difference between conventional FGS and [38] is that, instead of coding the difference between the original video and the base layer reconstruction, the enhancement layer is generated "blindly", without knowing the base layer. Therefore, the stringent requirement of FGS coding that the base layer is always available losslessly at the decoder/receiver can be loosened, since an error-concealed version of the base layer can still be used in the joint WZ decoder. Therefore, besides providing SNR scalability, the scheme proposed in [38] is robust with respect to channel losses. Neither [28] nor [38] give explicit indication about how to allocate the redundancy bits across bitplanes and/or DCT coefficients. Although the encoder does not know the exact realization of the reconstructed frame, it can try to characterize the effect of channel errors (i.e., packet losses) in statistical terms, in order to perform optimal bit allocation. This idea has been pursued in [18,19,8], where a PRISM-like auxiliary stream is encoded for FEP, and rate-allocation is performed at the encoder by exploiting the information provided by the ROPE algorithm. The latter is able to estimate the distortion expected at the decoder due to packet losses. While [18] uses a rather coarse distortion estimation algorithm, a more accurate solution is proposed in [8].

Distributed video coding has been applied to error-resilient MPEG-2 video broadcasting in [24,25], where a systematic lossy source channel coding framework (SLEP, systematic lossy error protection) is proposed. An MPEG-2 video bitstream is transmitted over an error-prone channel without error protection. In addition, a supplementary bitstream is generated using distributed video coding tools, which consists of a coarsely quantized video bitstream obtained using a conventional hybrid video coder, applying Reed–Solomon codes, and transmitting only the parity symbols. In the event of channel errors, the decoder decodes these parity symbols using the error-prone conventionally decoded MPEG-2 video sequence as side information. The SLEP scheme has also been extended to the H.264/AVC video coding standard [26]. The redundant slices feature can be used to generate redundant video descriptions on which Reed Solomon Slepian–Wolf coding can be applied. Using flexible macroblock ordering (FMO), a region-of interest (ROI) can be selected in each video frame. The SLEP scheme can be applied preferentially on the ROI, while allowing the less important background to be protected by conventional decoder-based error concealment schemes. Based on the SLEP framework, the scheme proposed in [13] performs UEP assigning different amounts of parity bits between motion information and transform coefficients. This work has been further improved in [14] where the number of parity bits for both motion information and transform coefficients is adaptively tuned based on the content of each frame. The parity bitrate adaptation is made comparing the sum of absolute differences (SAD) of each frame with some experimentally obtained thresholds. This approach shares some similarities with the one presented in this paper, since they both use WZ coding in the transform domain to protect the primary video bitstream. With respect to [14], we propose a more sophisticated rate allocation algorithm, based on the estimated induced channel distortion. Therefore, we provide a finer granularity in performing UEP, not only at the frame level, but also at the subband and bitplane level.

The WZ video codec described in [9] has been extensively studied in the literature, especially when targeting low encoding complexity and multi-view coding. In this codec scheme, a feedback channel is used in order to obtain the rate flexibility in adapting to the changing statistics between the side information and the frame to be encoded. The error resilient properties of an improved version of the same codec [4] have been recently investigated in [32,23]. In [32] the key frames are encoded with an H.264/AVC encoder enabling the FMO tool and using the concealment algorithms implemented in the decoder reference software. Channel redundancy is added to the key frames bitstream by the means of Reed–Solomon codes, while the parity bits relative to the WZ frames are supposed to be sent unprotected. The reported results suggest that, at low-to-medium packet loss rates (PLRs) (typically $\leqslant 10\%$), a conventional H.264/AVC bitstream protected with Reed–Solomon codes outperforms the WZ codec. Conversely, at higher PLRs (i.e., $\geqslant 15\%$), the WZ codec gives better performance, especially at high bitrates. The work in [23] considers packet losses afflicting both the key frame and the WZ frames. Experimental results show that the codec in [4] is highly sensitive to errors in the key frames, while it provides graceful degradation when WZ packets get lost.

Distributed video coding principles have also been applied in the area of robust image coding. As an example, [6] proposes an algorithm that exploits turbo codes for sources with memory. This way, it is possible to take into account the statistical dependency within and across bitplanes, conventionally addressed by context based arithmetic coding in JPEG2000. The scheme is shown to be highly robust when bitwise errors affect the bitstream.

3. Error resilient scheme architecture

The overall architecture of the proposed error resilient scheme is depicted in Fig. 1. The operations performed, respectively, at the transmitter and at the receiver side are summarized in Algorithms 1 and 2.

Algorithm 1. Operations performed at the transmitter side
Require: Coded bitstream
Require $n = 0$
1: **for** each frame X_n **do**
2: Decode X_n up to spatial domain
3: Use the reconstructed frame $\hat{X}_n$, the motion vectors and the mode decisions to estimate the auxiliary bitstream rate of j-th bitplane of the l-th DCT subband: $R_n^{l,j}$
4: Calculate the syndrome bits for $\hat{X}_n$
5: Send the coded data and syndrome bits packets through the noisy channel
6: $n = n + 1$
7: **end for**

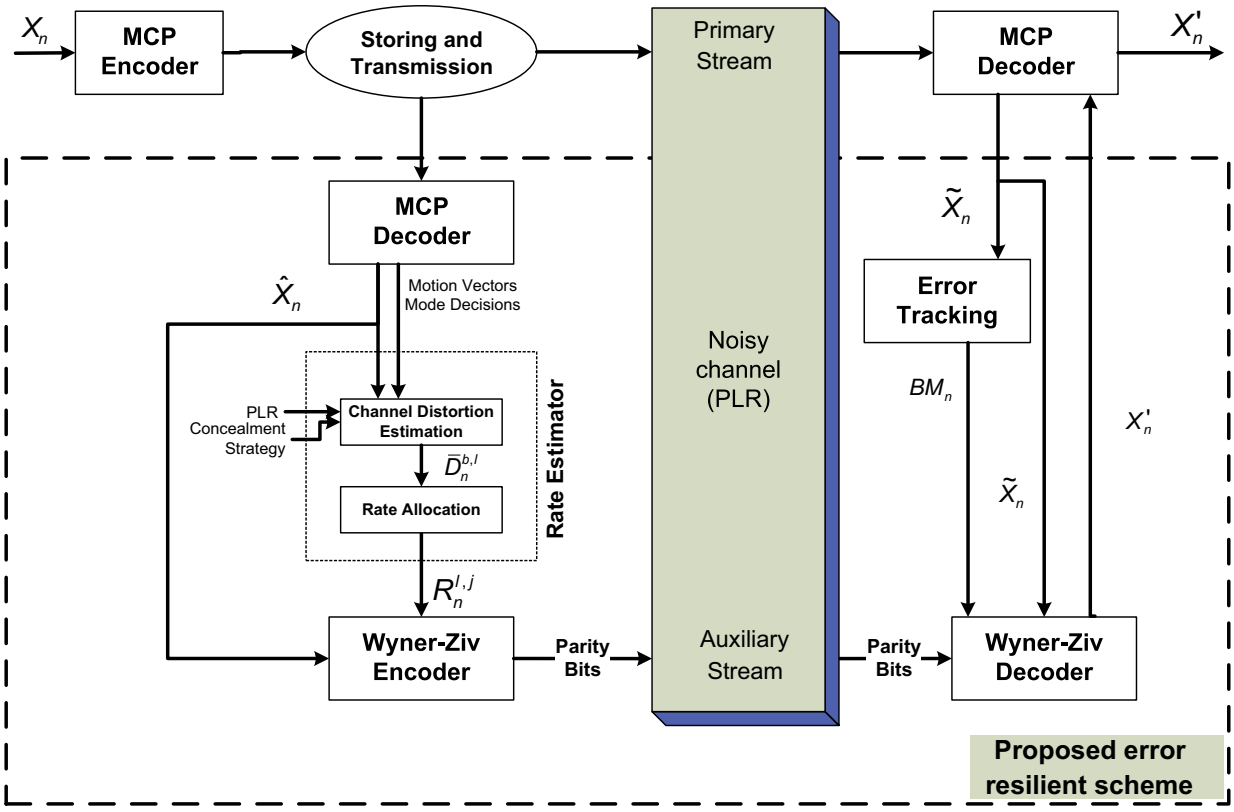

Fig. 1. Block diagram of the proposed error resilient scheme. Refer to Algorithms 1 and 2.

Algorithm 2. Operations performed at the receiver side
Require $n = 0$
for each frame X_n **do**
  Decode the received packets
  Apply the error tracking algorithm to produce the binary map BM_n
  if at least one packet of the current frame has been lost **then**
    Apply motion-compensated error concealment
  end if
  if at least one block in the binary map is flagged as "noisy" or "potentially noisy" **then**
    Feed the concealed frame $\tilde{X}_n$ and the binary map BM_n into the Wyner–Ziv decoder
    Use the received syndrome bits to correct $\tilde{X}_n$ into X'_n
    Insert X'_n into the reference frame buffer of the motion-compensated predictive decoder
  end if
  $n = n + 1$
end for

The rate estimator module comprises two sub-modules: the channel induced distortion estimation and the rate allocation. The former provides the expected distortion $\bar{D}_n^{b,l}$ due to channel losses for each DCT coefficient l in each block b in frame n. The WZ encoder works in the DCT domain and it is similar to the one described in [1], although LDPC codes are used instead of turbo codes. The reconstructed frame $\hat{X}_n$ is divided into blocks of size 8×8 pixels and the DCT transform is applied to these blocks. The transform coefficients are grouped together to form subbands. Each subband is then quantized and the corresponding bitplanes are independently coded using an LDPC encoder. Experimental evidence shows that channel noise can be well approximated by means of a Laplacian distribution. Exploiting the Laplacian model, the rate $R_n^{l,j}$ of the j-th bitplane of the l-th DCT subband can be obtained as described in Section 5. A packet switched network is assumed, characterized by a PLR equal to p.

At the receiver side the MCP decoder decodes the correctly received packets. If any packet has been dropped, the decoder applies the error concealment algorithm over lost data to produce the frame $\tilde{X}_n$. At the same time, the error tracking (ET) module flags the concealed blocks as "noisy" while those blocks that depend (e.g., by motion compensation) on concealed data in previous frames as "potentially noisy". These flags are organized into the Binary Map BM_n. Finally, the WZ decoder works in the transform domain and uses the received syndrome bits as well as the prior information of the binary map BM_n to reconstruct a cleaner version of the frame, X'_n, where the concealed frame $\tilde{X}_n$ serves as side information (see Section 6).

We emphasize that the proposed error resilient scheme is independent from the specific MCP codec used to encode the bitstream. Only the rate allocation algorithm needs to be adjusted to the specific codec adopted. In Section 7 we provide experimental results obtained by encoding the primary stream with the H.264/AVC [37] video codec.

4. End-to-end distortion estimation at the encoder side

In order to optimally allocate bits in the redundant WZ coded stream, the encoder needs to compute an estimate

of the channel induced distortion in the DCT domain. This section illustrates the distortion estimation algorithm adopted in our scheme. The algorithm performs a sequence of two steps:

(1) Channel induced distortion estimation in the pixel domain.
(2) Transformation of the channel induced distortion from the pixel to the DCT domain.

The estimate provided in the pixel domain is computed by means of the ROPE algorithm, whilst for the estimate in the DCT domain the proposed EDDD (expected distortion of DCT coefficients) algorithm is used. Since EDDD represents an extension of the ROPE algorithm, Section 4.1 briefly reviews the pixel-domain version of ROPE [42]. Section 4.2 describes how to compute the channel induced distortion in the DCT domain with the EDDD algorithm.

4.1. Overview of the ROPE algorithm

Let f_n^i and $\hat{f}_n^i$ denote, respectively, the original and the reconstructed luminance (or chrominance) component of a pixel at spatial location i in the n-th frame at the encoder side, and let $\tilde{f}_n^i$ denote the corresponding pixel at the receiver side. Assuming the mean square error (MSE) as distortion measure, the end-to-end distortion, due to both quantization and channel losses, at spatial location i in frame n is given by

$$d_n^i = (f_n^i - \tilde{f}_n^i)^2. \tag{1}$$

Supposing a packet-switched based network, the value provided by Eq. (1) depends on the PLR at which the channel drops transmitted packets and also on the concealment strategy employed at the decoder side. The work in [42] proposes a recursive method (ROPE) to estimate the decoder side distortion due to quantization, error propagation and error concealment in the spatial domain. The algorithm is designed to operate over motion-compensated hybrid predictive codecs, including MPEG-x and H.26x codecs [27]. In these video coding schemes, each frame is partitioned into 16×16 pixel blocks called *macroblocks* (MBs). Each MB can be intra or inter coded. Coded MBs are gathered in packets, each comprising a constant number of MBs, and transmitted through the channel. With this setting, the PLR equals the pixel loss rate.

At the decoder side, the packets are decoded. If some of these packets have been dropped, the concealment algorithm is applied. The PLR and the concealment algorithm are the only *a priori* knowledge that the ROPE algorithm needs to know in order to provide the decoder distortion estimate.

The ROPE algorithm is now briefly summarized, distinguishing between intra and inter coded MBs. In the following, a simple error concealment strategy is assumed: the motion vector (MV) of a missing MB is replaced by the one of the above MB. If the MB above is missing too, the MV is set to zero. The pixels pointed by the concealed MV replace the pixels of the lost MB.

Below we briefly summarize the ROPE algorithm as described in [42]:

(1) *Preliminaries*: Since the loss pattern is known only at the decoder side, the quantity $\tilde{f}_n^i$ is characterized as a random variable; the distortion d_n^i is the expectation over all possible channel realizations:

$$d_n^i = E[(f_n^i - \tilde{f}_n^i)^2]. \tag{2}$$

Developing Eq. (2), we can highlight the dependency of d_n^i from the first and the second order moments of the random variable $\tilde{f}_n^i$:

$$d_n^i = E[(f_n^i - \tilde{f}_n^i)^2] = (f_n^i)^2 - 2 \cdot f_n^i \cdot E[\tilde{f}_n^i] + E[(\tilde{f}_n^i)^2]. \tag{3}$$

The ROPE equations provide a recursive formulation to calculate the first and the second order moment of $\tilde{f}_n^i$. The PLR is denoted by p.

(2) *End-to-end distortion in intra coded MBs*: The considered MB can be either correctly received or lost. Furthermore, in this case the MB can be concealed in two ways: with a concealed MV, if the MB above has been correctly received, or with a zero MV, if the MB above has been lost too. More precisely:
   (a) If the MB to be decoded has been correctly received, $\tilde{f}_n^i = \hat{f}_n^i$ with probability $(1-p)$.
   (b) If the MB to be decoded is lost, but the MB above has been correctly received, $\tilde{f}_n^i = \tilde{f}_{n-1}^{i''}$, where i'' is the pixel location pointed by the concealed MV. The probability is the product between the probability of the event of loosing the MB being decoded, p, and the probability of receiving the above MB, $(1-p)$. Thus the overall probability of this event is $p(1-p)$.
   (c) If both the MB to be decoded and the one above have been lost, $\tilde{f}_n^i = \tilde{f}_{n-1}^i$ with probability p^2.

Combining these three cases, the recursive equations for the first and the second order moments of $\tilde{f}_n^i$ can be derived:

$$E[\tilde{f}_n^i] = (1-p) \cdot (\hat{f}_n^i) + p(1-p) \cdot E[\tilde{f}_{n-1}^{i''}] + p^2 \cdot E[\tilde{f}_{n-1}^i] \tag{4}$$

$$E[(\tilde{f}_n^i)^2] = (1-p) \cdot (\hat{f}_n^i)^2 + p(1-p) \cdot E[(\tilde{f}_{n-1}^{i''})^2] + p^2 \cdot E[(\tilde{f}_{n-1}^i)^2]. \tag{5}$$

(3) *End-to-end distortion in inter coded MBs*: As for intra coded MBs, an inter-macroblock can be received or lost, and if this is the case, its concealment can be done either with the concealed or with the zero MV. With reference to the intra-macroblock case, the only difference is when the MB data is actually received (with probability $(1-p)$). Received data consist of the quantized prediction error $\hat{e}$ and the MV that points to the pixel location i'. Therefore, the reconstructed pixel at the decoder is given by $\tilde{f}_n^i = \hat{e}_n^i + \tilde{f}_{n-1}^{i'}$. It is possible to derive the recursive equations for the first and the

second order moments of $\tilde{f}_n^i$:

$$E[\tilde{f}_n^i] = (1-p)\cdot(\hat{e}_n^i + E[\tilde{f}_{n-1}^{i'}]) + p(1-p) \times E[\tilde{f}_{n-1}^{i''}] + p^2 \cdot E[\tilde{f}_{n-1}^i], \quad (6)$$

$$\begin{aligned} E[(\tilde{f}_n^i)^2] &= (1-p)\cdot E[(\hat{e}_n^i + \tilde{f}_{n-1}^{i'})^2] \\ &\quad + p(1-p)\cdot E[(\tilde{f}_{n-1}^{i''})^2] \\ &\quad + p^2 \cdot E[(\tilde{f}_{n-1}^i)^2] \\ &= (1-p)\cdot((\hat{e}_n^i)^2 + E[(\tilde{f}_{n-1}^{i'})^2] \\ &\quad + 2\cdot\hat{e}_n^i \cdot E[\tilde{f}_{n-1}^{i'}]) \\ &\quad + p(1-p)\cdot E[(\tilde{f}_{n-1}^{i''})^2] \\ &\quad + p^2 \cdot E[(\tilde{f}_{n-1}^i)^2]. \end{aligned} \quad (7)$$

In order to handle MVs with sub-pixel precision, the extension of ROPE described in [3] has been integrated into the system.

4.2. The distortion estimate in the DCT domain

In the proposed scheme, WZ coding is applied in the DCT domain. In order to perform optimal bit allocation, the end-to-end distortion needs to be estimated in the transform domain. In [19], the recursive equations of ROPE (4)–(7) are applied directly in the DCT domain. The main issue with this approach regards how to relate coefficients at time n with those at time $n-1$ by motion compensation. In [19] MVs are coarsely quantized with a step size equal to the MB size. The predictor for the current MB is matched with its nearest MB in the reference frame. This approximation leads to a significant loss of accuracy [7].

In this work, we adopt an extension of ROPE in the DCT domain previously proposed by the same authors [7], (EDDD—expected distortion of DCT coefficients). The basic idea consists of running ROPE with sub-pixel accuracy in the spatial domain. Then, the end-to-end distortion is translated from the spatial to the DCT domain. Let $\tilde{F}_n^{b,l}$ denote the l-th coefficient of the b-th DCT block in frame n, $l = 1,\ldots,B^2$, $b = 1,\ldots,N_b$, where B is the DCT block size and $N_b = N/B^2$ is the number of DCT blocks in a frame with N pixels. Due to channel losses, $\tilde{F}_n^{b,l}$ is considered as a random variable. In fact, it is a linear combination of random variables, each representing a pixel value inside a DCT block. As in the spatial domain, the first and second order moments are needed to calculate $D_n^{b,l}$:

$$\begin{aligned} D_n^{b,l} &= E[(F_n^{b,l} - \tilde{F}_n^{b,l})^2] \\ &= (F_n^{b,l})^2 - 2\cdot F_n^{b,l}\cdot E[\tilde{F}_n^{b,l}] + E[(\tilde{F}_n^{b,l})^2]. \end{aligned} \quad (8)$$

Let T_l, $l = 1,\ldots,B^2$, denote the l-th 2D-DCT basis function applied to a block of size $B\times B$. Since the DCT is a linear transform:

$$\tilde{F}_n^{b,l} = \sum_{r=1}^{B^2} T_l(r)\cdot\tilde{f}_n^{b,r}, \quad (9)$$

the first order moment of $\tilde{F}_n^{b,l}$, is given by

$$E[\tilde{F}_n^{b,l}] = \sum_{r=1}^{B^2} T_l(r)\cdot E[\tilde{f}_n^{b,r}], \quad (10)$$

while the second order moment is

$$\begin{aligned} E[(\tilde{F}_n^{b,l})^2] &= \sum_{r=1}^{B^2}\sum_{s=1}^{B^2} T_l(r)\cdot T_l(s)\cdot(E[\tilde{f}_n^{b,r}\tilde{f}_n^{b,s}] \\ &\quad - E[\tilde{f}_n^{b,r}]\cdot E[\tilde{f}_n^{b,s}]) + (E[\tilde{F}_n^{b,l}])^2. \end{aligned} \quad (11)$$

In order to compute (11), we need to provide an estimate of $E\left[\tilde{f}_n^{b,r}\tilde{f}_n^{b,s}\right]$, where r and s denote two arbitrary pixels within a $B\times B$ block. Instead of keeping track of all the $B^2(B^2-1)/2$ correlation terms $E[\tilde{f}_n^{b,r}\tilde{f}_n^{b,s}]$ between any pair of pixels (r,s), only $2B(B-1)$ terms, when r and s are adjacent pixels, are estimated by means of the following recursive equations analogous to (4) and (6), respectively, for the intra and inter coded MBs:

$$E[\tilde{f}_n^r\tilde{f}_n^s] = (1-p)\cdot(\hat{f}_n^r\hat{f}_n^s) + p(1-p)\cdot E[\tilde{f}_{n-1}^{r'}\tilde{f}_{n-1}^{s'}] + p^2\cdot E[\tilde{f}_{n-1}^r\tilde{f}_{n-1}^s], \quad (12)$$

$$\begin{aligned} E[\tilde{f}_n^r\tilde{f}_n^s] &= (1-p)\cdot E[(\hat{e}_n^r + \tilde{f}_{n-1}^{r'})(\hat{e}_n^s + \tilde{f}_{n-1}^{s'})] \\ &\quad + p(1-p)\cdot E[(\tilde{f}_{n-1}^{r''}\tilde{f}_{n-1}^{s''})] \\ &\quad + p^2\cdot E[(\tilde{f}_{n-1}^r\tilde{f}_{n-1}^s)] \\ &= (1-p)\cdot(\hat{e}_n^r\hat{e}_n^s + \hat{e}_n^r E[\tilde{f}_{n-1}^{s'}] + \hat{e}_n^s E[\tilde{f}_{n-1}^{r'}] \\ &\quad + E[\tilde{f}_{n-1}^{r'}\tilde{f}_{n-1}^{s'}]) + p(1-p)\cdot E[\tilde{f}_{n-1}^{r''}\tilde{f}_{n-1}^{s''}] \\ &\quad + p^2\cdot E[\tilde{f}_{n-1}^r\tilde{f}_{n-1}^s], \end{aligned} \quad (13)$$

where r' (s') denotes the pixel location in frame $n-1$ referenced by the received MVs, while r'' (s'') the one referenced by the concealed MVs.

In order to compute (11), we need to provide an estimate of $E\left[\tilde{f}_n^r\tilde{f}_n^s\right]$ also when r and s are not adjacent pixels, as illustrated in Fig. 2. To address this issue, we

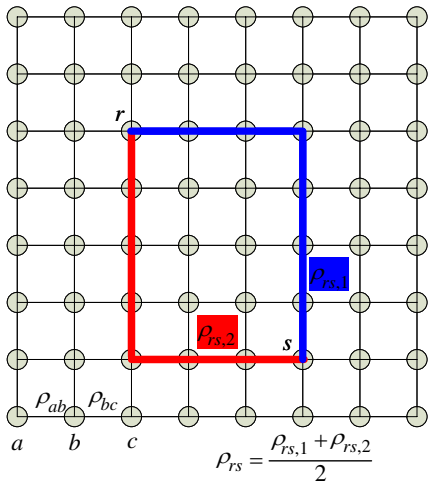

Fig. 2. Computation of correlation terms between non-adjacent pixels r and s.

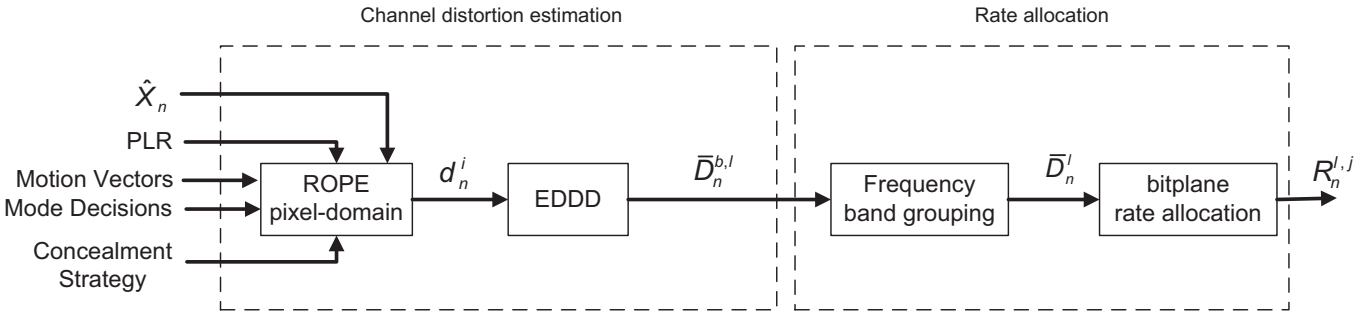

Fig. 3. The rate estimator module.

adopt the following relationship:

$$\rho_{rs} = \frac{E[\tilde{f}^r_{n-1}\tilde{f}^s_{n-1}] - \mu_r\mu_s}{\sigma_r\sigma_s} \Rightarrow E[\tilde{f}^r_{n-1}\tilde{f}^s_{n-1}] = \rho_{rs}\cdot\sigma_r\sigma_s + \mu_r\mu_s, \quad (14)$$

where μ represents the mean and σ the standard deviation of a reconstructed pixel, obtained with ROPE. The term ρ_{rs} is the correlation coefficient between pixels r and s; its value is estimated assuming a first order autoregressive model between any pair of adjacent pixels r_1 and r_2:

$$\tilde{f}^{r_1}_{n-1} = \rho_{r_1r_2}\cdot\tilde{f}^{r_2}_{n-1} + w_1, \quad (15)$$

where w_1 denotes the white noise term. Thus, ρ_{rs} is given as the average of the products of the correlation coefficients corresponding to the pixels lying on the paths $\rho_{rs,1}$ and $\rho_{rs,2}$ as shown in Fig. 2.

The ROPE–EDDD algorithm allows to compute the end-to-end distortion due to quantization, packet losses and error propagation, i.e.,

$$D_n^{b,l} = E[(F_n^{b,l} - \tilde{F}_n^{b,l})^2]. \quad (16)$$

Instead, in this work we are interested in estimating the distortion between the noiseless and noisy reconstructed coefficients at the decoder, i.e.,

$$\bar{D}_n^{b,l} = E[(\hat{F}_n^{b,l} - \tilde{F}_n^{b,l})^2], \quad (17)$$

in order to determine the rate allocation in the auxiliary stream.

By replacing $F_n^{b,l}$ with $\hat{F}_n^{b,l}$ in (8), $\bar{D}_n^{b,l}$ is readily computed. We notice that in this case we do not need to have access to the original frames X_n, but only to the noiseless reconstructed frames $\hat{X}_n$. In case the proposed technique is not applied at encoding time, the cost of doing this is that the video sequence needs to be completely decoded at the transmitter end in order to run ROPE–EDDD. Fig. 3 shows that the channel distortion estimation module receives in input the reconstructed frame at time n, $\hat{X}_n$, the MVs and the mode decisions for this frame, the network PLR p and the knowledge of the error concealment algorithm adopted at the decoder.

5. WZ encoding and channel rate allocation

The WZ encoder is similar to the one presented in [1]. A block representation of the WZ encoder is shown in Fig. 4. To generate the WZ bitstream, the reconstructed frame $\hat{X}_n$ is partitioned into blocks of dimension $B\times B$ with $B=8$ (we will denote the generic block with b); over these blocks the DCT transform is applied. The transform coefficients are then grouped together to form the coefficients subbands $\hat{F}_n^l = [\hat{F}_n^{1,l}, \hat{F}_n^{2,l}, \ldots, \hat{F}_n^{N_b,l}]$, where l denotes the subband number and N_b is the number of transform coefficients in each subband. Each subband $\hat{F}_n^l$ is then quantized using the same midtread scalar quantizer with step δ and the corresponding bitplanes are independently coded using an LDPC encoder. The quantizer step δ is determined based on the dynamic range of DCT coefficients, which is equal to $[-2048, +2048]$ for a 8×8 DCT and on the target number of bitplanes J to be encoded, $\delta = 2*2048/2^J$. We use regular rate-adaptive LDPC accumulated codes [33] with degree three source node distribution. We transmit the syndrome bits relative to the first 16 DCT subbands, while the systematic bits are discarded. We start encoding the most significant bitplane. Then, encoding proceeds with the remaining bitplanes. The encoder also transmits the average distortion computed by the *rate estimator* module, i.e., $\bar{D}_n^l$,since it is exploited by the turbo decoder, as explained in the next section.

Since there is no feedback channel available, the encoder needs to estimate the number of syndrome bits for each bitplane j of subband l in frame n. In order to simplify the notation, in the following we discard the frame index n and the subband index l, denoting $X=\hat{F}_n^l$ and $Y=\tilde{F}_n^l$, respectively, as the source and the side information. Also, we assume the following additive noise model:

$$Y = X + Z, \quad (18)$$

where X and Z are assumed statistically independent. The distribution of the reconstructed DCT coefficients X can be well approximated by a Laplacian distribution [12]

$$p_X(x) = \frac{\alpha_x}{2}e^{-\alpha_x|x|}, \quad (19)$$

where the parameter α_x is related to the subband coefficient variance σ_x^2 by $\alpha_x = \sqrt{2/\sigma_x^2}$. Let σ_z^2 denote the average distortion obtained from the ROPE–EDDD estimate for the current DCT subband in the current frame

$$\sigma_z^2 = \frac{1}{N_b}\sum_{b=1}^{N_b}\bar{D}^b. \quad (20)$$

In our experiments, we verified that also the distribution of the correlation noise Z can be modeled by a Laplacian

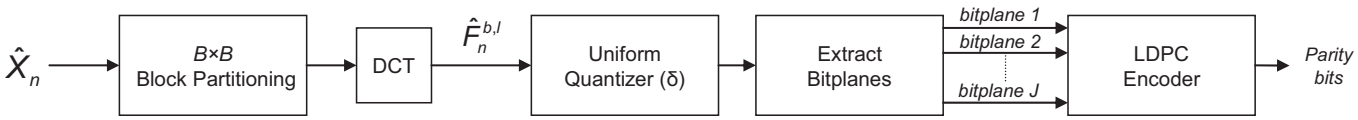

Fig. 4. Block diagram of the WZ encoder.

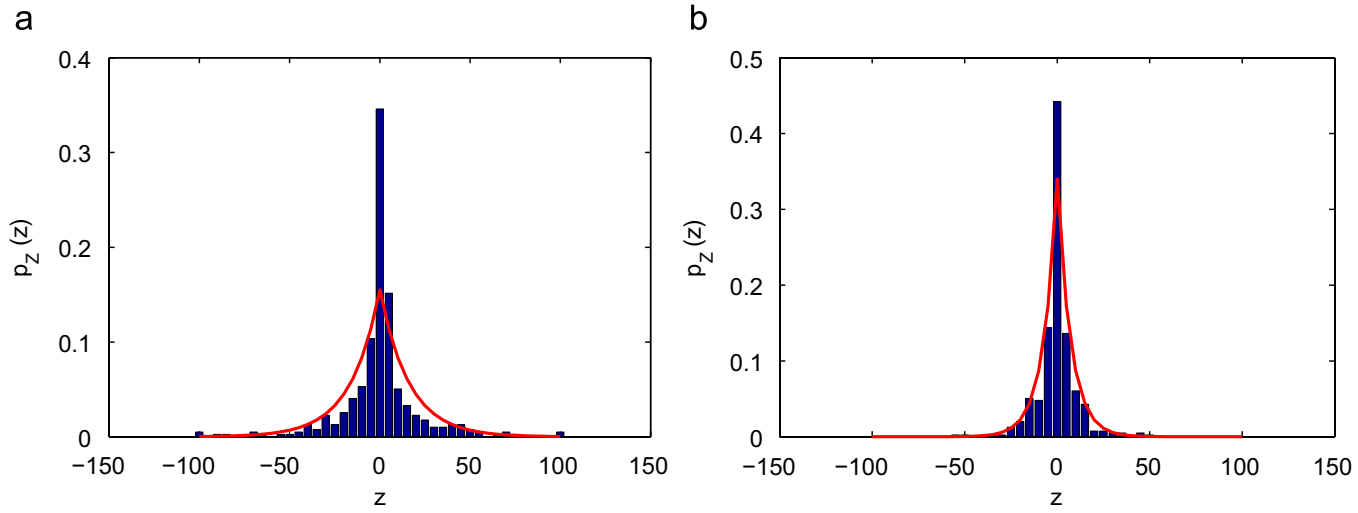

Fig. 5. Laplacian fitting of the correlation noise in the DCT domain. (a) DC coefficient and (b) AC coefficient number 10 in zig-zag scanning order.

distribution $p_Z(z)$ with parameter $\alpha_z = \sqrt{2/\sigma_z^2}$. As an example, Fig. 5 illustrates the fitting of the Laplacian model on the correlation noise for two DCT coefficients in a sample frame of the *Foreman* sequence affected by errors.

The rate allocation algorithm receives in input the source variance σ_x^2, the correlation noise variance σ_z^2, the quantization step size δ and the number of bitplanes to be encoded J and returns the average number of bits needed to decode each bitplane R^j, $j = 1,\ldots,J$. The Shannon's lower bound on the number of bits is given by

$$R^j \geqslant H(x^j|Y,x^{j-1},x^{j-2},\ldots,x^1) \quad \text{(bits/sample)}, \tag{21}$$

where x^j denotes the j-th bitplane of the source X. In fact, as detailed in the next section, LDPC decoding of bitplane j exploits the knowledge of the real-valued side information Y as well as previously decoded bitplanes $x^{j-1},x^{j-2},\ldots,x^1$. The value of R^j from Eq. (21) can be readily computed by numerical integration. In fact, the expression of the entropy in (21) can be written as

$$\begin{aligned} &H(x^j|Y,x^{j-1},x^{j-2},\ldots,x^1) \\ &\quad = H(x^j|Y,Q) = \sum_{q=1}^{2^{j-1}} p(q) H(x^j|Y,Q=q) \\ &\quad = \sum_{q=1}^{2^{j-1}} p(q) \int_{-\infty}^{+\infty} p_{Y|q}(y) H(x^j|Y=y,Q=q)\,\mathrm{d}y, \end{aligned} \tag{22}$$

where Q denotes the quantization bin index obtained decoding up to bitplane $j-1$. The value of $H(x^j|Y=y,Q=q)$ represents the entropy of the binary source x^j when the side information assumes the specific value y and the source X is known to be within quantization bin q, i.e.,

$$\begin{aligned} &H(x^j|Y=y,Q=q) \\ &\quad = -p_0\log_2 p_0 - (1-p_0)\log_2(1-p_0), \end{aligned} \tag{23}$$

and

$$\begin{aligned} p_0 &= \Pr\{x^j = 0|Y=y,Q=q\} \\ &= \frac{\int_{L_q}^{(L_q+U_q)/2} p_X(x)p_Z(x-y)\,\mathrm{d}x}{\int_{L_q}^{U_q} p_X(x)p_Z(x-y)\,\mathrm{d}x}, \end{aligned} \tag{24}$$

where L_q and U_q are the lower and upper thresholds of the quantization bin q. Since $p_Z(\cdot)$ is Laplacian, the value of (23) can be computed analytically.

The expression $p_{Y|q}(y)$ in (22) represents the marginal distribution of Y, conditioned on $X \in q$, and it can be obtained, according to the additive correlation model (18), from the knowledge of the joint distribution $p_{XY}(x,y) = p_X(x)p_Z(x-y)$, i.e.,

$$p_{Y|q}(y) = \frac{\int_{L_q}^{U_q} p_{XY}(x,y)\,\mathrm{d}x}{\int_{L_q}^{U_q} p_X(x)\,\mathrm{d}x} = \frac{\int_{L_q}^{U_q} p_X(x)p_Z(x-y)\,\mathrm{d}x}{\int_{L_q}^{U_q} p_X(x)\,\mathrm{d}x}. \tag{25}$$

Finally, Eq. (22) can be evaluated by means of numerical integration over y.

As explained in the experimental results section, we will consider QCIF video sequences with 8×8 DCT, resulting in non-ideal channel codes with a finite sequence length ($N_b = 396$). Because of this non-ideality, the lower bound in (21) is not attained. Indeed, a rate overhead of approximately $\Delta R = 0.1$ (bit/sample) is introduced. This value has been determined experimentally and it achieves a good balance between the rate

overhead and the probability of successful decoding. Therefore, the rate allocation algorithm presented here outputs $R^j = H(x^j|Y, x^{j-1}, x^{j-2}, \ldots, x^1) + \Delta R$. Also, note that the baseline version of the ROPE algorithm, as briefly summarized in Section 4.1, considers only basic coding tools, such as intra/inter coding modes. Instead, the state-of-the-art H.264/AVC video coding standard provides more advanced tools, including intra-prediction, deblocking filter, etc. In addition, different non-normative solutions can be adopted at the decoder side concerning the motion-compensated concealment algorithm. Unlike the work in [39], which explicitly models the effect of intra-prediction and deblocking, here we apply the extension of the baseline version of ROPE [3], in order to simplify the distortion estimation procedure. We noticed that this slightly over-estimates the distortion measured at the decoder: this is mostly due to the fact that the concealment algorithm adopted in our simulations [29] is much more sophisticated than the one assumed by the baseline ROPE at the encoder. To take this into account, we correct the distortion provided by ROPE–EDDD for the l-th DCT coefficient by a factor $\alpha_l < 1$. The values of α_l are empirically determined on the basis of training data. Simulations over a large amount of data show that these coefficients can be assumed as independent with respect to the considered video sequence. Finally, we have verified in our simulations that the effect of neglecting intra-prediction and deblocking is negligible and it can be taken into consideration in the parameter α_l.

6. WZ decoding

At the receiver, the concealed reconstructed frame $\tilde{X}_n$ (see Fig. 1) is used by the WZ decoder as side information. At this side the actual error pattern is known exactly, therefore an ET module uses this information to determine which blocks might be affected by errors. Apart from the blocks belonging to lost slices, also those blocks that depend on previously corrupted blocks are flagged as "potentially noisy". The error tracker produces, for each frame, a binary map BM_n that indicates if the reconstructed block at the decoder might differ from the corresponding one at the encoder. The algorithm is similar to the one presented in [5], with the important difference that in our case ET is performed at the decoder only. The LDPC decoder can take advantage from this information by adaptively setting the conditional probability to 1 for those coefficients that are certainly correct. This means that the LDPC decoder totally trusts the side information in these cases. For the other coefficients (i.e., the *noisy* and *potentially noisy* ones) the conditional probabilities are calculated using (24).

The WZ decoder (see Fig. 6) takes the concealed frame $\tilde{X}_n$, partitions it into blocks of dimension $B \times B$ and transforms it using a block-based DCT. The transform coefficients are then grouped together to form the side information subbands $\tilde{F}_n^l$. The received syndrome bits are used to "correct" each coefficient $\tilde{F}_n^l$ into F'^l_n. LDPC decoding is applied at the bitplane level, starting from the most significant bitplane of each subband. In the proposed system, the LDPC decoder exploits the received distortions estimated at the encoder $\bar{D}_n^l$ in order to describe the correlation statistics between $\hat{F}_n^{b,l}$ and the side information $\tilde{F}_n^{b,l}$. Therefore, adaptivity is guaranteed both at subband and frame level. After LDPC decoding, some residual errors might occur if the number of received syndrome bits is not sufficient for the exact reconstruction of every bitplane. Error detection at the LDPC decoder is performed by comparing the received syndrome bits with a syndrome generated from the decoded bitplane. In this case, if decoding of bitplane j fails, reconstruction is based on bitplanes $1, \ldots, j-1$ only. As before, by denoting with q the quantization bin index obtained decoding up to bitplane $j-1$, optimal reconstruction is obtained by computing the centroid of the quantization interval (L_q, U_q) exploiting the Laplacian model

$$E[X|X \in (L_q, U_q), Y = \tilde{F}_n^{b,l}] = \frac{\int_{L_q}^{U_q} x\, p_X(x) p_Z(x - \tilde{F}_n^{b,l})\, dx}{\int_{L_q}^{U_q} p_X(x) p_Z(x - \tilde{F}_n^{b,l})\, dx}, \quad (26)$$

where $p_Z(z)$ has been defined in the previous section, parameterized by the received distortion estimate $\bar{D}_n^l$. In evaluating (26), we need the knowledge of σ_x^2, which is not directly available at the decoder. Assuming the statistical independence between X and Z and the additive noise model (18), we obtain $\sigma_x^2 = \sigma_y^2 - \sigma_z^2$. In the previous expression σ_y^2 is estimated from the received DCT coefficients and $\sigma_z^2 = \bar{D}_n^l$ as before.

Finally, the inverse-DCT is performed by the WZ decoder and the reconstructed frame $X'(n)$ is copied into the buffer of the MCP decoder, to serve as reference frame at time $n+1$. In this way, the amount of drift propagated to successive frames is reduced.

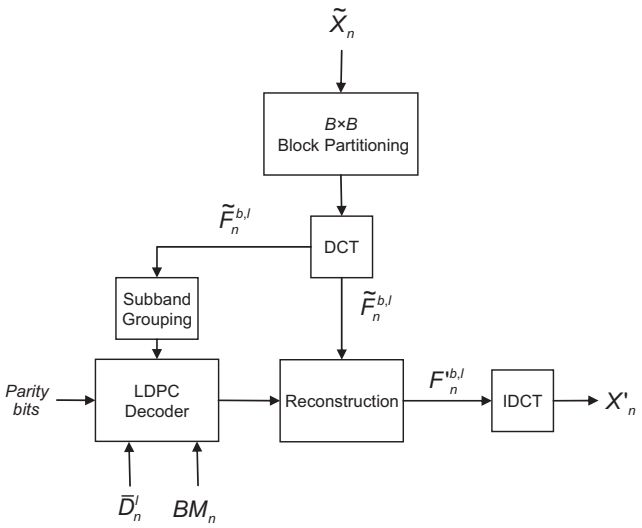

Fig. 6. Block diagram of the WZ decoder.

7. Experimental results

We carried out several experiments in order to validate the proposed scheme. First, we present the source coding

Table 1
Source coding conditions for the test video sequences

Sequence name	Spatial resolution	Rate (kbps)	Frame rate (Hz)	GOP length	Number of frames	Average PSNR all frames (dB)
Foreman	QCIF	128	15	15	150	35.5
Soccer	QCIF	128	15	15	150	33.5
Coastguard	QCIF	128	15	15	150	31.9

conditions and the network packetization strategy. Second, the comparison setup is briefly discussed and finally the experimental results are reported and commented.

7.1. Source coding conditions and packetization issues

In our simulations three QCIF video sequences, namely *Foreman*, *Soccer* and *Coastguard*, are compressed using the H.264/AVC reference software (JM13.2) encoder with baseline profile [11]. The source coding parameters for all these video sequences are listed in Table 1. As for the H.264/AVC encoder the following parameters have been used:

- Maximum number of reference frames: 1.
- MB partitions for motion estimation: enabled.
- Rate-distortion optimization: high complexity mode [11].
- Early skip detection mode decision: enabled.
- Motion estimation algorithm: enhanced predictive zonal search (EPZS).

Each frame is coded using nine slices, each consisting of 11 MBs. In order to improve the efficiency of the decoder error concealment algorithm, the FMO tool has been enabled using the dispersed MB to slice group mapping, with two slice groups, as described in [37]. We have used the error concealment algorithm implemented in the H.264/AVC decoder reference software [29]. The packetization follows the specifications of the real-time transfer protocol (RTP) [36] where each packet contains a coded slice. Finally, the considered PLRs are 3%, 5%, 10%, 20% and the loss patterns have been generated according to [35].

7.2. Comparison setup

The proposed scheme has been compared with other widely adopted error resiliency tools: adaptive MB intra refresh and FEC by means of Reed–Solomon codes.

As for adaptive intra-refresh, the MBs are selected on the basis of the channel induced distortion estimate provided by the ROPE algorithm (see Section 4.1). In this setup, for each inter-coded frame, its MBs are sorted in descending order according to expected distortion, and the first M MBs are chosen to be intra coded. The number M is determined in order to spend the same average rate required by the proposed scheme. We notice that this technique can be applied also at transcoding time. In this scenario, the transcoder encodes the MBs selected on the basis of the ROPE estimate as intra, whilst, for those MBs that were originally inter coded, it exploits the existing MVs allowing computational savings.

Table 2
Additional WZ bitrate (in % of the main stream rate) sent to protect the primary stream

	Additional WZ bitrate (%)			
Sequence name	PLR = 3%	PLR = 5%	PLR = 10%	PLR = 20%
Foreman	18	20	28	32
Soccer	20	22	27	29
Coastguard	19	25	28	32

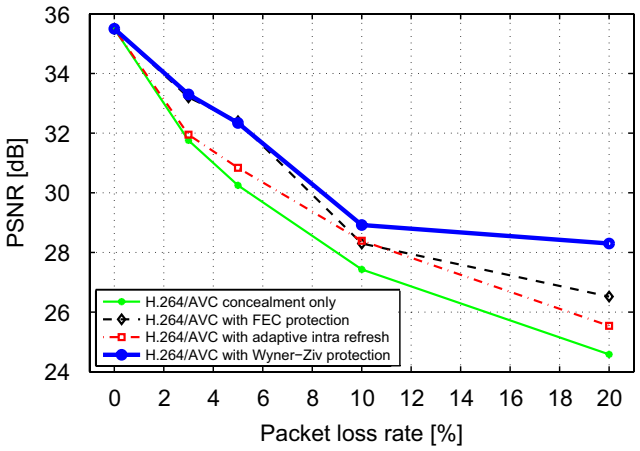

Fig. 7. PSNR vs. PLR for the *Foreman* sequence.

The FEC scheme adopts (N, K) Reed–Solomon channel codes. In this case the number of data slices, i.e., K, is fixed and set equal to 9, while the number of redundant slices, i.e., $(N-K)$ is chosen to have the same total bitrate of the proposed WZ scheme.

7.3. Experimental results and discussion

We simulated the transmission of the test sequences by averaging the PSNR values obtained over 30 channel realizations. For all the experiments, we protected the first six bitplanes (i.e., $J = 6$, see Section 5) and the first sixteen DCT subbands in diagonal scanning order (i.e., $l = 1, \ldots, 16$, see Section 5). The corresponding rate overhead due to the encoding of the auxiliary WZ stream is indicated in Table 2, expressed as a percentage of the main stream bitrate (128 kbps).

Figs. 7–9 show the performance of the proposed scheme in terms of robustness to packet losses compared with adaptive intra-refresh and FECs. We notice that for low values of PLR (3–5%) the proposed scheme achieves the same performance as a FEC based scheme. This is expected,

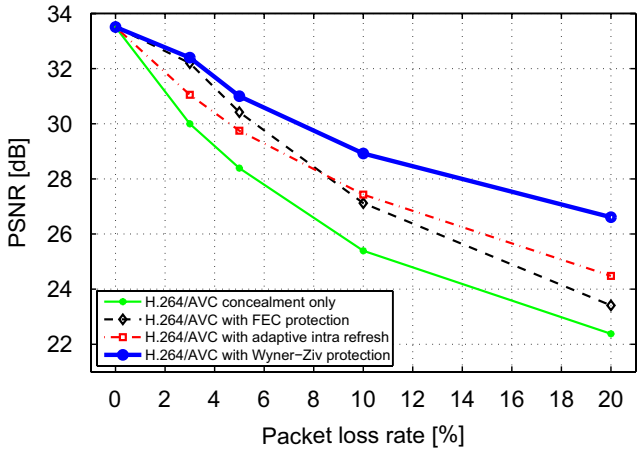

Fig. 8. PSNR vs. PLR for the *Soccer* sequence.

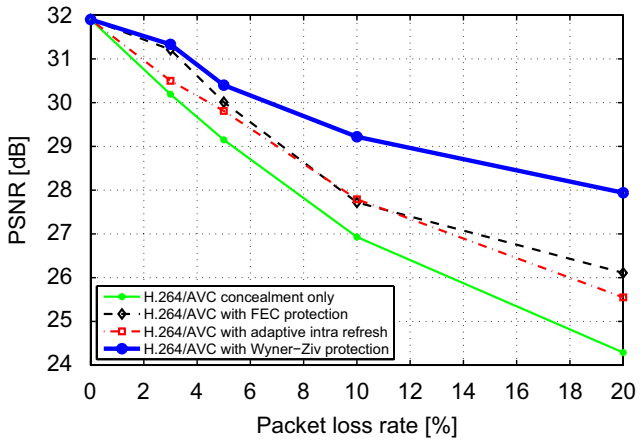

Fig. 9. PSNR vs. PLR for the *Coastguard* sequence.

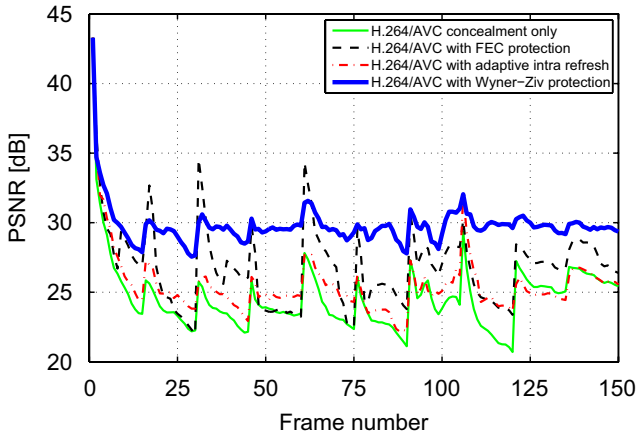

Fig. 10. PSNR tracks for the *Foreman* sequence when the PLR is equal to 20%.

since FECs are particularly efficient at low PLR, when up to $N-K$ packet losses per frame can be perfectly recovered. As the PLR increases (10–20%), the proposed scheme outperforms a FEC based scheme by up to 3.2 dB for the *Soccer* sequence at 20% PLR. We can justify this behavior observing that the performance of FEC codes rapidly degrades once the error correction capability is exceeded. Conversely, the

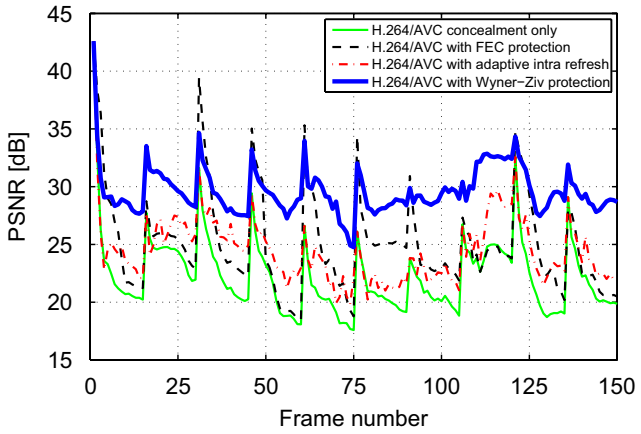

Fig. 11. PSNR tracks for the *Soccer* sequence when the PLR is equal to 20%.

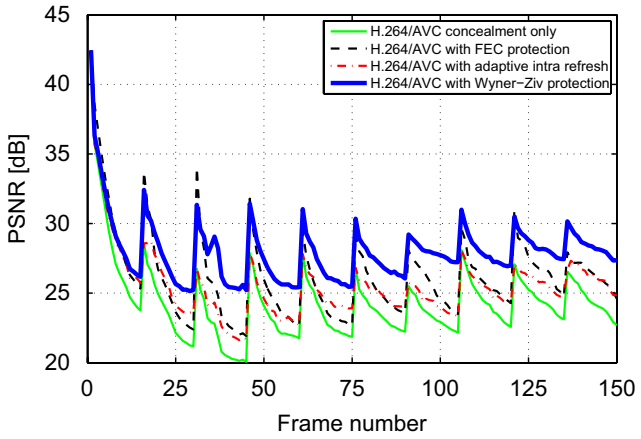

Fig. 12. PSNR tracks for the *Coastguard* sequence when the PLR is equal to 20%.

proposed scheme exhibits a more graceful degradation and reduces the effect of drift propagation: this is demonstrated by temporal PSNR tracks of Figs. 10–12.

The proposed WZ scheme outperforms adaptive intra-refresh at all PLRs. At 3% PLR, the average coding gain is about 1.35 dB for the *Foreman* sequence and increases to up to 2.76 dB at 20% PLR.

Visual quality inspection of the reconstructed sequences confirms the objective quality measurements. As an example, Fig. 13 illustrates the 104-th frame of the *Coastguard* sequence, which is the last frame within its GOP. In this case, adaptive intra refresh is unable to cope with drift propagation. FECs provide a better result, recovering most of the details of the moving object. We notice that the proposed WZ scheme significantly reduces the effect of drift, preserving the details in the picture, both in the moving object and in the background.

8. Conclusions

In this paper, we propose a video coding scheme where an auxiliary bitstream encoded according to distributed source coding principles is used to correct errors introduced by packet losses and improve the quality of the

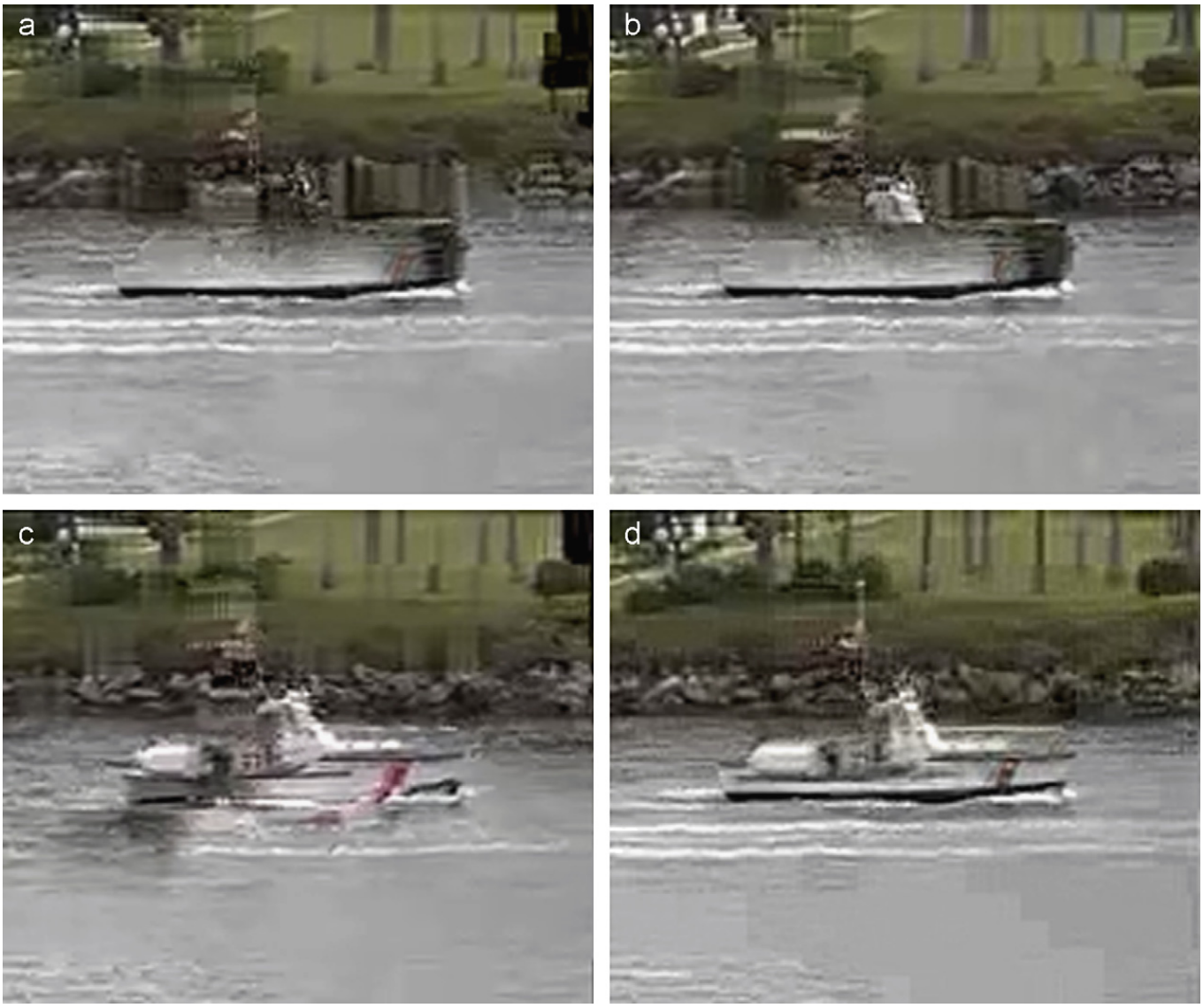

Fig. 13. Subjective image quality evaluation for the 104-th frame of the *Coastguard* sequence when the PLR is equal to 20% (concealment only) (adaptive intra refresh) (FEC protection) (Wyner–Ziv protection).

reconstructed frames. Experimental results reveal that the proposed scheme has comparable or better performance than more conventional solutions based on FEC codes or adaptive intra-macroblock refresh. The novelty and the advantages of the proposed scheme concern three central facts: first, the channel redundancy can be added over an already coded bitstream, only requiring local decoding for the calculation of the channel induced distortion estimate; second, since the syndrome bit rates are allocated at the transmitter side, the proposed scheme does not use a feedback channel and thus it can satisfy low end-to-end delay requirements; third, since the proposed scheme does not require the original video data to perform UEP, it is particularly suitable for real video coding applications.

References

[1] A. Aaron, S. Rane, E. Setton, B. Girod, Transform-domain Wyner–Ziv codec for video, San Jose, CA, USA, January 2004.

[2] R. Bernadini, M. Naccari, R. Rinaldo, M. Tagliasacchi, S. Tubaro, P. Zontone, Error concealment using a DVC approach for video streaming applications, in: European Signal Processing Conference, Poznan, Poland, September 2007.

[3] V. Bocca, M. Fumagalli, R. Lancini, S. Tubaro, Accurate estimate of the decoded video quality: extension of ROPE algorithm to half-pixel precision, in: Picture Coding Symposium, San Francisco, CA, USA, December 2004.

[4] C. Brites, J. Ascenso, F. Pereira, Improving transform domain Wyner–Ziv video coding performance, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Toulouse, FR, May 2006.

[5] N. Färber, B. Girod, Feedback-based error control for mobile video transmission, Proc. IEEE 87 (October 1997) 1707–1723.

[6] M. Fresia, G. Caire, Combined error protection and compression with turbo codes for image transmission using a JPEG2000-like architecture, in Proceedings of the International Conference on Image Processing, Atlanta, USA, October 2006.

[7] M. Fumagalli, M. Tagliasacchi, S. Tubaro, Expected Distortion of DCT—Coefficients in Video Streaming over Unreliable Channel, Lecture Notes in Computer Science, vol. 3893/2006, Springer, Berlin, Heidelberg, April 2006.

[8] M. Fumagalli, M. Tagliasacchi, S. Tubaro, Drift reduction in predictive video transmission using a distributed source coded side-channel, in: Proceedings of the International Conference on

Acoustics, Speech, and Signal Processing, Toulouse, France, May 2006.
[9] B. Girod, A. Aaron, S. Rane, D. Rebollo Monedero, Distributed video coding, Proc. IEEE 93 (January 2005) 71–83.
[10] ITU-T, Information Technology—Coding of Audio-Visual Objects—Part 10: Advanced Video Coding, Final Draft International Standard, ISO-IEC FDIS 14 496-10, 2003.
[11] Joint Video Team (JVT), H.264/AVC Reference Software Version JM13.2, Downloadable at: ⟨http://iphome.hhi.de/suehring/tml/download/⟩.
[12] E.Y. Lam, J.W. Goodman, A mathematical analysis of the DCT coefficient distributions for images, IEEE Trans. Image Process. 9 (October 2000) 1661–1666.
[13] L. Liang, P. Salama, E.J. Delp, Unequal error protection using Wyner–Ziv coding for error resilience, in: Visual Communications and Image Processing, Proceedings of SPIE, San Jose, CA, USA, January 2007.
[14] L. Liang, P. Salama, E.J. Delp, Adaptive unequal error protection based on Wyner–Ziv coding, in: Picture Coding Symposium, Lisbon, PT, November 2007.
[15] A. Majumdar, J. Chou, K. Ramchandran, Robust distributed video compression based on multilevel coset codes, in: Proceedings of the 37th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, November 2003.
[16] A. Majumdar, R. Puri, P. Ishwar, K. Ramchandran, Complexity/performance trade-offs for robust distributed video coding, in: Proceedings of the International Conference on Image Processing, Genova, Italy, 2005.
[17] A. Majumdar, K. Ramchandran, Video multicast over lossy channels based on distributed source coding, in: Proceedings of the International Conference on Image Processing, Singapore, October 2004.
[18] A. Majumdar, J. Wang, K. Ramchandran, Drift reduction in predictive video transmission using a distributed source coded side-channel, in: ACM Multimedia, New York, USA, October 2004.
[19] A. Majumdar, J. Wang, K. Ramchandran, H. Garudadri, Robust video transmission over a lossy network using a distributed source coded auxiliary channel, in: Picture Coding Symposium, San Francisco, CA, USA, December 2004.
[20] R. Puri, A. Majumdar, P. Ishwar, K. Ramchandran, Distributed video coding in wireless sensor networks, IEEE Signal Process. Mag. 23 (July 2006) 94–106.
[21] R. Puri, A. Majumdar, K. Ramchandran, PRISM: a video coding paradigm with motion estimation at the decoder, IEEE Trans. Image Process. 16 (October 2007) 2436–2448.
[22] R. Puri, K. Ramchandran, PRISM: a new robust video coding architecture based on distributed compression principles, in: Allerton Conference on Communication, Control and Computing, Urbana-Champaign, IL, October 2002.
[23] J. Quintas Pedro, L. Ducla Soares, C. Brites, J. Ascenso, F. Pereira, C. Bandeirinha, S. Ye, F. Dufaux, T. Ebrahimi, Studying error resilience performance for a feedback channel based transform domain Wyner–Ziv video codec, in: Picture Coding Symposium, Lisbon, PT, November 2007.
[24] S. Rane, A. Aaron, B. Girod, Systematic lossy forward error protection for error—resilient digital video broadcasting—a Wyner–Ziv coding approach, in: Proceedings of the International Conference on Image Processing, Singapore, October 2004.
[25] S. Rane, A. Aaron, B. Girod, Error-resilient video transmission using multiple embedded Wyner–Ziv descriptions, in: Proceedings of the International Conference on Image Processing, Genova, Italy, September 2005.
[26] S. Rane, P. Baccichet, B. Girod, Modelling and optimization of a systematic lossy error protection system based on H.264/AVC redundant slices, in: Picture Coding Symposium, Beijing, China, April 2006.
[27] I.E.G. Richardson, Video Codec Design, Wiley, New York, 2002.
[28] A. Sehgal, A. Jagmohan, N. Ahuja, Wyner–Ziv coding of video: an error-resilient compression framework, IEEE Trans. Multimedia 6 (2004) 249–258.
[29] G.J. Sullivan, T. Wiegand, K.-P. Lim, Joint model reference encoding methods and decoding concealment methods, Technical Report JVT-I049, Joint Video Team (JVT), September 2003.
[30] M. Tagliasacchi, A. Majumdar, K. Ramchandran, A distributed-source-coding based robust spatio-temporal scalable video codec, in: Picture Coding Symposium, San Francisco, CA, USA, December 2004.
[31] M. Tagliasacchi, A. Majumdar, S. Tubaro, K. Ramchandran, Robust wireless video multicast based on a distributed source coding approach, EURASIP Signal Process. J. 86 (November 2006) 3196–3211.
[32] C. Tonoli, M. Dalai, P. Migliorati, R. Leonardi, Error resilience performance evaluation of a distributed video codec, in: Picture Coding Symposium, Lisbon, PT, November 2007.
[33] D.P. Varodayan, A. Aaron, B. Girod, Rate-adaptive codes for distributed source coding, EURASIP Signal Process. J. 86 (November 2006) 3123–3130 (Special section on distributed source coding).
[34] Y. Wang, S. Wenger, J. Wen, A.K. Katsaggelos, Error resilient video coding techniques, IEEE Signal Process. Mag. 17 (4) (July 2000) 61–82.
[35] S. Wenger, Error patterns for internet experiments, Technical Report, Joint Video Team (JVT), October 1999.
[36] S. Wenger, H.264/AVC over IP, IEEE Trans. Circuit Syst. Video Technol. 13 (July 2003) 645–656.
[37] T. Wiegand, G.J. Sullivan, G. Bjøntegaard, A. Luthra, Overview of the H.264/AVC video coding standard, IEEE Trans. Circuits Syst. Video Technol. 13 (7) (July 2003) 560–576.
[38] Q. Xu, V. Stankovic, X. Xiong, Layered Wyner–Ziv coding for transmission over unreliable channels, EURASIP Signal Process. J. 86 (November 2006) 3212–3225.
[39] H. Yang, K. Rose, Advances in recursive per-pixel end-to-end distortion estimation for robust video coding in H.264/AVC, IEEE Trans. Circuits Syst. Video Technol. 17 (7) (July 2007) 845–856.
[40] C. Yeo, K. Ramchandran, Robust distributed multi-view video compression for wireless camera networks, in: Visual Communications and Image Processing, Proceedings of SPIE, San Jose, CA, USA, January 2007.
[41] C. Yeo, J. Wang, K. Ramchandran, View synthesis for robust distributed video compression in wireless camera networks, in: Proceedings of the International Conference on Image Processing, San Antonio, TX, USA, September 2007.
[42] R. Zhang, S.L. Regunathan, K. Rose, Video coding with optimal inter/intra-mode switching for packet loss resilience, IEEE J. Sel. Areas Commun. 18 (6) (June 2000) 966–976.