Multimed Tools Appl (2018) 77:7327–7355
DOI 10.1007/s11042-017-4635-8

A joint correlation noise estimation and decoding algorithm for distributed video coding

Yaser Mohammad Taheri[1] · **M. Omair Ahmad**[1] · **M. N. S. Swamy**[1]

Received: 11 August 2016 / Revised: 26 January 2017 / Accepted: 20 March 2017 /
Published online: 1 April 2017

Abstract Distributed video coding is relatively a novel video coding paradigm that enables a lower complex video encoding compared to conventional video coding schemes, at the expense of a higher-complexity decoder. Improving the rate-distortion and coding efficiency is a challenging problem in distributed video coding. Using a suitable correlation noise model along with an accurate estimation of its parameter can lead to an improved rate-distortion performance. In a distributed video codec, the Wyner-Ziv frames are not available at the decoder. In addition, the correlation noise is not stationary and its statistics vary within each frame and in its corresponding transform coefficient bands. Hence, the estimation of the correlation noise model parameter is not a feasible task. In this paper, a new decoder is proposed to estimate the correlation noise parameter and carry out the decoding process progressively and recursively on an augmented factor graph. In the proposed decoder, a recursive message passing algorithm is used for decoding the bitplanes corresponding to each DCT band in a WZ frame, and simultaneously, for estimating and refining the correlation noise distribution parameter. To approximate the posterior distribution of the correlation noise parameter, and consequently, derive a closed-form expression for the messages on the augmented factor graph, a variational Bayes algorithm is employed. Extensive simulations are carried out to show that using the proposed decoder leads to considerable improvement in the rate-distortion performance of the distributed video codec, particularly on video sequences with fast motions.

✉ M. Omair Ahmad
omair@ece.concordia.ca

Yaser Mohammad Taheri
y_moh@ece.concordia.ca

M. N. S. Swamy
swamy@ece.concordia.ca

[1] Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, H3G 1M8, Canada

Keywords Distributed video coding · Wyner-Ziv frame · Variational Bayes · Correlation noise

1 Introduction

In recent times, many up-stream applications, such as wireless surveillance video and mobile camera, have emerged wherein the size and battery life of the transmitting device are of paramount importance. For such applications, a low complexity encoder is desirable at the expense of a highly complex decoder [12]. A new video coding paradigm, known as distributed video coding (DVC), has been proposed to fulfill this requirement [17]. In this video coding paradigm, the source statistics are exploited only at the decoder. DVC is a new coding scheme that is not yet fully developed, but is now receiving more attention from the research community [1–4]. The theoretical for DVC is based on Slepian-Wolf [29] and Wyner-Ziv [34] theorems. According to the Slepian-Wolf theorem, if two correlated sources X and Y are encoded separately and decoded jointly, we can achieve a minimum coding rate, which is the same as that of joint encoding and joint decoding. Wyner and Ziv extended the Slepian-Wolf theorem for lossy coding with a decoder side information (SI). The Wyner-Ziv (WZ) coding technique has been widely used in DVC. Most of the DVC schemes developed in the literature are based on the Stanford architecture [1], which is one of the first practical implementations of DVC. In these DVC schemes, the video sequences are first divided into Group of Pictures (GOPs). The first frame of each GOP, called a key frame, is encoded using the conventional intra-frame encoder, for example, H.264/AVC in the intra-mode. The remaining frames in a GOP, called WZ frames, are encoded based on channel codes such as turbo codes or LDPC codes. In the transform-domain DVC [2], a 4×4 discrete cosine transform (DCT) is applied on each WZ frame and the corresponding coefficients organized into 16 DCT coefficient bands X_i, $i = 1, 2, ..., 16$. After quantizing each coefficient band into different levels depending on the target quality, bitplanes are extracted for each quantized coefficient band. Subsequently, each bitplane goes through a rate-adaptive channel encoder, to produce some error-correcting bits (parity bits or syndrome bits). These bits are stored in a buffer and transmitted incrementally upon a request by the decoder received through a feedback channel. At the decoder, after decoding the key frames, an SI frame, as an approximation to the current WZ frame, is usually generated by a motion- compensated interpolation or extrapolation of the decoded key frames and the previously decoded WZ frames. The side information is then used in the channel decoder along with the buffered error-correcting bits requested via a feedback channel, in order to decode the bitpalanes corresponding to each DCT band of the current WZ frame and finally, to reconstruct the WZ frame. In DVC, the ability to model the statistical dependency between the WZ information and SI generated at the decoder has a significant impact on the coding efficiency and the rate-distortion (RD) performance. The DICSOVER codec, a state-of-the-art and practical implementation of the transform-domain DVC based on the Stanford approach, was developed in 2007 by Artigas et al. [3]. Figure 1 shows the architecture of the Discover codec. X_{WZ} and X represent the WZ frame and its corresponding DCT coefficient bands respectively. X_P and X_F are two successive decoded key frames that are used for generating the side information frame Y_{SI}. Moreover, Y is the DCT coefficient bands of the SI frame Y_{SI}. In the Discover codec, a Laplacian distribution is used in decoder to model the correlation noise between the DCT band of SI and the corresponding DCT band in the WZ frame. The Laplacian distribution parameter is estimated online for each DCT band, as well as for each of the coefficients in the various bands. The Log-likelihood ratio (LLR) for each

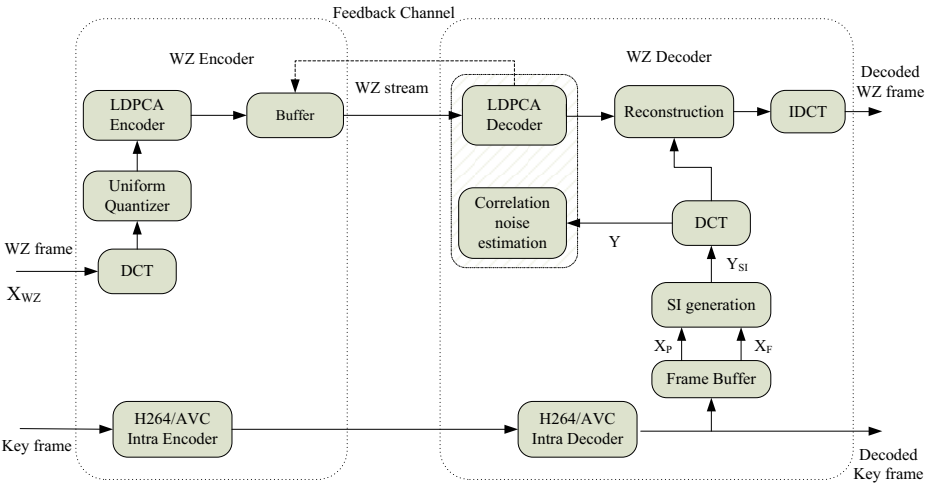


Fig. 1 DISCOVER codec [3]

bit is calculated by considering the previously decoded bitplanes, the side information and the correlation noise model (Laplacian distribution in the case of DISCOVER codec). Rate-adaptive LDPC accumulative (LDPCA) codec is employed by DISCOVER. In the LDPCA decoder, the bitplanes are decoded from the most significant one to the least significant, based on the calculated LLR intrinsic values and the syndrome bits sent by the encoder. If the decoder fails to decode the current bitplane, then it requests for more syndrome bits through a feedback channel. After successfully decoding all the bitplanes associated with the DCT bands, the bitplanes are grouped together to form the corresponding quantized DCT coefficients. Then, inverse DCT is applied to the quantized coefficients to construct the decoded WZ frame.

In DVC, the parameters of the correlation noise model should be estimated as accurately as possible to improve the overall RD performance and the coding efficiency.

The correlation noise parameters can be computed offline at the encoder using the original WZ frame and the estimated side information; however, the encoder has to extract the side information by using a motion estimation procedure, thus making it complex. The correlation noise parameters can also be calculated offline by employing training methods using several video sequences [2, 34]; however, in this case the same correlation noise model is used for the corresponding DCT bands of any given video sequence and the non-stationary behavior of the correlation noise is not taken into account.

These parameters can also be estimated online at the decoder without having access to the original WZ frames, a realistic solution that is used in practice. For the pixel-domain WZ video coding, Brites et al. [7] have proposed several online schemes that make use of the temporal correlation between frames to estimate the correlation noise at different levels, frame, block and pixel. They estimate the correlation noise parameters for block and pixel levels by using the spatial correlation within each frame and obtained estimates, which are more accurate than that using frame-level estimation. In 2008, Brites and Pereira extended the work in the pixel-domain WZ video coding [7] to the transform domain WZ video coding by estimating the correlation noise model parameters in the DCT band level as well as at the coefficient level [6]. In 2009, Haung and Forchhammer [19] improved the method proposed in [6] by considering the cross-band correlation and using a classification map that

is refined after each DCT band is decoded. Esmaili and Cosman [13, 14] proposed a method to estimate the correlation noise parameters by separating and classifying the blocks of each frame based on the quality and accuracy of the side information. After determining the class for each specific block, a Laplacian parameter value is assigned to each of the blocks, using a lookup table. In some of the correlation noise estimation methods, the information on the previously-decoded bands is used to improve the decoding of the current band [15, 20, 27]. Thus, the previously decoded bands are used to improve the estimation of the correlation noise in the succeeding bands.

In most of the online methods [6, 7, 13–15, 19, 20, 27], the estimation process is performed before the Slepian-Wolf decoder starts to decode the bitplanes. Therefore, the estimated parameters for the correlation noise are held constant, that is, they are not modified during the decoding of each DCT band. The soft information of each bitplane corresponding to a DCT band is available after every iteration of the LDPC decoder. This information can be used at the decoder to estimate and refine the correlation noise parameters during the decoding process. In [24], a parallel LDPC decoding is used to decode and estimate the correlation noise parameters on a factor graph. In this algorithm, the non-stationary characteristic of the correlation noise through the DCT bands is not taken into account, and just one parameter is estimated for each DCT band. In [28] and [33], a particle filter-based message passing algorithm for decoding and adaptively estimating the correlation noise parameters has been proposed. As a stochastic method is used for the message passing, it may lead to unpredictable results; further, it is slow as it requires a large number of iterations.

In this paper, in order to overcome the aforementioned limitations, a new message passing algorithm, based on variational Bayes, is proposed for decoding the bitplanes corresponding to each DCT band in a WZ frame, and simultaneously for estimating and refining the correlation noise parameter. In Section 2, using an augmented factor graph, a parallel decoding of several bitplanes as well as Bayesian estimation of the correlation noise parameter is briefly reviewed. In Section 3, the proposed message passing algorithm on the augmented factor graph is presented. Variational Bayes method is employed to approximate the posterior distribution of the correlation noise parameter, which is used to derive a closed form expression for the messages on the augmented factor graph. In Section 4, the performance of the proposed algorithm is studied in the frame work of a DVC codec using several video sequences. Conclusion is given in Section 5.

2 Bayesian estimation of the correlation noise parameter in a parallel LDPCA decoder

As the correlation noise distribution in DVC is defined at a symbol or coefficient level, all the corresponding bitplanes are required to be available for them to be decoded simultaneously on an augmented factor graph in order to estimate the parameter of the correlation noise [24, 31]. Therefore, a parallel LDPCA decoder is used. As a consequence, cross correlation between the bitplanes is utilized to improve the decoding performance of DVC [24]. The parameters of the correlation noise distribution are unknown and need to be estimated during the decoding process dynamically and progressively. One way of estimating the unknown parameters ϕ is by using the maximum likelihood estimation (MLE) method which seeks the parameters that maximize the likelihood function $P(D|\phi)$ given the observation D [10]. Maximum likelihood estimation has been used for estimating the channel and correlation noise parameters in distributed source coding (DSC) [32, 36] and DVC

problems, [11]. In [32], it is used to estimate the cross-over probability for binary symmetric channel (BSC) modeling of the channel in DSC. It has been used for estimating the correlation noise parameter during the decoding process in DVC [11, 36]. One of the drawbacks with MLE is that the entire probability mass is used to assign probabilities to the observed data. Further, MLE performs poorly when the sample size is small. One way to overcoming these drawbacks is to add a prior distribution for ϕ, which allows us to adjust and control as to how the probability mass could be distributed between the observed and unobserved data. Employing the Bayes rule, we can use such a prior distribution for ϕ so that a posteriori distribution, conditioned on the data D, can be derived as: $P(\phi|D) = P(\phi)P(D|\phi)/Z$, where Z is a normalization factor. In maximum a posteriori (MAP) estimation, we look for the parameters ϕ that maximize the posterior distribution $P(\phi|D)$. MLE and MAP are point estimation methods that yield fixed values for ϕ. Consequently, any information regarding the uncertainty of the parameters is not taken into account. To address this problem, the Bayesian estimation is used. In this approach, all possible values for ϕ are considered by defining a probability distribution for ϕ. Hence, in this approach, parameter estimation is equivalent to calculating the posterior distribution of ϕ. Moreover, Bayesian estimation performs better than MLE when the sample size is small.

Suppose $Y = \{y_1, y_2, ..., y_N\}$, N being the number of 4×4 blocks in frame, is a DCT coefficient band obtained by grouping the DCT coefficients of the side information frame and $X = \{x_1, x_2, ..., x_N\}$ is the corresponding DCT coefficient band for the current WZ frame quantized uniformly to 2^β levels, where β is the number of bitplanes extracted from the quantized symbols of DCT coefficient band X, and decoded jointly using the LDPCA decoders.

To take into account the non-stationary characteristic of the correlation noise in each DCT coefficient band in the DVC problem, a parameter θ_j is assigned to each block of DCT coefficients with length M, where $j = 1, 2, ..., N/M$ [33]. As M is selected to be relatively small, Bayesian estimation is preferred for estimating the parameter θ_j. Considering only jth block of DCT coefficients, the posterior distribution for parameter θ_j given $Y_j = \{y_1^j, y_2^j, ..., y_M^j\}$, M DCT coefficients in the corresponding DCT band Y of the generated side information frame, can be written as [28]

$$P(\theta_j|Y_j) = \frac{1}{L_j} p(\theta_j) \prod_{i=1}^{M} P(y_i^j|\theta_j) \tag{1}$$

where L_j is a normalization factor. Replacing $P(y_i^j|\theta_j)$ by $\sum_{x_i^j} P(y_i^j, x_i^j|\theta_j)$ where x_i^j is the coefficient in the DCT band of the WZ frame corresponding to y_i^j, (1) gets transformed to

$$\begin{aligned} P(\theta_j|Y_j) &= \tfrac{1}{L_j} P(\theta_j) \textstyle\prod_{i=1}^{M} \sum_{x_i^j} P(y_i^j, x_i^j|\theta_j) \\ &= \tfrac{1}{L_j} P(\theta_j) \textstyle\prod_{i=1}^{M} \sum_{x_i^j} P(y_i^j|x_i^j, \theta_j) P(x_i^j) \end{aligned} \tag{2}$$

where the summation is over all the values that x_i^j can take. To find the posterior distribution, the corresponding factor graph [23] is first obtained. In the factor graph, a message along the edge from node a to node b is represented by $\mu_{a\to b}$. The likelihood function $P(y_i^j|x_i^j, \theta_j)$ in (2) is represented by the factor node $f_i^j(y_i^j, x_i^j, \theta_j)$ in the factor graph, while the prior

distribution for x_i^j, $P(x_i^j)$ by the message $\mu_{x_i^j \to f_i^j}(x_i^j)$ from the variable node x_i^j to the factor node f_i^j. As a consequence, the posterior distribution (2) can be rewritten as

$$P(\theta_j|Y_j) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \sum_{x_i^j} f_i^j(y_i^j, x_i^j, \theta_j) \mu_{x_i^j \to f_i^j}(x_i^j) \tag{3}$$

We can identify the sum $S_i^j = \sum_{x_i^j} f_i^j(y_i^j, x_i^j, \theta_j) \mu_{x_i^j \to f_i^j}(x_i^j)$ to be the output message $\mu_{f_i^j \to \theta^j}(\theta_j)$ going from the factor node $f_i^j(y_i^j, x_i^j, \theta_j)$ to the variable node θ_j in the factor graph shown in Fig. 2.

Therefore, the posterior distribution in (3) can be written as

$$P(\theta_j|Y_j) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \mu_{f_i^j \to \theta_j}(\theta_j) \tag{4}$$

We now introduce a factor node g_j so that the prior distribution of θ_j , $P(\theta_j)$, can be denoted by the message $\mu_{g_j \to \theta_j}(\theta_j)$. As a consequence, (3) may be rewritten as

$$P(\theta_j|Y_j) = \frac{1}{L_j} \mu_{g_j \to \theta_j}(\theta_j) \prod_{i=1}^{M} \mu_{f_i^j \to \theta^j}(\theta_j) \tag{5}$$

Without loss of generality, we assume that the above equation is normalized so that the posterior distribution in (3) may be written as

$$P(\theta_j|Y_j) = \mu_{g_j \to \theta^j}(\theta_j) \prod_{i=1}^{M} \mu_{f_i^j \to \theta_j}(\theta_j) \tag{6}$$

The expression in (6) shows that the posterior distribution of θ_j given Y_j can be calculated as the product of all the M incoming messages from the factor nodes f_i^j, $i = 1, 2, ..., M$ to variable node θ_j and the message $\mu_{g_j \to \theta_j}$ coming from the factor node g_j. Hence, the posterior distribution, $P(\theta_j|Y_j)$ given by (6) can be represented by the factor graph shown in Fig. 3.

After using 2^β level quantizer for the DCT coefficient band X, the quantization indices of that DCT band turn into β bitplanes $B_c = \{b_{1c}, b_{2c}, ..., b_{Nc}\}$, $c = 1, 2, ..., \beta$. Here, β

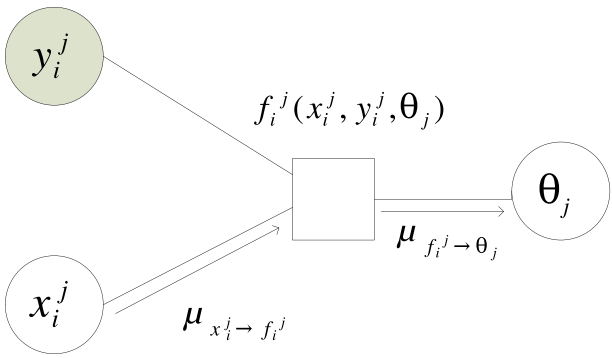


Fig. 2 Factor graph whose output message $\mu_{f_i^j \to \theta^j}(\theta_j)$ is the sum S_i^j

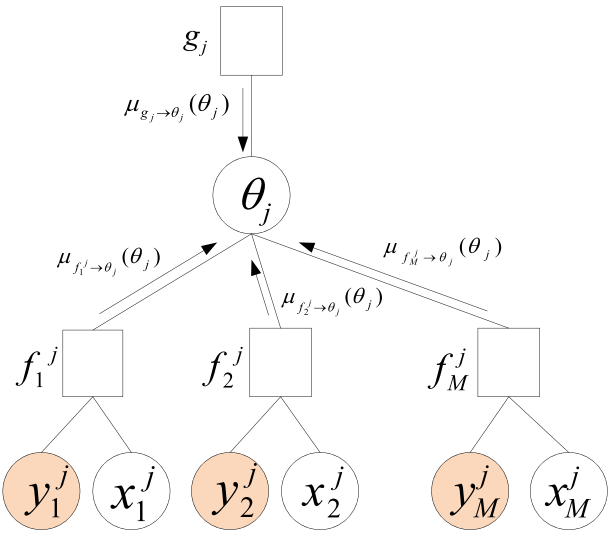


Fig. 3 Factor graph representing the posterior distribution $P(\theta_j|Y_j)$ given by (6)

LDPCA decoders are used in parallel to decode all the bitplanes B_c. The belief propagation (BP) decoding algorithm [35], a well-known iterative decoding algorithm, is used on the factor graphs of the LDPCA decoders to obtain the log-likelihood ratios (LLR) for each bit b_{ic} in the bitplane B_c. The factor graphs of each of the LDPCA decoders used for decoding the bitplanes are augmented by the factor graph shown in Fig. 3 representing $P(\theta_j|Y_j)$ for $j = 1, 2, ..., N/M$. The augmented LDPCA decoder is obtained as shown in Fig. 4.

The boxes in Fig. 4 represent the LDPCA decoder graphs constructed for simultaneous decoding of the various bitpalnes. The LDPC decoder graph for each bitplane B_c consists of N source nodes and T_c syndrome nodes corresponding to T_c accumulated syndrome bits sent to the decoder, as shown in Fig. 5. It should be noted that at the encoder, N accumulated syndrome bits are produced for each bitplane according to the LDPC encoder graph structure and the concatenated accumulator as explained in [30]. These N accumulated syndrome bits are stored in a buffer and sent to the decoder incrementally at the request of the

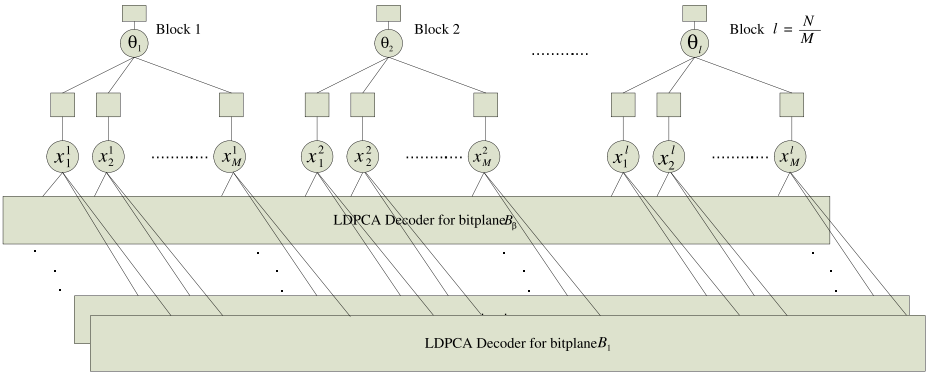


Fig. 4 The augmented decoder

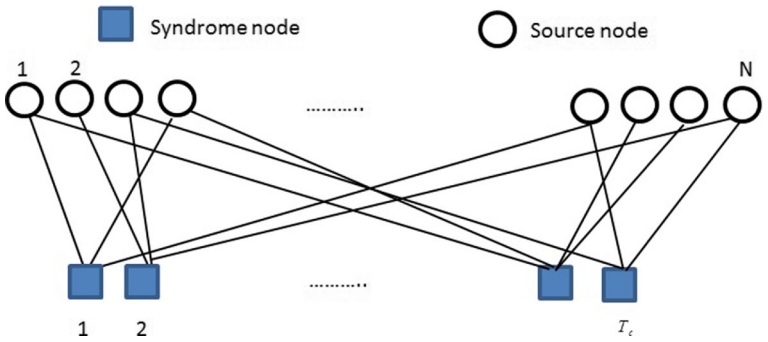


Fig. 5 LDPCA decoder graph for the bitplane B_c

decoder. Based on the number of accumulated syndrome bits T_c ($T_c < N$) received at the decoder for each bitplane, the LDPCA decoder graph for that bitplane gets updated.

Details of the Block j in Fig. 4 are shown in Fig. 6, where b_{ic}^j, $c = 1, 2, ..., \beta$, represents the c-th bit corresponding to the quantized symbol of DCT coefficient x_i^j. The message $\mu_{b_{ic}^j \to x_i^j}$ is calculated from LLR of the bit b_{ic}^j obtained using the BP algorithm that passes the messages back and forth between the source and syndrome nodes in the LDPCA decoder graph for B_c using (2) and (3) of [30]. Hence, the message $\mu_{x_i^j \to f_i^j}$ is obtained as the product of the messages $\mu_{b_{ic}^j \to x_i^j}$, $c = 1, 2, ..., \beta$.

It is prohibitively expensive to calculate the posterior distribution $P(\theta_j|Y_j)$ as given by (5). Also, we need a simple and closed form distribution for $P(\theta_j|Y_j)$ to derive messages for

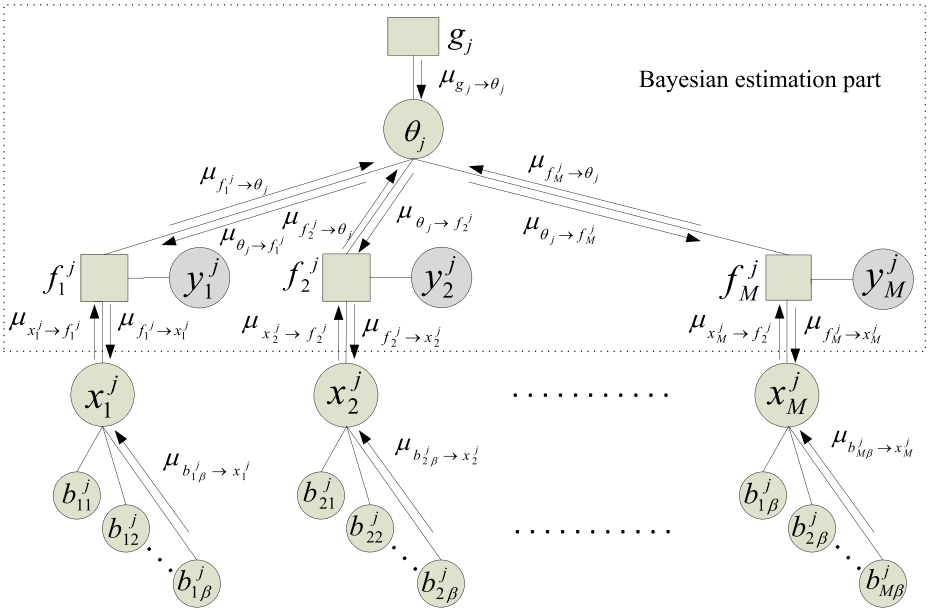


Fig. 6 Factor graph of block j

the message passing algorithm. In view of these, the posterior distribution $P(\theta_j|Y_j)$ needs to be approximated by a simple distribution, such as a distribution from the exponential family.

3 New decoding algorithm based on VB

In this section, a new recursive message passing algorithm is proposed to decode all the bit-planes corresponding to each of the DCT bands. The proposed recursive algorithm consists of three main modules:

1. VB algorithm to approximate the posterior distribution
2. Message update
3. Parallel LDPCA decoding process.

These three modules are explained in detail in Sections 3.1, 3.2 and 3.3, respectively, and the complete recursive algorithm is given in Section 3.4.

3.1 VB algorithm to approximate the posterior distribution

It was seen in Section II that the posterior distribution $P(\theta_j|Y_j)$ given by (3) consists of $2^{\beta M}$ terms and further, that it does not have a closed form. Hence, calculation of the posterior distribution is extremely expensive. Sampling or particle methods, such as the Markov Chain Monte Carlo (MCMC) method, are used frequently for the approximation of the posterior distribution [18, 33]. These methods are stochastic approximation methods [9], and have high computational costs. In addition, results using any of these methods vary for each run on the same data set. Another class of methods used for approximation is deterministic in nature and is much faster than the sampling methods. The main idea behind the deterministic methods is to find a distribution function that is as close as possible to the true posterior distribution. Variational Bayes is a well-known deterministic method that is used to approximate the true posterior distribution [5, 16].

In a general Bayesian problem, one of the objectives is to find $P(Z|X)$, where Z denotes all the unknown parameters and the hidden variables and X the observed variables. Since the exact calculation of $P(Z|X)$ is prohibitively expensive, it is necessary to find an approximation for $P(Z|X)$. It is known that for a given distribution $q(Z)$, the log marginal probability of X can be decomposed as [16]

$$lnP(X) = l(q) + KL(q \,||\, p) \tag{7}$$

where

$$l(q) = \int q(Z) ln \left\{ \frac{P(X, Z)}{q(Z)} \right\} dZ \tag{8a}$$

$$KL(q \,||\, p) = \int q(Z) ln \left\{ \frac{q(Z)}{P(Z \mid X)} \right\} dZ \tag{8b}$$

$KL(q \,||\, p)$ being the Kullback-Leibler (KL) divergence that quantifies the similarity between the two distributions $q(Z)$ and $P(Z \mid X)$, and $l(q)$ being the lower bound for $lnP(X)$. In order for $q(Z)$ to be an approximation of $P(Z \mid X)$ and at the same time be a tractable distribution, a restricted family of distributions is considered for $q(Z)$. In fact, we try to restrict $q(Z)$ to be a tractable distribution that is flexible enough to provide a proper approximation to the true posterior distribution. Then, the members of this distribution

family are found for which the KL divergence in (8b) is minimized. It is equivalent to maximizing the lower bound $l(q)$ with respect to $q(Z)$.

Suppose the elements of Z are partitioned into S disjoint subsets Z_n, $n = 1, 2, ..., S$. Then, we assume that $q(Z)$ can be factorized as [16]

$$q(Z) = \prod_{i=1}^{S} q_n(Z_n) \tag{9}$$

The objective is to find the distribution $q(Z)$ that leads to the largest lower bound $l(q)$. As shown in [23], the variational optimization of $l(q)$ with respect to the m-th factor, $q_m(Z_m)$, can be obtained using

$$lnq_m(Z_m) = E_{n\neq m}\,[ln\ P(X,\ Z)] + C \qquad 1 < m < S \tag{10}$$

where C is a constant, and $E_{n\neq m}\,[ln\ P(X,\ Z)]$=$\int P(X, Z) \prod_{n\neq m} q_n(Z_n)\mathrm{dZ}_n$.

Equation (10) represents the conditions that will maximize the lower bound $l(q)$ and consequently, minimize the KL divergence with respect to m-th factor, $q_m(Z_m)$. Solving (10) for $q_m(Z_m)$, $m = 1, 2, .., S$, leads to the distribution $q(Z)$ as an approximation to the posterior distribution $P(Z \mid X)$.

The above method is used in our proposed one to approximate the posterior distribution $P(\theta_j|Y_j)$ derived in Section 2 and consequently, simplify the message structure on the augmented LDPC decoder in the j^{th} block illustrated in Fig. 6. In order to use the variational Bayes method, we use a set of hidden variables $H_j = \{h_1^j, h_2^j, ..., h_M^j\}$, where each h_i^j, $i = 1, 2, ..., M$ is a K-length vector ($K = 2^\beta$). Let $Z = \{H_j, \theta_j\}$, where θ_j is an unknown parameter. Hence, the variational factorization given by (9) can be performed, for $S = 2$ as follows, by letting $Z_1 = H_j$ and $Z_2 = \theta_j$ in (9)

$$q(Z) = q(H_j, \theta_j) = q_1(Z_1)q_2(Z_2) = q_1(H_j)q_2(\theta_j) \tag{11}$$

where $q_2(\theta_j)$ is the variational approximation for $P(\theta_j|Y_j)$. After the factorization, the optimization (10) is used for both the factors by considering the observed variables X in (10) to be the side information Y_j in our problem. Hence, two equations for VB algorithm can be presented as

$$lnq_1(H_j) = E_{\theta_j}[LnP(Y_j, H_j, \theta_j)] + C_1 \tag{12a}$$

$$lnq_2(\theta_j) = E_{H_j}[LnP(Y_j, H_j, \theta_j)] + C_2 \tag{12b}$$

where the joint distribution $P(Y_j, H_j, \theta_j)$ in (12a) and (12b) can be written as

$$P(Y_j, H_j, \theta_j) = P(\theta_j)P(Y_j|H_j, \theta_j)P(H_j) = P(\theta_j)\prod_{i=1}^{M} P(y_j^i|h_j^i, \theta_j)P(h_j^i) \tag{13}$$

To determine $q_1(H_j)$ and $q_2(\theta_j)$ from (12), we first need to find an expression for $P(Y_j, H_j, \theta_j)$ in (13).

For each WZ frame in the encoder, all of the coefficients in a specific DCT coefficient band have been uniformly quantized to $K = 2^\beta$ level to generate the quantized symbols. At the decoder, since the DCT coefficients of WZ frame, x_i^js, are not available, we use a partially decoded coefficient obtained by minimum mean square error (MMSE) reconstruction $w_{ik}^j = E(x_i^j|y_i^j, I_k, \lambda_j)$, where $k = 1, 2, ..., K$, I_k is k^{th} quantization interval and λ_j is the initial value for the parameter of correlation noise distribution. In view of this, for each side information DCT coefficient y_i^j extracted in the decoder, a hidden variable vector h_i^j is considered as a k-length binary vector with elements $h_{i1}^j, h_{i2}^j, ..., h_{iK}^j$. This

vector has only one element equal to 1 and the rest are all zero. For each observation y_i^j, the position of 1 in each vector h_i^j is determined by the quantization interval index (quantized symbol) so that if $x_i^j \in I_d$, $1 < d < K$, only the d^{th} element of vector h_i^j is 1, i.e., $h_i^j = [0, 0, ..., 1, 0, 0, ..., 0]$. By considering this feature for the hidden variable vectors h_i^j, $P(y_i^j|h_i^j, \theta_j)$ and $P(h_i^j)$ can be written as $P(y_i^j|h_i^j, \theta_j) = \prod_{k=1}^{K} P(y_i^j|x_i^j = w_{ik}^j, \theta_j)^{h_{ik}^j}$ and $P(h_i^j) = \prod_{k=1}^{K}(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))^{h_{ik}^j}$ respectively. Using the expressions for $P(h_i^j)$ and $P(y_i^j|h_i^j, \theta_j)$, (12a) can be rewritten as

$$\begin{aligned} lnq_1(H_j) &= E_{\theta_j}[lnP(Y_j|H_j, \theta_j) + lnP(H_j)] + C_1 \\ &= E_{\theta_j}\left[ln\prod_{i=1}^{M}\prod_{k=1}^{K}(P(y_i^j|x_i^j = w_{ik}^j, \theta_j))^{h_{ik}^j}\right. \\ &\quad \left. + ln\prod_{i=1}^{M}\prod_{k=1}^{K}(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))^{h_{ik}^j}\right] + C_1 \\ &= E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j\{ln(P(y_i^j|x_i^j = w_{ik}^j, \theta_j))\right. \\ &\quad \left. + ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))\}\right] + C_1 \end{aligned} \tag{14}$$

Similarly, (12b) can be rewritten as

$$\begin{aligned} lnq_2(\theta_j) &= E_{H_j}[lnP(Y_j|H_j, \theta_j) + lnP(\theta_j)] + C_2 \\ &= E_{H_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j\{ln(P(y_i^j|x_i^j = w_{ik}^j, \theta_j))\}\right] + lnP(\theta_j) + C_2 \end{aligned} \tag{15}$$

In distributed video coding, the correlation noise which is the difference between each DCT coefficient band of the WZ frame and the corresponding one in the side information frame is often modeled by Gaussian [25] or Laplacian [22] distribution. In the following subsections, we consider Gaussian and Laplacian distributions for the correlation noise model to solve (14) and (15) simultaneously in order to find $q_2(\theta_j)$ as an approximation to $P(\theta_j|Y_j)$.

1) Gaussian distribution for correlation noise model

Assuming Gaussian distribution for the correlation noise, we can express the probability $P(y_i^j|x_i^j = w_{ik}^j, \theta_j)$ in (14) and (15) as

$$P(y_i^j|x_i^j = w_{ik}^j, \theta_j) = \frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}} e^{\frac{-(y_i^j - w_{ik}^j)^2}{2}\theta_j} \tag{16}$$

Substituting (16) in (14) and after some simplification, it can be shown that,

$$\begin{aligned} lnq_1(H_j) &= E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{ ln\left(\frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}} e^{\frac{-(y_i^j - w_{ik}^j)^2}{2}\theta_j}\right)\right.\right. \\ &\quad \left.\left. + ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))\right\}\right] + C_1 \\ &= E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{ \frac{1}{2} ln\theta_j + \frac{1}{2} ln\frac{1}{2\pi} - \frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j \right.\right. \\ &\quad \left.\left. + ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))\right\}\right] + C_1 \\ &= \sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j ln\rho_{ik} + C_1 \end{aligned} \tag{17}$$

where

$$ln\rho_{ik} = \frac{1}{2} E_{\theta_j}\left[ln\theta_j\right] - \frac{1}{2} ln2\pi - E_{\theta_j}\left[\frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j\right] + ln(\underset{x_i^j \to f_i^j}{\mu(x_i^j = w_{ik}^j)}) \tag{18}$$

Let the normalized value of ρ_{ik} be denoted by r_{ik}. Then,

$$r_{ik} = \rho_{ik} / \sum_{k=1}^{K} \rho_{ik} \tag{19}$$

From (18), it can be concluded that

$$q_1(H_j) = \prod_{i=1}^{M}\prod_{k=1}^{K} r_{ik}^{h_{ik}^i} \tag{20}$$

Also, the update (15) for $q_2(\theta_j)$ can be obtained as follows

$$\begin{aligned} lnq_2(\theta_j) &= E_{H_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{ lnP(y_i^j | x_i^j = w_{ik}^j, \theta_j)\right\}\right] + lnP(\theta_j) + C_2 \\ &= \sum_{i=1}^{M}\sum_{k=1}^{K} \frac{1}{2} E_{h_{ik}^j}[h_{ik}^j] ln(\theta_j) - \sum_{i=1}^{M}\sum_{k=1}^{K} E_{h_{ik}^j}[h_{ik}^j] \frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j \\ &\quad + lnP(\theta_j) + C_2 \end{aligned} \tag{21}$$

As $E_{h_{ik}^j}[h_{ik}^j] = p(h_{ik}^j = 1) = r_{ik}$, (21) can be rewritten as

$$lnq_2(\theta_j) = \sum_{i=1}^{M}\sum_{k=1}^{K} \frac{1}{2} r_{ik} ln(\theta_j) - \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} \frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j + lnP(\theta_j) + C_2 \tag{22}$$

If the prior distribution $P(\theta_j)$ is considered as a gamma distribution with parameter a_0 and b_0, that is,

$$P(\theta_j) = Gama(\theta | a_0, b_0) = \frac{1}{\Gamma(a_0)} b_0^{a_0} \theta_j^{a_0 - 1} e^{-b_0\theta_j} \tag{23}$$

Then

$$lnP(\theta_j) = b_0^{a_0}\theta_j(a_0 - 1)ln\theta_j - b_0\theta_j + \alpha \tag{24}$$

where $\alpha = \frac{1}{\Gamma(a_0)} b_0^{a_0}$ is a constant. Then, by substituting (24) in (22), $lnP(\theta_j)$ can be simplified as

$$lnq_2(\theta_j) = \left(\frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 - 1\right) ln(\theta_j) - \left(\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik}\frac{(y_i^j - w_{ik}^j)^2}{2} + b_0\right)\theta_j + C_3 \tag{25}$$

By comparing (25) and (24), it is obvious that the $q_2(\theta_j)$, the variational approximation of the true posterior distribution, would be in the form of a gamma distribution with parameters a and b as follows,

$$\begin{aligned} a &= \frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 = \frac{1}{2}M + a_0 \\ b &= \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik}\frac{(y_i^j - w_{ik}^j)^2}{2} + b_0 \end{aligned} \tag{26}$$

Then, by using the gamma distribution with parameters a and b obtained in (26), ρ_{ik} can be calculated from (18). Consequently, after normalizing ρ_{ik} using (19), r_{ik} can be obtained as

$$r_{ik} = \frac{\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j) e^{\left(\frac{-(y_i^j - w_{ik}^j)^2}{2}\frac{a}{b}\right)}}{\sum_{k=1}^{K} \mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j) e^{\left(\frac{-(y_i^j - w_{ik}^j)^2}{2}\frac{a}{b}\right)}} \tag{27}$$

In the first iteration of the VB algorithm, we consider $a = a_0$ and $b = b_0$ for the parameters of the gamma distribution. The value obtained for r_{ik} is then substituted in (26) to find the new value for b. The new parameters for gamma distribution is now used in (27) to obtain a new value for r_{ik}. This procedure is repeated iteratively until there is almost no change in the value of b. The gamma distribution with the parameters a and b so obtained, i.e., $Gama(\theta_j|a, b)$ is considered as the distribution approximating the posterior distribution.

2) Laplacian for correlation noise model

Assuming Laplacian distribution for the correlation noise, we can express the probability $P(y_i^j|x_i^j = w_{ik}^j, \theta_j)$ in (14) and (15) as

$$P(y_i^j|x_i^j = w_{ik}^j, \theta_j) = \frac{\theta_j}{2} e^{-|y_i^j - w_{ik}^j|\theta_j} \tag{28}$$

The VB method explained above for the Gaussian distribution can be also applied for the Laplacian distribution. In this case, the approximation of the posterior distribution is also a gamma distribution with parameters a and b as given below.

$$\begin{aligned} a &= \frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 = \frac{1}{2}M + a_0 \\ b &= \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik}|y_i^j - w_{ik}^j| + b_0 \end{aligned} \tag{29}$$

Then, using the gamma distribution with the above parameters, r_{ik} can be obtained as

$$r_{ik} = \frac{\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j) e^{-|y_i^j - w_{ik}^j|\frac{a}{b}}}{\sum_{k=1}^{K} \mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j) e^{-|y_i^j - w_{ik}^j|\frac{a}{b}}} \tag{30}$$

Just as in the case of VB in Gaussian distribution, the values for r_{ik} and the parameters of the gamma distribution, a and b, are obtained iteratively until there is almost no more change in the value of b. The gamma distribution with the parameters a and b so obtained, i.e., $Gama(\theta_j|a, b)$ is considered as the approximate distribution for the posterior distribution.

3.2 Message update

After obtaining the approximation for the posterior distribution $P(\theta_j|Y_j)$, the message $\mu_{f_i^j \to x_i^j}(x_i)$, representing the probability that the partially decoded coefficient is w_{ik}^j or equivalently $x_i^j \in I_k$, $k = 1, 2, ..., K$, is calculated based only on the information from the Bayesian estimation part shown in the factor graph of Fig. 6. If the correlation noise is Gaussian, then the message $\mu_{f_i^j \to x_i^j}(x_i)$ can be calculated as

$$\begin{aligned}
\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j) &= \int_{\theta_j} \frac{1}{\Gamma(a)} b^a \theta_j^{a-1} e^{-b\theta_j} \frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}} e^{\frac{-\theta_j}{2}(y_i^j - w_{ik}^j)^2} d\theta_j \\
&= \int_0^{\infty} \frac{1}{\Gamma(a)\sqrt{2\pi}} b^a \theta_j^{a-\frac{1}{2}} e^{-\theta_j \left[\frac{\left(y_i^j - w_{ik}^j\right)^2}{2} + b \right]} d\theta_j
\end{aligned} \tag{31}$$

Then, after some mathematical simplification, $\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j)$ can be expressed as

$$\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j) = \frac{\Gamma(a + \frac{1}{2})}{\Gamma(a)\sqrt{2\pi}} b^a \left[b + \frac{\left(y_i^j - w_{ik}^j\right)^2}{2} \right]^{-\left(a+\frac{1}{2}\right)} \tag{32}$$

On the other hand, if the correlation noise has a Laplacian distribution, then the message $\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j)$ can be obtained as

$$\begin{aligned}
\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j) &= \int_{\theta_j} \frac{1}{\Gamma(a)} b^a \theta_j^{a-1} e^{-b\theta_j} \frac{\theta_j}{2} e^{-\theta_j |y_i^j - w_{ik}^j|} d\theta_j \\
&= \int_0^{\infty} \frac{1}{2\Gamma(a)} b^a \theta_j^{a-\frac{1}{2}} e^{\left[-\theta_j |y_i^j - w_{ik}^j| + b\right]} d\theta_j
\end{aligned} \tag{33}$$

which after simplification, can be written as

$$\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j) = \frac{1}{2} b^a \left[b + |y_i^j - w_{ik}^j| \right]^{-(a+1)} \tag{34}$$

The updated messages from each of the blocks are then returned into LDPCA decoders for the bitplanes $B_1, B_2, ..., B_\beta$ (See Fig. 4) to start decoding with more accurate soft

information. Hence, all the decoders have new and more precise knowledge about the correlation noise parameter, leading to a more efficient decoding after performing regular belief propagation in the LDPCA decoder.

3.3 Parallel LDPCA decoding process

To decode the bitplanes $B_1, B_2, ..., B_\beta$ (see Fig. 4) using the BP algorithm in the LDPCA decoders, the log-likelihood ratio (LLR) for each bit b_{ic}^j in the bitplanes $B_1, B_2, ..., B_\beta$ has to be obtained. First, the messages $\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j)$ obtained in Section 3.2 are used to calculate the messages $\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j)$ from node x_i^j to the corresponding bit nodes b_{ic}^j, using the procedure given in [24]. Then, $\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j)$ is exploited to compute the initial LLR for each bit b_{ic}^j as

$$L_{ic}^j = log \frac{\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j = 1)}{\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j = 0)} \tag{35}$$

After a pre-specified number of iterations for the BP algorithm in the LDPCA decoders, LLR for each bit b_{ic}^j is obtained as $l_{ic}^j = L_{ic}^j + \sum l_{ic}^{j,v}$, where L_{ic}^j is calculated using (35), $l_{ic}^{j,v}$ is the LLR value received through the v^{th} edge ($v = 1, 2, ...V_i$) from the syndrome node to the node b_{ic}^j after a pre-defined number of iterations and V_i is the number of syndrome nodes connected to the node b_{ic}^j. Then, b_{ic}^j is decoded as 1 if $l_{ic}^j > 0$ and as zero otherwise. Next, the LDPCA syndrome and 8-bit cyclic redundancy check (CRC) summations are used in the decoder to determine whether or not the LDPCA decoding has been successful [3].

3.4 The recursive message passing algorithm

Figure 7 shows the proposed bitplane decoder consisting of the three modules explained in Sections 3.1, 3.2 and 3.3. The arrows in this figure indicate the interactions amongst the three modules.

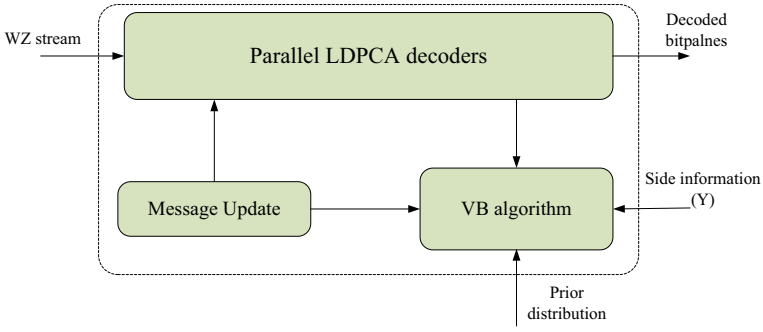


Fig. 7 Proposed bitplanes decoder

The recursive message passing algorithm is described below.

Step 1 - The messages $\mu_{x_i^j \to f_i^j}(x_i^j)$ in Fig. 6 are first calculated using the messages $\mu_{b_{ic}^j \to x_i^j}(b_{ic}^j)$ received by the node x_i^j from the bit nodes b_{ic}^j, $c = 1, 2, ..., \beta$ so that $\mu_{x_i^j \to f_i^j}(x_i^j) = \prod_{c=1}^{\beta} \mu_{b_{ic}^j \to x_i^j}(b_{ic}^j)$.

Step 2 - Using the messages, $\mu_{x_i^j \to f_i^j}(x_i^j)$, and the partially decoded coefficients w_{ik}^j for $k = 1, 2, .., K$, an approximation for the posterior distribution of each correlation noise parameter θ_j is calculated using the VB algorithm, as explained in Section 3.1.

Step 3 - The approximated posterior distribution for each correlation noise parameter θ_j is used to calculate the messages $\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j)$ from the factor nodes f_i^j to the variable nodes x_i^j, as explained in Section 3.2.

Step 4 - The messages $\mu_{f_i^j \to x_i^j}(x_i^j = w_{ik}^j)$ thus obtained are then used to calculate the messages $\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j)$ from the node x_i^j to the bit nodes b_{ic}^j, $c = 1, 2, ..., \beta$. The initial LLRs L_{ic}^j are then calculated using (35) and employed in the LDPCA decoders to decode all the bitplanes, as explained in Section 3.3.

Step 5 - By using the LDPCA syndrome and 8-bit CRC summations as mentioned in Section 3.3, we check if all the bitplanes have been decode correctly.

Step 6 - if all the bitplanes are decoded correctly, the decoding process stops; otherwise Steps 1- 5 are repeated for a pre-specified number of iterations.

After applying the above algorithm, if any of the LDPCA decoder fails to decode its bitplane correctly, then the corresponding decoder requests more syndrome bits from the encoder, as is done in algorithms. Then, this decoder modifies its factor graph, and the proposed recursive message passing algorithm is applied again. This whole process is repeated until all the LDPCA decoders successfully decode all their bitplanes. The LDPCA decoder and the correlation noise estimation blocks in the DISCOVER codec shown in Fig. 1 are now replaced by the proposed decoder shown in Fig. 7, and the resulting modified architecture for the transform-domain distributed video codec is shown in Fig. 8.

4 Simulation results

In this section, we study through extensive experimentation the effect on the rate-distortion performance of the DISCOVER codec, when it is modified as in Fig. 8 by incorporating the bitplane decoder of Fig. 7, which is based on the proposed joint estimation and decoding method. We then compare the rate-distortion (RD) performance of this modified DISCOVER codec with that of the original codec [3] that uses the online correlation noise estimation proposed in [6]. For the simulations, *Foreman* (150 frames), Coastguard (150 frames) and Hall (150 frames) and Soccer (150 frames) video sequences with 15Hz frame rate and QCIF format are employed; one frame of each of these sequences is shown in Fig. 9. The key frames are encoded using the intra coding mode of the H.264/AVC codec, JM 9.5 [21]. Eight RD points corresponding to the eight 4×4 quantization matrices $Q_1, Q_2, , Q_8$, the same as the ones used in the DISCOVER codec [6],

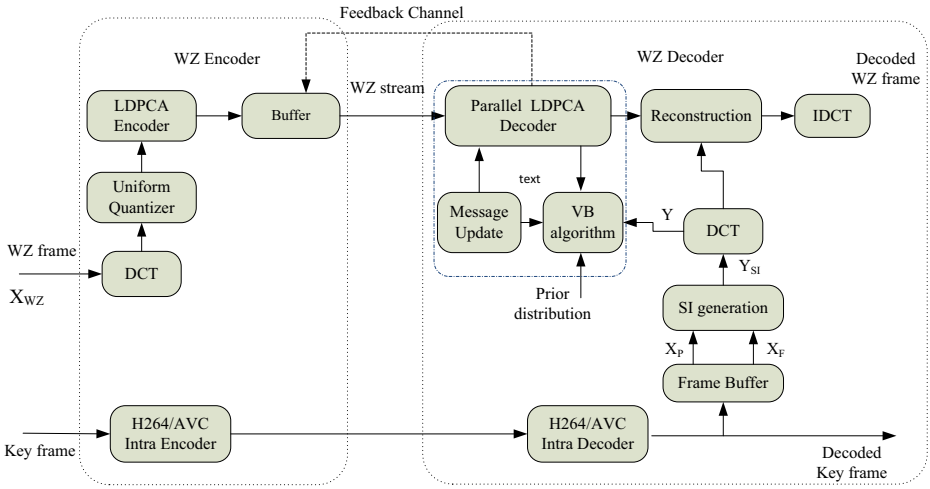


Fig. 8 Modified architecture for DVC

are considered. Moreover, the QP values in H.264/AVC (in intra mode) are set to the same values as used for the key frames in the DISCOVER codec [6]. Also, only the luminance component (Y) of the video frames is considered in our simulation for the rate-distortion evaluation. In our setup, the Laplacain distribution is used to model the correlation noise in each block of the DCT coefficients of length M=99 in the corresponding DCT band. Then, the proposed message passing algorithm based on VB is used to decode all the bitplanes simultaneously in each of the DCT bands. The maximum number of iterations used for the recursive message passing algorithm in the proposed decoder before requesting for more syndrome bits is three. Carrying out further iterations would only increase the execution time without adding any noticeable improvement in the performance. Also, the belief propagation algorithm inside the LDPCA decoders runs for 100 iterations to decode the bitplanes in each DCT band of each of the Wyner-Ziv frames.

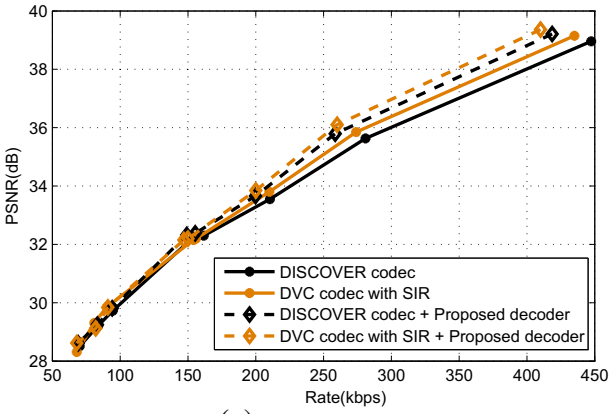
Table 1 gives the relative average savings (%) in bitrate and the improvement in PSNR (dB) of the proposed codec over that of the DISCOVER codec for WZ frames as well as for all frames, computed using the *Bjøntegaard* metric [8]. Considering the GOP of size 2 (Having one WZ frames between two successive key frames), it can be seen that for the *Foreman* sequence the proposed method leads to an average bitrate savings of 5.53% and 11.45% for all frames and WZ frames, respectively. The corresponding savings are 3.21% and 6.11% for the Hall sequence, 4.79% and 9.73% for the Coastguard sequence, and 8.23% and 15.71% for the soccer sequence. As for the PSNR, the proposed codec shows an average improvement of 0.31 dB and 0.29 dB, 0.27 dB and 0.48 dB for the *Foreman*, *Hall*, *Coastguard* and *Soccer* sequences in WZ frames. Further, there is an improvement of 0.16 dB, 0.14 dB, 0.12 dB and 0.26 dB in *Foreman*, *Hall*, *Coastguard* and *Soccer* sequences respectively, for all frames. Hence, on an average, we observe that our method leads to 10.75% saving in bitrate for WZ frames and 5.44% bitrate saving for all frames of the sequences. Moreover, on an average, the improvements in PSNR values are 0.33 dB and 0.17dB in WZ frames and all frames, respectively.

Fig. 9 One frame of each of the sequences, *Foreman*, *Hall* and *Coastguard* and *Soccer*

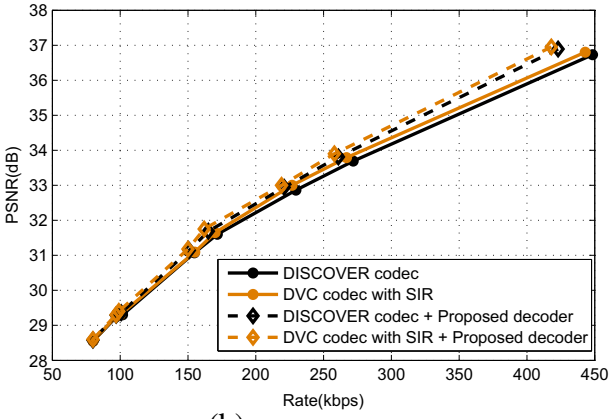
For GOP of size 4 (having 3 WZ frames between two successive key frames), it also can be seen from Table 1 that for the *Foreman* sequence the proposed method leads to an average bitrate savings of 8.41% and 10.68% for all frames and WZ frames, respectively. The corresponding savings are 5.26% and 7.26% for the Hall sequence, 6.67% and 9.56% for the Coastguard sequence, and 7.76% and 11.21% for the soccer sequence. As for the PSNR, the proposed codec shows an average improvement of 0.33 dB and 0.35 dB, 0.26 dB and 0.41 dB for the *Foreman*, *Hall*, *Coastguard* and *Soccer* sequences in WZ frames. Further, there is an improvement of 0.24 dB, 0.22 dB, 0.19 dB and 0.25 dB in *Foreman*, *Hall*, *Coastguard* and *Soccer* sequences respectively, for all frames. Hence, on an average, we observe that our method leads to 9.67% saving in bitrate for WZ frames and 7.02% bitrate saving for all frames of the sequences. Moreover, on an average, the improvements in PSNR values are 0.34 dB and 0.22dB in WZ frames and all frames, respectively.

Table 1 The relative bit-rate savings (%) and improvement in PSNR values (dB) over that of discover codec, computed using the Bjøntegaard metric

	GOP=2				GOP=4			
	WZ frames		All frames		WZ frames		Allframes	
	$\triangle R(\%)$	△PSNR (in dB)	△R (%)	△PSNR (in dB)	△R (%)	△PSNR (in dB)	△R (%)	△PSNR (in dB)
Foreman	11.45	0.31	5.53	0.16	10.68	0.33	8.41	0.24
Coastguard	9.73	0.27	4.79	0.12	9.56	0.26	6.67	0.19
Hall	6.11	0.29	3.21	0.14	7.26	0.35	5.26	0.22
Soccer	15.71	0.48	8.23	0.26	11.21	0.41	7.76	0.25
Average	10.75	0.33	5.44	0.17	9.67	0.34	7.02	0.22



(a) *Foreman* sequence

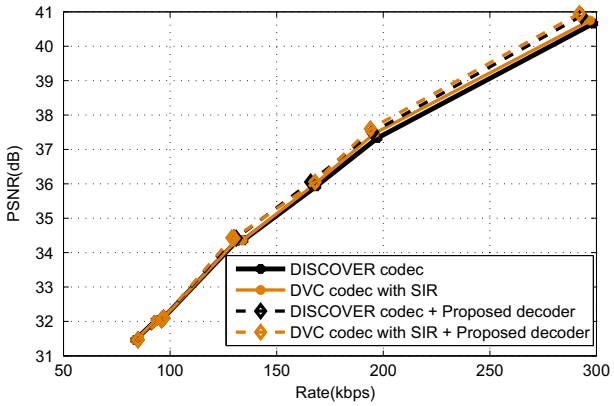


(b) *Coastguard* sequence

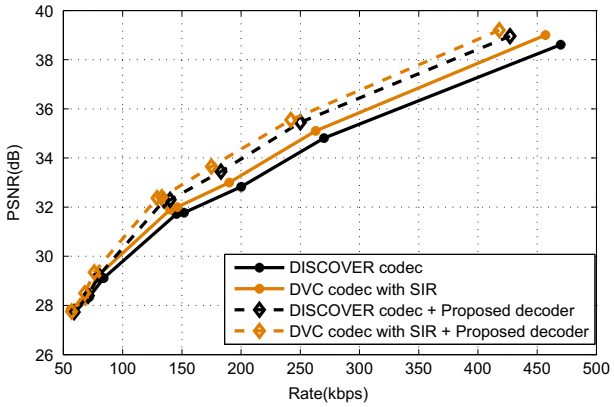
Fig. 10 RD performance for GOP size of 2

Figures 10 and 11 show the overall RD performance for the four sequences *Foreman*, *Coastguard*, *Hall* and *Soccer* using the DISCOVER codec and the modified DISCOVER codec for GOP of size 2 and 4. It is seen from these figures that the modified DISCOVER codec exhibits an overall better RD performance than the original codec. Further, it is clear from these figures and Table 1 that this improvement in the RD performance is even more pronounced in the case of *Foreman* and *Soccer* sequences where there are fast motions.

The proposed decoder can also be used in other transform-domain distributed video coding schemes which have the same architecture as the DISCOVER codec, namely, those based on the Stanford approach. For instance, if the proposed decoder is employed in the distributed video codec with the side information refinement (SIR) proposed in [26], then using Bjøntegaard metric, the relative bitrate savings (%) and PSNR improvement (in dB) are obtained as shown in Table 2. It can be see that for GOP of size 2, on an average, the modified DVC codec, on average, leads to 8.79% and 4.11% savings in the bitrate for WZ frames and all frames, respectively, for the aforementioned sequences. Moreover, on an



(c) *Hall* sequence

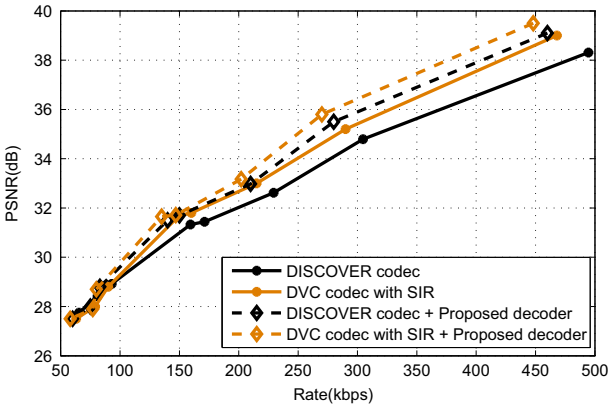


(d) *Soccer* sequence

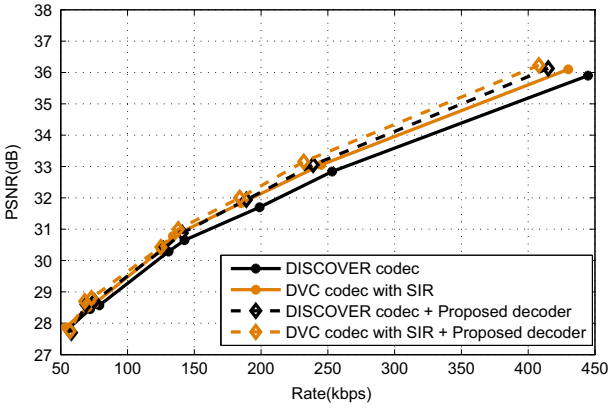
Fig. 10 (continued)

average, there is an improvement of 0.19 dB and 0.1 dB in PSNR for WZ frames and all frames, respectively. Further, for GOP of size 4, the corresponding savings in the bitrate are 7.51% and 5.63% for WZ frames and all frames, respectively. Also, there is an improvement of 0.17 dB and 0.11 dB in PSNR for WZ frames and all frames, respectively. The RD performance of DVC codec with SIR [26] and its modified version using the proposed decoder are also shown in Figs. 10 and 11. It is clear from these figures that an improved RD performance is achieved by using the proposed decoder.

Some screenshots from the decoded Wyner-Ziv frames of *Foreman* and *Soccer* sequences are shown in Figs. 12 and 13 respectively, which demonstrate the improvement in the quality of the decoded frames resulting from using the DISCOVER codec modified by the proposed decoder over the original DISCOVER codec. For these video sequences, GOP size of 2 and the quantization matrix Q_6 corresponding to 6^{th} RD point are considered.



(a) *Foreman* sequence

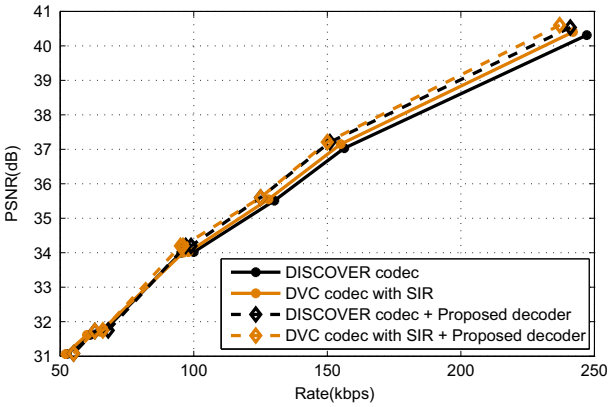


(b) *Coastguard* sequence

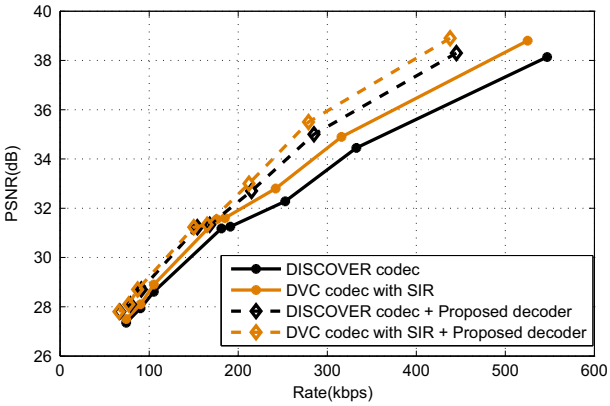
Fig. 11 RD performance for GOP size of 4

In the DISCOVER codec as well as in the DVC codec with SIR [26], the correlation noise is considered to have a Laplacian distribution and its distribution parameter is obtained online using the method proposed in [6] before the LDPCA decoder starts decoding the bitplanes. Therefore, the correlation noise parameter is kept fixed during the decoding of each WZ frame. However, in our proposed decoder, the estimation of the correlation noise parameter is refined recursively during the decoding of each DCT coefficient band. This has led to a more accurate estimation of the correlation noise parameter and consequently, to a better RD performance.

In addition, we investigate the relation between the amount of motion in the video sequences and the improvement in the RD performance resulting from using the proposed decoder. The mean of the magnitudes of forward motion vectors (MVs) between each pair of the successive frames and the average of these values over the entire video sequence for



(c) *Hall* sequence



(d) *Soccer* sequence

Fig. 11 (continued)

Table 2 The relative bit-rate savings (%) and improvement in PSNR values (dB) over that of codec with SIR [26], computed using the Bjøntegaard metric

	GOP=2				GOP=4			
	WZ frames		All frames		WZ frames		Allframes	
	$\triangle R(\%)$	△PSNR (in dB)	△R (%)	△PSNR (in dB)	△R (%)	△PSNR (in dB)	△R (%)	△PSNR (in dB)
Foreman	9.31	0.18	4.69	0.11	8.18	0.21	6.29	0.16
Coastguard	7.46	0.15	3.19	0.08	6.11	0.13	4.93	0.06
Hall	5.69	0.14	2.27	0.05	4.9	0.09	3.01	0.04
Soccer	12.71	0.32	6.29	0.17	10.86	0.26	8.31	0.19
Average	8.79	0.19	4.11	0.1	7.51	0.17	5.63	0.11

(a) DISCOVER codec modified by the proposed decoder. Decoded frame: PSNR=32.6 dB, 15575 bits

(b) DISCOVER codec. Decoded frame: PSNR=31.99 dB, 17086 bits

Fig. 12 47^{th} decoded frame of the *Foreman* sequence

each of the *Foreman*, *Coastguard*, *Hall* and *Soccer* sequences are shown in Fig. 14. It is clear from this figure that the video sequences *Soccer*, *Foreman*, *coastguard* and *Hall* have in that order the highest to lowest motion contents. From Figs. 10 and 11, it can be seen that the improvements in the RD performance resulting from using the two DVC codecs modified by the proposed decoder, respectively, over the two original DVC codecs are more significant in video sequences having higher motion content.

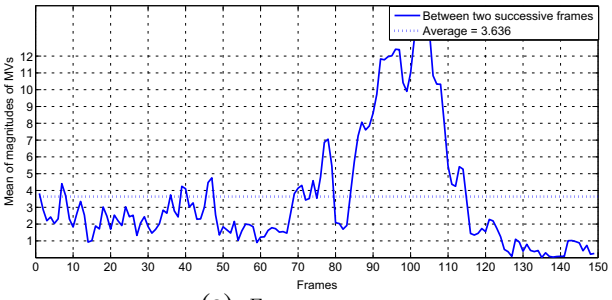
The hardware used for our simulation is a personal computer with Core i5 CPU at 2.7 GHz, and 8-GB RAM. Windows 7 operating system is used and the codec is implemented using the Visual Studio C++ v10.0 compiler in release mode on one CPU core. The execution time (in seconds) to decode each of the four video sequences (with GOP size of 2 and quantization matrix Q1) is shown in Table 3 for the original DISCOVER codec and the DISCOVER codec modified by the proposed decoder.

It should be mentioned that a parallel or multi-threaded programming on a multi-core processor or on a GPU can be used to reduce the execution time of the algorithm and speed up the decoding process significantly.

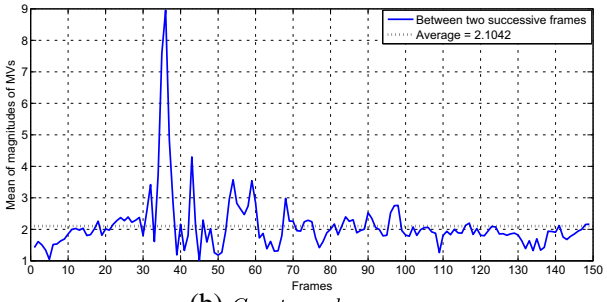
(a) DISCOVER codec modified by the proposed decoder. Decoded frame: PSNR=32.45dB, 18646 bits

(b) DISCOVER codec. Decoded frame: PSNR=31.51 dB, 20614 bits

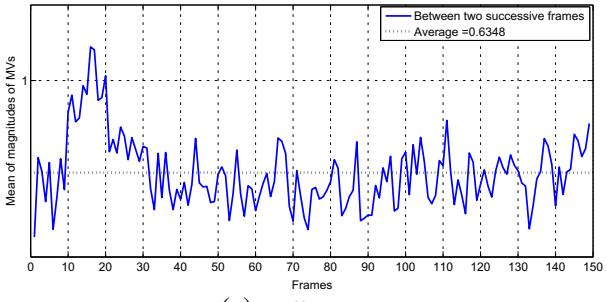
Fig. 13 85^{th} decoded frame of the *Soccer* sequence



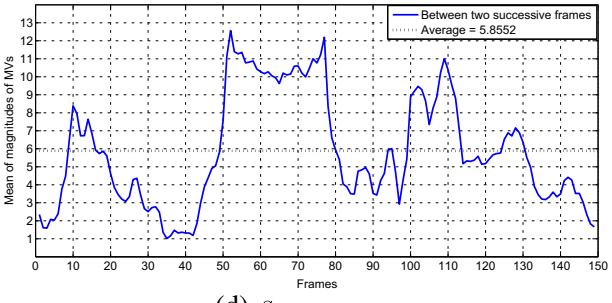
(a) *Foreman* sequence



(b) *Coastguard* sequence



(c) *Hall* sequence



(d) *Soccer* sequence

Fig. 14 Measurement of motion in the video sequences

Table 3 Execution time (in seconds) for decoding the video sequences with GOP size of 2 and quantization matrix Q1

	Execution time (s)	
sequences	DISCOVER codec	DISCOVER codec modified by the proposed decoder
Foreman	664	1354
Coastguard	489	1097
Hall	391	951
Soccer	1132	1721

5 Conclusion

Distributed video coding is a coding paradigm based on Slepian-Wolf and Wyner-Ziv theorems. In this paper, we have investigated the problem of obtaining the correlation noise parameter in the DVC decoder in order to improve the rate-distortion performance and the coding efficiency in a distributed video codec. Having a more accurate information about the correlation noise leads to a better decoding performance and consequently, a higher coding efficiency and a better rate-distortion performance of the video codec. Since the decoder does not have access to the current encoded WZ frame located at the encoder and the correlation noise is non-stationary, it is difficult to model the correlation noise and obtain its parameter accurately. To overcome these difficulties, a recursive algorithm based on variational Bayes has been proposed to estimate and refine the correlation noise distribution parameter while decoding simultaneously all the bitplanes corresponding to the current DCT band on an augmented factor graph. Unlike most of the DVC schemes in which the parameter of the correlation noise distribution is obtained before the decoding of each DCT coefficient band of the WZ frame, in our proposed decoder, the estimation of the correlation noise parameter is refined during the decoding of each DCT coefficient band. This has resulted in obtaining more accurate information about the correlation noise. The proposed decoder has then been used in the DISCOVER codec, one of the most popular codecs designed based on the Stanford approach. It has been shown through extensive simulations that the DISCOVER codec using the proposed decoder exhibits a performance that is better that of the original codec, particularly on sequences with fast motions. Such an improvement in the performance has also been observed when the proposed decoder is used in the DVC codec with SIR [36], which also has an architecture designed based on the Stanford approach. This leads us to believe that proposed decoder can be used to improve the performance of any codec whose architecture is based on the Stanford approach.

Acknowledgments This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada and in part by the Regroupement Stratégique en Microélectronique du Québec (ReSMiQ).

References

1. Aaron A, Zhang R, Girod B (2002) Wyner-Ziv coding for motion video. In: Asilomar conf. on signals, systems and computers. Pacific Grove, pp 240–244

2. Aaron A, Rane S, Setton E, Girod B (2004) Transform-domain Wyner–Ziv codec for video. In: SPIE Visual communications and image processing conf. San Jose
3. Artigas X, Ascenso J, Dalai M, Klomp S, Kubasov D, Ouaret M (2007) The DISCOVER codec: architecture, techniques and evaluation. In: Proc. of picture coding symp., Portugal, pp 1–4
4. Ascenso J, Brites C, Pereira F (2006) Content adaptive Wyner-Ziv video coding driven by motion activity. In: ICIP IEEE int. conf. on image processing. USA, pp 605–608
5. Bishop C (2006) Pattern recognition and machine learning
6. Brites C, Pereira F (2008) Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding. IEEE Trans Circ Syst Video Technol 18(9):1177–1190
7. Brites C, Ascenso J, Pereira F (2006) Studying temporal correlation noise modeling for pixel based Wyner-Ziv video coding. In: ICIP IEEE int. conf. on image process, pp 273–276
8. Bjøntegaard G (2001) Calculation of average PSNR differences between RD curves. Tech. Rep., 13th VCEGM33 Meeting. Austin
9. Carbonetto P, King M, Hamze F (2006) A stochastic approximation method for inference in probabilistic graphical models
10. Chick SE (2006) Subjective probability and Bayesian methodology. In: Henderson S, Nelson B (eds) Handbook in operations research and management science: simulation. Elsevier
11. Deligiannis N et al. (2014) Maximum likelihood Laplacian correlation channel estimation in layered Wyner-Ziv coding. In: IEEE Transactions on Signal Processing, pp 892–904
12. Dufaux F, Gao W, Tubaro S, Vetro A (2009) Distributed video coding: trends and perspectives. In: EURASIP on image and video processing (special issue on DVC)
13. Esmaili GR, Cosman P (2009) Correlation noise classification based on matching success for transform domain Wyner-Ziv video coding. In: ICASSP IEEE int. conf. on acoustics, speech, and signal process. Taipei, pp 801–904
14. Esmaili GR, Cosman P (2011) Wyner-Ziv video coding with classified correlation noise estimation and key frame coding mode selection. IEEE Trans Image Process 99
15. Fan X, Au O, Cheung N (2009) Adaptive correlation estimation for general Wyner-Ziv video coding. In: Proc. IEEE Int. conf. image process. (ICIP), pp 1409–1412
16. Fox C, Roberts S (2011) A tutorial on variational Bayesian inference. Artif Intell Rev:1–13
17. Guo M, Lu Y, Wu F, Li SP, Gao W (2007) Distributed video coding with spatial correlation exploited only at the decoder. In: ISCAS IEEE int. symp. on circuits and systems. New Orleans, pp 41–44
18. Haug AJ (2005) A tutorial on Bayesian estimation and tracking techniques applicable to nonlinear and non-Gaussian Processes. The Mitre Corporation, McLean, Virginia, pp 1–52
19. Huang X, Forchhammer S (2009) Improved virtual channel noise model for transform domain Wyner-Ziv video coding. In: IEEE International conference on acoustics, speech and signal processing, pp 921–924
20. Huang X, Forchhammer S (2012) Cross-band noise model refinement for transform domain Wyner–Ziv video coding. Signal Process Image Commun:16–30
21. Joint Video Team (JVT) reference software. [Online]. Available at: http://iphome.hhi.de/suehring/tml/index.htm
22. Kubasov D, Lajnef, Guillemot K (2007) A hybrid encoder/decoder rate control for a Wyner-Ziv video codec with a feedback channel. In: IEEE MultiMedia signal processing workshop, pp 251–254
23. Loeliger H-A (2004) An introduction to factor graphs. IEEE Signal Process Mag 21.1:28–41
24. Luong H, Huang X, Forchhammer S (2011) Parallel iterative decoding of transform domain Wyner-Ziv video using cross bitplane correlation. In: Proc. IEEE int. conf. image process. (ICIP). Belgium, pp 2633–2636
25. Macchiavello B, Mukherjee D, Querioz R. L. (2009) Iterative side-information generation in a mixed resolution wyner-ziv framework. IEEE Trans Circ Syst Video Technol 19(10):1409–1423
26. Martins R et al. (2009) Refining side information for improved transform domain Wyner-Ziv video coding. In: IEEE Transactions on circuits and systems for video technology, pp 1327–1341
27. Meyer P, Westerlaken R, Gunnewiek R, Lagendijk R (2005) Distributed source coding of video with non-stationary side-information. Proc SPIE 5960:857–866
28. Stankovic L, Stankovic V, Wang S, Cheng S (2009) Correlation estimation with particle-based belief propagation for distributed video coding. In: Proc. IEEE int. conf. acoust. speech signal process. (ICASSP), pp 1505–1508
29. Slepian JD, Wolf JK (1973) Noiseless coding of correlated information sources. IEEE Trans Inf Theory IT-19:471–480
30. Varodayan D, Aaron A, Girod B (2006) Rate-adaptive codes for distributed source coding, vol 86
31. Varodayan D, Chen D, Flierl M, Girod B (2008) Wyner-Ziv coding of video with unsupervised motion vector learning. Elsevier Signal Process Image Commun 23(5):369–378

32. Velotiaray T-Z, Roumy A, Guillemot C (2011) Maximum likelihood BSC parameter estimation for the Slepian-Wolf problem. IEEE Commun Lett:232–234
33. Wang S, Cui L, Stankovic L, Stankovic V, Cheng S (2012) Adaptive correlation estimation with particle filtering for distributed video coding. IEEE Trans Circuits Syst Video Technol 22(5):649–658
34. Wyner AD, Ziv J (1976) The rate-distortion function for source coding with side information at the decoder. IEEE Trans Inf Theory IT-22(1):1–10
35. Yedidia JS, Freeman WT, Weiss Y (2003) Understanding belief propagation and its generalizations. Explor Artif Intell New Millen 8:236–239
36. Zia A, Reilly JP, Shahram S (2007) Distributed parameter estimation with side information: a factor graph approach. IEEE Int Symp Inf Theory:2556–2560

Yaser Mohammad Taheri received the B.Sc. degree from University of Tehran, Tehran, Iran, in 2006 and the M.Sc. degree from Shiraz University, Shiraz, Iran, in 2010, both in Electrical and Computer Engineering. He is currently working toward the Ph.D. degree in Electrical and Computer Engineering at Concordia University, Montreal, QC, Canada. He has been a Research Associate in Signal Processing Group at Concordia University from September 2010 where he has worked on development of distributed video coding (DVC) and improving its rate-distortion performance. His research areas of interest include image processing, video processing and machine learning with application in video coding.

M. Omair Ahmad received the B.Eng. degree from Sir George Williams University, Montreal, QC, Canada, and the Ph.D. degree from Concordia University, Montreal, QC, Canada, both in electrical engineering. From 1978 to 1979, he was a Faculty Member with the New York University College, Buffalo, NY, USA. In September 1979, he joined the Faculty of Concordia University as an Assistant Professor of computer science. He joined the Department of Electrical and Computer Engineering, Concordia University, where he was the Chair with the department from June 2002 to May 2005 and is currently a Professor. He holds the Concordia University Research Chair (Tier I) in Multimedia Signal Processing. He has published extensively in the area of signal processing and holds four patents. His current research interests include the areas of multidimensional filter design, speech, image and video processing, non-linear signal processing, communication DSP, artificial neural networks, and VLSI circuits for signal processing. He was a Founding Researcher at Micronet from its inception in 1990 as a Canadian Network of Centers of Excellence until its expiration in 2004. Previously, he was an Examiner of the order of Engineers of Quebec. Dr. Ahmad was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART I: FUNDAMENTAL THEORY AND APPLICATIONS from June 1999 to December 2001. He was the Local Arrangements Chairman of the 1984 IEEE International Symposium on Circuits and Systems. In 1988, he was a member of the Admission and Advancement Committee of the IEEE. He has served as the Program Co-Chair for the 1995 IEEE International Conference on Neural Networks and Signal Processing, the 2003 IEEE International Conference on Neural Networks and Signal Processing, and the 2004 IEEE International Midwest Symposium on Circuits and Systems. He was a General Co-Chair for the 2008 IEEE International Conference on Neural Networks and Signal Processing. He is the Chair of the Montreal Chapter IEEE Circuits and Systems Society. He is a recipient of numerous honors and awards, including the Wighton Fellowship from the Sandford Fleming Foundation, an induction to Provosts Circle of Distinction for Career Achievements, and the Award of Excellence in Doctoral Supervision from the Faculty of Engineering and Computer Science of Concordia University. Dr. Ahmad is a Fellow of the Institute of Electrical and Electronics Engineers.

M. N. S. Swamy received the B.Sc. (Hons.) degree in mathematics from Mysore University, India, in 1954, the Diploma degree in electrical communication engineering from the Indian Institute of Science, Bangalore, in 1957 and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Saskatchewan, Saskatoon, Canada, in 1960 and 1963, respectively. He was conferred in 2009 the title of Honorary Professor at National Chiao Tong University in Taiwan. He is presently a Research Professor in the Department of Electrical and Computer Engineering at Concordia University, Montreal, QC, Canada, where he served as the Chair of the Department of Electrical Engineering from 1970 to 1977, and Dean of Engineering and Computer Science from 1977 to 1993. During that time, he developed the Faculty into a research-oriented one from what was primarily an undergraduate Faculty. Since July 2001, he holds the Concordia Chair (Tier I) in Signal Processing. He has also taught in the Electrical Engineering Department of the Technical University of Nova Scotia, Halifax, and the University of Calgary, Calgary, as well as in the Department of Mathematics at the University of Saskatchewan. He has published extensively in the areas of number theory, circuits, systems and signal processing, and holds five patents. He is the coauthor of six books and three book chapters. He was a founding member of Micronet from its inception in 1990 as a Canadian Network of Centers of Excellence until its expiration in 2004, and also its coordinator for Concordia University. Dr. Swamy is a Fellow of the Institute of Electrical and Electronics Engineers, Fellow of the Institute of Electrical Engineers (United Kingdom), the Engineering Institute of Canada, the Institution of Engineers (India), and the Institution of Electronic and Telecommunication Engineers (India). In 2008, Concordia University instituted the M.N.S. Swamy Research Chair in Electrical Engineering as a recognition of his research contributions. He was inducted in 2009 to the Provosts Circle of Distinction for career achievements. He has served the IEEE in various capacities such as the President-Elect in 2003, President in 2004, Past-President in 2005, Vice President (Publications) during 20012002, Vice-President in 1976, Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I from June 1999 to December 2001, Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS during June 1985 to May 1987, Program Chair for the 1973 IEEE CAS Symposium, General Chair for the 1984 IEEE CAS Symposium, Vice-Chair for the 1999 IEEE Circuits and Systems (CAS) Symposium, and a member of the Board of Governors of the CAS Society. He is the recipient of many IEEE-CAS Society awards, including the Education Award in 2000, Golden Jubilee Medal in 2000, and the 1986 Guillemin-Cauer Best Paper Award. He is the Editor-in-Chief of the journal Circuits, Systems and Signal Processing (CSSP) since 1999. Recently CSSP has instituted a best paper award in his name.