Iterative Two-Dimensional Error Concealment for Low-Complexity Wireless Video Uplink Transmitters

Yongkai Huo, Tao Wang, Robert G. Maunder and Lajos Hanzo

School of ECS, University of Southampton, SO17 1BJ, United Kingdom.

Tel: +44-23-8059 3125, Fax: +44-23-8059 4508

Email: {yh3g09, tw08r, lh}@ecs.soton.ac.uk, http://www-mobile.ecs.soton.ac.uk

Abstract

Since joint source-channel decoding is capable of exploiting the residual redundancy in the encoded source signals for improving the attainable error resilience, it has attracted substantial attention. Motivated by the principle of exploiting the source redundancy at the receiver, in this treatise we study the application of iterative Error Concealment (EC) for low-complexity uplink video communications, where the video signal is modelled by a first-order Markov process. Firstly, we derive reduced-complexity formulas for the first-order Markov modelling aided source decoding. Then we propose a bit-based iterative EC algorithm, where a horizontal and a vertical source decoder are employed for exchanging their information using the iterative decoding philosophy. This scheme may be combined with low-complexity video codecs, provided that they retain some of the redundancy residing in the video signals and are capable of estimating the softbit information representing each bit of the video pixels. As application examples, we test our proposed two-dimensional iterative EC in both Wyner-Ziv video coded and uncompressed video transmission scenarios. Finally, we benchmark the attainable system performance against the existing first-order Markov process based softbit source decoding scheme, where the softbit decoding is performed

The financial support of the RC-UK under the auspices of the India-UK Advanced Technology Centre (IU-ATC) and that of the EU under the CONCERTO project is gratefully acknowledged.

by a one-dimensional Markov model aided decoder, as well as by the existing pixel-domain Wyner-Ziv video scheme. Our simulation results show that E_b/N_0 improvements in excess of 6 dB are attainable by the proposed technique in uncompressed video home-networking applications. Furthermore, up to 21.5% bitrate reduction is achieved by employing our proposed iterative error concealment technique in a Wyner-Ziv video coding scheme.

I. INTRODUCTION

Shannon's separation theorem [1] states that reliable transmission can be accomplished by separate source coding using lossless entropy codes and channel coding under the idealized assumption of Gaussian channels and potentially infinite encoding/decoding delay and complexity. However, a finite-delay, finite-complexity source encoder fails to remove all the redundancy residing in the source signals, such as audio and video. Hence, joint source-channel coding (JSCC) [2] was proposed for additionally exploiting the source statistics for the sake of improved error concealment. Furthermore, Görtz [3], [4] proposed the iterative source and channel decoding (ISCD) philosophy, which performed turbo-like iterative decoding by exchanging extrinsic information between the source encoder and decoder. Softbit source decoding (SBSD) [5] was proposed for error concealment of speech signals by Fingscheidt and Vary using softbit information, where the correction of adjacent speech source samples were modeled by a first-order Markov process. Afterwards, Adrat and Vary [6], [7] developed SBSD for iterative source-channel decoding using both forward- and backward-oriented calculations.

Similar to speech, joint source channel decoding of image and video signals also attracted substantial attention. For example, Kliewer, Görtz and Mertins [8], [9] proposed an ISCD scheme for images modelled by a Markov Random Field (MRF) by exploiting that any pixel can exchange extrinsic information with its eight neighbors for exploiting the residual spatial correlations residing in the encoded image. In [10], Kliewer proposed a novel symbol-based soft-input *a-posteriori* probability (APP) decoder for packetized variable length coded (VLC) [11] source signals transmitted over wireless channels, where the Markov-modelled residual redundancy generated after source encoding was exploited for achieving an improved error protection. In the context of distributed source coding, the authors of [12] proposed an error-resilient JSCC scheme using a Slepian-Wolf (SW) codec, which exploited the knowledge of both

the channel statistics and the correlation between specific video frames and their reference frames. In [13], an H.264 [14] video telephone scheme was proposed using ISCD. Firstly, the H.264 bitstream was segmented into the partitions A, B and C [14]. Then the three partitions were encoded by variable code-rate short block codes (SBC), which artificially imposed redundancy on the H.264-encoded bitstreams. This artificial redundancy was then exploited by the softbit source decoder, which performs iterative decoding by exchanging extrinsic information with the channel decoder. In [15], a double low-density parity-check (LDPC) code was proposed for joint source and channel coding, where two concatenated LDPC codes were employed as the source LDPC and channel LDPC, respectively. At the receiver, the source LDPC and channel LDPC performed joint decoding by exchanging extrinsic information.

On the other hand, as a post-enhancement technique conceived for standard video compression schemes, error concealment (EC) methods [16], [17] have also attracted substantial attention for the sake of delivering flawless video over error-prone networks, whilst using diverse video codecs, such as MPEG-1/2/4, H.263, H.264/AVC [18]. The family of EC schemes may be categorized into spatial-domain (SD) algorithms, temporal-domain (TD) algorithms and hybrid algorithms that exploited both SD and TD information [19]. The SD algorithms conceal the errors using the information within the same frame. As a design-alternative, Chen *et al. [20]* proposed to embed the motion vectors (MV) of a macroblock (MB) into other MBs within the same frame. When a MB is lost at the receiver, its MVs embedded into other MBs will be retrieved for EC. In contrast to SD, the TD EC schemes exploit the inter-frame correlation for EC. The authors of [21] proposed the so-called adaptive error concealment order determination (AECOD) method for TD EC, which adaptively determined the processing order of MBs in a contiguous error-infested region by analyzing the external boundary patterns of the MBs in its neighborhood. For concealing the loss of an entire frame, in hierarchical B-frame based scalable video coding, in [22] the MVs of the lost frame were derived directly from the correctly received previous and, future, video pictures. The hybrid EC schemes jointly exploit the information gleaned from both the successive and the current frame. In [23], the authors utilized both SD interpolation and TD prediction for mitigating the effects of corruption. The conventional EC methods typically conceal the effects of corrupted blocks by exploiting

the knowledge of surrounding blocks, but they tend to precipitate errors, when the corrupted blocks belong to a larger contiguous region. A novel two-stage EC scheme was proposed in [24], where first, a novel spatio-temporal boundary matching algorithm was employed for reconstructing the lost MVs. As the second stage, instead of directly copying the reference MB to represent the recovered pixel values, a partial differential equation (PDE) based algorithm was invoked for smoothing the reconstructed pixels. The authors of [25] proposed to restore lost blocks from both the SD and TD surrounding information using a Gaussian mixture based model, which was obtained offline. However, a common feature of these methods is that they tend to operate in a post-decoding mode of video compression schemes, hence they cannot be readily invoked in uncompressed or low-compression, low-complexity video scenarios for the sake of further quality improvements.

The traditional video coding methods such as MPEG and the ITU-T H.26x codecs impose a high encoder complexity by the discrete cosine transform (DCT) transform, motion compensation, etc which may become excessive in video sensor networks, mobile camera phones and wireless personal area networks (WPAN) [26], for example. Hence in this treatise, we focus our attention on low-complexity, Wyner-Ziv video coding [27] and uncompressed WPAN video scenarios [26].

Suppose X and Y are two independently and identically distributed (i.i.d.) binary sequences, which may be generated either at the same or different locations. The Slepian-Wolf (SW) Theorem [28] states that it is possible to compress X and Y independently using two separate encoders and then to jointly decode them at the receiver, using a rate similar to that as though X and Y were encoded jointly, i.e. as a single sequence, just like in traditional video coding schemes, such as the ITU-T H.26x codecs. Hence the SW philosophy may be viewed as the complement of using a high-complexity encoder and a low-complexity decoder. Since it replies on either uncompressed or on simple independent encoding combined with joint decoding. Hence our method is eminently suitable for the mobile-station (MS) to base-station (BS) uplink (UL). By contrast, the high-complexity H.264-style encoders combined with low-complexity decoding and more suitable for the BS-to-HS downlink (DL), since the BS can afford a higher complexity than the MS. This philosophy was also exploited in the LTE wireless system, where the BS

can afford a higher complexity. To elaborate a little further, we may minimize $(R_X + R_Y)$ to $H(X,Y)$ by jointly decoding X and Y at the receiver, where R_X and R_Y are the minimum rates required for transmitting the source X and Y separately, while $H(X,Y)$ is the joint entropy of X and Y. Wyner and Ziv (WZ) [29] further extended this theory to lossy coding scenarios. Motivated by the emerging source encoding requirements of low-complexity video sensor networks, Wyner-Ziv video coding, also known as distributed video coding (DVC), has attracted substantial attention in recent years, as documented both by Girod and his team [27], as well as by Puri *et al.* [30]. In [31], an efficient channel-quality-aware source-coding algorithm was proposed by Sehgal *et al.*, which combined predictive encoding and the Wyner-Ziv paradigm. Wyner-Ziv coding may be deemed to be a robust video transmission technique conceived for error-infested channels. A small amount of additional side-information, termed as coset information is transmitted for the sake of correcting the "errors" within specific video frames, referred to as "peg frames" [31]. Brites and Pereira [32] proposed a more realistic WZ video coding approach, which performs online estimation of the channel-induced noise (CN) model parameters at the decoder. The method of Brites and Pereira [32] is applicable to both pixel-domain and to transform-domain WZ video codecs. For pixel-domain WZ (PDWZ) video coding, three levels of granularity were proposed, namely frame-, block- and pixel-level granularity. For transform-domain WZ (TDWZ) video coding, DCT-band and DCT-coefficient level granularity was proposed. A context-adaptive Markov random field aided reconstruction algorithm was proposed by Zhang *et al.* [33], which exploits the spatio-temporal correlation by modelling the WZ frames. The Slepian-Wolf and Wyner-Ziv theorems state that the sequences X and Y must be jointly decoded in order to approach the theoretical bound of the joint source entropy $H(X,Y)$. Hence it is possible to apply iterative decoding principles for further exploiting both the intra-frame and inter-frame source correlation at the receiver.

The emerging 60 GHz wireless personal area networks (WPAN) within the IEEE 802.15.3c standard family [34], [35] is designed for short-range (<10 m) transmission of very-high-speed (>2 Gb/s) multimedia data to computer terminals and consumer appliances centered around an individual person's workspace, such as in residential rooms, offices, etc. The WirelessHD specification [36], [37], as another WPAN stan-

dard, increases the maximum data rate to 28 Gb/s, which supports the transmission of either compressed or uncompressed digital high definition (HD) multimedia signals. On the other hand, the transmission of compressed rather than uncompressed video may impose problems in wireless multimedia applications. Firstly, the processing time generates an intrinsic latency, which may violate the delay endget of delay-sensitive applications, such as interactive gaming. Secondly, some video quality degradation is inevitable and remains unrecoverable at the receiver. Thirdly, compressed video streaming is limited to devices, where identical compression techniques must be employed. A transcodec converting between compressed video formats is required, when a device has to relay the received video stream to another device employing a different compression technique. This may increase both the cost and complexity, hence the transmission of uncompressed video was investigated in [26], [38].

A common feature in low-complexity video coding scenarios, such as distributed video coding for example, is that a high amount of redundancy is present in the transmitted video stream, which manifests itself in terms of a high adjacent-pixel correlation and should be exploited at the receiver for concealing the pixel errors in the reconstructed video. Motivated by the congenial principle of iterative JSCC, we design a EC technique, where the video-domain redundancy is exploited for iteratively concealing the errors in the softbit video signals. Against this background, our novel contributions are:

1. We conceive a two-dimensional iterative EC technique, which exchanges extrinsic information between the rows and columns of a video frame. Since the technique is based on the philosophy of the iterative decoding principle, it may also be used for iterative joint source-channel decoding.

2. A substantial benefit of our technique is that it may be combined with low-complexity video codecs, where typically *there is source residual redundancy residing in the video signals. A further prerequisite for the scheme's operation is that the corresponding video decoder is capable of estimating the softbit information of each bit representing the video pixels*.

3. We apply our proposed technique both in a Wyner-Ziv video coding system and in an uncompressed video transmission scenario for, improving the reconstructed video quality.

4. Finally, we reduce the complexity of the one-dimensional decoding algorithms derived by Vary and

his team [6], [7] by designing a novel trellis representation of the Markov process and by deriving its decoding rules.

This rest of this paper is organized as follows. In Section II, we introduce the general architecture of the EC model, which uses the proposed Iterative Horizontal-Vertical Scanline Model (IHVSM) decoder. In Section III, we present the technique of decoding the first-order Markov process as well as the associated iterative decoding principle, which will be employed in the IHVSM decoder introduced in Section II. Then our EC model will be utilized and simulated in two representative video transmission scenarios using low-complexity video transceivers in Section IV. Specifically, the system architecture and the performance of the proposed IHVSM decoder applied in both a pixel-domain Wyner-Ziv coding scheme and in an uncompressed video application will be detailed in Section IV-A and IV-B, respectively. Finally, our conclusions are offered in Section V.

II. Iterative Error Concealment Model

A one-dimensional iterative system model has been proposed and analyzed by Vary and his team [4]–[7] in the context of audio signals. Since their model was designed for one-dimensional signals, it cannot directly exploit the two-dimensional (2D) correlation of practical video signals. In this section, we will detail the iterative EC model conceived for 2D frames video, for the intuitive characterization of our EC model. In this section, we initially focus our attention on the receiver side. The details of applying the 2D iterative EC technique in two specific video transmission scenarios will then be illustrated in Section IV. Let us commence by stipulating the following assumptions:

- x_i: the m-bit pattern of pixel scanned from the original video pixels at time instant i, which is expressed as $\{x_i(0), \cdots, x_i(m-1)\} = x_i \binom{m-1}{0}$;
- m: the number of bits in each m-bit pattern x_i of pixels;
- $X_m = \{0, 1, \cdots, 2^m - 1\}$: the set of all possible values in an m-bit pattern x_i;
- $x_0^t = x_0, \cdots, x_t$: the bit patterns of the 1^{st} frame of the original video consisting of $(t+1)$ m-bit patterns during the time interval spanning from 0 to t;
- $y_0^t = y_0, \cdots, y_t$: potentially error-infested bit pattern of the 1^{st} frame;

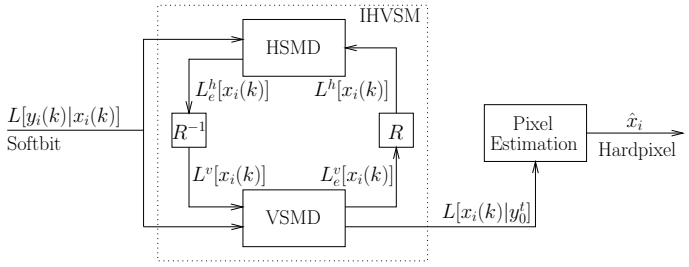

Fig. 1. Iterative error concealment architecture using IHVSM, where R represents reordering of the video pixels.

The model of 2D iterative EC is displayed in Fig. 1, which only accepts the soft bit input and generates the corresponding hard-decision-based pixel as output. We will discuss the details of 2D iterative EC techniques below.

A. Softbit[1] *Input*

Let us now focus our attention on the 1^{st} original video frame carrying $(t+1)$ consecutive and hence correlated bit patterns, eg. $x_0, \cdots, x_t$. Due to the channel-impairments, the receiver can only reconstruct the error-infested version of $x_0, \cdots, x_t$. Again, the 2D iterative EC shown in Fig. 1 only accepts softbit information as its input. Then, the receiver has to estimate the softbit information of each bit representing the original pixels[2] $x_0, \cdots, x_t$, namely $y_0, \cdots, y_t$, which carry the error-infested bit sequence $y_0\binom{m-1}{0}, \cdots, y_t\binom{m-1}{0}$. The softbit information represented by the log-likelihood ratios (LLR) may be readily derived from the softbit patterns $y_0, \cdots, y_t$, yielding $L\left[y_i\left(k\right)|x_i\left(k\right)\right]$ in Fig. 1, which indicates the receiver's confidence in the original m-bit pixel $x_i\left(k\right)$.

B. Error Concealment

Given the softbit LLR input $L\left[y_i\left(k\right)|x_i\left(k\right)\right]$ of Fig. 1, our 2D iterative EC may be invoked for mitigating the effects of channel-errors on the error-infested m-bit sequence of $(t+1)$ pixels $y_0\binom{m-1}{0}, \cdots, y_t\binom{m-1}{0}$. Two stages are involved in the EC process, namely the Iterative Horizontal-Vertical Scanline Model (IHVSM) based decoding and the related pixel estimation, as seen in Fig. 1.

[1]Softbit source decoding was proposed in [5]. Instead of expressing a bit as 0 or 1, a softbit represents one bit of information using a floating-point number, indicating our confidence in that bit.

[2]The ability to estimate the soft value of a bit constitutes an essential condition of applying our EC model.

1) IHVSM based decoding: At the first stage, two similar source decoders are employed, namely the Horizontal Scanline Model Decoder (HSMD) operating in the horizontal direction and the Vertical Scanline Model Decoder (VSMD) proceeding in the vertical direction. Let us now assume that we receive H horizontal scanlines and V vertical scanlines at the receiver. Then the IHVSM decoding may be performed based on the $(H \times V)$-pixel block, which is represented by the $(H \cdot V)$-m-softbit patterns y_i. Here we consider the HSMD as an example for highlighting the decoding process. The m-softbit-based pixel value y_i is used for generating the systematic LLR information $L\left[y_i\left(k\right)|x_i\left(k\right)\right]$, which is then input to the HSMD without the assistance of any channel decoding. However, our technique may be conceived with arbitrary channel codecs. The HSMD also exploits the *a-priori* LLR information $L^h\left[x_i\left(k\right)\right]$ generated from the extrinsic information $L_e^v\left[x_i\left(k\right)\right]$[3] provided by the vertical decoder. The horizontal decoder independently performs source-modelling aided soft decoding of each of the H horizontal scanlines. Hence, following horizontal decoding, the extrinsic LLR information $L_e^h\left[x_i\left(k\right)\right]$ may be generated for all H scanlines. Then the relevant $(H \times V)$-line extrinsic information block will be appropriately reordered in the block R^{-1} of Fig. 1 for generating the *a-priori* information $L^v\left[x_i\left(k\right)\right]$ for the vertical decoder.

For further illustration of the IHVSM decoding process, consider the example of the horizontal and vertical scanlines displayed in Fig. 2, where the pixels connected by the solid/dashed arrows belong to a horizontal/vertical Markov process. Specifically, the 5 pixels of the first row in Fig. 2 are modelled by a 5-pixel Markov process h_1. Let us assume that the 5 pixel values available at the receiver may be expressed as $y_{1,1}, \cdots, y_{1,5}$ of Fig. 2, each of which consists of m soft values indicating the decoder's confidence in each of the m systematic bits[4]. The corresponding *a-priori* LLR information $L^h\left[x_{1,1}\binom{m-1}{0}\right], \cdots, L^h\left[x_{1,5}\binom{m-1}{0}\right]$ for the 5 pixels is provided by the VSMD decoder. The $(5 \times m)$ soft values representing the 5 pixels can exchange extrinsic information with each other, when the HSMD decoder is processing scanline h_1. Similarly, the pixel confidences of all 5 horizontal scanlines may be

[3]The first-order Markov process aided decoding algorithm will be detailed in Section III, along with the associated iterative decoding principle.

[4]The m soft values may be acquired in different ways in different applications. In the application of DVC, the m soft values of the pixels in the WZ frames are estimated from the key frames, which are intra-coded and transmitted to the receiver [27].

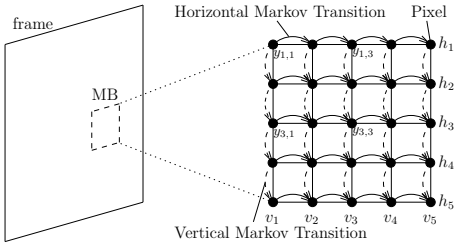

Fig. 2. The structure of horizontal scanlines and vertical scanlines and their corresponding Markov processes. $y_{i,j}$ represents the received pixel at the position (i,j) in the MB, which corresponds to the error-infested version of original pixel $x_{i,j}$.

improved by performing HSMD on $h_1, \cdots, h_5$, respectively. Note that the pixels only shared their information within the horizontal scanlines in the HSMD process. After HSMD decoding, the extrinsic information provided by the (5×5) pixels, namely the $(5\times5\times m)$ soft values of $L_e^h\left[x_{1,1}\binom{m-1}{0}\right], \cdots, L_e^h\left[x_{5,5}\binom{m-1}{0}\right]$ are generated. Then, these $(5\times5\times m)$ extrinsic information values will be reordered into vertical scanlines by the block R^{-1} of Fig. 1, which will be utilized as *a-priori* information by the VSMD decoder. Then the VSMD decoder will decode the vertical Markov process v_1 based on the systematic information $y_{1,1}, \cdots, y_{5,1}$, whilst also exploiting the *a-priori* LLR information $L^v\left[x_{1,1}\binom{m-1}{0}\right], \cdots, L^v\left[x_{5,1}\binom{m-1}{0}\right]$ reordered by R^{-1}. Similar to the HSMD, the VSMD will improve the decoder's confidence related to the pixels by exchanging extrinsic information in the vertical direction. After VSMD processing of the vertical scanlines $v_1, \cdots, v_5$, respectively, a full iteration of the IHVSM scheme is completed and the HSMD will exploit the extrinsic information fed back by the VSMD decoder in order to start the next iteration. This iterative process will terminate after the affordable number of iterations. Observe that any two pixels of a MB are related to each other according to our Markov model and can exchange information with each other with the aid of the IHVSM decoder. Moreover, the IHVSM decoder performs decoding on a block by block basis, as defined in the system.

2) Pixel Estimation: Following the IHVSM decoding process, the relevant *a-posteriori* information is generated at the parameter estimation stage by summing the systematic LLR information $L\left[y_i\left(k\right)|x_i\left(k\right)\right]$, the *a-priori* LLR information $L^v\left[x_i\left(k\right)\right]$ and the extrinsic LLR information $L_e^h\left[x_i\left(k\right)\right]$, where the *a-posteriori* information $L\left[x_i|y_0^t\right]$ may be exploited either by the minimum mean square error (MMSE) decoder or the maximum a *posteriori* probability (MAP) decoder for estimating the m-bit pattern x_i and ultimately to output the estimated original pixel $\hat{x}_i$ [5], which may be formulated as

- MAP estimator

$$\hat{x}_i = \arg\max_{\forall x_i \in X_m} p(x_i|y_0^t); \tag{1}$$

- MMSE estimator

$$\hat{x}_i = \sum_{x_i \in X_m} x_i \cdot p(x_i|y_0^t). \tag{2}$$

Both the MMSE estimator of Eq. (2) and the MAP estimator of Eq. (1) may be deemed to be symbol-based estimators, since they are based on the *a-posteriori* probability $p(x_i|y_0^t)$ of the m-bit pattern x_i, which is conditioned on the received frame of bit patterns y_0^t. However, since the specific bits $x_i(0), \cdots, x_i(m-1)$ of the pixels are independent of each other, if the bit-based *a-posteriori* probability $p\left[x_i(k)|y_0^t\right]$ is provided by the iterative decoding process, the symbol-based *a-posteriori* probability $p(x_i|y_0^t)$ in Eq. (1) and Eq. (2) may be derived from the bit-based information as their product:

$$p(x_i|y_0^t) = \prod_{k=0}^{m-1} p\left[x_i(k)|y_0^t\right]. \tag{3}$$

Furthermore, the symbol-based MMSE estimation rule may also be derived from the bit-based probability $p\left[x_i(k)|y_0^t\right]$ as follows

$$\begin{aligned}\hat{x}_i &= \sum_{x_i \in Xm} x_i \cdot p\left(x_i|y_0^t\right) = \sum_{x_i \in Xm} x_i \cdot \prod_{k=0}^{m-1} p\left[x_i(k)|y_0^t\right] \\ &= \sum_{k=0}^{m-1} 2^k \cdot p\left[x_i(k)=1|y_0^t\right] \sum_{\substack{x_i \in X_m \\ x_i(k)=1}} \prod_{\substack{l=0 \\ l \neq k}}^{m-1} p\left[x_i(l)|y_0^t\right].\end{aligned}$$

It may be readily shown that $\sum\limits_{\substack{x_i \in X_m \\ x_i(k)=1}} \prod\limits_{\substack{l=0 \\ l \neq k}}^{m-1} p\left[x_i(l)|y_0^t\right] = 1$, hence the symbol-based MMSE estimator $\hat{x}_i$ can be derived from the bit-based probabilities as

$$\hat{x}_i \;\; = \;\; \sum_{k=0}^{m-1} 2^k \cdot p\left[x_i(k)=1|y_0^t\right]. \tag{4}$$

Finally, the hard pixel $\hat{x}_i$ of Fig. 1 may be returned as the final estimate of the original pixel x_i.

III. Markov Modelled Softbit Source Decoding with Reduced Complexity

In this section, we focus our attention on the decoding of the first-order Markov process as well as on its iterative decoding principle. The employment of first-order Markov modelling aided softbit source

$$L\left[x_i\left(k\right)|y_0^t\right] = \log \frac{\sum\limits_{\substack{x_i \in X_m \\ x_i(k)=0}} \beta_i\left(x_i\right) \cdot \gamma_i\left(x_i\right) \cdot \sum\limits_{x_{i-1} \in X_m} p\left(x_i|x_{i-1}\right) \cdot \alpha_{i-1}\left(x_{i-1}\right)}{\sum\limits_{\substack{x_i \in X_m \\ x_i(k)=1}} \beta_i\left(x_i\right) \cdot \gamma_i\left(x_i\right) \cdot \sum\limits_{x_{i-1} \in X_m} p\left(x_i|x_{i-1}\right) \cdot \alpha_{i-1}\left(x_{i-1}\right)} \tag{5}$$

decoding (FOMM-SBSD) was discussed in [5]–[7]. The *a-posteriori* log-likelihood ratio of bit $x_i(k)$ conditioned on y_0^t was derived in [6], [7], which may be expressed as in Eq. (5), where the components γ, α and β are defined by Adrat and Vary in [7]. In this section, firstly we propose our novel trellis representation of the first-order Markov source-process in Section III-A. We will then detail the decoding rules of our trellis in Section III-B, which is a reduced complexity version of the technique proposed by Vary's team [7]. Finally, we will present our extrinsic information formula invoked for iterative decoding in Section III-C, while the computational complexity imposed is analyzed in Section III-D.

A. Trellis Representation of First-Order Markov Chain

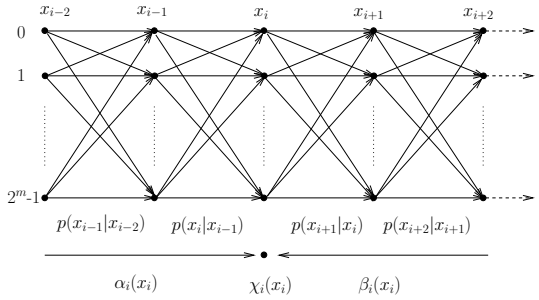

Fig. 3. Trellis of first-order Markov process for BCJR decoding, where $p\left(x_{i+1}|x_i\right)$ is the Markov transition probability.

The traditional trellis representation is detailed in [39]. In contrast to the traditional one, the trellis of the first-order Markov process is shown in Fig. 3, where the m-bit pattern x_i represents the trellis state at time instant i and the probability of transition from state x_i to state x_{i+1} is represented by $p\left(x_{i+1}|x_i\right)$, which is the state transition of the related Markov process. For the m-bit pattern x_i, there are 2^m possible states.

B. BCJR Decoding of First-Order Markov Chain

The existing *a-posteriori* probability generation rule conceived for first-order Markov modeled SBSD is shown in Eq. (5). However, we will demonstrate that the complexity imposed may be reduced by invoking our novel trellis introduced in Section III-A. Let us initially follow the procedure of the classic BCJR

[40] based determination rule of the *maximum a-posteriori probability*. At the receiver, the *a-posteriori* probability of the m-bit pattern $x_i, x_i \in X_m$ conditioned on the specific received frame of m-bit patterns $y_0, \ldots, y_t$ may be expressed as

$$p\left(x_i|y_0^t\right) = \frac{p\left(x_i \wedge y_0^t\right)}{p\left(y_0^t\right)}, \tag{6}$$

where the joint probability $p\left(x_i \wedge y_0^t\right)$ of the m-bit pattern x_i and of the received frame y_0^t may be further formulated as

$$p\left(x_i \wedge y_0^t\right) = \beta_i\left(x_i\right) \cdot \chi_i\left(x_i\right) \cdot \alpha_i\left(x_i\right). \tag{7}$$

In Eq. (7), the components α, β, χ are defined as follows

$$\begin{aligned} \alpha_i\left(x_i\right) &= p\left(y_0^{i-1} \wedge x_i\right) \\ \beta_i\left(x_i\right) &= p\left(y_{i+1}^t|x_i\right) \\ \chi_i\left(x_i\right) &= p\left(y_i|x_i\right). \end{aligned} \tag{8}$$

In Eq. (8), the symbol-based channel information $\chi_i\left(x_i\right) = p\left(y_i|x_i\right)$ may be calculated from the bit-based channel information as

$$\chi_i\left(x_i\right) = C_{\chi_i} \cdot \exp \sum_{k=0}^{m-1} \frac{x_i(k)}{2} \cdot L\left[y_i(k)|x_i(k)\right], \tag{9}$$

where C_{χ_i} is the normalization factor, which solely depends on y_i. Furthermore, similar to the forward recursion calculation in BCJR, the component $\alpha_i\left(x_i\right)$ in Eq. (8) may be formulated as

$$\alpha_i\left(x_i\right) = \sum_{x_{i-1} \in X_m} \chi_{i-1}\left(x_{i-1}\right) \cdot p\left(x_i|x_{i-1}\right) \cdot \alpha_{i-1}\left(x_{i-1}\right).$$

Similarly, the backward recursion calculation of the component $\beta_i\left(x_i\right)$ in Eq. (8) is given by:

$$\beta_i\left(x_i\right) = \sum_{x_{i+1} \in X_m} \beta_{i+1}\left(x_{i+1}\right) \cdot \chi_{i+1}\left(x_{i+1}\right) \cdot p\left(x_{i+1}|x_i\right).$$

The determination of the bit-based *a-posteriori* LLR from the symbol-based *a-posteriori* probability $p\left(x_i|y_0^t\right)$ was presented in [7]. Similarly, the bit-based *a-posteriori* LLR $L\left[x_i\left(k\right)|y_0^t\right]$ may be formulated as

$$L\left[x_i\left(k\right)|y_0^t\right] = \ln \frac{\sum\limits_{\substack{x_i(k)=0 \\ x_i \in X_m}} \beta_i\left(x_i\right) \cdot \chi_i\left(x_i\right) \cdot \alpha_i\left(x_i\right)}{\sum\limits_{\substack{x_i(k)=1 \\ x_i \in X_m}} \beta_i\left(x_i\right) \cdot \chi_i\left(x_i\right) \cdot \alpha_i\left(x_i\right)}. \tag{10}$$

We have now derived the final rule of determining the bit-based *a-posteriori* probability LLR, which is represented in Eq. (10). In contrast to the existing determination rule formulated in Eq. (5), the inner summation $\sum p\left(x_i|x_{i-1}\right)\cdot\alpha_{i-1}\left(x_{i-1}\right)$ is avoided in Eq. (10), which reduces the computational complexity.

Furthermore, the Jacobian logarithm [39] may be readily applied for deriving the log-domain representation of our algorithm.

C. Extrinsic Information Exchange for Iterative Decoding

A limitation of the formulas provided in Section III-B is that they cannot be directly used for iterative decoding, since they cannot exploit the *a-priori* LLR information $L\left[x_i(k)\right]$, which was generated from the *extrinsic* information gleaned from the other decoder involved in the turbo-like iterative decoding process [41]. The rules of iterative source and channel decoding were derived by Vary and his team in [6], [7]. To make use of the *a-priori* LLR information $L\left[x_i(k)\right]$, the combined bit-based log-likelihood information[5] may be utilized as [7]

$$\gamma_i\left(x_i\right)=\exp\sum_{k=0}^{m-1}\frac{\bar{x}_i(k)}{2}\cdot\left\{L\left[x_i(k)\right]+L\left[y_i(k)|x_i(k)\right]\right\}, \tag{11}$$

where the symbol-based m-bit information γ is the combination of the bit-based log-likelihood *a-priori* information $L\left[x_i(k)\right]$ and of the channel information $L\left[y_i(k)|x_i(k)\right]$. We note in this context that γ of Eq. (11) contains more valuable information than the channel information χ. By replacing χ with γ in Eq. (11) we have the following formula:

$$L\left[x_i\left(k\right)|y_0^t\right]=\ln\frac{\sum\limits_{\substack{x_i\in X_m\\ x_i(k)=0}}\beta_i\left(x_i\right)\cdot\gamma_i\left(x_i\right)\cdot\alpha_i\left(x_i\right)}{\sum\limits_{\substack{x_i\in X_m\\ x_i(k)=1}}\beta_i\left(x_i\right)\cdot\gamma_i\left(x_i\right)\cdot\alpha_i\left(x_i\right)}. \tag{12}$$

Moreover the symbol-based *a-posteriori* probability of the first and last m-bit patterns may be expressed as follows

$$\begin{aligned} p\left(x_0\wedge y_0^t\right)&=\beta_0\left(x_0\right)\cdot\gamma_0\left(x_0\right)\cdot p\left(x_0\right)\\ p\left(x_t\wedge y_0^t\right)&=\gamma_t\left(x_t\right)\cdot\alpha_t\left(x_t\right). \end{aligned} \tag{13}$$

[5]Similar to Eq. (9), a constant normalization factor is neglected since it will be canceled during the calculation.

$$L\left[x_i\left(k\right)|y_0^t\right] = L\left[x_i(k)\right] + L\left[y_i(k)|x_i(k)\right] + \ln \frac{\sum\limits_{\substack{x_i \in X_m \\ x_i(k)=0}} \beta_i\left(x_i\right) \cdot \gamma_i^{[ext]}\left[x_i(k)\right] \cdot \alpha_i\left(x_i\right)}{\sum\limits_{\substack{x_i \in X_m \\ x_i(k)=1}} \beta_i\left(x_i\right) \cdot \gamma_i^{[ext]}\left[x_i(k)\right] \cdot \alpha_i\left(x_i\right)} = L\left[x_i(k)\right] + L\left[y_i(k)|x_i(k)\right] + L$$

(14)

Similar to the BCJR decoding technique of classic turbo codes [39], the bit-based *a-posteriori* LLR $L\left[x_i\left(k\right)|y_0^t\right]$ may be split into three components, namely the *a-priori* information $L\left[x_i(k)\right]$, the channel information $L\left[y_i(k)|x_i(k)\right]$ and the extrinsic information $L_e\left[x_i(k)\right]$. Specifically, the bit-based *a-posteriori* information $L\left[x_i\left(k\right)|y_0^t\right]$ may be formulated as in Eq. (14), where the extrinsic information component $\gamma_i^{[ext]}\left[x_i(k)\right]$ may be expressed as

$$\gamma_i^{[ext]}\left[x_i(k)\right] = \exp \sum_{l=0, l\neq k}^{m-1} \frac{\bar{x}_i(l)}{2} \cdot \left\{L\left[x_i(l)\right] + L\left[y_i(l)|x_i(l)\right]\right\}.$$

D. Complexity Analysis

The complexity of the first-order Markov process based decoder can be attributed to the calculation of $\gamma_i\left(x_i\right)$, $\alpha_i\left(x_i\right)$, $\beta_i\left(x_i\right)$ and $L\left[x_i\left(k\right)|y_0^t\right]$. For the existing Markov process based decoding technique of Eq. (5) [7], the computational costs are on the order of $O\left(2^m \cdot t \cdot m\right)$, $O\left(2^{2m} \cdot t\right)$ and $O\left(2^{2m} \cdot t\right)$ for γ, α and β, respectively. Furthermore, in Eq. (5), the inner summation $\sum p\left(x_i|x_{i-1}\right) \cdot \alpha_{i-1}\left(x_{i-1}\right)$ imposes $O\left(2^{2m} \cdot t\right)$ operations. Hence, the calculation of $L\left[x_i\left(k\right)|y_0^t\right]$ has a complexity of $O\left(2^{2m} \cdot t + 2^m \cdot t \cdot m\right)$. Therefore the complexity of the existing Markov process based decoding algorithm is on the order of $O\left(3 \cdot 2^{2m} \cdot t + 2 \cdot 2^m \cdot t \cdot m\right)$. Similarly, our proposed BCJR decoder of the first-order Markov process has a complexity of $O\left(2 \cdot 2^{2m} \cdot t + 2 \cdot 2^m \cdot t \cdot m\right)$. Generally speaking, the proposed decoding technique achieves a 33% computational complexity reduction.

Note that the decoding complexity of the first-order Markov process increases exponentially with the number of bits per symbol m. Hence, a quantizer may be employed for striking a tradeoff between the complexity imposed and the attainable performance.

IV. Performance Analysis

Again, low-complexity video transceivers exhibit a high amount of residual redundancy in the video signals received at the receiver. In this section, we will consider two representative video transmission scenarios, namely a Wyner-Ziv video coding [27] and an uncompressed video transmission scenario. In both of the scenarios, we will firstly present the system configuration, including the system parameters and the key techniques employed.

Note that the peak to-signal-noise ratio (PSNR) metric is employed for quantifying the reconstructed video quality in this section. To avoid having infinite PSNR values when a video is perfectly reconstructed, we artificially set the total averaged mean squared error (MSE) value between the reconstructed and the original frame to a minimum value of 1. This is justified, since the same technique is employed in the H.264 reference software JM. Hence the maximum unimpaired video PSNR that may be obtained at the receiver is about 48.1 dB.

A. IHVSM aided Wyner-Ziv Video Coding

In the Wyner-Ziv video coding schemes of Girod *et al.* [27], as well as of Brites and Pereira [32], the side information (SI) of the WZ frame is estimated from the so-called key frames, which are intra-coded by standard codecs, such as H.264 for example. Furthermore, a part of the parity bits of WZ frames is transmitted to the receiver upon request for the sake of assisting the receiver to recover the WZ frames. However, the pixel-correlation within WZ frames is not exploited. In this section, we will combine our EC scheme with the the turbo-coding aided WZ codec [32], in order to iteratively conceal the softbit errors returned by the turbo decoder, thereby implicitly reducing the requested bitrate.

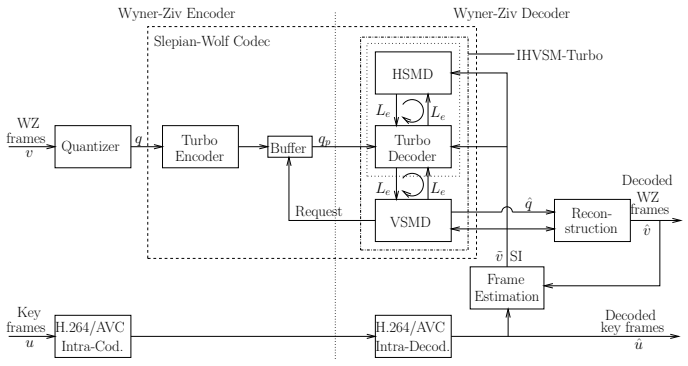

Fig. 4. System architecture of our proposed IHVSM-PDWZ system, which is based on the PDWZ system proposed in [32].

1) System Configuration: Generally, two types of techniques exist in the literature for encoding the WZ frames, namely pixel-domain WZ (PDWZ) [27], [32] and transform-domain WZ (TDWZ) [32] video coding. The PDWZ codec imposes a lower complexity on the encoder than the TDWZ scheme, hence - as expected - it exhibits a worse rate-distortion (RD) performance. In this section, we apply the proposed IHVSM algorithm in the context of a PDWZ scheme for improving the attainable RD performance. The Wyner-Ziv coding scheme advocated in [32], [42] is based on the structure proposed in [43], [44], with a number of further refinements proposed. We opted for the PDWZ system of [32] as our benchmarker, while some further details are illustrated in [42]. Note that the proposed IHVSM may also be employed in other WZ video coding schemes provided that the video decoders at the receiver are capable of estimating the softbit information of the video pixels.

Let us now proceed by detailing the turbo-coded (IHVSM-Turbo) PDWZ system of Fig. 4. At the transmitter, the key frame u is intra-coded by the H.264/AVC video codec, whilst the WZ frame v is uniformly quantized for generating the 2D signal q. Then the bit-pattern q is encoded by the classic turbo codec [41], where only the parity bits q_p will be buffered and transmitted to the receiver. At the receiver, the key frame is reconstructed using the H.264/AVC intra-decoder. The decoded key frame $\hat{u}$ will then be utilized by the frame estimator[6] of Fig. 4 for estimating the side information (SI) frame $\tilde{v}$. Then, our IHVSM-Turbo decoder of Fig. 4 will perform three-stage iterative decoding [45] using the estimated SI frame $\tilde{v}$ and the received parity bits q_p. For the IHVSM-Turbo scheme, the SI frame $\tilde{v}$ may be used as the systematic information of the corresponding WZ frame v, which will be exploited by all the three decoding components, namely the HSMD, the turbo codec, and the VSMD of Fig. 4. For the HSMD and the VSMD components, the technique introduced in Section III-C is utilized for iterative decoding. Furthermore, the HSMD and the turbo codec constitute the inner decoding component, while the VSMD is the outer one. Similarly to most proposals found in the literature [27], [32], the "request-and-decode"[7]process of [27] is repeated, until an acceptable probability of symbol error is reached. Finally, the quantized WZ frame is

[6]More details of the frame estimation process can be found in [32]. Here we focus on our proposed IHVSM-Turbo scheme.

[7]When failing in decoding a WZ frame, the receiver sends a request to the transmitter, which will transmit additional parity bits to the receiver. Then the receiver re-decodes the WZ frame upon the accumulated parity bits.

	Hall	Foreman	Coastguard
Representation	YUV 4:2:0	YUV 4:2:0	YUV 4:2:0
Format	QCIF	QCIF	QCIF
Bits Per Pixel	8	8	8
FPS	15	30	15
GOP	2	2	2
Number of Frames	165	300	150
Channel	Perfect	Perfect	Perfect

TABLE I

TABLE OF PARAMETERS EMPLOYED FOR THE VIDEO SEQUENCES.

recovered as $\hat{q}$, which will be utilized together with the estimated SI frame $\tilde{v}$ for reconstructing the WZ frame $\hat{v}$.

In the following, we benchmark the proposed IHVSM-PDWZ scheme's performance against that of the PDWZ system of Brites and Pereira [32], where the traditional turbo codec was employed. Moreover, the corresponding results recorded for the conventional intra-frame and inter-frame video coding scheme employing the H.263 and H.264 codecs are also provided as baseline benchmarker curves in Section IV-A.2. The RD performance of the well-known transform-domain DISCOVER [46] codec will also be included for visualizing the gap between the pixel domain WZ and transform domain WZ. Three sequences are tested, namely the Hall monitor, the Foreman and the Coastguard sequences, whose parameters are shown in Table I. In all the experiments, the group of picture (GOP) parameter was set to 2; the bitrate/PSNR of both the WZ and the key frames was also taken into account. The RD results recorded for both the PDWZ and IHVSM-PDWZ schemes in Fig. 4 were parameterized by the number of WZ coded bitplanes, which was set to m =1, 2, 3 or 4 most significant bit (MSB) planes, as usual in the pixel-domain WZ video coding literature. This was arranged by invoking the uniform quantizer shown in Fig. 4. In [42], each bitplane of the MSB planes was transmitted separately and each bitplane was then refined based on the previously decoded bitplanes [47]. However, in our system, all bitplanes of the MSB planes were transmitted together, which allowed us to reduce the number of "request-and-decode" processes defined in [27]. The remaining parameters of our system were identical to those in [32], [42]. Note that

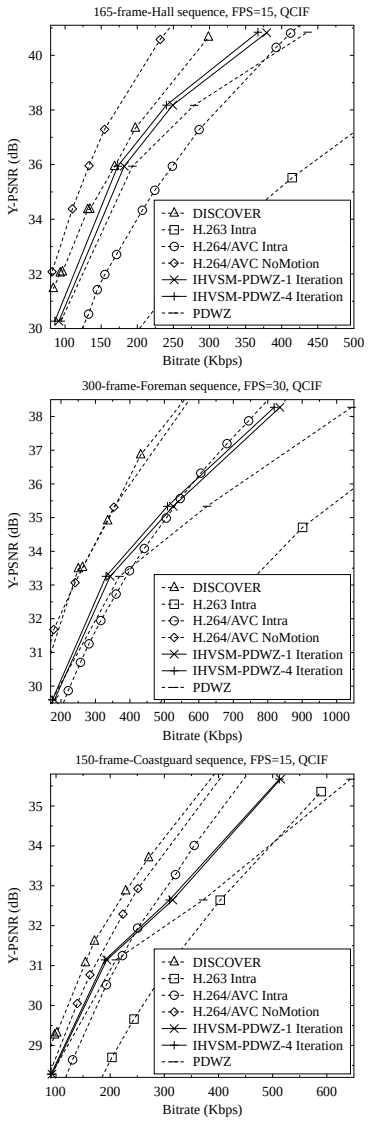

Fig. 5. Rate-distortion performance comparison of the IHVSM-PDWZ codec and the benchmarkers, H.263 Intra, H.264/AVC Intra, H.264/AVC NoMotion, PDWZ [32] and DISCOVER [46]. Hall sequence, Foreman sequence and Coastguard sequence.

for the IHVSM-PDWZ schemes, the size of the Markov Model's State Transition Table (MMSTT) was of $(2^m \times 2^m)$ elements. Specifically, for the Hall sequence associated with $m = 4$, the size of the MMSTT side information was (16×16) floating-point values for 165 QCIF frames[8].

2) Numerical Results: The simulation results recorded for the three sequences are displayed in Fig. 5. Observe from Fig. 5 that for the Hall sequence the IHVSM-PDWZ using a single iteration reduces the bitrate by 3%, 5.9%, 10.7% and 13% in comparison to the PDWZ scheme for $m = 1, \cdots, 4$ MSB

[8]In our simulations, the transmitter scans all adjacent pixel pairs and records their difference, which may be modelled by a Laplacian distribution. The required MMSTT is then generated at the receiver using the received Laplace parameters. Furthermore, since the key frames are available at the decoder, an alternative technique is to estimate the MMSTT from the adjacent key frames.

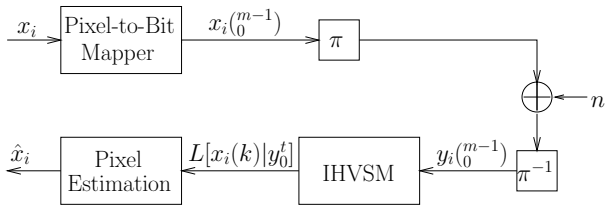

Fig. 6. Transmission of uncompressed video through wireless channel.

planes, respectively. When $I = 4$ iterations were employed by the IHVSM-PDWZ, the bitrate reduction was increased to 8.5%, 10.5%, 13.8% and 15.8% for $m = 1, \cdots, 4$ MSB planes, respectively. This is due to the fact that our IHVSM scheme is capable of estimating and correcting the extrinsic information gleaned from the turbo codec during each iteration. Similarly, observe from Fig. 5 that for the Foreman sequence the IHVSM-PDWZ using $I = 4$ iterations reduces the bitrate by 0%, 10.5%, 18.4%, 21.5% for $m = 1, \cdots, 4$ MSB planes, respectively. For the Coastguard sequence, the IHVSM-PDWZ using $I = 4$ iterations reduced the bitrate by 1.1%, 9.6%, 16.8%, 20.4% for $m = 1, \cdots, 4$ MSB planes, respectively. For all the test sequences, the bitrate reduction increased with the number of MSB planes. The reason for this is that the receiver is more unlikely to be able to estimate the SI of WZ frame accurately from the received key frames, while our IHVSM scheme is capable of effectively concealing the errors in the turbo decoding process. As suggested by the simulation results, the IHVSM-PDWZ is unable to substantially reduce the bitrate by increasing the number of iterations. However, the IHVSM-PDWZ fails to match the performance of the transform-domain DISCOVER codec.

B. Uncompressed Video Transmission

In this section uncompressed video will be transmitted though a Rayleigh channel. The demodulator estimates the softbit information at the receiver, thereby meeting the essential prerequisite we emphasized in Section II for the application of our EC technique. Hence our EC model will be employed for concealing the error effects imposed on the video frames.

1) System Configuration: In this section, we consider the scenario of uncompressed video transmission, which may be employed for home networking [26]. The system's architecture is displayed in Fig. 6. At

	Akiyo	Coastguard	Football
Representation	YUV 4:2:0	YUV 4:2:0	YUV 4:2:0
Format	QCIF	QCIF	QCIF
Bits Per Pixel	8	8	8
FPS	15	15	15
Number of Frames	30	30	30
Bitrate	524 kbps	1924 kbps	3168 kbps
"Natural" Code Rate	1/8.7	1/2.37	1/1.44
Channel	Unc-Ray	Unc-Ray	Unc-Ray
Modulation	BPSK	BPSK	BPSK

TABLE II

TABLE OF PARAMETERS EMPLOYED FOR THE VIDEO SEQUENCES. $Unc\text{-}Ray$ STANDS FOR UNCORRELATED RAYLEIGH.

time instant i, the transmitter has to convey a video pixel x_i, which is mapped to the m-bit pattern $x_i \binom{m-1}{0}$. This pixel-to-bit mapper may include a quantization function [18]. We treat $(t+1)$ consecutive and hence correlated m-bit patterns eg. $x_0, \cdots, x_t$ as a frame. Let us consider the first 2D video frame for example, which will be interleaved by a bit-based interleaver of length $(t+1) \cdot m$. Then the signals are transmitted to the receiver through a Rayleigh channel without channel coding using BPSK modulation. At the receiver, the demodulator will generate the softbit information of the video pixels, which will be input to our proposed EC model, namely to the *IHVSM* and *pixel estimation* blocks of Fig. 6. The reconstructed hard-decision-based pixel $\hat{x}_i$ can then be acquired after the pixel estimation stage of Fig. 6. Note that for the sake of low-complexity, the system does not employ any channel encoder. However, a channel encoder may be invoked for error protection, while the corresponding iterative decoding may then be performed by a three-stage decoder [45].

We compare the performance of our scheme recorded for the video sequences Akiyo, Coastguard and Football against that of the existing system model. Video sequences stored in (176×144)-pixel quarter common intermediate format (QCIF) and 4:2:0 YUV representation are employed. Moreover, the horizontal and vertical decoders perform iterative decoding based on (8×8)-pixel blocks. Each QCIF luminance frame is divided into (22×18) blocks and each QCIF chroma frame is divided into (11×9)

blocks. Note that for simplicity, the uncompressed video bits are transmitted through an uncorrelated non-dispersive Rayleigh channel using BPSK modulation. These parameters are summarized in Table II. The MMSTT of the first-order Markov model was derived from the original video pixels, which may be utilized by the horizontal and vertical source decoders as side-information for improving the achievable error resilience. We mainly rely on two types of curves for characterizing the video quality, namely the peak signal-to-noise ratio (PSNR) versus the channel SNR per bit, namely E_b/N_0 curves and the bit error ratio (BER) versus E_b/N_0 curves.

Shannon's channel capacity theorem [1] was proposed for the transmission of i.i.d source. Hence, to be in line with the channel capacity theory, we have to consider the true entropy of the video sequence, when calculating the energy efficiency per bit. More explicitly, any redundancy inherent in the encoded sequence has to be taken into account by shifting the BER vs E_b/N_0 curves to the right, regardless, whether the redundancy is natural source redundancy or whether it was artificially imposed by channel coding. In our case, substantial redundancy resides in the video source signal, since here we do not employ any video encoder. Hence the true amount of non-redundant information transmitted to the receiver is given by the entropy of the video sequence. Assuming that the total uncompressed size of a video file is S_r bits and the entropy of this video source file is S_e, we might interpret the raw video file as being "naturally" losslessly encoded from S_e i.i.d bits, to generate S_r bits where the code rate is $r = S_e/S_r$. According to Shannon's source coding theorem [1], the entropy of the video source file represents the lowest achievable rate at which the source may be losslessly represented. Hence in our simulations the E_b/N_0 (dB) value is calculated as $E_b/N_0 = 10\log_{10}\frac{E_b S_r}{N_0 S_e}$. However, no widely recognized technique exists for quantifying the entropy of a realistic video source. As a practical solution, we opted for using the near-lossless coding mode of the H.264 codec [14], [18] to encode the source video for the sake of approximating its entropy. The "natural" code rates of the Akiyo, Coastguard and Football sequences used in our simulations are listed in Table II. Quantitatively, we found that the "natural" code rates of the three sequences were $1/8.7$, $1/2.37$ and $1/1.44$ for the scenario considered, which corresponds to the maximum achievable compression ratios of 8.7, 2.37 and 1.44, respectively. The corresponding bitrates and PSNR values are

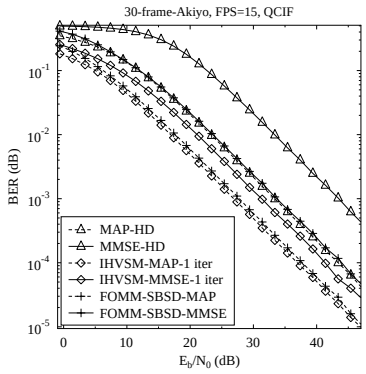

Fig. 7. BER vs E_b/N_0 comparison of MAP- and MMSE-based pixel estimation for a Rayleigh channel. Akiyo sequence.

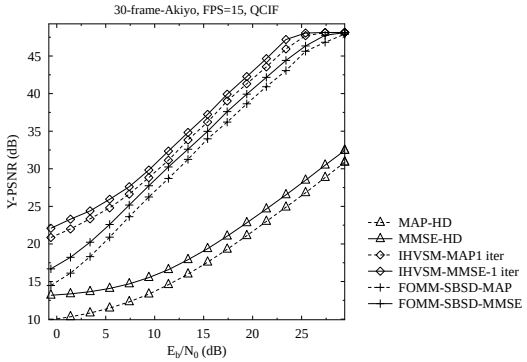

Fig. 8. Y-PSNR vs E_b/N_0 comparison of MAP- and MMSE-based pixel estimation for a Rayleigh channel. Akiyo sequence.

also shown in Table II, where the PSNR values correspond to the maximum quality of the error-freely source decoded video at the receiver.

	MAP-HD	MMSE-HD	FOMM-SBSD	IHVSM
Dimension	1	1	1	2
MIN Bits to Decode	1	8	64	4096
Side information	None	None	1×MMSTT	1×MMSTT

TABLE III

COMPARISON OF IHVSM AND THE BENCHMARKERS: MAP-HD [5], MMSE-HD [5] AND FOMM-SBSD [7].

We will benchmark the performance of our system against three schemes, namely against that of the MAP-based and MMSE-based hard decoding schemes (MAP-HD/MMSE-HD), where no softbit source decoding is employed, as well as against the FOMM-SBSD scheme, where no IHVSM decoding is employed. In the latter case only one of the two decoders is activated. A brief comparison of the four schemes is shown in Table III.

2) Numerical Results: Firstly, the BER versus E_b/N_0 performance comparison of the MAP based and MMSE based pixel estimation using the Akiyo sequence is presented in Fig. 7, while the corresponding Y-PSNR versus E_b/N_0 results are presented in Fig. 8. As suggested by both the BER and Y-PSNR figures, the IHVSM-MAP decoder substantially outperforms both the FOMM-SBSD-MAP and the MAP-HD. Specifically, at BER of (3×10^{-2}) the IHVSM-MAP outperforms the FOMM-SBSD-MAP by about 0.65 dB, while the MAP-HD is outperformed by about 6.3 dB. Similarly, at a Y-PSNR of 37.5 dB the IHVSM-MAP outperforms the FOMM-SBSD-MAP by about 2.5 dB. Moreover, the IHVSM-MMSE decoder significantly outperforms both the FOMM-SBSD-MMSE and the MMSE-HD. Specifically, at a BER of 10^{-2} the IHVSM-MMSE outperforms the FOMM-SBSD-MMSE by about 2.5 dB. Similarly, at a Y-PSNR of 40 dB the IHVSM-MMSE outperforms the FOMM-SBSD-MMSE by about 2 dB in terms of the E_b/N_0 required.

Observe from Fig. 7 and Fig. 8 that we can achieve a BER of 3×10^{-3} at E_b/N_0 of 11.9 dB and 15.6 dB for the MAP-aided and MMSE-based IHSVM, respectively. In contrast to this, we can achieve a Y-PSNR of 40 dB at an E_b/N_0 of 18.1 dB and 17.4 dB for the MAP-aided and MMSE-based IHSVM, respectively. We may conclude that the MAP estimator is capable of achieving a lower BER, while the MMSE estimator may achieve an increased PSNR. Hence, to achieve an increased PSNR, the MMSE based pixel estimation should be employed.

To provide further insights, we present simulation results for two high-dynamic video sequences in Fig. 9, namely for the Coastguard and Football sequences. Again, the MMSE based estimator is capable of achieving a higher PSNR than the MAP based estimator. We only present the Y-PSNR versus E_b/N_0 curves in Fig. 9 using the MMSE based estimator. As suggested by Fig. 9, the IHVSM-MMSE substantially outperforms both the FOMM-SBSD-MMSE and the MMSE-HD. Specifically, at a Y-PSNR of 40 dB the IHVSM-MAP outperforms the FOMM-SBSD-MAP by about 6 dB and 3.5 dB for the Coastguard and Football sequences, respectively. However, for the IHVSM scheme, the Y-PSNR decreases upon increasing the number of iterations at low E_b/N_0 values. This is due to the fact that the first-order Markov process is unable to perfectly match the statistics of the real video, since the pixel values change dramatically

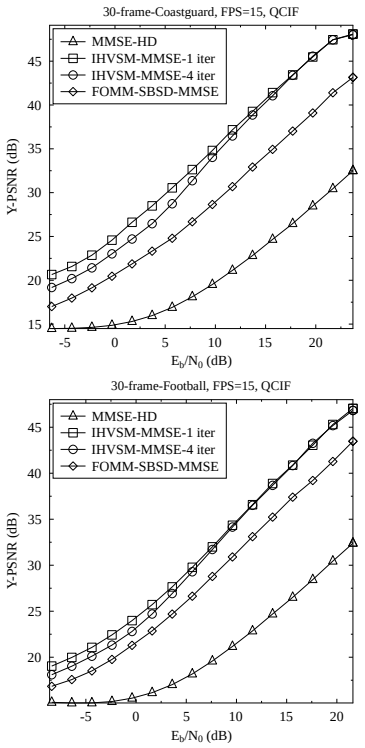

Fig. 9. Y-PSNR vs E_b/N_0 for a Rayleigh channel when the MMSE is employed for pixel estimation. Coastguard and Football sequences.

at the scene borders in real video scenes. The visual comparison of the decoded frames for the Akiyo, Coastguard and Football sequences are shown in Fig. 10.

Fig. 10. Comparison of the first frames recorded at E_b/N_0 of 9.4 dB, 3.7 dB and 1.6 dB for the Akiyo, Coastguard and Football sequences, decoded by MMSE-HD, FOMM-SBSD and IHVSM. MMSE-based pixel estimation is employed.

V. Conclusions

In this paper, we proposed a symbol-based model of iterative source decoding for video receiver enhancements, which may be combined with channel decoding. Furthermore, a reduced-complexity first-order Markov model based source decoder was derived. Iterative decoding was performed by exchanging extrinsic information between two source decoders, namely the HSMD and the VSMD. Our simulation

results showed that the proposed system model substantially improves the achievable error resilience and may facilitate a transmit power reduction in excess of 6 dB in uncompressed video coding. Furthermore, by incorporating the IHVSM scheme into WZ video coding, we were able to reduce the bitrate by up to 21.5% compared to the PDWZ benchmarker system.

Our future work will focus on iterative decoding exchanging extrinsic information between the source decoder and channel decoder.

REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, June and October 1948.

[2] K. Sayood and J. Borkenhagen, "Use of residual redundancy in the design of joint source/channel coders," *IEEE Transactions on Communications*, vol. 39, pp. 838–846, June 1991.

[3] N. Görtz, "Joint source channel decoding using bit-reliability information and source statistics," *International Symposium on Information Theory*, p. 9, August 1998.

[4] N. Görtz, "On the iterative approximation of optimal joint source-channel decoding," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 1662–1670, September 2001.

[5] T. Fingscheidt and P. Vary, "Softbit speech decoding: A new approach to error concealment," *IEEE Transaction on Speech and Audio Processing*, vol. 9, pp. 240–251, March 2001.

[6] M. Adrat, R. Vary, and J. Spittka, "Iterative source-channel decoder using extrinsic information from softbit-source decoding," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2653–2656, May 2001.

[7] M. Adrat and P. Vary, "Iterative source-channel decoding: Improved system design using Exit charts," *EURASIP Journal on Applied Signal Processing*, vol. 2005, pp. 928–941, January 2005.

[8] J. Kliewer, N. Görtz, and A. Mertins, "On iterative source-channel image decoding with Markov random field source models," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. iv–661–iv–664, August 2004.

[9] J. Kliewer, N. Görtz, and A. Mertins, "Iterative source-channel decoding with Markov random field source models," *IEEE Transactions on Signal Processing*, vol. 54, pp. 3688–3701, October 2006.

[10] J. Kliewer and R. Thobaben, "Iterative joint source-channel decoding of variable-length codes using residual source redundancy," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 919–929, May 2005.

[11] L. Hanzo, R. G. Maunder, J. Wang, and L.-L. Yang, *Near-Capacity Variable-Length Coding: Regular and Exit-Chart Aided Irregular Designs*. John Wiley & Sons Ltd, 2010.

[12] Y. Zhang, C. Zhu, and K.-H. Yap, "A joint source-channel video coding scheme based on distributed source coding," *IEEE Transactions on Multimedia*, vol. 10, pp. 1648–1656, December 2008.

[13] Nasruminallah and L. Hanzo, “EXIT-chart optimized short block codes for iterative joint source and channel decoding in H.264 video telephony,” *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 4306–4315, October 2009.

[14] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, *ITU-T Rec. H.264/ISO/IEC 14496-10 AVC: Advanced Video Coding for Generic Audiovisual Services*, March 2010.

[15] M. Fresia, F. Pérez-Cruz, H. Poor, and S. Verdú, “Joint source and channel coding,” *IEEE Signal Processing Magazine*, vol. 27, pp. 104–113, November 2010.

[16] Y. Wang and Q.-F. Zhu, “Error control and concealment for video communication: a review,” *Proceedings of the IEEE*, vol. 86, pp. 974–997, May 1998.

[17] Y. Guo, Y. Chen, Y.-K. Wang, H. Li, M. Hannuksela, and M. Gabbouj, “Error resilient coding and error concealment in scalable video coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 781–795, June 2009.

[18] L. Hanzo, P. Cherriman, and J. Streit, *Video Compression and Communications: From Basics to H.261, H.263, H.264, MPEG2, MPEG4 for DVB and HSDPA-Style Adaptive Turbo-Transceivers*. New York: John Wiley, 2007.

[19] Y. Zhang, X. Xiang, D. Zhao, S. Ma, and W. Gao, “Packet video error concealment with auto regressive model,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 12–27, January 2012.

[20] S. Chen and H. Leung, “A temporal approach for improving intra-frame concealment performance in H.264/AVC,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 422–426, March 2009.

[21] X. Qian, G. Liu, and H. Wang, “Recovering connected error region based on adaptive error concealment order determination,” *IEEE Transactions on Multimedia*, vol. 11, pp. 683–695, June 2009.

[22] X. Ji, D. Zhao, and W. Gao, “Concealment of whole-picture loss in hierarchical B-picture scalable video coding,” *IEEE Transactions on Multimedia*, vol. 11, pp. 11–22, January 2009.

[23] S.-C. Hsia, S.-C. Cheng, and S.-W. Chou, “Efficient adaptive error concealment technique for video decoding system,” *IEEE Transactions on Multimedia*, vol. 7, pp. 860–868, October 2005.

[24] Y. Chen, Y. Hu, O. Au, H. Li, and C. W. Chen, “Video error concealment using spatio-temporal boundary matching and partial differential equation,” *IEEE Transactions on Multimedia*, vol. 10, pp. 2–15, January 2008.

[25] D. Persson, T. Eriksson, and P. Hedelin, “Packet video error concealment with Gaussian mixture models,” *IEEE Transactions on Image Processing*, vol. 17, pp. 145–154, February 2008.

[26] H. Singh, J. Oh, C. Kweon, X. Qin, H.-R. Shao, and C. Ngo, “A 60 GHz wireless network for enabling uncompressed video communication,” *IEEE Communications Magazine*, vol. 46, pp. 71–78, December 2008.

[27] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, “Distributed video coding,” *Proceedings of the IEEE*, vol. 93, pp. 71–83, January 2005.

[28] D. Slepian and J. Wolf, “Noiseless coding of correlated information sources,” *IEEE Transactions on Information Theory*, vol. 19, pp. 471–480, July 1973.

[29] A. Wyner and J. Ziv, “The rate-distortion function for source coding with side information at the decoder,” *IEEE Transactions on Information Theory*, vol. 22, pp. 1–10, January 1976.

[30] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A video coding paradigm with motion estimation at the decoder," *IEEE Transactions on Image Processing,*, vol. 16, pp. 2436–2448, October 2007.

[31] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv coding of video: an error-resilient compression framework," *IEEE Transactions on Multimedia*, vol. 6, pp. 249–258, April 2004.

[32] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, pp. 1177–1190, September 2008.

[33] Y. Zhang, H. Xiong, Z. He, S. Yu, and C. W. Chen, "Reconstruction for distributed video coding: A context-adaptive Markov random field approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, pp. 1100–1114, August 2011.

[34] R. Fisher, "60 GHz WPAN standardization within IEEE 802.15.3c," in *International Symposium on Signals, Systems and Electronics, 2007. ISSSE '07*, pp. 103–105, July 30-August 2 2007.

[35] C. Park and T. Rappaport, "Short-range wireless communications for next-generation networks: UWB, 60 GHz millimeter-wave WPAN, and ZigBee," *IEEE Wireless Communications*, vol. 14, pp. 70–78, August 2007.

[36] http://www.wirelessHD.org, *WirelessHD Specification Overview*, October 2007.

[37] D. Pepe and D. Zito, "60-GHz transceivers for wireless HD uncompressed video communication in nano-era CMOS technology," in *15th IEEE Mediterranean Electrotechnical Conference (MELECON)*, pp. 1237–1240, April 2010.

[38] S.-E. Hong and W. Y. Lee, "Flexible unequal error protection scheme for uncompressed video transmission over 60GHz multi-Gigabit wireless system," in *Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–6, July 31-August 4 2011.

[39] L. Hanzo, T.H.Liew, B.L.Yeap, R. Tee, and S. Ng, *Turbo Coding, Turbo Equalisation and Space-Time Coding*. New York: John Wiley, 2011.

[40] L.R. Bahl and J. Cocke and F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimising symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[41] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo codes," in *Proceedings of the International Conference on Communications*, (Geneva, Switzerland), pp. 1064–1070, May 1993.

[42] C. Brites, *Exploiting Correlation Noise Modeling in Wyner-Ziv Video Coding*. PhD thesis, Instituto Superior Técnico, Technical University of Lisbon, 2011.

[43] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002*, vol. 1, pp. 240–244, November 2002.

[44] A. Aaron, E. Setton, and B. Girod, "Towards practical Wyner-Ziv coding of video," in *Proceedings of International Conference on Image Processing, ICIP 2003*, vol. 3, pp. 869–872, September 2003.

[45] Nasruminallah and L. Hanzo, "Near-capacity H.264 multimedia communications using iterative joint source-channel decoding," *IEEE Communications Surveys and Tutorials*, vol. 14, pp. 538–564, Second Quarter 2012.

[46] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Picture Coding Symposium*, vol. 6, (Lisbon, Portugal), November 2007.

[47] S. Cheng and Z. Xiong, “Successive refinement for the Wyner-Ziv problem and layered code design,” *IEEE Transactions on Signal Processing*, vol. 53, pp. 3269–3281, August 2005.