Improving the Rate-Distortion Performance in Distributed Video Coding

Yaser Mohammad Taheri

A Thesis
In the Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy (Electrical and Computer Engineering) at
Concordia University
Montreal, Quebec, Canada

August 2017

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: Yaser Mohammad Taheri

Entitled: Improving the Rate-Distortion Performance in Distributed Video Coding

and submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY (Electrical & Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. Deborah Dysart-Gale

_____ External Examiner
Dr. Douglas O'Shaughnessy

_____ External to Program
Dr. Chun-Yi Su

_____ Examiner
Dr. William E. Lynch

_____ Examiner
Dr. Wei-Ping Zhu

_____ Supervisor
Dr. M.O. Ahmad

_____ Supervisor
Dr. M.N.S. Swamy

Approved by _____
Dr. Wei-Ping Zhu, Graduate Program Director

September 7, 2017
Date of Defence

_____
Dr. Amir Asif, Dean, Faculty of Engineering and Computer Science

ABSTRACT

Improving the Rate-Distortion Performance in Distributed Video Coding

Yaser Mohammad Taheri, Ph. D.
Concordia University, 2017.

Distributed video coding is a coding paradigm, which allows encoding of video frames at a complexity that is substantially lower than that in conventional video coding schemes. This feature makes it suitable for some emerging applications such as wireless surveillance video and mobile camera phones. In distributed video coding, a subset of frames in the video sequence, known as the key frames, are encoded using a conventional intra-frame encoder, such as H264/AVC in the intra mode, and then transmitted to the decoder. The remaining frames, known as the Wyner-Ziv frames, are encoded based on the Wyner-Ziv principle by using the channel codes, such as LDPC codes. In the transform-domain distributed video coding, each Wyner-Ziv frame undergoes a 4x4 block DCT transform and the resulting DCT coefficients are grouped into DCT bands. The bitplaines corresponding to each DCT band are encoded by a channel encoder, for example an LDPCA encoder, one after another. The resulting error-correcting bits are retained in a buffer at the encoder and transmitted incrementally as needed by the decoder. At the decoder, the key frames are first decoded. The decoded key frames are then used to generate a side information frame as an initial estimate of the corresponding Wyner-Ziv frame, usually by employing an interpolation method. The difference between the DCT band in the side information frame and the corresponding one in the Wyner-Ziv

frame, referred to as the correlation noise, is often modeled by Laplacian distribution. A soft-input information for each bit in the bitplane is obtained using this correlation noise model and the corresponding DCT band of the side information frame. The channel decoder then uses this soft-input information along with some error-correcting bits sent by the encoder to decode the bitplanes of each DCT band in each of the Wyner-Ziv frames. Hence, an accurate estimation of the correlation noise model parameter(s) and generation of high-quality side information are required for reliable soft-input information for the bitplanes in the decoder, which in turn leads to a more efficient decoding. Consequently, less error-correcting bits need to be transmitted from the encoder to the decoder to decode the bitplanes, leading to a better compression efficiency and rate-distortion performance.

The correlation noise is not stationary and its statistics vary within each Wyner-Ziv frame and within its corresponding DCT bands. Hence, it is difficult to find an accurate model for the correlation noise and estimate its parameters precisely at the decoder. Moreover, in existing schemes the parameters of the correlation noise for each DCT band are estimated before the decoder starts to decode the bitplanes of that DCT band and they are not modified and kept unchanged during decoding process of the bitplanes. Another problem of concern is that, since side information frame is generated in the decoder using the temporal interpolation between the previously decoded frames, the quality of the side information frames is generally poor when the motions between the frames are non-linear. Hence, generating a high-quality side information is a challenging problem.

This thesis is concerned with the study of accurate estimation of correlation noise model parameters and increasing in the quality of the side information from the standpoint of improving the rate-distortion performance in distributed video coding.

A new scheme is proposed for the estimation of the correlation noise parameters wherein the decoder decodes simultaneously all the bitplanes of a DCT band in a Wyner-Ziv frame and then refines the parameters of the correlation noise model of the band in an iterative manner. This process is carried out on an augmented factor graph using a new recursive message passing algorithm, with the side information generated and kept unchanged during the decoding of the Wyner-Ziv frame. Extensive simulations are carried out showing that the proposed decoder leads to an improved rate-distortion performance in comparison to the original DISCOVER codec and in another DVC codec employing side information frame refinement, particularly for video sequences with high motion content.

In the second part of this work, a new algorithm for the generation of the side information is proposed to refine the initial side information frame using the additional information obtained after decoding the previous DCT bands of a Wyner-Ziv frame. The simulations are carried out demonstrating that the proposed algorithm provides a performance superior to that of schemes employing the other side information refinement mechanisms. Finally, it is shown that incorporating the proposed algorithm for refining the side information into the decoder proposed in the first part of the thesis leads to a further improvement in the rate-distortion performance of the DVC codec.

ACKNOWLEDGEMENTS

It is my pleasure to express my deep gratitude and thanks to my supervisors, professor M.O. Ahmad and professor M.N.S Swamy for their continuous guidance and support throughout the course of this research. Their valuable suggestions and positive responses have been very useful, and were among the major reasons that enabled me to pursue my research. I consider my experience with them, as my supervisors, a very rich one, from which I have learned a lot, and I would like to thank them especially for that.

Special thanks and gratitude are due to my dear wife for her patience, encouragement and continuous support for me. I would like also to thank my parents and other family members who supported me and were available in times of need and eased the hardships of my life. Special gratitude to my mother and my father who are the first inspiration for me in the field of research work.

TABLE OF CONTENTS

List of Figures

List of Tables

List of Symbols

B_c	c^{th} bitplane in a DCT band
b_{ci}	i^{th} bit in c^{th} bitplane
c_j	j^{th} check node
D	observation
$H(X)$	Entropy of X
H_j	Hidden variable matrix for j^{th} block
I_k	k^{th} quantization bin interval
K	Number of quantization bins
L	Normalization factor
L^i	Initial Log-likelihood ratio for i^{th} bit (intrinsic LLR)
l^i	Log-likelihood ratio for i^{th} bit
$\ell(q)$	Variational lower bound for q
M	Number of DCT coefficients in each block
N	Number of DCT coefficients in each DCT band
Q_k	k^{th} Quantization matrix
$q(z)$	Variational probability or approximation for $P(Z \mid D)$
R_X	Rate for encoding sequence X
$\Re(x, y)$	Residual between backward and forward motion compensated frames

s	Syndrome bits
S	Number of the disjoint set in variational Bayes algorithm
t_c	Number of the received syndrome bits for bitplane B_c
T	DCT frame obtained by applying 4x4 DCT to $\Re(x, y)$
T_i	i^{th} DCT band of T
X_{DCT}	A DCT band of a WZ frame
X_F	Future decoded frame
X_P	Past decoded frame
X_{DCT}^{P}	DCT of a backward motion compensated frame
X_{DCT}^{F}	DCT of a forward motion compensated frame
w	Possible values for the reconstructed DCT coefficient
Y	A sequence correlated to X or side information of X
Y_{SI}	Side information frame
Z	Set of hidden and unknown parameters
z_i	i^{th} disjoint subset of set Z
β_k	Number of bitplanes in k^{th} DCT band
δ_{b_k}	Distance between the k^{th} coefficient of $\lvert T_b \rvert$ and the its mean
$\mu_{a\to b}$	Message from node a to node b on the augmented factor graph
η	Number of non-zero elements in a Quantization matrix
θ_j	Correlation noise parameter for block j

List of Abbreviations

AVC	Advanced video coding
BP	Belief propagation
BSC	Binary symmetric channel
CRC	Cyclic redundancy check
CNPR	Correlation noise parameter refinement
DCT	Discrete cosine transform
DSC	Distributed source coding
DVC	Distributed video coding
GOP	Group of pictures
i.i.d	independent and identically distributed
KL	Kullback-Leibler
LDPC	Low-density parity check
LDPCA	Low-density parity check accumulated
LLR	Log likelihood ratio
MAP	Maximum a posteriori
MCMC	Markov chain Monte Carlo
MLE	Maximum likelihood Estimation
MCFI	Motion compensated frame interpolation
MCI	Motion compensated interpolation
MMSE	Minimum mean square error

MPEG	Moving Picture Experts Group
QCIF	Quarter common intermediate format
QP	Quality parameter
PSNR	Peak signal-to-noise ratio
RD	Rate distortion
SI	Side information
SIR	Side information refinement
SRSIF	Successive refinement of side information frame
SW	Slepian-Wolf
VB	Variational Bayes
WZ	Wyner-Ziv

CHAPTER 1

Introduction

Video coding plays an important role in both storage and transmission of video data in applications such as high definition TV, video broadcasting and video on demand. In the commonly used and highly efficient hybrid video coding paradigm, used in ISO MPEG-x and ITU-T H.26x standards [1], [2], video data are compressed by employing transform and predictive coding. In these video coding schemes, the encoder exploits spatial and temporal redundancies among the frames to perform efficient compression. In conventional video codecs, such as MPEG-4 and H264/AVC, the computational complexity of the encoder is 5 to 10 times higher than that of the decoder [3]. This high complexity of the encoder results mostly from the computationally expensive motion estimation and compensation algorithm. Hence, the conventional video coding schemes are well-suited for broadcasting applications where video stream is encoded only once at the transmitter but decoded at many different receivers. In contrast, there are a number of applications, such as wireless surveillance video and mobile camera, in which the size and power efficiency of the transmitting devices are very important. In this kind of applications, it is preferred to have a lightweight encoding even at the expense of a highly complex decoder while having the compression efficiency close to that of conventional video coding schemes [4]. To meet the requirements of these emerging applications, a new video coding paradigm, known as distributed video coding (DVC) or Wyner-Ziv video coding, has been proposed. In the next subsection, some of the applications that can beneficially adapt this video coding scheme, are briefly discussed.

1.1 Some Possible Applications of Distributed Video Coding

The following are a few examples of potential applications of distributed video coding where the power-efficiency and complexity of the encoder is important [5].

• **Wireless low-power surveillance:** Nowadays, surveillance systems are almost universally used. In a video surveillance system, such as the one in Figure 1.1, multiple cameras are installed at different locations to cover a specific region for the purpose of security. Since the number of encoders is much higher in comparison to only one decoder located at the central station, the cost of the system can be reduced if low-complexity encoders are used in the cameras. Distributed video coding is well suited for this scenario, since it results in shifting the complexity from the encoders to the decoder.

Figure 1.1 DVC application, in a wireless surveillance system [6]

• **Video conferencing with mobile devices:** Another application that can take advantage of the distributed video coding paradigm is wireless video communication among a number of camera phones. In this application, since battery life and power consumption depends on the complexity of both the encoder and decoder of a camera phone, a low-complexity encoder as well as a low complexity decoder is required at each terminal. A low-complexity encoder can be achieved by using Wyner-Ziv encoding at the transmitting terminal and by having a high-complexity decoder at the base station. The bit streams received in the base station can then be transcoded into H.26x bit stream and transmitted to the receiving terminal that can, therefore, be required to have only a low complexity decoder. Hence, in this application, the encoder and decoder in each terminal would have low-complexity.

• **Visual sensor networks**: Distributed video coding can also be used in visual sensor network applications, such as urban traffic management system. In a typical visual sensor network, as shown in Figure 1.2, a set of power-efficient camera nodes collect the visual information at different locations and the network performs distributed and collaborative data processing [7]. By using the distributed video coding, the sensor camera nodes can be made to have simple encoders and have the complexity transferred to only a few central nodes for carrying out the decoding. This feature would be very beneficial in view of the resource limitation in terms of the battery life at the network nodes.

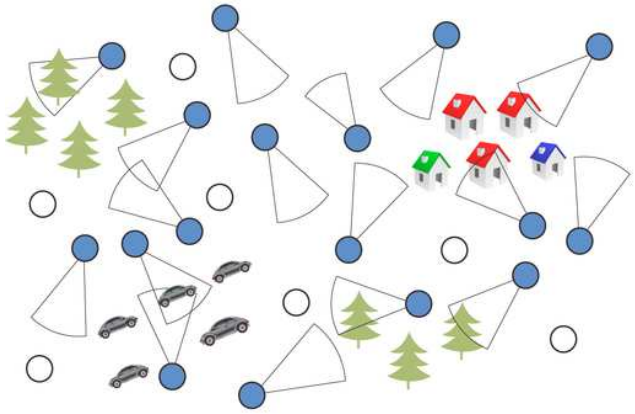


Figure 1.2 DVC application, visual sensor network [8].

The distributed video coding paradigm originates from the more general concept of distributed source coding. The distributed source coding is an important concept in the fields of information theory and communication. In the next subsection, the distributed source coding and its underlying theory are explained.

1.2 Distributed Source Coding

Distributed source coding (DSC) refers to a coding scheme that encodes two or more correlated sequences separately, but decodes them jointly. In this coding paradigm, an independent sequence is transmitted from each encoder to a single decoder, which jointly decodes all the received sequences using the correlation and dependency information between them. In other words, the source statistics is not used at the encoder and exploited only in the decoder [9]. Hence, this results in having a much simpler encoder at the expense of a complex decoder. It is different from the traditional coding and compression schemes that carry out joint encoding, leading to a complex encoder [3].

The distributed source coding is based on two important theorems called Slepian-Wolf theorem [10] and Wyner-Ziv theorem [11].

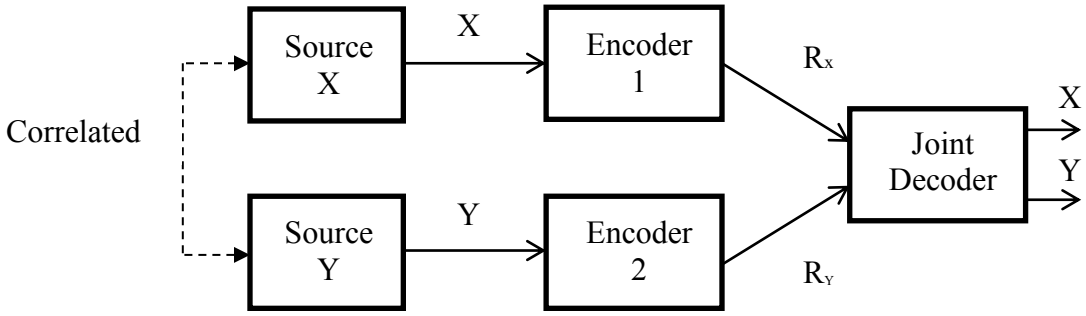


Figure 1.3 Separate encoding and joint decoding

1.2.1 Slepian-Wolf theorem

The Slepian-Wolf theorem [10] addresses the problem of independent encoding of two statistically dependent random sequences, X and Y, each being independent and identically distributed (i.i.d). This encoding is different from the joint encoding used in the conventional predictive coding solutions. Slepian and Wolf were the first ones to show that in this case, these dependent and correlated sequences, X and Y can be decoded, with a very small probability of error, requiring the bitrates R_X and R_Y to be lower than their individual entropies, $H(X)$ and $H(Y)$, respectively. According to the Slepian-Wolf theorem, if a joint decoding is performed, the minimum rate to encode the two correlated sources independently is the same as the minimum rate achievable in the joint encoding. Figure 1.3 shows two dependent and correlated source sequences *X* and *Y,* each being i.i.d and separately encoded with rates R_X and R_Y, respectively, but jointly

decoded by exploiting the dependency and correlation between them. Slepian and Wolf established the bounds on these rates given by

$$R_X > H(X \mid Y) \tag{1.1}$$

$$R_Y > H(Y \mid X) \tag{1.2}$$

$$R_X + R_Y > H(X,Y) \tag{1.3}$$

where $H(X \mid Y)$ and $H(Y \mid X)$ are conditional entropies and $H(X,Y)$ is the joint entropy for X and Y. Equation (1.3) states that, even with separate encoding, the total rate, $R = R_X + R_Y$, can achieve the joint entropy $H(X,Y)$, just as for the joint encoding in conventional coding schemes. The region corresponding to Equations (1.1) - (1.3) is shown in Figure 1.4. Since two dependent sequences that are encoded independently can be decoded with very small probability of error using a joint decoder [12], the Slepian-Wolf coding is usually called the lossless distributed source coding.

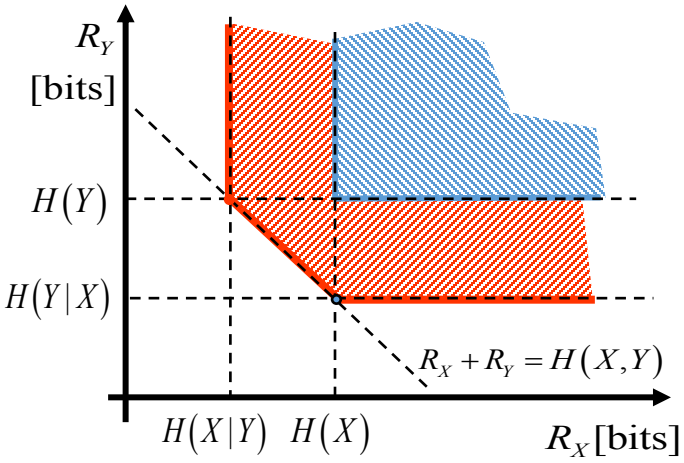


Figure 1.4 Slepian-Wolf rate region

1.2.2 Wyner-Ziv theorem

Wyner and Ziv [11] investigated a specific case of Slepian-Wolf coding, that is, the case of encoding the sequence *X* considering that the statistically dependent sequence *Y*, called the side information, is available at the decoder and used to decode *X*, as shown in Figure 1.5. This coding is also known in the literature as lossy compression with decoder side information [13]. The term lossy compression comes from the fact that Wyner and Ziv considered a distortion *d* between the sequence *X* to be encoded, and its decoded version, *X'*. Letting $R^{WZ}_{X|Y}(d)$ to denote the rate of encoding the sequence *X* when the side information *Y* is available only at the decoder and $R_{X|Y}(d)$ to denote the rate of encoding the sequence *X* when *Y* is available at the encoder, then there is a rate loss of $R^{WZ}_{X|Y}(d) - R_{X|Y}(d) \geq 0$ when the side information *Y* is not available at the encoder. Wyner and Ziv have shown that by having the Gaussian memoryless sequences and by considering a mean-squared error distortion, $R^{WZ}_{X|Y}(d) - R_{X|Y}(d) = 0$ is achieved. In other words, the overall bit rate can still achieve the joint entropy $H(X,Y)$, even if the side information *Y* is not available at the encoder.

The WZ theorem constitutes the basis of the distributed video coding, in which, a video frame *X* at the encoder is considered to be the source and a frame *Y* generated at the decoder using previously decoded frames is considered as side information.

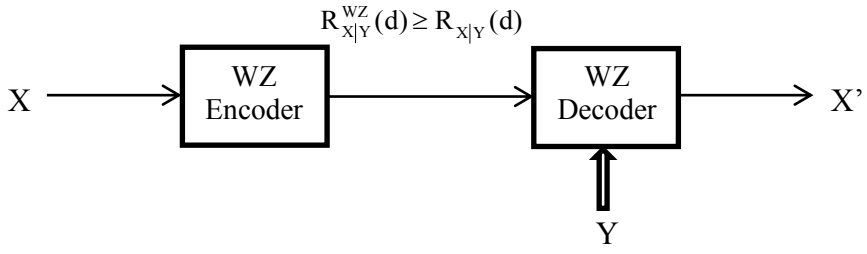


Figure 1.5 Lossy compression with decoder side information.

1.3 Distributed Video Coding

Distributed video coding, also called the Wyner-Ziv video coding, is a video coding paradigm that takes advantage of the distributed source coding principle enunciated by the Slepian-Wolf and Wyner-Ziv theorems for compressing video sequences. In a DVC paradigm, the computationally intensive temporal prediction and estimation are avoided at the encoder; instead, the temporal redundancy is exploited at the decoder. Thus, the complexity is shifted from the encoder to the decoder. Practical solutions for the distributed video coding were first introduced by two different groups from the University of California, Berkeley and Stanford University in 2002, after some significant advances were made in the preceding years in the channel coding problem and near-Shannon capacity error correction codes, such as turbo and LDPC codes. The Berkeley distributed video coding [14] is based on a block-based coding with motion estimation at the decoder. In contrast, the Stanford distributed video coding architecture [15] is a frame-based coding using error correcting codes such as turbo codes and low-density parity-check (LDPC) codes. The scheme of [15] in comparison to that of [14] also uses a feedback channel for the rate control at the decoder. The rate-distortion performance of the Stanford DVC codec is quite significant compared with the Berkley's [14]. Hence, the Stanford DVC architecture has become the most popular distributed video coding design in the research community. In view of this, in this thesis, the DVC codec (Wyner-Ziv video codec) based on the Stanford architecture is considered.

The Stanford distributed video coding scheme was proposed in 2002, first for the pixel domain [15] and then was extended in 2004 to the transform domain [16]. In the transform-domain DVC, the spatial correlation within each frame and the temporal

correlation among adjacent frames are used to improve the rate-distortion performance. It is because of this superiority, the transform–domain DVC codec is considered a state-of-art DVC codec.

1.3.1 Transform–domain distributed video Coding

The most well-known and popular transform-domain DVC codec based on Stanford architecture, the DISCOVER codec, was developed by a European research group [17]. The architecture of this transform-domain DVC codec is shown in Figure 1.6.

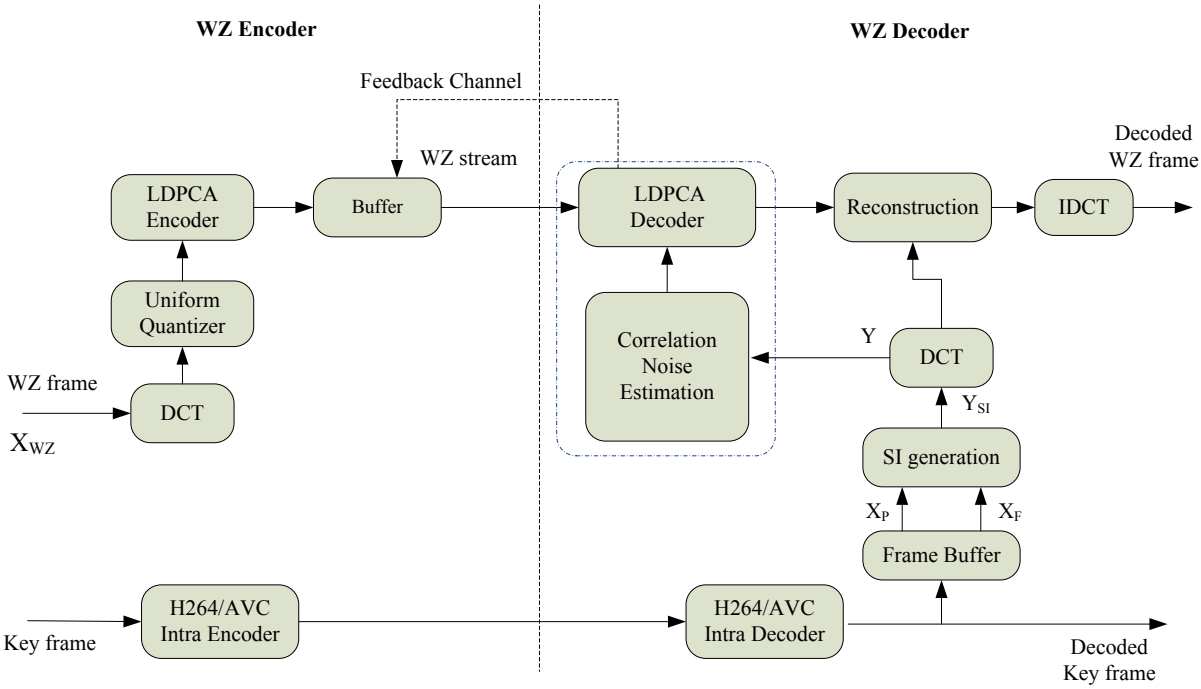


Figure 1.6 DISCOVER codec, a DVC codec based on the Stanford-based transform-domain DVC architecture

In the transform-domain DVC codec based on Stanford architecture, encoding process is carried out as follows:

1- A fixed size of GOP is used to split a video sequences into two sets of frames, a set of Key frames and another set of Wyner-Ziv (WZ) frames.

The first frame in each GOP of the video sequence is a key frame and the rest are the WZ frames.

2- The key frame are encoded by using a conventional intra-frame encoding such as H.264/AVC Intra encoding.

3- A Wyner-Ziv frame is partitioned into N non-overlapped blocks, each of size 4×4. The discrete cosine transform (DCT) is then applied individually to each block.

4- The DCT coefficients of all the blocks in the Wyner-Ziv frame with the same frequency are grouped together in a zig-zag order to provide 16 DCT coefficient bands (one DC and 15 AC bands) each of size *N*, as shown in Figure 1.7.

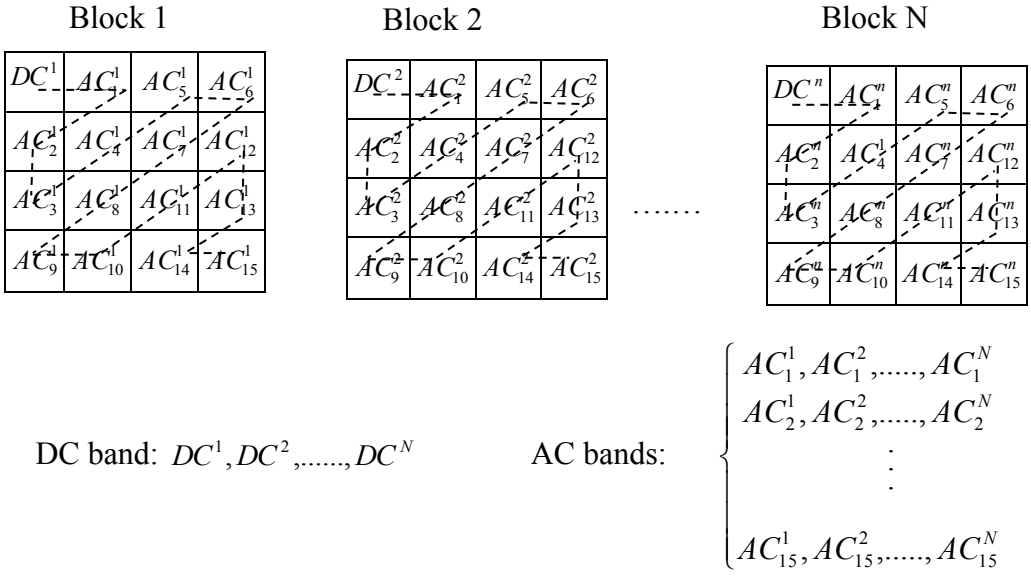


Figure 1.7 4×4 DCT blocks and constructed DCT bands

5- The DCT coefficients in the b^{th} DCT band, $b = 1,2,...,16$, are quantized by a uniform quantizer with the number of quantization bins K_b to obtain quantized symbols. Depending on the target quality of the decoded video, different quantization matrices are used as shown in Figure 1.8, where their elements indicate the number of quantization levels K_b used for b^{th} DCT bands. For example, if the quantization matrix Q_3 is chosen, 32 quantization levels for the first DCT band (DC band), 8 quantization levels for the 2nd and 3rd DCT bands, 4 quantization levels for the 4th ,5th and 6th DCT bands, and with no quantization for the remaining bands would be used. A zero in quantization matrix indicates that the corresponding band is neither quantized nor encoded.

Q_1

16	8	0	0
8	0	0	0
0	0	0	0
0	0	0	0

Q_2

32	8	0	0
8	0	0	0
0	0	0	0
0	0	0	0

Q_3

32	8	4	0
8	4	0	0
4	0	0	0
0	0	0	0

Q_4

32	16	8	4
16	8	4	0
8	4	0	0
4	0	0	0

Q_5

32	16	8	4
16	8	4	4
8	4	4	0
4	4	0	0

Q_6

64	16	8	8
16	8	8	4
8	8	4	4
8	4	4	0

Q_7

64	32	16	8
32	16	8	8
16	8	8	4
8	8	4	0

Q_8

128	64	32	16
64	32	16	8
32	16	8	4
16	8	4	0

Figure 1.8 Eight quantization matrices for quantizing the DCT coefficients used in the transform-domain DVC [18]

5- Each quantized symbol in the DCT band b is binarized using $\beta_b = \log_2 K_b$ number of bits. The bits with the same significance in each binarized symbol of b^{th} DCT band are then grouped together to form β_b bitplanes. Each bitplane is then fed to a rate-adaptive low-density parity check accumulate (LDPCA) encoder [19] to be encoded, starting from the most significant bitplane to the least significant one. Then, the corresponding accumulated syndrome bits and 8 cyclic redundancy check (CRC) bits [20] corresponding to each bitplaneare are stored in a buffer.

The decoding process at the decoder is carried out as follows:

1- The Key frames are first decoded by a conventional intra-frame decoder such as the H264/AVC intra decoder.

2- A side information frame Y_{SI} corresponding to each WZ frame is generated using the previously decoded nearest past and future WZ frame(s) or Key frame(s). A motion compensation interpolation (MCI) framework is usually employed to generate a side information frame. In the DISCOVER codec, an advanced hierarchical motion compensation interpolation procedure is carried out in order to generate the side information (SI) frame. The SI frame Y_{SI} is considered to be a noisy version of the corresponding WZ frame or an initial estimate of this WZ frame.

3- A 4x4 DCT of the type employed at the encoder for a WZ frame is then applied to the side information frame in order to obtain the DCT coefficients. These DCT coefficients are then arranged into DCT bands in the same way as done in Step 3 of the encoding.

4- The difference between a DCT coefficient of the WZ frame and the corresponding one of the SI frame is called the correlation noise and it is assumed to have a Laplacian distribution. The parameter of this Laplacian distribution is then estimated online at the decoder.

5- After receiving the first batch of the accumulated syndrome bits corresponding to each bitpalane of a WZ frame's DCT band from the encoder, a factor graph for the LDPCA decoder is constructed in the DVC decoder to decode this bitplane. For each bit within the bitplane, a soft-input information is obtained as the conditional probability of that bit being 0 or 1 given the corresponding DCT coefficient of the SI frame and the previously decoded bitplanes. This conditional probability is calculated using the distribution of the correlation noise. The soft-input information for the bits within the bitplane is then fed to the LDPCA decoder to start decoding the bitplane. If the decoder fails to decode the bitplane under a pre-specified criterion, more accumulated syndrome bits from the encoder are requested by the decoder through a feed-back channel. This decoding procedure is carried out for all the β_b bitplanes corresponding to the b^{th} DCT band of the WZ frame.

6- All the β_b decoded bitplanes corresponding to b^{th} DCT band of the WZ frame are converted into a band of symbols corresponding to the b^{th} DCT band. The value of each symbol determines the quantization bin in which the corresponding DCT coefficient is placed. Each DCT coefficient in b^{th} DCT band of the WZ frame are then reconstructed using the corresponding quantization bin, the correlation noise model and the DCT coefficients of the SI frame [21].

7- The reconstructed DCT coefficients of a DCT band are placed in the corresponding locations of the corresponding 4×4 blocks in order to make a reconstructed DCT frame. This process is actually carried out in an order reverse to that carried out in Step 3 of the encoding.

8- A 4×4 inverse DCT is finally applied to each of the blocks obtained in Step 7 in order to obtain the decoded Wyner-Ziv frame.

1.4 Improving the Rate-Distortion Performance in DVC: A Literature Review

The distributed video coding has not yet achieved the same compression efficiency and rate-distortion performance as that of the conventional hybrid video coding, such as H.264/AVC. In order to use the DVC codec in the applications described in Section 1.1, the compression efficiency of the DVC coding needs to be at least as high as that of the other video coding schemes with a low-complexity encoder such as H.264/AVC Intra.

Hence, one of the main challenges in DVC is to improve the RD performance so as to outperform the other video coding solutions, without impacting adversely the complexity of the encoder. The correlation noise model and the side information play important roles on the coding efficiency and RD performance of DVC codecs. In this section, some of the techniques that focus on the correlation noise and side information in order to improve the coding efficiency and RD performance in DVC are reviewed.

Correlation noise model: The difference between the corresponding coefficients of a WZ frame and the associated SI frame is referred to as the correlation noise. The accuracy of modeling this correlation noise at the decoder has a considerable impact on the RD performance and coding efficiency of a DVC codec. An accurate modeling of this correlation noise is essential for an efficient use of the error-correcting bits sent by the encoder, which consequently leads to a lower bit rate and a better RD performance. In distributed video coding, a number of research activities have been undertaken to improve the coding efficiency and RD performance by developing techniques to obtain a more accurate correlation noise model.

A comprehensive study for the relationship between the coding efficiency and the correlation noise model has been carried out in [22]. In this paper, it has been shown that the choice of the correlation noise model and its parameter have significate effect on the coding efficiency and RD performance. In [23], a study on compression bound and coding efficiency using different correlation models has been conducted. The Laplacian distribution is often used in the literature to model the correlation noise in both the transform-domain DVC codecs and the pixel-domain DVC codecs [24], [25]. In [26],

Laplacian distribution has been validated for the correlation noise model with a goodness-of-fit test.

The parameters of the correlation noise can be computed offline at the encoder using the original WZ frame and the side information frame. In this method, since the side information frame is generated at the encoder by carrying out a motion estimation procedure, the complexity of the encoder is increased. The correlation noise parameters can also be calculated offline by employing a training method using a set of video sequences [27] [28]. In this case, however, the same parameter for the correlation noise model is used for all the DCT bands irrespective of the frame of the video sequence. Thus, in this scheme, the non-stationary behavior of the correlation noise is not taken into account.

The parameter values of the correlation noise model have also been estimated online at the decoder that does not have access to the original WZ frames, which is a realistic practical solution [29]-[36]. For the pixel-domain distributed video coding, Brites *et. al* [29] have proposed several online schemes that make use of the temporal correlation between the frames to estimate the correlation noise at the level of frame, block or pixel. They have estimated the correlation noise parameters at the block and pixel levels by using the spatial correlation within each frame, which are more accurate than that obtained at the frame-level. In 2008, Brites and Pereira extended their work [29] on the pixel-domain distributed video coding to the transform-domain by estimating the correlation noise model parameters at the DCT band and coefficient levels [30]. In 2009, Haung and Forchhammer [31] improved the method proposed in [30] by considering the cross-band correlation and using a classification map that is refined after each DCT band

is decoded. Esmaili and Cosman [32], [33] 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 of the individual blocks, a value for the Laplacian distribution parameter is assigned to all of the blocks belonging to a specific class using a lookup table. In some methods for estimating the correlation noise parameter in the transform-domain DVC, the information on the previously decoded DCT bands is used to improve the estimation of the correlation noise in the succeeding bands, which consequently leads to improving the decoding of the successive bands progressively [34], [35]. In [36], 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 within a DCT band is not taken into account, that is, only one parameter is estimated for the entire the DCT band. In [37] and [38], 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 in the message-passing algorithm, it may lead to unpredictable results. Further, the method is slow, since it requires a large number of iterations.

Side information generation: RD performance and compression efficiency of a DVC codec also depends on the quality of the side information generated at the decoder. The side information is usually generated based on motion estimation interpolation and extrapolation methods using the previously decoded nearest past and future WZ frame(s) or Key frame(s). Higher the quality of the side information frame, more the similarity between the side information frame at the decoder and the corresponding Wyner-Ziv frame at the encoder. Thus, having high-quality side information frame would result in

requiring a smaller number of error correcting bits to be transmitted to the decoder for the decoding of the Wyner-Ziv frames, and consequently a better RD performance. As a result, some research efforts have focused on improving the RD performance by obtaining a high-quality side information to be used in the decoder. In 2005, Ascenso *et. al* proposed a motion compensation frame interpolation (MCFI) scheme [39] to generate a side information frame that has been used frequently in DVC codecs in the literature [40]-[42]. The authors of [39] subsequently improved their side information generation framework by incorporating the concepts of using hierarchical block-size and an adaptive search range [43]. In the DISCOVER DVC codec, a side information approach similar to that in [43] was adapted. Also, in [44] and [45] symmetric optical flows between frames has been used to improve the quality of side information frame in the DVC codecs.

There are some other side information generation schemes [46]-[48] that are based on transmitting a hash information (hash codes) to the decoder in order to facilitate the decoder to generate a high-quality side information. Hash codes, in fact, contain some critical information about the original WZ frame. Aaron *et*. *al* [46] proposed sending a hash information consisting of a small subset of the quantized DCT coefficients of the blocks of a WZ frame to the decoder along with the Wyner-Ziv bits to help the decoder to generate a higher-quality side information frame. In [47], Ascenso *et. al* proposed a bidirectional hash motion estimation method, in which some quantized DCT bands are selected adaptively at the encoder, and then transmitted to the decoder to improve the motion estimation accuracy. In [48], the hash information is transmitted for the blocks in the regions of a frame where they are difficult to be interpolated reliably to generate the

side information at the decoder. A genetic algorithm is then used to exploit the hash information and select the best candidate blocks.

In the above-described techniques, the side information frame generated and used for the decoding is kept unchanged as the decoding process continues. However, as the decoding process progresses, more decoded information become available that can be used to refine the side information. Based on this observation, some machine learning methods have been proposed to improve the SI frame after decoding each bitplane or DCT band. In 2008, Varodayan *et. al* [49] proposed an unsupervised learning method to update the motion fields during the LDPC decoding. The side information is refined after the syndrome bits are received by the LDPC decoder. In [50], Martins *et. al* proposed a learning-based side information refinement algorithm, in which the side information is refined and gradually improved after the decoding each DCT band. This leads to having a more accurate side information, and consequently, a better RD performance.

1.5 Problem Statement

From the literature review carried out in the previous section, it is clear that the existing techniques for improving the rate-distortion performance of distributed video coding rely either on an accurate estimation of the correlation noise model or on the quality of the side information frame.

The drawback of some of the DVC schemes in which the RD performance is enhanced by improving the estimation of the correlation noise model is that the parameters of the model are estimated prior to the decoding and their values are not refined during the course of decoding the WZ frame, that is, the information obtained

from the decoding of the previous DCT bands of the WZ frame is not used for a more accurate estimation of the correlation noise parameters in the current DCT band. In other schemes, even though the correlation noise parameters for a DCT band are refined based on the information from the previously decoded DCT bands, these refined parameters are kept constant and not further modified during the decoding of that DCT band. In addition, in most of the DVC schemes, the non-stationary characteristic of the correlation noise within a DCT band is not adequately taken into account. Consequently, the performance of the correlation noise estimation, in general, degrades for that DCT band.

On the other hand, the DVC schemes in which the RD performance is enhanced by relying on improving the quality of the side information frame suffers from a poor-quality generation of such a frame when the motion between the decoded frames used for the generation of the side information frame is non-linear or the GOP size is large. A disadvantage of some of the other DVC schemes is that regardless of the method used to generate a good-quality side information frame before the decoding starts, this frame is kept unchanged during the decoding of all the DCT bands of the corresponding WZ frame. In the other words, the information obtained from the decoding of the DCT bands is not employed to refine the side information used for the decoding of the succeeding DCT bands. In some other DVC schemes, the side information frame is updated after decoding each DCT band. In these schemes, after modifying the DCT coefficients of a SI frame using the reconstructed DCT coefficients of the previous DCT band of the corresponding WZ frame, this SI frame is used as a reference for re-estimation of the motion vectors employed to generate the updated side information. Since the side information is generated using the average of the forward and backward motion

compensated frames, these techniques result in unreliable motion vectors for generating the updated side information.

1.6 Thesis Objective

The objective of this thesis is to develop techniques for improving the rate-distortion performance of distributed video coding by focusing on the estimation of correlation noise model and the quality of the side information frame. Within this theme, the work of this thesis is carried out in two streams.

In the first part of this thesis, a new decoder is proposed by developing a technique for the estimation and refinement of the correlation noise parameters as well as by introducing a novel procedure for the decoding of the DCT coefficients. In the proposed decoder, the process of refining the correlation noise parameters and decoding the bitplanes of a DCT band are carried out alternately in a recursive manner with a view to progressively improve the overall decoding performance. The progressive improvement in the decoding performance is achieved by devising a scheme in which the decoding result of one iteration in the form of a soft information on all the bitplanes of a DCT band is passed on to an augmented factor graph, which in turn, refines the correlation noise parameters to be used by next iteration of decoding.

In the second part of the thesis, a new algorithm is designed for a progressive improvement in the decoding of successive DCT bands by refining the side information frame based on the information on all the previously decoded DCT bands. The progressive improvement in the side information is achieved by employing all the DCT

band coefficients decoded up to that point of the decoding process in the forward and backward motion compensated frames, which are then utilized to obtain the current version of the refined side information frame.

1.7 Organization of the Thesis

The thesis is organized as follows.

In Chapter 2, some background material necessary for the development of the research problem undertaken in this thesis is presented. Channel coding and its role in the distributed coding (Wyner-Ziv coding) are explained. The LDPCA coding as an example of channel coding used in the transform-domain DVC, particularly in the DISCOVER codec, is described and its performance illustrated. Estimation of the correlation noise model parameters and generation of side information in the DISCOVER codec, and their significance in a DVC codec are explained in detail. The metrics used to evaluate the performance and efficiency of DVC codecs are briefly described.

In Chapter 3, a new decoder is proposed to carry out alternately the processes of refinement of the correlation noise parameters and decoding of a DCT band of the WZ frame in a recursive manner in order to improve the overall decoding efficiency and the RD performance of a DVC codec. A Bayesian estimator is used to estimate the correlation noise parameters by deriving the posterior distribution for each block of symbols in a DCT band, given the side information, and by using the messages received from a set of parallel LDPCA decoders. The posterior distribution is approximated to a gamma distribution by using the variational Bayes algorithm. A scheme is devised to update the messages returned to the set of LDPCA decoders by using the approximated distribution. Finally, an algorithm that incorporates these processes is presented. The

efficiency of the proposed decoder is evaluated by incorporating it in two transform-domain DVC codecs and examining their RD performance. The performance of the modified DVC codecs is compared with that of the original codecs on several video sequences with different motion characteristics.

In Chapter 4, an algorithm for carrying out a progressive refinement of the side information frame is proposed by exploiting the information on all the previously decoded DCT bands with a view to improve the RD performance in a DVC codec. A method is developed to modify the DCT coefficients in the backward and forward motion compensated frames by using the reconstructed DCT coefficients within the previously decoded DCT bands. Then, a scheme is devised to carry out the motion estimation between the updated backward motion compensated frame and the past decoded frame as well as between the updated forward motion compensated frame and the future decoded frame by using a hierarchical block search mechanism in order to find reliable motion vectors. Finally, these vectors are used to generate the refined side information for decoding the current DCT band. The efficiency of the proposed method is evaluated by employing the proposed algorithm in a DVC codec and examining its RD performance. The performance of the resulting DVC codecs is compared with that of the original codec on several video sequences. As a final study of this chapter, improvement in the rate-distortion performance of the DVC codec is investigated by incorporating the proposed algorithm for refining the side information into the decoder proposed in Chapter 3.

Finally, in Chapter 5, concluding remarks highlighting the contributions of the thesis and suggestions for some further investigation of topics related to the work of this thesis are provided.

CHAPTER 2

Background Material

2.1 Introduction

In this chapter, the background material for the work undertaken in this thesis is presented. In Section 2.2, the use of channel codes, in particular LDPC codes in distributed coding, is explained. In section 2.3, a commonly used technique for the generation of side information used in the transform-domain DVC codec is described. In Section 2.4, modeling of the correlation noise and a well-known method for calculating the parameters of the distribution of the correlation noise is presented.

2.2 Channel Codes for Distributed Coding

Most of the practical Wyner-Ziv codecs are implemented using channel codes such as Turbo codes [51] and LDPC codes [52]. The distributed video coding, also known as Wyner-Ziv video coding, is implemented using these advanced channel codes. In a typical channel coding problem, the channel noise is added to the transmitted data during transmission. Hence, an error correcting information is added to the original data at the encoder in order to protect them against the channel errors. However, in a distributed coding problem, the Wyner-Ziv encoder transmits only the error correcting information to the decoder; since the noisy version Y of the original signal X, called the side information, is already available at the decoder. The ratio of the error correcting bits sent

by the Wyner-Ziv encoder to the number of bits in the sequence X indicates the compression rate.

2.2.1 LDPC codes for distributed coding

A low-density parity-check (LDPC) code is a linear error-correcting code that was originally invented by Gallager [53] in 1963. This code had been forgotten for several decades until it was rediscovered in 1996 by Mackay [54] after the invention of the Turbo codes. Compared to other channel codes, LDPC codes are more suited for distributed source coding applications [55], [56]. The application of LDPC codes for the distributed source coding problem was first suggested by Murayama [57]. Unlike the conventional channel coding problem, in [57], a syndrome sequence is determined for encoding each source sequence by using a low-density parity-check matrix. Leveris *et. al* [58] viewed this problem differently by considering one of the two statistically dependent sources considered in [57] to be available at the decoder, but by using the same syndrome-based approach as in [57]. In view of this, a system for compression of a source sequence, as shown in Figure 2.1 was considered in [58], where the source X and the side information Y are assumed to be two correlated random variables with the latter being available at the decoder. The sequence Y is considered to be transmitted to the decoder with a rate equal

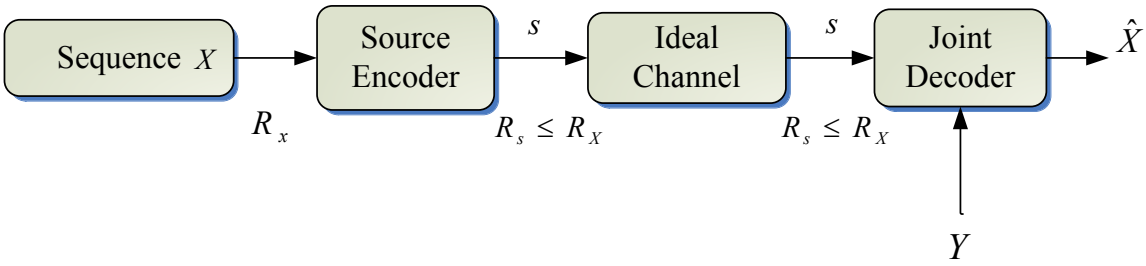


Figure 2.1 A system for compressing the source sequence X using the side information Y [58].

to the entropy of Y, $H(Y)$. Hence, according to Slepian-wolf theorem [10], the sequence X can be transmitted at a rate $R > H(X|Y)$.

In order to use the channel codes, the system in Figure 2.1 can be equivalently represented by the system shown in Figure 2.2, where the correlation between sequences X and Y is modeled as a communication channel (a virtual channel) so that the sequences X and Y are, respectively, the input and distorted output of the channel and S is a compressed version of the codeword X. If X and Y are assumed to be uniform binary codes and the correlation between X and Y is modeled by a binary symmetric channel (BSC), then S represents the syndrome bits of the corresponding binary sequence X. In view of this model, regular and irregular LDPC syndrome codes were proposed in [58] for the distributed source coding problem. It was also shown in [58] that the LDPC codes provide better results compared to that provided by the turbo codes for the Gaussian and binary symmetric correlation channels.

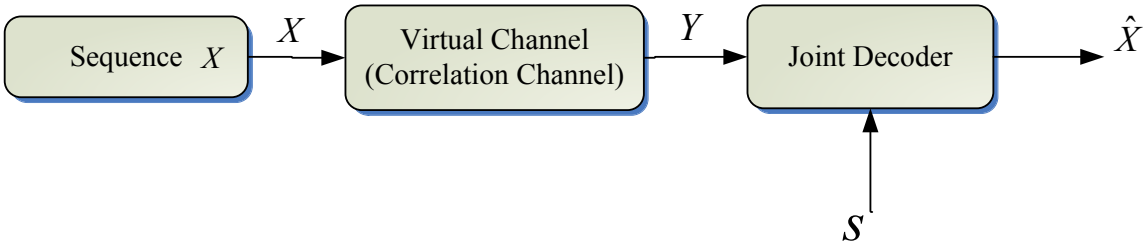


Figure 2.2. The equivalent correlation channel.

Encoding

An LDPC syndrome code is determined by a randomly-generated sparse parity-check matrix H or equivalently by the corresponding bipartite factor graph. This graph consists of a set of n variable nodes representing the n bits of a codeword X and another set of m nodes, called the check nodes, for the m syndrome bits. For encoding the n-length binary codeword X, the matrix H is multiplied with the binary codeword $X = [x_1, x_2, ..., x_n]$, to get the corresponding *m*-length syndrome S. On a factor graph, it is equivalent to a binary summation of all the values of the variable nodes connected to the same check node. An example of this encoding procedure for $n = 7$ and $m = 4$ is shown in Figure 2.3.

$$s = HX = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix}$$

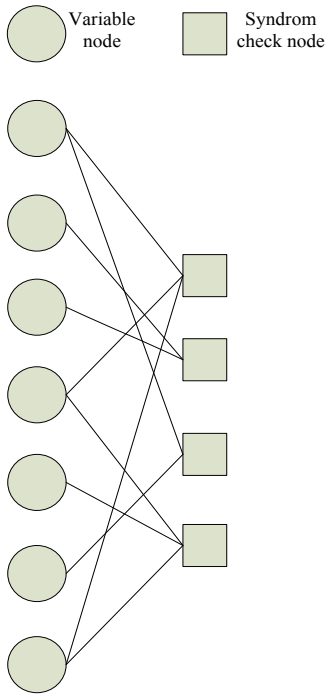


Figure 2.3 An example of encoding for LDPC syndrome codes for $n = 7$ and $m = 4$.

Decoding

On the decoder side, the *n*-length code $X = [x_1, x_2, ..., x_n]$ is decoded by an iterative message-passing decoding algorithm, called the belief propagation (BP) or sum-product algorithm, on a bipartite factor graph, given the $m-$length syndrome $s = [s_1, s_2, ..., s_m]$ and n-length sequence $Y = [y_1, y_2, ..., y_n]$. The sequence Y is considered to be encoded using a conventional encoding method and is decoded perfectly (i.e., without any error) at the decoder. Figure 2.4 shows an example of a decoding graph.

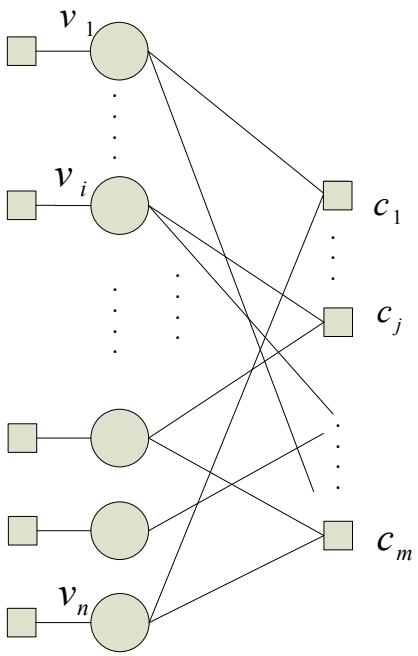
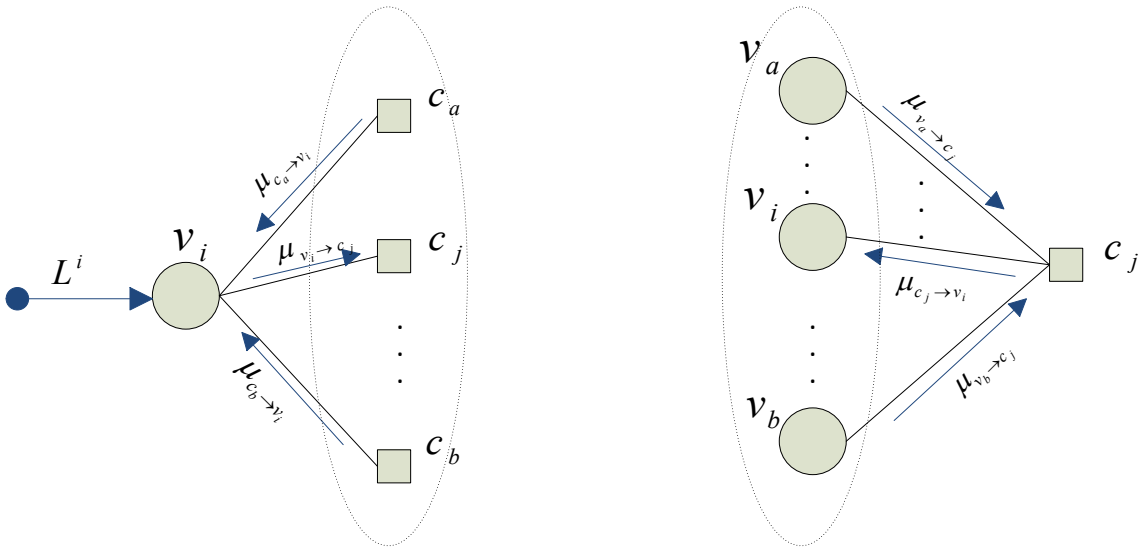


Figure 2.4 An example of a decoding graph.

Let $\mu_{v_i \to c_j}$ denote the outgoing message from the variable node v_i to the check node c_j and $\mu_{c_j \to v_i}$ denote that from the check node c_j to the variable node v_i . These

messages represent the log-likelihood ratio (LLR) values that are passed to each node by an edge of the graph. An iterative message-passing algorithm for decoding



(a) (b)

Figure 2.5 Message passing through the edges of a decoding graph.

the codeword X is carried out on the decoding graph by following the steps given below.

Step 1 - The variable node v_i, $i = 1,2,..,n$, is initialized with a soft-input information (a *LLR* value) obtained from the channel (virtual channel in the DSC case).

$$L^i = \log \frac{P(x_i \mid y_i = 1)}{P(x_i \mid y_i = 0)} \tag{2-1}$$

For example, if the correlation between X and Y are modeled as a binary symmetric channel with $p_e = P(x_i \neq y_i)$, then

$$L^i = \log \frac{P(x_i \mid y_i = 1)}{P(x_i \mid y_i = 0)} = (1 - 2y_i) \log \frac{1 - p_e}{p_e} \tag{2-2}$$

Step 2- A message $\mu_{v_i \to c_j}$ passing on the edge of the graph from the i^{th} variable node v_i to j^{th} check node c_j is obtained by the summation of all the incoming messages to the variable node v_i except the message from the check node c_j (see Figure 2.5 a) as

$$m_{v_i \to c_j} = L^i + \sum_{h \in n(v_i) \setminus \{c_j\}} \mu_{h \to v_i} \tag{2-3}$$

where $n(v_i)$ denotes the set of all the nodes connected to the variable node v_i, $n(v_i) \setminus \{c_j\}$ denotes all the nodes, except c_j, connected to node v_i and $\mu_{h \to v_i}$ represents the message from the check node $h \in n(v_i) \setminus \{c_j\}$ to the variable node v_i. It should be noted that all the $\mu_{h \to v_i}$'s have a zero value in the first iteration.

Step 3- A message $\mu_{c_j \to v_j}$ from the j^{th} check node c_j to the i^{th} variable node v_i is obtained after carrying out a processing on all the incoming messages to the check node c_j except the message from variable node v_i (see Figure 2.5 b) as

$$\tanh\left(\frac{\mu_{c_j \to v_j}}{2}\right) = (1 - 2s_j) \prod_{k \in n(c_j) \setminus \{v_i\}} \tanh\left(\frac{\mu_{k \to c_j}}{2}\right) \tag{2-4}$$

where $n(c_j)$ denotes all the variable nodes connected to the check node c_j and $n(c_j) \setminus \{v_i\}$ denotes all the nodes, except v_i, connected to the check node c_j and $\mu_{k \to c_j}$ represents the message from the variable node $k \in n(c_j) \setminus \{v_i\}$ to the check node c_j.

The message passing process given by steps 2 and 3 is carried out between each pair of a variable node and a check node that are connected by an edge of the graph until the change in the LLR value is negligible or a pre-specified number of iterations is reached.

Step 4- After the last iteration, the decoded bit $\hat{x}_i$, $i = 1,2,..,n$, is obtained as

$$\hat{x}_i = \begin{cases} 1, & if \quad L^i + \sum_{h \in n(v_i)} \mu_{h \to v_i} \geq 0 \\ 0, & otherwise \end{cases} \tag{2-5}$$

2.2.2 Rate-adaptive LDPC accumulated (LDPCA) code

The syndrome-based LDPC codes for a fixed rate distributed source coding as described in the previous subsection, if used for a rate-adaptive distributed coding, the syndrome bits need to be transmitted incrementally from the encoder to the decoder. However, in this case for achieving a high compression codes, the performance would not be satisfactory. The reason behind this is that the corresponding decoding graph would contain some unconnected or singly connected nodes. Varodayan *et. al* [59] presented a method to construct rate-adaptive LDPC codes, called the low-density parity-check accumulated (LDPCA) code, for a distributed source coding, which performs well even for a high compression rate. The LDPCA encoder consists of a syndrome-based

LDPC and an accumulator as shown in Figure 2.6. In the LDPCA encoder, the source bits $X = [x_1,\ x_2\ ,..., x_n]$ are first binary summed at the syndrome nodes to provide syndrome bits $s_1, s_2, ..., s_n$. Then, these syndrome bits are undergone a modulo 2 accumulation to generate the accumulated syndrome bits $a_1, a_2, ..., a_n$. The encoder keeps these accumulated syndrome bits in a buffer and transmits them to decoder incrementally as requested.

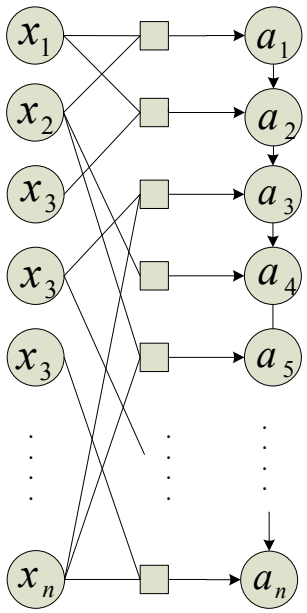


Figure 2.6 An example of LDPCA encoder.

The LDPCA decoder modifies its decoding graph after the receipt of each increment of the accumulated syndrome bits. Then, the BP algorithm is applied to this decoder graph in a way as explained in the previous subsection in order to decode the source bits $X = [x_1, x_2, ...,\ x_n]$. In order to validate the performance of LDPCA codes, a simulation is carried out to obtain the coding rate for the source sequence X with lengths of 396 and 6336 by using the regular and irregular LDPCA codes and considering an i.i.d. binary symmetric statistics between the source sequence X and the side information Y.

Figures 2.7 and 2.8 show the encoding rate points corresponding to different values of the channel error probability $p_e = P(x_i \neq y_i)$ for the sequence X with lengths 396 and 6336, respectively. The Slepian-Wolf bound for the rate of the source sequence X is also shown in these figures. It can be seen from these figures that by using LDPCA codes, we can achieve a rate R_X for encoding the sequence X that is close to the Slepian-Wolf bound $H(X \mid Y)$.

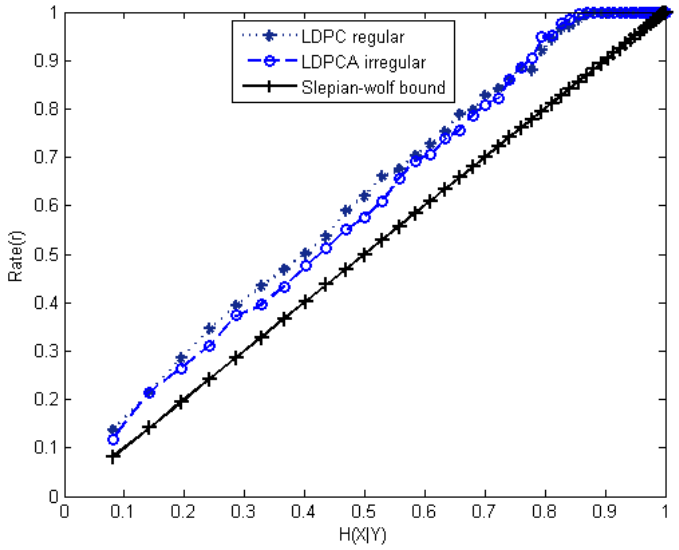


Figure 2.7 Performance of the rate-adaptive codes of length 396 with the i.i.d binary symmetric statistics.

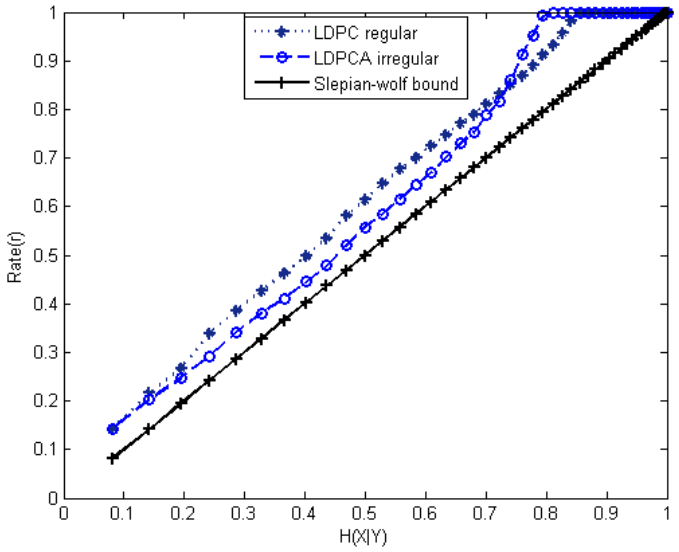


Figure 2.8 Performance of the rate-adaptive codes of length 6336 with the i.i.d binary symmetric statistics.

In most of the practical Stanford-based DVC solutions, including the well-known DISCOVER codec, the same LDPCA codes as described above are used for encoding each bitplane of a DCT band of the WZ frame. The accumulated syndrome bits are generated in the LDPCA encoder for each of the bitplanes of a DCT band and stored in a buffer. They are transmitted incrementally as per the decoder request. At the decoder, after receiving each increment of the syndrome bits, the corresponding LDPCA decoding graph is generated and then the decoding procedure using the BP message passing, as described in Section 2.2.1, is used to decode the bitplane. The soft-input information for the LDPCA decoder is obtained using the side information and the correlation noise distribution as explained in the following two sections.

2.3 Generation of Side Information

The quality of the side information frame at the decoder significantly influences the rate-distortion performance of the distributed video coding. A higher quality side information leads to a higher correlation (i.e., more similarity) between the side information at the decoder and the corresponding WZ frame at the encoder and thus results in requiring the encoder to send fewer syndrome bits to the decoder in order to decode the WZ frame. The side information is often generated at the decoder as an intermediate frame between the two previously decoded adjacent frames by employing a scheme of motion compensated interpolation between these two past and future frames. Typically, a block-matching criterion is used to find the best symmetric motion vectors for each block in the intermediate frame with respect to the previously decoded past and future frames. Then, the average of the motion compensated blocks from the two adjacent frames (past and future) is used to obtain the interpolated frame as the side information frame used for decoding the WZ frame.

Even though there are some limitations in the motion compensated interpolation (MCI) scheme especially for the video sequences with complex and nonlinear motions, this scheme still leads to a more consistent RD performance in DVC compared to that provided by other techniques used for generating a side information frame. In view of this, a side information generation technique based on MCI was adapted and integrated into the DISCOVER codec. In fact, an advanced MCI technique depicted by the block diagram given in Figure 2.9 has been used in the DISCOVER codec. In this DVC codec, by considering the past and future decoded frames, X_P and X_F (reference frames), an

interpolated frame, i.e, the side information frame for the corresponding WZ frame, is obtained following the steps of the procedure given below

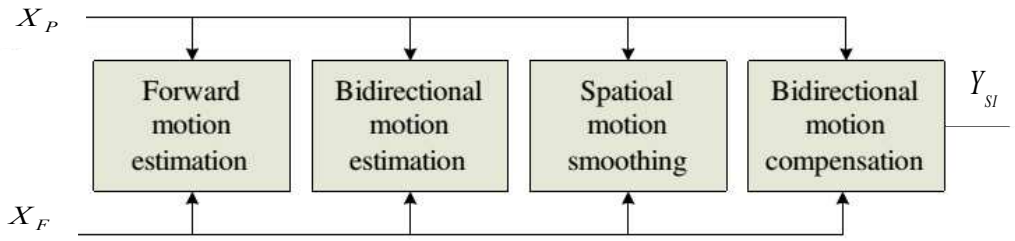


Figure 2.9 Side information generation in the DISCOVER codec [16].

1) Frames X_P and X_F are filtered by a 3x3 low-pass mean filter in order to improve the reliability of the motion vectors. A block-matching algorithm is then used for the forward motion estimation that determines the motions from the frame X_P to the frame X_F. In this algorithm, a full search motion estimation using the mean absolute difference (MAD) criterion is performed to find the motion vector field.
2) A bidirectional motion estimation is carried out in order to find a pair of symmetric motion vectors from each block in the intermediate frame to the two matched blocks in the frames X_p and X_F by employing a hierarchical coarse-to-fine approach with an adaptive search rage. At first, a large block of size (16×16) is used to deal with the fast motions. Then, a smaller block of size (8×8) is used to achieve a higher precision. Finally, a spatial motion-smoothing algorithm [17] based on weighted vector median filters is applied to obtain the smoothed pair of symmetric motion

vectors, denoted by $(\frac{v_x}{2},\frac{v_y}{2})$ and $(-\frac{v_x}{2},-\frac{v_y}{2})$, for each block in the interpolated frame, with subscripts x and y representing the pixel position in the interpolated frame. It is to be noted that the motion vector between the two matched blocks in the past and future frame is (v_x, v_y).

3) A bidirectional motion compensation is performed between X_P and X_F using the symmetric motion vectors obtained in step 2 in order to generate the interpolated frame as the side information frame Y_{SI} given by

$$Y_{SI}(x,y) = \frac{X_P\left(x-\frac{v_x}{2}, y-\frac{v_y}{2}\right) + X_F\left(x+\frac{v_x}{2}, y+\frac{v_y}{2}\right)}{2} \qquad (2\text{-}6)$$

Figures 2.9 and 2.10 show the side information frames generated using the above technique for the 9th and 41st frames (two WZ frames) of the *Foreman* sequence with a QCIF spatial resolution and 15-Hz frame rate. By considering *GOP* = 2, the side information frame for the 9th frame of the *Foreman* sequence is obtained using its adjacent decoded Key frames, namely the 8th and 10th frames. Further, the side information frame for the 41st frame (a WZ frame) is generated using its adjacent decoded Key frames, namely the 40th and 42nd frames.

(a) (b)

Figure 2.9 WZ frame (9th frame) of the *Foreman* sequence. (a) Original frame at the encoder. (b) The corresponding side information frame (*PSNR = 30.63 dB*)

(a) (b)

Figure 2.10 WZ frame (41st frame) of the *Foreman* sequence. (a) Original frame at the encoder. (b) The corresponding side information frame (*PSNR = 26.66 dB*)

To evaluate the quality of the side information frame in comparison to the corresponding WZ frame, the peak signal to noise ratio (PSNR) criterion is used. As shown in Figures 2.9 and 2.10, the *PSNR* values for the side information frames obtained for the 9th and 41st frames (WZ frames) are 30.63 dB and 26.66 dB, respectively, indicating that the quality of side information frame generated by the side information generation technique, as explained above, depends on the complexity and linearity of the motions between the reference frames.

In general, for video sequences that consist of simple and slow motions, the quality of the side information frames is, on average, higher than that of those with fast and complex motions. Thus, the DVC codec has better performance in low-motion video sequences containing relatively static scenes.

2.4 Correlation Noise Estimation

In the transform-domain distributed video coding, the correlation noise is a term used to describe the statistical residual between a DCT coefficient of the WZ frame and the DCT coefficient at the corresponding side information frame. This correlation noise is modeled in the distributed video coding scheme by a probability distribution.

As mentioned in Chapter 1, a suitable distribution to model the correlation noise should be chosen and then its parameters be estimated precisely in order to improve the RD performance in the distributed video coding. This correlation noise model is used to convert the side information into a soft input information needed for the LDPCA decoding. Hence, using a more accurate correlation noise model with precise parameters should lead to a more reliable soft-input information for the LDPCA decoder thus resulting in an improved decoding efficiency, and consequently, in a higher overall coding efficiency.

Most of the Stanford-based distributed video coding schemes, such as the one used in the DISCOVER codec, employ a Laplacian distribution to model the correlation noise. Even though there are some other accurate correlation noise models, e.g., the generalized Gaussian distribution, the Laplacian distribution provides a good trade-off between the accuracy and complexity [22].

By considering a specific DCT band X_{DCT} of a WZ frame and the corresponding DCT band Y_{DCT} of the SI frame Y_{SI}, the correlation noise $X_{DCT} - Y_{DCT}$ can be modeled by a Laplacian distribution as

$$f(X_{DCT}(u) - Y_{DCT}(u)) = \frac{\alpha}{2} e^{-\alpha(X_{DCT}(u) - Y_{DCT}(u))} \quad (2\text{-}7)$$

where, $X_{DCT}(u)$ is u^{th} transform coefficient in the DCT band X_{DCT} of the the WZ frame and $Y_{DCT}(u)$ is the corresponding transform coefficient in the DCT band Y_{DCT} of the SI frame, and α is the parameter of the distribution given as

$$\alpha = \sqrt{\frac{2}{\sigma^2}} \quad (2\text{-}8)$$

σ^2 being the variance of the residual $X_{DCT} - Y_{DCT}$ between a DCT coefficient band of the WZ frame and the corresponding DCT band of the SI frame.

Since each DCT band has different statistics and is independent of the other DCT bands, a different parameter should be considered for the correlation noise model for each DCT band. Figures 2.11 and 2.12 show the actual (empirical) histograms of the DCT band residual in two different DCT bands (the 6th and 10th DCT bands) for the 2nd frame of the *Foreman* sequences with a spatial QCIF resolution and the frame rate of 15 Hz. The plots of the Laplacian distribution with parameter value obtained using (2-8) are also shown in these figures.

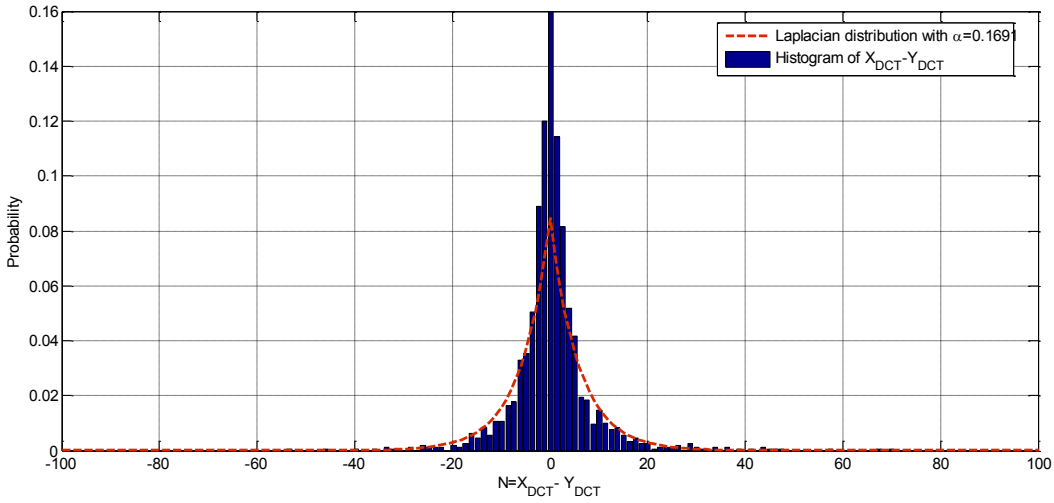


Figure 2.11. Histogram of correlation noise $X_{DCT} - Y_{DCT}$ for the 6th DCT band in the 2nd frame of the *Foreman* sequence.

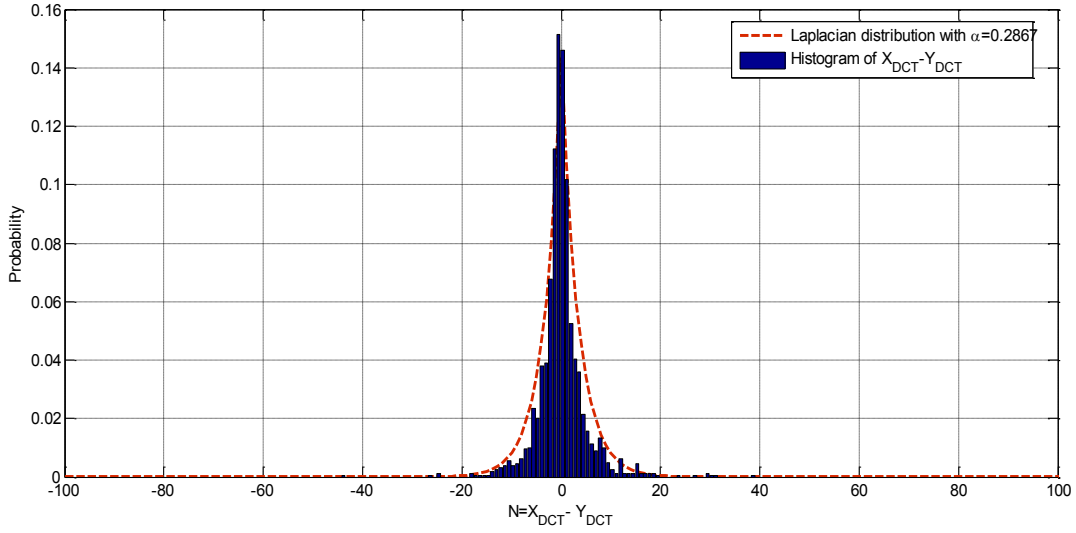


Figure 2.12. Histogram of correlation noise $X_{DCT} - Y_{DCT}$ for the 10th DCT band in the 2nd frame of the *Foreman* sequence.

The use of the actual DCT band residual ($X_{DCT} - Y_{DCT}$) between the DCT coefficient band of WZ frame and that of the corresponding side information frame to find the

correlation noise is not practical, and not used in a DVC scheme, since the WZ frame and its DCT bands are not available at the decoder. Moreover, the correlation noise is not stationary within each DCT band, and consequently, use of the same parameter for the Laplacian distribution to model the correlation noise for all the DCT coefficients in that band would actually lead to a lower performance.

In [30], Brites *et al.* proposed a solution for an online estimation of the Laplacian distribution parameter for the transform-domain Stanford-based DVC codecs in both the band and coefficient levels. In their work, the following technique has been used to estimate a different Laplacian distribution parameter for each DCT band of a frame.

1) Residual frame $\Re$ between the same forward and backward motion compensated frames as those used to generate the side information frame, is computed as

$$\Re\ (x,y) = \frac{X_P\left(x-\frac{v_x}{2}, y-\frac{v_y}{2}\right) - X_F\left(x+\frac{v_x}{2}, y+\frac{v_y}{2}\right)}{2} \tag{2-9}$$

where $\left(\frac{v_x}{2}, \frac{v_y}{2}\right)$ and $\left(-\frac{v_x}{2}, -\frac{v_y}{2}\right)$ are the same symmetric bidirectional motion vectors as those used in (2-1) to generate the side information.

2) A 4x4 DCT block is applied to the residual frame $\Re$ in order to obtain the DCT coefficient frame T.

3) The variance σ_b^2 of b^{th} DCT band of T is computed as

$$\sigma_b^2 = E\left(T_b^{\,2}\right) - \left(E\left(|T_b|\right)\right)^2 \tag{2-10}$$

where T_b is the b^{th} DCT band extracted from the DCT frame T and $|T_b|$ represents

a band with absolute of the coefficients in T_b.

4) The Laplacian parameter α_b of the b^{th} DCT band is given by

$$\alpha_b = \sqrt{\frac{2}{\sigma_b^2}} \tag{2-11}$$

5) The distance δ_{b_k} between the k^{th} coefficient of $|T_b|$ and the mean:

$$m_b = \frac{1}{N}\sum_{i=1}^{N} |T_{b_k}| \tag{2-12}$$

is computed as

$$\delta_{b_k} = |T_{b_k}| - m_b \tag{2-13}$$

6) The parameter α_{b_k} for k^{th} coefficient of the b^{th} DCT band is obtained as

$$\alpha_{b_k} = \begin{cases} \alpha_b, & if\ \delta_{b_k}{}^2 \le \alpha_b{}^2 \\ \sqrt{\dfrac{2}{\delta_{b_k}^2}}, & \text{otherwise} \end{cases} \tag{2-14}$$

A value of $\delta_{b_k}{}^2$ less than $\alpha_b{}^2$ implies that the k^{th} coefficient of the b^{th} DCT band corresponds to a well-interpolated area (high quality SI). In this case, then the parameter α_b obtained in step 4 (2-11) for the DCT band is a reliable estimation for the parameter α_{b_k} In contrast, a value of $\delta_{b_k}{}^2$ larger than $\alpha_b{}^2$, indicates that the corresponding block is

not correctly interpolated and the quality of SI obtained for that block is low. Hence, the band-level parameter α_b is not the best estimate. In this case, the distance $\delta_{b_k}^{\ 2}$ is used instead of $\sigma_b^{\ 2}$ in (2-11) to indicate less confidence in the DCT coefficients in blocks that are not interpolated correctly during the SI generation procedure (low-quality blocks). The Laplacian distribution with the parameter α_{b_k} obtained in (2-14) for each DCT band coefficient in a specific WZ frame is then used to obtain the soft input information for decoding the corresponding bitplaines on an LDPCA decoding graph.

2.5 Summary

The purpose of this chapter has been to present some background material on the main modules in the distributed video coding. The channel codes, particularly the LDPCA codes in the distributed video coding, along with the message passing algorithm on decoding graphs has been explained in detail. The importance of the side information frame and correlation noise model in the RD performance of the distributed video coding has been emphasized. Finally, a commonly used method for the generation of side information and another one for the estimation of the correlation noise parameter in the transform-domain distributed video coding have been described.

CHAPTER 3

Joint Correlation Noise Estimation and Decoding

3.1 Introduction

As mentioned earlier, the correlation noise model greatly influences the coding efficiency and the rete-distortion performance of distributed video coding. In other words, having a more accurate information about the correlation noise leads to a better decoding performance and consequently, a superior rate-distortion performance. In view of this, after a suitable choice of a distribution to model the correlation noise, the parameters of this distribution need to be estimated precisely in order to improve the RD performance in the distributed video coding. In this chapter, we have investigated the problem of obtaining the correlation noise parameter in the DVC decoder in order to improve the rate-distortion performance and coding efficiency in distributed video coding. Since, the decoder does not have access to the encoded WZ frame at the encoder and the correlation noise is non-stationary, it is difficult to model the correlation noise and to obtain its parameters accurately. In most of the online methods [29]-[34], the estimation of the correlation noise parameters is performed before the Wyner-Ziv decoder starts to decode the bitplanes of each DCT band of a WZ frame and the estimated parameters are kept constant, that is, they are not modified during the decoding of each DCT band. As the soft information on each bitplane corresponding to a DCT band is available during the decoding process at the decoder, this information could be used to refine the correlation

noise parameters during the decoding process. In view of this, a recursive algorithm and a decoder based on variational Bayes is proposed in this chapter in order to refine the correlation noise distribution parameter while decoding all the bitplanes corresponding to the current DCT band on an augmented decoding factor graph [60, 61]. In Section 3.2, using an augmented factor graph, a parallel decoding of several bitplanes as well as the Bayesian estimation of the correlation noise parameter is briefly reviewed. In Section 3.3, a message passing algorithm on the augmented factor graph is proposed. The variational Bayes method is employed to approximate the posterior distribution of the correlation noise parameter. This approximated distribution is then used to derive a closed-form expression for the messages on the augmented factor graph. In Section 3.4, the performance of the proposed algorithm is experimentally studied in the framework of a DVC codec using several video sequences.

3.2 Bayesian Estimation of Correlation Noise Parameters in Parallel LDPCA Decoders

As the correlation noise distribution in DVC is defined at the 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 [36], [49]. Therefore, parallel LDPCA decoders need to be used. As a consequence, cross correlation between the bitplanes is utilized to improve the decoding performance of DVC [36]. 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 $P(D|\varphi)$ for an observation D. The maximum likelihood estimation has been used for estimating the channel and correlation noise parameters in distributed source coding (DSC) [62] and distributed video coding (DVC) problems [63], [64]. In [62], it has been used to estimate the cross-over probability for binary symmetric channel (BSC) modeling of the channel in DSC, while for estimating the correlation noise parameter during the decoding process in DVC [63], [64]. 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 of overcoming these drawbacks is to add a prior distribution for φ, which allows to adjust and control the distribution of the probability mass function of the unobserved data relative to that of observed 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 $P(\varphi|D) = P(D|\varphi)P(\varphi)/L$, where L is a normalization factor. In maximum a posteriori (MAP) estimation, we look for the parameters φ that maximize the posterior distribution $P(\varphi|D)$. 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, wherein all the possible values for φ are considered by defining a probability distribution for φ. Hence, in this approach, the parameter estimation is equivalent to calculating the posterior distribution of φ. Also, the Bayesian estimation performs better than MLE when the sample size is small.

Suppose $Y_{DCT} = \{y_1, y_2, ..., y_N\}$, N being the number of 4x4 blocks in the side information frame, is the DCT coefficient band obtained from the side information frame constructed in a DVC decoder and $X_{DCT} = \{x_1, x_2, ..., x_N\}$ is the corresponding DCT coefficient band of the current WZ frame quantized to 2^{β} levels, where β is the number of bitplanes for the quantized coefficient band X_{DCT}. These bitplanes are jointly decoded using the LDPCA decoders. In order to take into consideration the non-stationary characteristic of the correlation noise in each DCT band in the proposed scheme, a parameter θ_j ($j = 1, 2, ..., N/M$) is assigned to each block of M symbols. As M is selected to be relatively small, the Bayesian estimation is more suitable for estimating the parameter θ_j. Considering only the j^{th} block of the DCT coefficients, the posterior distribution for the parameter θ_j given, the corresponding side information $Y_{DCT}^{j} = \{y_1^j, y_2^j, ..., y_M^j\}$ consisting of M DCT coefficients in the DCT band constructed from the current side information frame, can be written as

$$P\left(\theta_j \middle| Y_{DCT}^{j}\right) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} P(y_i^j \middle| \theta_j) \tag{3-1}$$

where L_j is a normalization factor. Replacing $P(y_i^j \middle| \theta_j)$ by $\sum_{x_i^j} P(y_i^j, x_i^j \middle| , \theta_j)$, where x_i^j is the coefficient in the DCT band of the WZ frame corresponding to y_i^j, (3-1) gets transformed to

$$P\left(\theta_j \middle| Y_{DCT}^j\right) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \sum_{x_i^j} P(y_i^j, x_i^j \middle| \theta_j)$$

$$= \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \sum_{x_i^j} P(y_i^j \middle| x_i^j, \theta_j) P(x_i^j) \tag{3-2}$$

where the summation is over all the 2^{β} values that x_i^j can assume. To find the posterior distribution, the corresponding factor graph is first obtained. In the factor graph, a message along the edge from a to b is represented by $\mu_{a \to b}$. The likelihood function $P(y_i^j \middle| x_i^j, \theta_j)$ in (3-2) 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 $\mu_{x_i \to f_i^j}(x_i^j)$ from the variable node x_i^j to the factor node f_i^j. As a consequence, the posterior distribution given by (3-2) can be rewritten as

$$P\left(\theta_j \middle| Y_{DCT}^j\right) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \sum_{x_i^j} f_i^j(y_i^j, x_i^j, \theta_j) \mu_{x_i \to f_i}(x_i^j) \tag{3-3}$$

We can identify the sum $S_i^j = \sum_{x_i^j} f_i^j(y_i^j, x_i^j, \theta_j) \mu_{x_i \to f_i^j}(x_i^j)$ 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 Figure 3.1. Therefore, the posterior distribution in (3-3) can be written as

$$P\left(\theta_j \middle| Y_{DCT}^j\right) = \frac{1}{L_j} P(\theta_j) \prod_{i=1}^{M} \mu_{f_i^j \to \theta_j}(\theta_j) \tag{3-4}$$

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

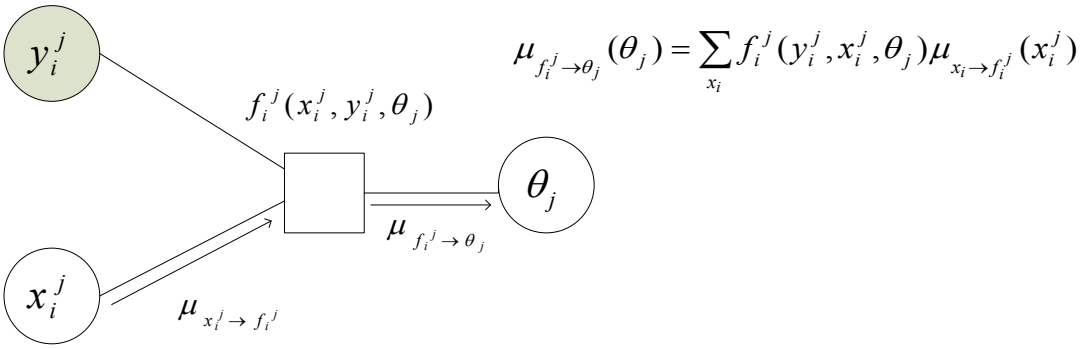


Figure 3.1 Factor graph with the output message $\mu_{f_i \to \theta_j}(\theta_j) = S_i^j$.

$$P\left(\theta_j \middle| Y_{DCT}^j\right) = \frac{1}{L_j}\mu_{g_j \to \theta_j}(\theta_j)\prod_{i=1}^{M}\mu_{f_i^j \to \theta_j}(\theta_j) \tag{3-5}$$

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

$$P\left(\theta_j \middle| Y_{DCT}^j\right) = \mu_{g_j \to \theta_j}(\theta_j)\prod_{i=1}^{M}\mu_{f_i^j \to \theta_j}(\theta_j) \tag{3-6}$$

The expression in (3-6) shows that the posterior distribution of θ_j given Y_{DCT}^j can be calculated as the product of all the M incoming messages from the factor nodes f_i, $i = 1, 2, \ldots M$, to the variable node θ_j and the message $\mu_{g_j \to \theta_j}(\theta_j)$ coming from the

factor node g_j. Hence, the posterior distribution $P\left(\theta_j \middle| Y_{DCT}^j\right)$ given by (3-6) can be represented by the factor graph shown in Figure. 3.2

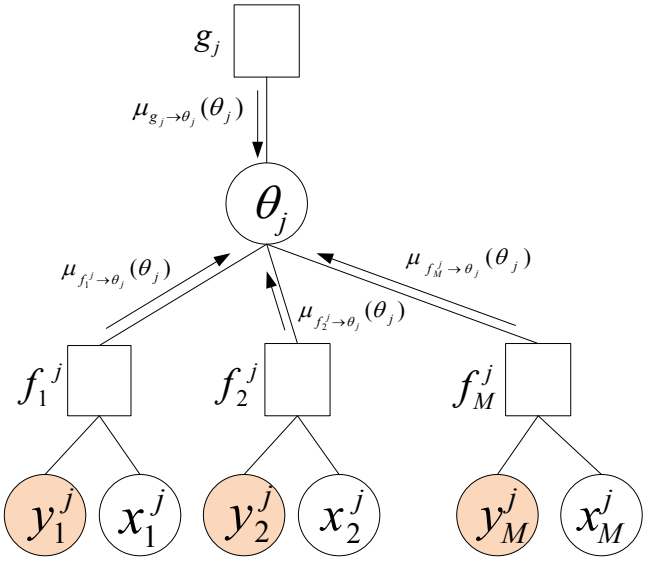


Figure 3.2 Factor graph representing the posterior distribution $P\left(\theta_j \middle| Y_{DCT}^j\right)$ given by

After using 2^β level quantizer for the DCT band X, the quantization indices of that DCT band turn into β bitplanes, $B_c = [b_{1c}, b_{2c}, ..., b_{Nc}]$, $c = 1, 2, \ldots, \beta$. A total of β LDPCA decoders are used in parallel to decode all the bitplanes. The belief propagation (BP) decoding algorithm, explained in Section 2.2, 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 B_c are augmented by the factor graph (Figure 3.2) representing $P\left(\theta_j \middle| Y_{DCT}^j\right)$ for $j = 1, 2, ..., N/M$. The augmented LDPCA decoder is obtained as shown in Figure 3.3. Rectangular blocks in this figure represent LDPCA decoder graphs each constructed for decoding a different bitpalne. The LDPC decoder graphs for each bitplane B_c consists of

N source nodes and t_c syndrome nodes corresponding to t_c accumulated syndrome bits received by the decoder, as shown in Figure 3.4. It should be noted that at the encoder, N accumulated syndrome bits are produced, as explained in Section 2.2, for each

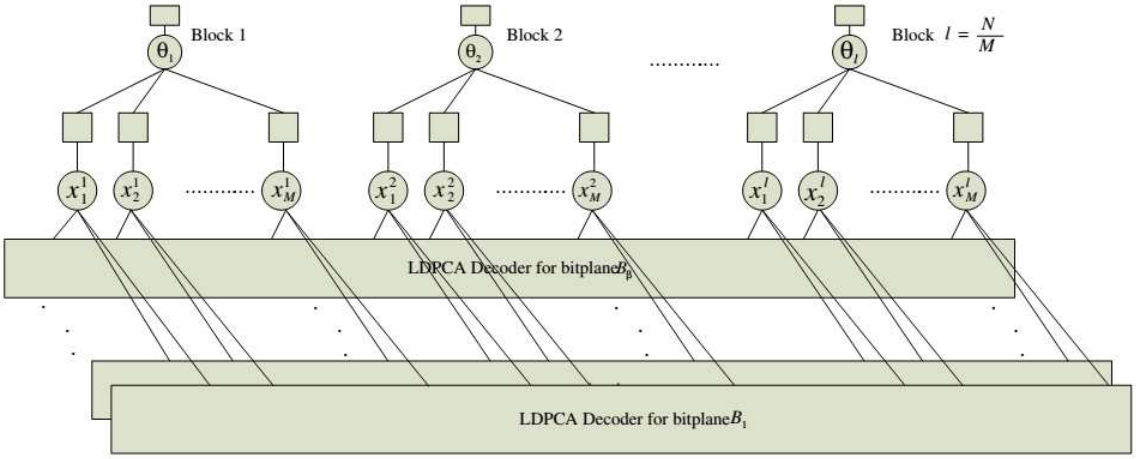


Figure 3.3 Augmented decoder factor graph.

bitplane according to the structure of LDPC encoder graph and the concatenated accumulator. These N bits are stored in a buffer and sent to the decoder incrementally at the request of the decoder. Based on the number of accumulated syndrome bits, t_c, received at the decoder for c^{th} bitplane, the LDPCA decoder graph for that bitplane gets updated. Note that $t_c \leq N$.

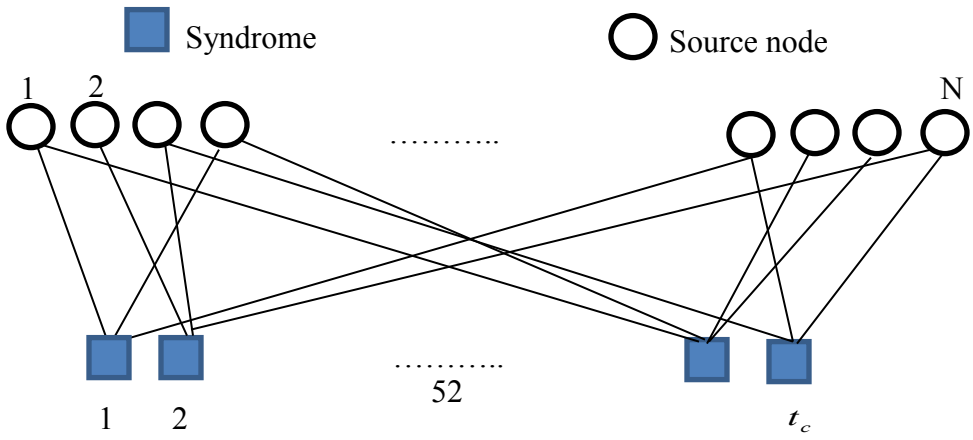


Figure. 3.4 LDPCA decoder graph for the bitplane B_c .

Details of Block j in Figure 3.3 are shown in Figure 3.5, where b_{ic}^{j}, $c = 1, 2, ..., \beta$, represents the c^{th} bit corresponding to the quantized DCT coefficient x_i^j. The message $\mu_{b_{ic}^{j} \to x_i^j}$ is calculated using LLR of the bit b_{ic}^{j} obtained using the BP algorithm in order to pass the messages back and forth between the source and syndrome nodes in the LDPCA decoder graph for the bitplane B_c using (2.3) and (2.4). Hence, the message $\mu_{x_i^j \to f_i}$ is obtained as the product of the messages $\mu_{b_{ic}^{j} \to x_i^j}$, $c = 1, 2, ..., \beta$

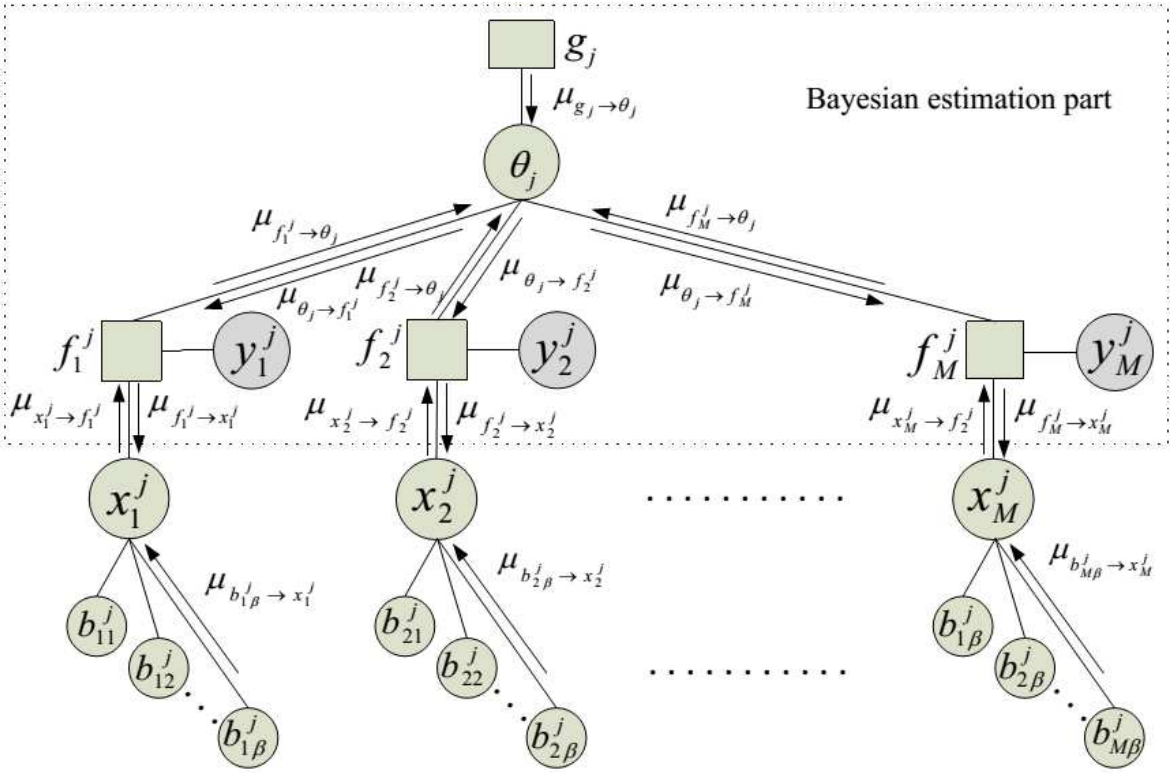


Figure 3.5 Factor graph for block j in Figure 3.3.

It is prohibitively expensive to compute the posterior distribution $P(\theta_j \mid Y_{DCT}^j)$ as given by (3-5), since it contains $2^{\beta M}$ terms resulting from the M factors on the right side of (3-5), where each factor is a sum of 2^{β} terms. Also, we need a simple and closed-form expression for the posterior distribution $P(\theta_j \mid Y_{DCT}^j)$ that can be used to derive expressions for the messages employed in the message passing algorithm. In view of these, the posterior distribution $P(\theta_j \mid Y_{DCT}^j)$ needs to be approximated by a simple distribution, such as a distribution from the exponential family.

3.3 New Decoding Algorithm Based on Variational Bayes

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

1- A module for the variational Bayes algorithm to approximate the posterior distribution

2- A module for the message updating

3- A module for carrying out parallel LDPCA decoding.

These three modules are explained next in detail followed by a complete description of the overall decoding algorithm.

3.3.1 Variational Bayes to approximate the posterior distribution

It was seen in Section 3.2 that the posterior distribution $P(\theta_j \mid Y_j)$ given by (3.5) consists of $2^{\beta M}$ terms and that it does not have a closed-form expression. Hence, the calculation of the posterior distribution is extremely expensive. Sampling methods, also known as particle methods, such as the Markov Chain Monte Carlo (MCMC) method, are frequently used for the approximation of the posterior distribution [65]. These methods are stochastic approximation methods [66], but still have high computational costs. In addition, results using any of these methods vary for each run of the algorithm. Another class of methods, known as deterministic approximation methods, are much faster than the stochastics ones. The main idea behind the deterministic methods is to find a distribution function that is as close to the true posterior distribution as possible.

Variational Bayes (VB) is a well-known deterministic method that is used to approximate the true posterior distribution [67], [68].

In a general Bayesian problem, one of the objectives is to find $P(Z \mid D)$, where Z denotes all the unknown parameters and the hidden variables, and D represents the observed variables. Since the exact calculation of $P(Z \mid D)$ is prohibitively expensive, it is necessary to find an approximation for $P(Z \mid D)$. It is known that for a given distribution $q(Z)$, the log marginal probability of D can be decomposed as [68]

$$\ln P(D) = KL(q \| p) + \ell(q) \tag{3-6}$$

where

$$KL(q \| p) = \int q(Z) \ln \left\{ \frac{q(Z)}{P(Z \mid D)} \right\} dZ \tag{3-7a}$$

$$\ell(q) = \int q(Z) \ln \left\{ \frac{P(D, Z)}{q(Z)} \right\} dZ \tag{3-7b}$$

In the above equation, the metric $KL(q \| p)$ is the Kullback-Leibler (KL) divergence that quantifies the similarity between the two distributions, $q(Z)$ and $P(Z \mid D)$, and $\ell(q)$ represents the lower bound for $\ln P(D)$. In order for $q(Z)$ to be an approximation of $P(Z \mid D)$ 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 (3-7b) is minimized. It is equivalent to maximizing the lower bound $\ell(q)$ with respect to $q(Z)$.

Suppose the elements of Z are partitioned into S disjoint subsets, z_n ($n = 1, \ldots, S$). Then, we assume that $q(Z)$ can be factorized as [68]

$$q(Z) = \prod_{n=1}^{S} q_n(z_n) \tag{3-8}$$

The objective is to find the distribution q(Z) that leads to the largest lower bound $\ell(q)$. As shown in [68], the variational optimization of $\ell(q)$ with respect to the m^{th} factor, $q_m(z_m)$, can be obtained using

$$\ln q_m(z_m) = E_{n \neq m}\left[\ln P(D, Z)\right] + C, \qquad 1 \leq m \leq S \tag{3-9}$$

where C is a constant, and $E_{n \neq m}\left[\ln P(D, Z)\right] = \int \ln P(D, Z) \prod_{n \neq m} q_n(z_n)\, dz_n$.

The above equation represents the condition for maximizing the lower bound $\ell(q)$, or equivalently, minimizing the KL divergence with respect to the m^{th} factor, $q_m(z_m)$. Solving (3-9) for $q_m(z_m)$, $m = 1, \ldots, S$, leads to a distribution $q(Z)$ that is an approximation of the posterior distribution $P(Z \mid D)$.

The above method is used in our proposed scheme to approximate the posterior distribution $P(\theta_j \mid Y_j)$ derived in Section 3.2, and consequently, the message structure on the augmented LDPC decoder in the j^{th} block illustrated in Figure 3.5 gets simplified. In order to use the variational Bayes method, a set of hidden variables $H_j = \{h_1^j, h_2^j, \ldots h_M^j\}$, where each h_i^j, $i = 1, 2, \ldots, M$, is a K-length vector ($K = 2^\beta$), is

employed. Let Z denotes the set $\{H_j, \theta_j\}$, where θ_j is the unknown parameter of the correlation noise model that needs to be estimated. The variational factorization given by (3-8) can now be performed, for $S=2$ by letting $z_1 = H_j$ and $z_2 = \theta_j$ as

$$q(Z) = q(H_j, \theta_j) = q_1(z_1) q_2(z_2) = q_1(H_j)\, q_2(\theta_j) \tag{3-10}$$

where $q_2(\theta_j)$ is the variational approximation for $P(\theta_j \mid Y_{DCT}^j)$. After the factorization, the optimization process using (3-9) is carried out for both the factors by considering the observed variable D to be the side information Y_{DCT}^j in our problem. Hence, in this case, the VB algorithm (3-9) can be represented by the following two equations:

$$\ln q_1(H_j) = E_{\theta_j}[\ln P(Y_{DCT}^j, H_j, \theta_j)] + C_1 \tag{3-11a}$$

$$\ln\ q_2(\theta_j) = E_{H_j}[\ln P(Y_{DCT}^j,\ H_j, \theta_j)] + C_2 \tag{3-11b}$$

where the joint distribution $P(Y_{DCT}^j, H_j, \theta_j)$ in (3-11) can be written as

$$P(Y_{DCT}^j, H_j, \theta_j) = P(\theta_j) P(Y_{DCT}^j \mid H_j, \theta_j) P(H_j) = P(\theta_j) \prod_{i=1}^{M} P(y_i^j \mid h_i^j, \theta_j) P(h_i^j) \tag{3-12}$$

To determine $q_1(\theta_j)$ and $q_1(H_j)$ from (3-11), we first need to find an expression for $P(Y_{DCT}^j, H_j, \theta_j)$ in (3-12).

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^j s, are not available, we use a partially decoded coefficient obtained by minimum mean square error (MMSE) reconstruction $w_{ik}^j = E[x_i^j \mid y_i^j, I_k, \lambda_j]$ where $k = 1,2,...,K$, I_k is k^{th} quantization interval and λ_j is the initial value of the correlation noise distribution parameter. In view of this, for each side information DCT coefficient y_i^j extracted in the decoder, a hidden variable vector $\boldsymbol{h}_i^j$ 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 zeros. For each observation y_i^j, the position of 1 in each vector $\boldsymbol{h}_i^j$ 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 the vector $\boldsymbol{h}_i^j$ is 1, i.e., $h_i^j = [\overbrace{0,0,..,0}^{d-1}, 1, \overbrace{0,0,....,0}^{K-d}]$. By considering this feature for the hidden variable h_i^j, $P(y_i^j \mid h_i^j, \theta_j)$ and $P(h_i^j)$ can be written as $P(y_i^j \mid h_i^j, \theta_j) = \prod_{k=1}^{K} P(y_i^j \mid x_i^j = w_{ik}^j, \theta_j)^{h_{ik}^j}$ and $P(h_i^j) = \prod_{k=1}^{K} (\mu_{x_i^j \to f_i}(x_i^j = w_{ik}^j))^{h_{ik}^j}$, respectively.

Using the expressions for $P(y_i^j \mid h_i^j, \theta_j)$ and $P(h_i^j)$, (3-11a) can be rewritten as

$$\begin{aligned}\ln q_1(H_j) &= E_{\theta_j}\left[\ln P(Y_j|H_j,\theta_j)+\ln P(H_j)\right]+C_1 \\ &= E_{\theta_j}\left[\ln\prod_{i=1}^{M}\prod_{k=1}^{K}\left(P(y_i^j|x_i^j=w_{ik}^j,\theta_j)\right)^{h_{ik}^j}+\ln\coprod_{i=1}^{M}\prod_{k=1}^{K}\left(\mu_{x_i^j\to f_i^j}(x_i^j=w_{ik}^j)\right)^{h_{ik}^j}\right]+C_1 \\ &= E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K}h_{ik}^j\left\{\ln\left(P(y_i^j|x_i^j=w_{ik}^j,\theta_j)\right)+\ln(\mu_{x_i^j\to f_i}(x_i^j=w_{ik}^j))\right\}\right]+C_1\end{aligned}$$

(3-13a)

Similarly, (3-11b) can also be rewritten as

$$\begin{aligned}\ln q_2(\theta_j) &= E_{H_j}\left[\ln P(Y_j|H_j,\theta_j)+\ln P(\theta_j)\right]+C_2 \\ &= E_{H_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K}h_{ik}^j\left\{\ln\left(P(y_i^j|x_i^j=w_{ik}^j,\theta_j)\right)\right\}\right]+\ln p(\theta_j)+C_2\end{aligned}$$

(3-13b)

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 or Laplacian distribution. In the following subsections, we consider the Gaussian and Laplacian distributions for the correlation noise model to solve (3-13a) and (3-13b) simultaneously in order to find $q_2(\theta_j)$ as an approximation to $p(\theta_j \mid Y_{DCT}^j)$.

A) Gaussian distribution for the correlation noise model

Assuming a Gaussian distribution for the correlation noise, we can express the probability $P(y_i^j|x_i^j=w_{ik}^j,\theta_j)$ in (3-13) as

$$P(y_i^j\,|\,x_i^j=w_{ik}^j,\theta_j)=\frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}}e^{-\frac{(y_i^j-w_{ik}^j)^2}{2}\theta_j}\qquad k=1,2,3,...,K \tag{3-14}$$

Substituting (3-14) in (3-13a) and after some simplification, it can be shown that

$$\ln q_1(H_j) = E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{\ln\left(\frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}} e^{-\frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j}\right) + \ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))\right\}\right] + C_1$$

$$= E_{\theta_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{\frac{1}{2}\ln\theta_j + \frac{1}{2}\ln\frac{1}{2\pi} - \frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j + \ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j))\right\}\right] + C_1$$

$$= \sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \ln\rho_{ik} + C_1 \tag{3-15}$$

where

$$\ln\rho_{ik} = \frac{1}{2}E_{\theta_j}[\ln\theta_j] - \frac{1}{2}\ln 2\pi - E_{\theta_j}\left[\frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j\right] + \ln(\mu_{x_i^j \to f_i^j}(x_i^j = w_{ik}^j)) \tag{3-16a}$$

Let the normalized value of ρ_{ik} denoted by r_{ik} be expressed as

$$r_{ik} = \rho_{ik} \Big/ \sum_{k=1}^{K}\rho_{ik} \tag{3-16b}$$

From (3-15), it can be concluded that

$$q_1(H_j) = \prod_{i=1}^{M}\prod_{k=1}^{K} r_{ik}^{h_{ik}^j} \tag{3-17}$$

Also, the update equation (3-13b) for $q_2(\theta_j)$ can be obtained as

$$\ln q_2(\theta_j) = E_{H_j}\left[\sum_{i=1}^{M}\sum_{k=1}^{K} h_{ik}^j \left\{\ln\left(P(y_i^j \middle| x_i^j = w_{ik}^j, \theta_j)\right)\right\}\right] + \ln p(\theta_j) + C_2$$

$$= \sum_{i=1}^{M}\sum_{k=1}^{K}\frac{1}{2}E_{h_{ik}^j}[h_{ik}^j]\ln\theta_j - \sum_{i=1}^{M}\sum_{k=1}^{K}E_{h_{ik}^j}[h_{ik}^j]\frac{(y_i^j - w_{ik}^j)^2}{2}\theta_j + \ln P(\theta_j) + C_2 \tag{3-18}$$

Since $E_{h_{ik}^j}[h_{ik}^j] = p(h_{ik}^j = 1) = r_{ik}$, (3-18) can be rewritten as

$$\ln q_2(\theta_j) = \sum_{i=1}^{M}\sum_{k=1}^{K}\frac{1}{2} r_{ik} \ln \theta_j - \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} \frac{(y_i^j - w_k)^2}{2}\theta_j + \ln P(\theta_j) + C_2 \quad (3\text{-}19)$$

If the prior distribution $P(\theta_j)$ is considered to be a gamma distribution, that is

$$P(\theta_j) = Gama\,(\theta_j \mid a_0, b_0) = \frac{1}{\Gamma(a_0)} {b_0}^{a_0} \theta_j^{a_0 - 1} e^{-b_0 \theta_j} \quad (3\text{-}20)$$

where a_0 and b_0 are the parameters of this distribution. Taking natural logarithm of both sides of the equation, we have

$$\ln P(\theta_j) = (a_0 - 1)\ln \theta_j - b_0 \theta_j + \alpha \quad (3\text{-}21)$$

where $\alpha = \ln \frac{1}{\Gamma(a_0)} {b_0}^{a_0}$ is a constant. By substituting (3-21) in (3-19), $\ln q_2(\theta_j)$ can be simplified as

$$\ln q_2(\theta_j) = \left(\frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 - 1 \right) \ln \theta_j - \left(\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} \frac{(y_i^j - w_{ik}^j)^2}{2} + b_0 \right) \theta_j + C_2' \quad (3\text{-}22)$$

By comparing (3-22) with (3-21), it is obvious that the variational approximation of the true posterior distribution, $q_2(\theta_j)$, would be in the form of a gamma distribution with parameters a and b given as

$$\begin{aligned} a &= \frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 = \frac{1}{2} M + a_0 \\ b &= \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} \frac{(y_i^j - w_{ik}^j)^2}{2} + b_0 \end{aligned} \quad (3\text{-}23)$$

Then, by using the gamma distribution with the parameters a and b obtained as above, ρ_{ik} can be calculated from (3-16a). Consequently, after normalizing ρ_{ik} using (3-16b), r_{ik} can be obtained as

$$r_{ik} = \frac{\mu_{x_i^j \to f_i^j}(w_{ik}^j) e^{\left(-\frac{(y_i^j - w_{ik}^j)^2}{2}\frac{a}{b}\right)}}{\sum_{k=1}^{K} \mu_{x_i^j \to f_i^j}(w_{ik}^j) e^{\left(-\frac{(y_i^j - w_{ik}^j)^2}{2}\frac{a}{b}\right)}} \qquad (3\text{-}24)$$

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 (3-23) to find the new value for b. The new parameters for gamma distribution is now used in (3-24) 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 can be considered as an approximation for the posterior distribution.

B) Laplacian distribution for the correlation noise model

Assuming a Laplacian distribution for the correlation noise, we can express the probability $P(y_i^j \left| x_i^j = w_k, \theta_j \right.)$ in (3-13) as

$$P(y_i^j \left| \, x_i^j = w_k, \theta_j \right.) = \frac{\theta_j}{2} e^{-\left| y_i^j - w_k \right| \theta_j} \qquad k = 1,2,3,\ldots,K \qquad (3\text{-}25)$$

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 by

$$a = \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik} + a_0 = M + a_0$$

$$b = \sum_{i=1}^{M}\sum_{k=1}^{K} r_{ik}\left|y_i^j - w_{ik}^j\right| + b_0 \tag{3-26}$$

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

$$r_{ik} = \frac{\mu_{x_i^j \to f_i^j}(w_{ik}^j) e^{-\left|y_i^j - w_{ik}^j\right|\frac{a}{b}}}{\sum_{k=1}^{K} \mu_{x_i^j \to f_i^j}(w_{ik}^j) e^{-\left|y_i^j - w_{ik}^j\right|\frac{a}{b}}} \tag{3-27}$$

Just as in the case of VB with the Gaussian distribution, the values for r_{ik} and the parameters of the gamma distribution, a and b, are also obtained iteratively until there is almost no change in the value of b. The gamma distribution with the parameters a and b so obtained, can be considered as an approximation for the posterior distribution.

3.3.2 Message update

After obtaining the approximation for the posterior distribution $P(\theta_j \left| Y_{DCT}^j \right.)$, the message $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$, representing 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 Figure 3.5. If the correlation noise is Gaussian, then the message $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$ can be calculated as

$$\mu_{f_i^j \to x_i^j}(w_{ik}^j) = \int_{\theta_j} \frac{1}{\Gamma(a)} b^{\,a} \theta_j^{\,a-1} e^{-b\,\theta_j} \frac{\theta_j^{\frac{1}{2}}}{\sqrt{2\pi}} e^{-\frac{\theta_j}{2}(y_i^j - w_{ik}^j)^2} d\theta_j$$
$$= \int_0^\infty \frac{1}{\Gamma(a)\sqrt{2\pi}} b^a \theta_j^{\,a-\frac{1}{2}} e^{-\theta_j\left[\frac{(y_i^j - w_{ik}^j)^2}{2} + b\right]} d\theta_j \tag{3-28}$$

Then, after performing the integration in this equation and some mathematical simplification, $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$ can be expressed in closed form as

$$\mu_{f_i^j \to x_i^j}(w_k) = \frac{\Gamma\left(a + \frac{1}{2}\right)}{\Gamma(a)\sqrt{2\pi}} b^a \left[b + \frac{(y_i^j - w_{ik}^j)^2}{2}\right]^{-\left(a+\frac{1}{2}\right)} \tag{3-29}$$

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

$$\mu_{f_i^j \to x_i^j}(w_{ik}^j) = \int_{\theta_j} \frac{1}{\Gamma(a)} b^{\,a} \theta_j^{\,a-1} e^{-b\,\theta_j} \frac{\theta_j}{2} e^{-\left|y_i^j - w_{ik}^j\right|\theta_j} d\theta_j$$
$$= \int_0^\infty \frac{1}{2\Gamma(a)} b^a \theta_j^{\,a} e^{-\theta_j\left[\left|y_i^j - w_{ik}^j\right| + b\right]} d\theta_j \tag{3-30}$$

which after carrying out the required integral operation, can be written as

$$\mu_{f_i^j \to x_i^j}(w_{ik}^j) = \frac{1}{2} b^a \left[b + \left|y_i^j - w_{ik}^j\right|\right]^{-(a+1)} \tag{3-31}$$

The updated messages from each of the blocks are then returned into LDPCA decoders for the bitplanes $B_1, B_2 \ldots . B_\beta$ (see Figure 3.3) to start decoding with more accurate soft information. Hence, all the decoders have new and more accurate knowledge about the correlation noise parameter, leading to a more efficient decoding after applying regular belief propagation algorithm in the LDPCA decoder.

3.3.3 Parallel LDPCA decoding process

To decode the bitplanes $B_1, B_2, \ldots, B_\beta$ (see Figure 3.3) using the BP algorithm in the LDPCA decoders, the log-likelihood ratio (LLR) for each bit b_{ic}^j in the bitplanes $B_1, B_2, \ldots, B_\beta$ needs to be obtained. First, the messages $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$ as obtained in Section 3.3.2 are used to calculate the message $\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j)$ from node x_i^j to the corresponding bit nodes b_{ic}^j using the procedure given in [49]. 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

$$L_{ic}^j = \log \frac{\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j = 1)}{\mu_{x_i^j \to b_{ic}^j}(b_{ic}^j = 0)} \tag{3-32}$$

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 $l_{ic}^j = L_{ic}^j + \sum_{v=1}^{V_i} l_{ic}^{j,v}$, where L_{ic}^j is calculated using (3-32), $l_{ic}^{j,v}$ is the LLR value received through the v^{th} edge ($v = 1, 2, \ldots, V_i$) from the

syndrome node to the node b_{ic}^{j} after a pre-specified 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 $l_{ic}^{j} \geq 0$ and as zero otherwise. Next, the LDPCA syndrome and 8-bit cyclic redundancy check (CRC) summation are used in the decoder to determine whether or not the LDPCA decoding has been successful [17].

3.3.4 Complete recursive message passing algorithm

Figure 3.6 shows the proposed decoder consisting of the three modules explained in Sections 3.3.1, 3.3.2 and 3.3.3. The arrows in this figure indicate the interactions amongst the three modules. We call this new decoder as the correlation noise parameter refinements (CNPR) based decoder.

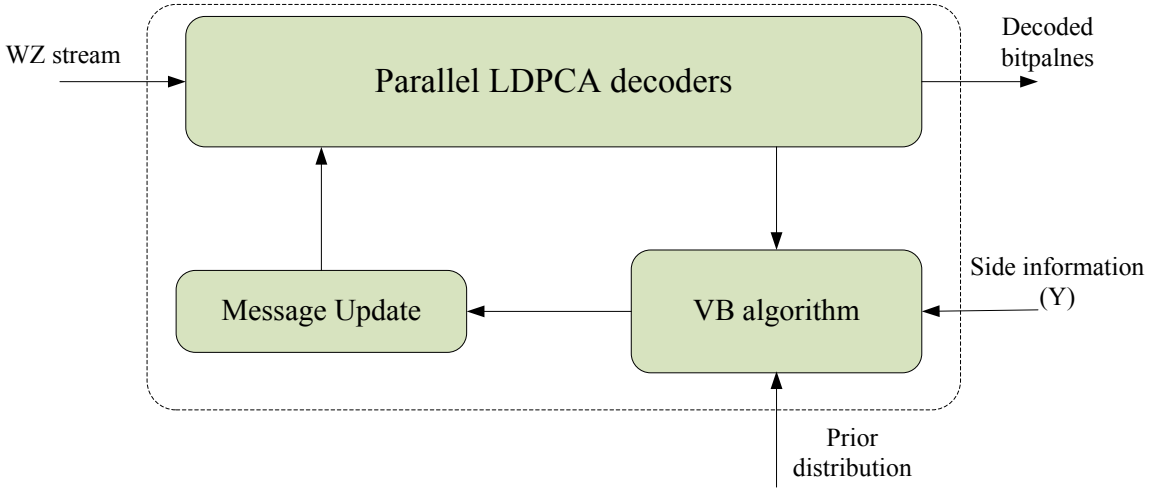


Figure 3.6 Proposed CNPR decoder.

The recu

Step 1- The messages $\mu_{x_i^j \to f_i^j}(w_k)$ ($j = 1,2,..., \frac{N}{M}$; $i = 1,2,..., M$) in Figure 3.5 are first calculated using the messages $\mu_{b_{ic}^j \to x_i^j}(b_{ic}^j)$ received by node x_i^j from the bit nodes, b_{ic}^j , $c = 1, 2,\ldots,\beta$, so that $\mu_{x_i^j \to f_i^j}(w_{ik}^j) = \prod_{c=1}^{\beta} \mu_{b_{ic}^j \to x_i^j}(b_{ic}^j)$.

Step 2- Using the messages, $\mu_{x_i^j \to f_i^j}(w_{ik}^j)$ 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 3.3.1.

Step 3- The approximated posterior distribution for each correlation noise parameter θ_j is used to calculate the messages $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$ from the factor nodes f_i^j to the variable nodes x_i^j, as explained in Section 3.3.2.

Step 4- The messages $\mu_{f_i^j \to x_i^j}(w_{ik}^j)$ obtained in Step 3 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 , $c = 1, 2,\ldots,\beta$. The initial LLRs L_{ic}^j are then calculated using (3-32) and employed in the LDPCA decoders to decode all the bitplanes, as explained in Section 3.3.3.

Step 5- By using the LDPCA syndrome bits and the 8-bit CRC summations as mentioned in Section 3.3.3, we check whether all the bitplanes have been decoded correctly.

Step 6- If all the bitplanes are decoded correctly or a pre-specified maximum number of iterations is reached, no further iterations of algorithm is carried out; otherwise Steps 1-6 are repeated.

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 and modifies its factor graph, as is generally done in other DVC schemes. Then, the proposed recursive message-passing algorithm is applied again. This process of requesting additional bits and applying the proposed algorithm is repeated until each LDPCA decoder successfully decodes its corresponding bitplane.

The LDPCA decoder and the correlation noise estimation blocks in the DISCOVER codec shown in Figure 1.6 are now replaced by the proposed CNPR decoder shown in Figure 3.6, and the resulting modified architecture for the transform-domain distributed video codec is shown in Figure 3.7. The modified parts of this architecture in this figure are shown in a dashed rectangular box.

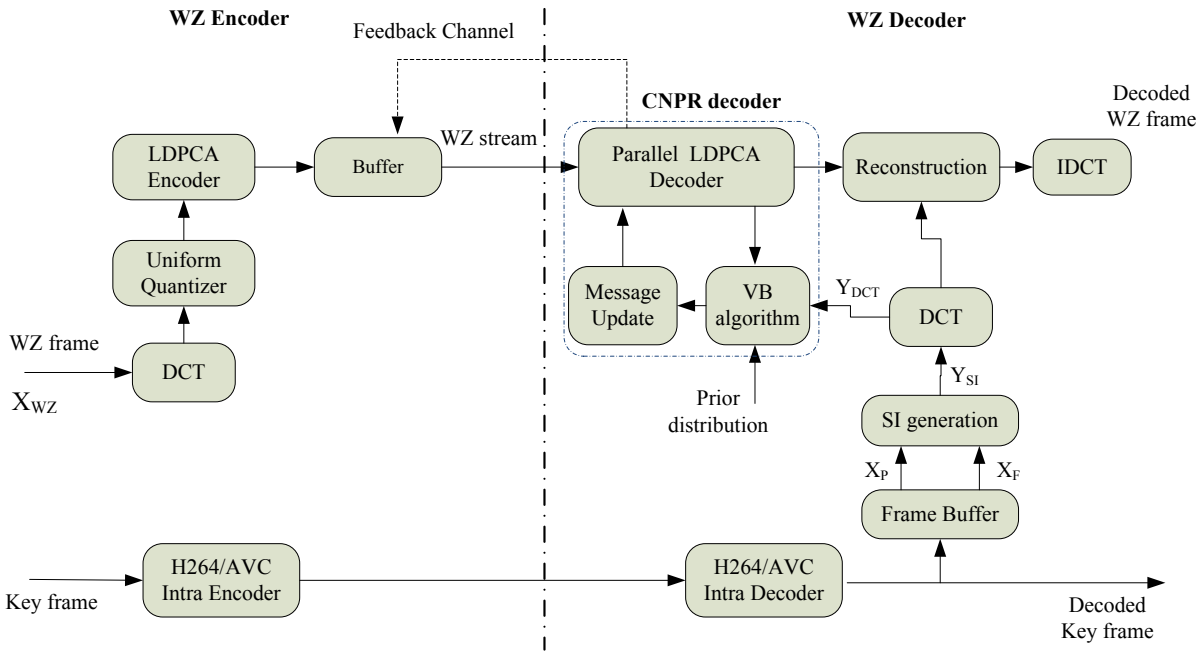


Figure 3.7. Modified architecture for the DVC codec

3.4 Simulation Results

In this section, we study through extensive experimentation the rate-distortion performance of the modified codec of Figure 3.7 using the proposed CNPR decoder shown in Figure 3.6 and compare the results with that of the two original DVC codecs, namely DISCOVER codec [17] and DVC codec with side information refinement in [50]. Both these latter codecs use the online correlation noise estimation method presented in [30]. For the simulations, the *Foreman*, *Coastguard*, *Hall*, and *Soccer* video sequences, each having 150 frames in QCIF format with *15-Hz* frame rate, are employed. Figure 3.8 shows a typical frame from each of these sequences. The key frames are encoded using the intra coding mode of the H.264 codec (JM 9.5) [69]. Eight RD points corresponding to the eight quantization matrices, $Q_1, Q_2, \ldots, Q_8$, that are the same as the ones used in the DISCOVER codec [17] and illustrated in Figure 1.8, are considered. The QP values in

Figure 3.8 A typical frame from each of the *Foreman*, *Hall*, *Coastguard* and *soccer* sequences.

H.264/AVC (in intra mode) are set to be the same as that used for the key frames in the DISCOVER codec [17]. Also, only the luminance component (Y) of the video frames is considered in our simulation for evaluating the rate-distortion performance. The Laplacain distribution is used to model the correlation noise in each block of $M = 99$ DCT coefficients in the corresponding DCT band. Then, the proposed CNPR decoder 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 CNPR 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 3.1 gives, for the DVC codec modified by the proposed CNPR decoder, the relative average savings (in %) in the bitrate and improvement in PSNR (in dB) (computed using Bjøntegaard metric [70]) over that of the DISCOVER codec corresponding to the WZ frame as well as for all the frames. With the GOP size of 2, that is, one WZ frame in between two successive key frames, it can be seen that for the *Foreman* sequence the proposed CNPR decoder results in average bitrate savings of 5.53% and 11.45% for all the frames and for only the WZ frames, respectively. The corresponding savings are 3.21% and 7.11% for the *Hall* sequence, 4.79% and 9.74% for the *Coastguard* sequence, and 8.23% and 15.71% for the *soccer* sequence. As for the PSNR, the proposed CNPR decoder results in average improvements of 0.31 dB, 0.29 dB, 0.27 dB and 0.58 dB for the *Foreman, Hall , Coastguard* and *Soccer* sequences,

Table 3.1- The relative bitrate savings (%) and improvement in PSNR (dB) over that of DISCOVER codec, computed using the BJØNTEGAARD metric

	GOP=2				GOP=4			
	WZ frames		All frames		WZ frames		All frames	
	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (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	7.11	0.29	3.21	0.14	7.26	0. 35	5.26	0.22
Soccer	15.71	0.58	8.23	0.26	12.13	0.41	9.53	0.25
Average	10.79	0.36	5.19	0.17	9.91	0.34	7.46	0.22

respectively, in the WZ frames. Further, there are improvements of 0.16 dB, 0.14 dB, 0.12 dB and 0.26 dB in the *Foreman, Hall, Coastguard* and *soccer* sequences, respectively, for all the frames. Hence, with the GOP size of 2, we observe that the codec

modified by the proposed decoder on average over all the sequences, leads to 10.79% savings in the bitrate for the sequence WZ frames and 5.19% savings in the bitrate for all the sequence frames.

Moreover, on average over all the sequences, the improvements in the PSNR values are 0.36 dB and 0.17dB in the WZ frames and all the frames, respectively.

For the GOP size of 4, that is, 3 WZ frames in between two successive key frames, it can also be seen from Table 3.1 that the proposed decoder leads to an average savings in bitrates of 8.41% and 10.68%, respectively for all the frames and the WZ frames of the *Foreman* sequence. The corresponding savings are 5.26% and 7.26% for the *Hall* sequence, 6.67% and 9.56% for the *Coastguard* sequence, and 9.53% and 12.13 % for the *soccer* sequence. As for the PSNR, the DVC codec modified with the proposed decoder shows average improvements of 0.33 dB, 0.35 dB, 0.26 dB and 0.41 dB for the WZ frames of the *Foreman, Hall, Coastguard* and *soccer* sequences, respectively. Further, there are improvements of 0.24 dB, 0.22 dB, 0.19 dB, and 0.25 dB in the *Foreman, Hall, Coastguard* and *soccer* sequences, respectively, for all the frames. Hence, for the GOP size of 4, on average over all the sequences, we observe that our proposed CNPR decoder results in 9.91% savings in the bitrate for the WZ frames and 7.46% savings in the bitrate for all the frames. Moreover, on average over all the sequences, the improvements in PSNR values are 0.34 dB and 0.22 dB for the WZ frames and all the frames, respectively.

The proposed CNPR decoder can also be used in other transform-domain distributed video coding schemes that have the same architecture as the DISCOVER codec, namely, those based on the Stanford approach. For instance, if the proposed CNPR decoder is employed in the DVC codec with the side information refinement in [50], then using

Bjøntegaard metric, the relative savings in the bitrate (in %) and the improvements in PSNR (in dB) are obtained as shown in Table 3.2. It can be seen from this table that for the GOP size of 2, on average over all the video sequences, namely, *Foreman*, *Coastguard*, *Hall*, and *Soccer*, the DVC codec with the CNPR decoder results in savings of 8.79% and 4.11% in the bitrates for the WZ frames and all the frames, respectively. Moreover, on average over these sequences, there are improvements of 0.19 dB and 0.10 dB in PSNR for the WZ frames and all the frames, respectively. On the other hand, for the GOP size of 4, on average over all the sequences, the DVC codec modified by the CNPR decoder results in savings of 7.51% and 5.63% in the bitrates for the WZ frames and all the frames, respectively; whereas, there are improvements of 0.17 dB and 0.11 dB in PSNR for the WZ frames and all the frames, respectively.

Table 3.2- The relative bitrate savings (%) and improvement in PSNR(dB) over that of DVC codec in [50] codec, computed using the BJØNTEGAARD metric

	GOP size 2				GOP size 4			
	WZ frames		All frames		WZ frames		All frames	
	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)
Foreman	9.31	0.18	4.69	0.16	8.18	0.21	6.29	0.16
Coastguard	7.46	0.15	3.19	0.12	6.11	0.13	4.93	0.06
Hall	5.69	0.14	2.27	0.14	4.9	0. 09	3.01	0.04
Soccer	12.71	0.32	6.29	0.26	10.86	0.26	8.31	0.19
Average	8.79	0.19	4.11	0.12	7.51	0.17	5.63	0.11

A rate-distortion (RD) curve obtained by fitting the RD points corresponding to the various quantization matrices could be a good illustration to indicate the rate-distortion performance of a DVC codec. In view of this, we obtain rate-distortion curves fitting the RD points corresponding to the eight quantization matrices shown in Figure 1.8 in order

to demonstrate the improvements in the RD performance in the DVC codecs modified by the proposed CNPR decoder over that of the original DVC codecs. Figures 3.9 to 3.12 show the overall RD performance for the *Foreman*, *Coastguard*, *Hall* and *soccer* sequences, respectively, using the two original DVC codecs and the corresponding codecs modified by the proposed decoder for the GOP size of 2. Figures 3.13 to 3.16 show the corresponding results for the GOP size of 4. It can be seen from these 8 figures that the DVC codecs modified by the proposed decoder exhibit an RD performance superior to that of the original DVC codecs. It is also seen that higher the bit rates, more the improvement in the RD performance. Further, it is noted that the improvements in the RD performance is even more pronounced in the case of the *Foreman* and *soccer* sequences in which there are faster and more complex motions. Even though the

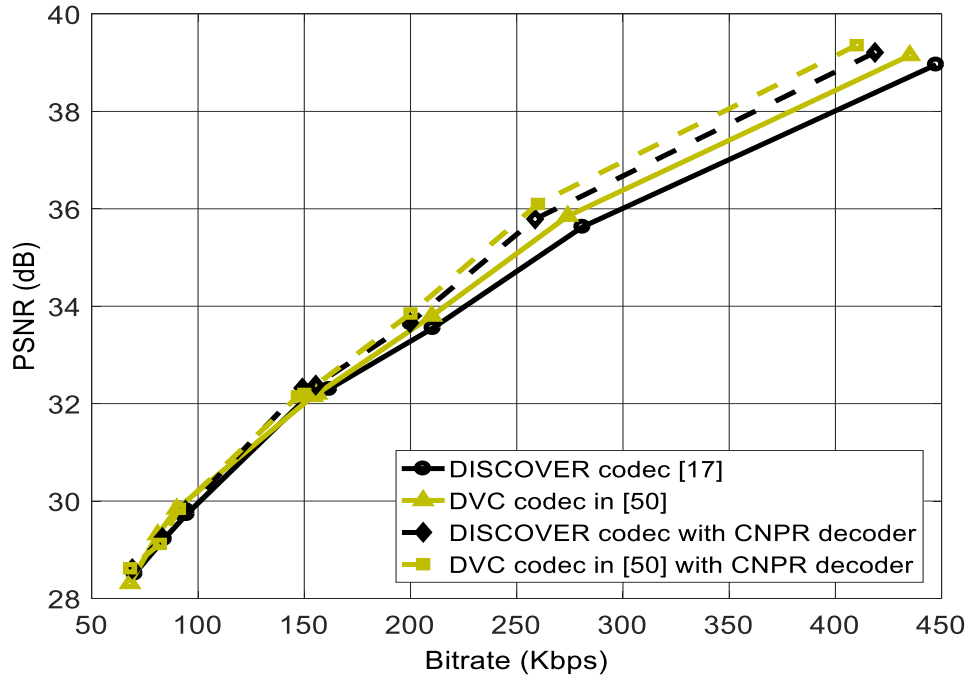


Figure 3.9 RD performance of the DVC codecs for the *Foreman* sequence (GOP size: 2)

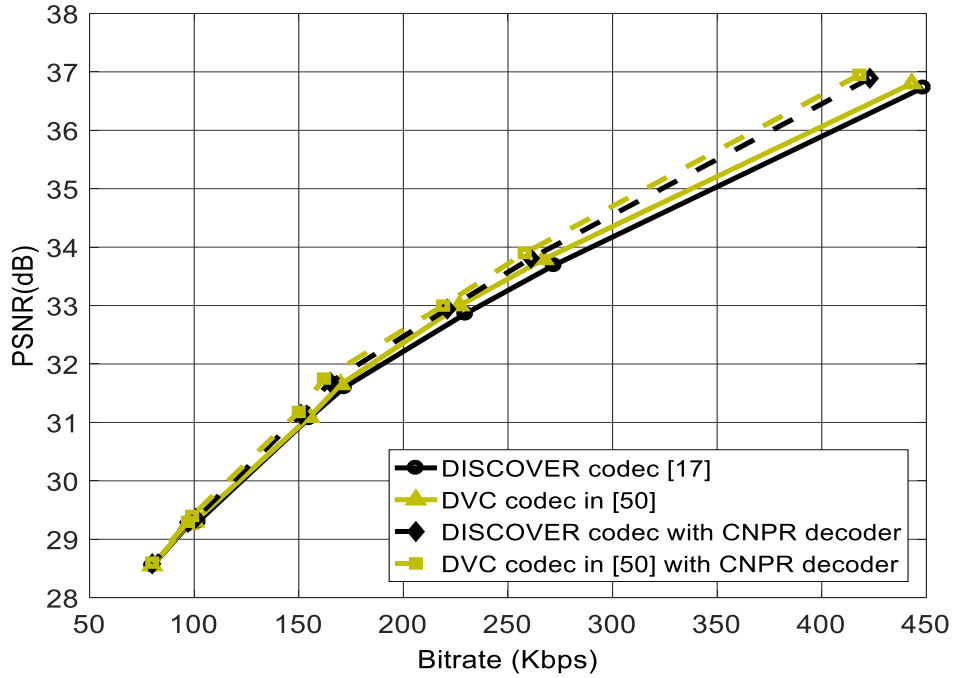


Figure 3.10 RD performance of the DVC codecs for the *Coastguard* sequence (GOP size: 2)

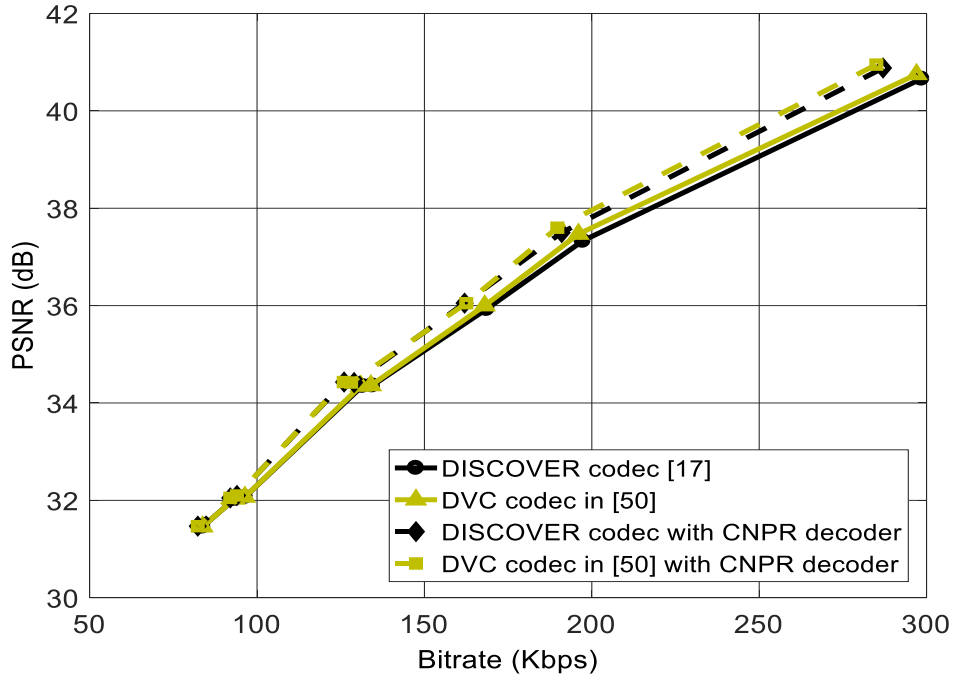


Figure 3.11 RD performance of the DVC codecs for the Hall sequence (GOP size: 2)

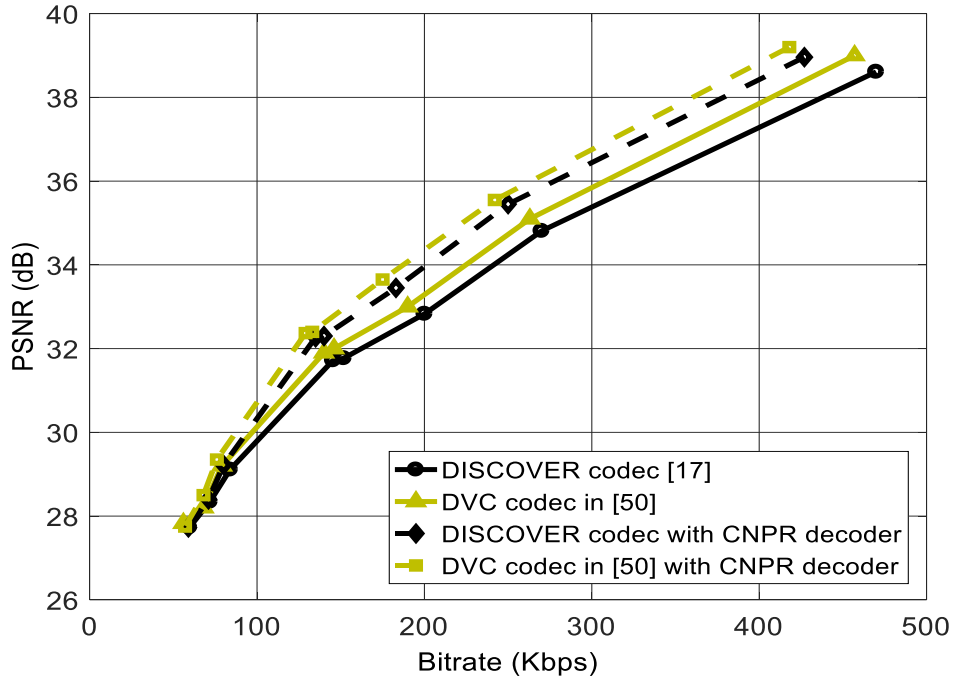


Figure 3.12 RD performance of the DVC codecs for the *Soccer* sequence (GOP size: 2)

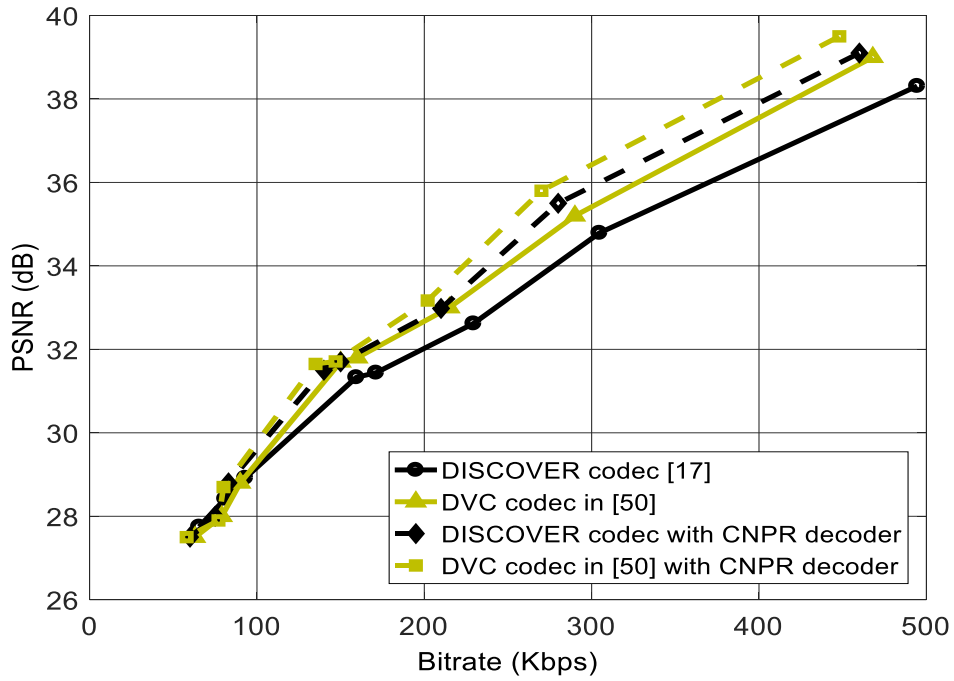


Figure 3.13 RD performance of the DVC codecs for the *Foreman* sequence (GOP size: 4)

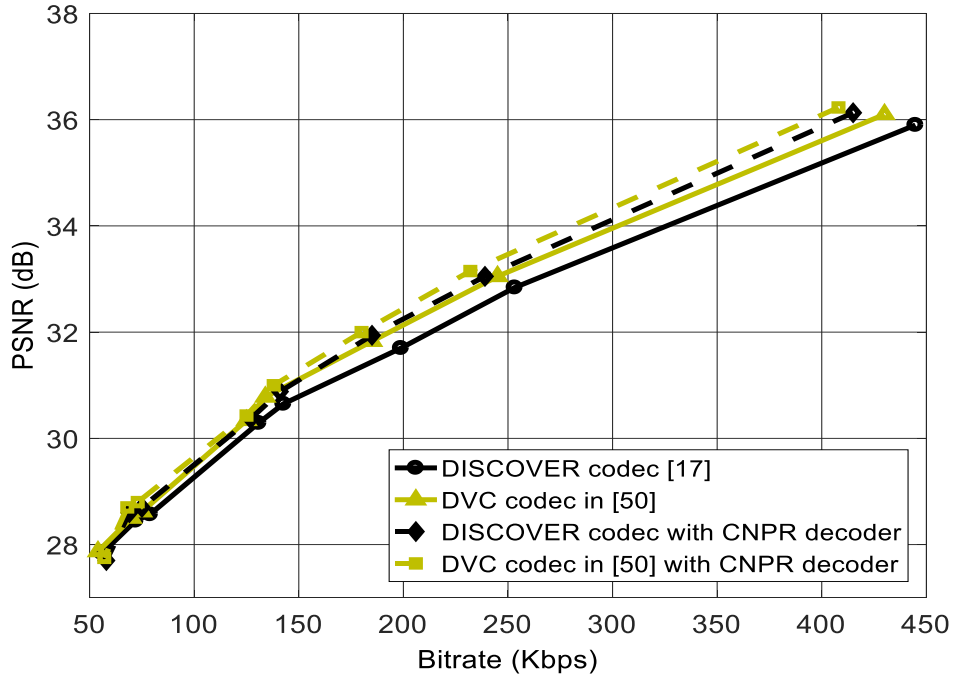


Figure 3.14 RD performance of the DVC codecs for the *Coastguard* sequence (GOP size: 4)

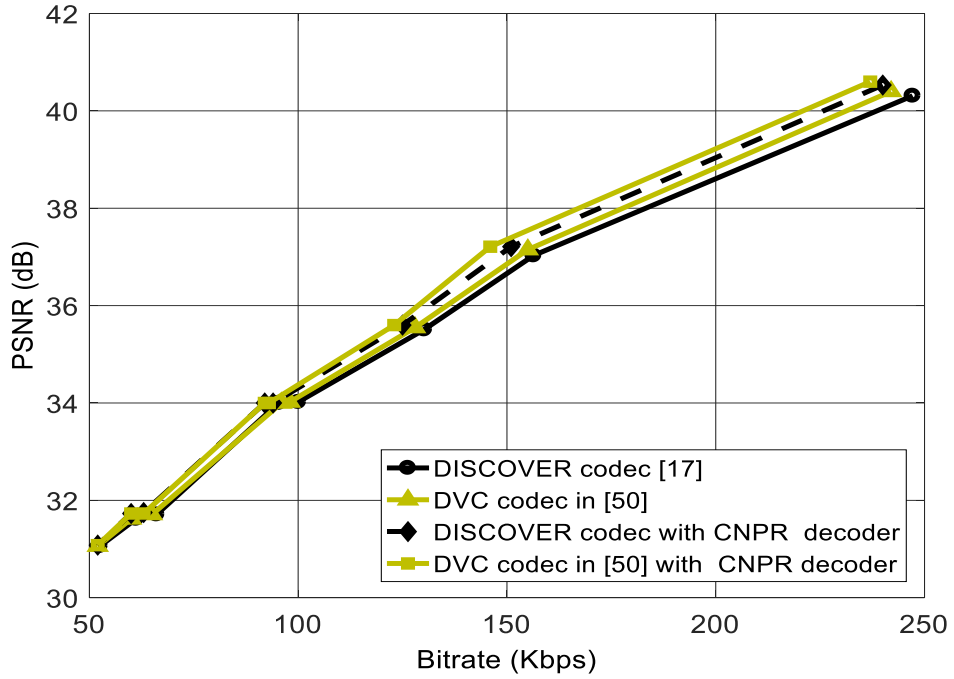


Figure 3.15 RD performance of the DVC codecs for the *Hall* sequence (GOP size: 4)

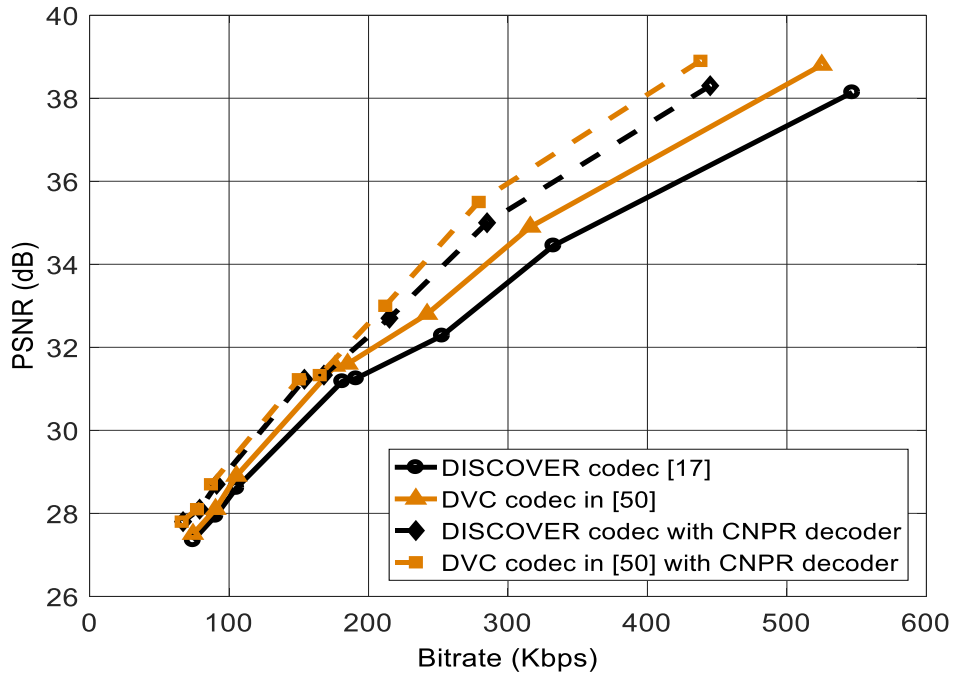


Figure 3.16 RD performance of the DVC codecs for the *Soccer* sequence (GOP size: 4)

performance of the codecs lowers by increasing the GOP size, it is observed by comparing Figures 3-9 – 3.12 with Figures 3.13 - 3.16 that the improvements in the RD performance of the modified codecs over the original ones is more with the higher GOP size.

Figure 3.17 shows two screenshots of a Wyner-Ziv frame of the *Foreman* sequence, decoded by the original DISCOVER codec and by its modified version. Similarly, Figure 3.18 shows again two screenshots, but this time corresponding to the *Soccer* sequence. By comparing the two screenshots in each of these figures, it is observed that the modification in the DISCOVER codec by the proposed decoder has a positive impact on the subjective quality of the decoded frame. Specifically, more details such as sharper edges can be seen in the screenshots resulting from the modified codec. It is seen from the values of the PSNR of the decoded frames and the number of bits required for the

decoding, as provided in the captions of these figures, that the modified codec results in a better-quality decoded frame while requiring a smaller number of bits for their decoding.

(a) (b)

Figure. 3.17: 47th decoded frame of the Foreman sequence, (GOP size :2 and quantization matrix Q6) (a) Decoded frame by the DISCOVER codec, PSNR=31.99 dB, number of bits required for decoding: 17086 bits. (b) Decoded frame by the modified DISCOVER codec, PSNR=32.6 dB, number of bits required for decoding:15575 bits

(a) (b)

Figure. 3.18: 85th decoded frame of the Soccer sequence, (GOP size :2 and quantization matrix Q6) (a) Decoded frame by the DISCOVER codec, PSNR=31.51 dB, number of bits required for decoding: 20614 bits. (b) Decoded frame by the modified DISCOVER codec, PSNR=32.45dB, number of bits required for decoding: 18646 bits

The superior performance of the modified codec can be attributed to the recursive refinement of the estimation of the correlation noise parameter by the proposed decoder.

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 Q_1 is given in Table 3.3 for the original and modified DISCOVER codecs. It is seen from the execution times given in this table that the performance improvement of the modified codec is achieved with an increase in the time complexity. However, it should be pointed that the time-complexity of the modified codec could be significantly reduced if a parallel or multi-threaded programming on a multi-core processor or GPU is employed to implement the proposed decoding scheme, which is inherently a parallel in nature.

Table 3.3 Execution time (in seconds) for decoding the video sequences with GOP size of 2 and quantization matrix Q_1

	Execution time (s)	
sequence	DISCOVER Codec	Modified DISCOVER codec
Foreman	664	1354
Coastguard	489	1097
Hall	391	951
Soccer	1132	1721

3.5 Summary

In this chapter, the problem of estimating the correlation noise parameter in the DVC codec has been investigated with a view to improve the rate-distortion performance and the coding efficiency in a distributed video coding. We have presented a new decoder by proposing a scheme for refining the correlation noise parameters in the decoder of a DVC codec. This new decoder has been designated as the correlation noise refinement (CNPR) based decoder. A recursive decoding algorithm based on variational Bayes has been proposed to estimate and refine the correlation noise distribution parameter while simultaneously decoding all the bitplanes corresponding to the current DCT band on an augmented factor graph. First, a preliminary discussion of parallel LDPCA decoding of the bitplanes of each of DCT bands, the Bayesian estimation of correlation noise parameters and the posterior distribution of correlation noise parameter for each block of symbols in a DCT band on an augmented factor graph have been conducted. Then, a recursive message passing algorithm on the augmented factor graph has been proposed. The variational Bayes method has been employed to approximate the posterior distribution of the correlation noise parameter as a gamma distribution by using side information and the messages received from a set of parallel LDPCA decoders. This approximated distribution is then used to derive a closed-form expression for the messages on the factor graph and to update the messages returned to this set of LDPCA decoders. The performance of the proposed decoding algorithm has been experimentally studied in the framework of a DVC codec using *Foreman, Coastguard, Hall* and *Soccer* sequences. The proposed decoder has then been used in the DISCOVER codec, one of the most popular codecs designed based on the Stanford approach, and in the DVC codec

with side information refinement. It has been shown through extensive experimentation that the DVC codecs modified by the proposed decoder exhibit a rate-distortion performance that is superior to that of the original codecs, particularly on sequences with fast motions. This leads us to believe that proposed decoder can be used to improve the performance of any other codec whose architecture is based on the Stanford approach.

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 and consequently a better rate-distortion performance.

CHAPTER 4

Side Information Frame Refinement

4.1 Introduction

As explained in Chapter 2, the quality of the side information frame influences the rate-distortion performance of distributed video coding. In other words, having a higher quality side information leads to a better decoding performance and consequently, a superior rate-distortion performance. In the previous chapter, we investigated the impact of an accurate estimation of the correlation noise distribution on the improvement of a rate-distortion performance in distributed video coding. In view of this, a new decoder called, the CNPR decoder, was proposed for decoding each DCT band and refining the

correlation noise parameters. In this chapter, we investigate the problem of improving the side information frame in the DVC decoder in order to improve the rate-distortion performance and coding efficiency in distributed video coding. In most of the existing solutions to this problem, the side information frame is generated once at the beginning before the Wyner-Ziv decoder starts to decode the DCT bands of a WZ frame, that is, the side information frame is kept unchanged after the decoding of each DCT band. However, as more information about the decoded frame becomes available after decoding each DCT band of the frame, this information could be utilized to refine the side information, which is employed to decode the next DCT band. In view of this, in this chapter an algorithm is proposed for successive refinements of the side information frame after decoding each DCT band in order to improve the rate distortion performance of the DVC codec [71]. In Section 4.2, the generation of the initial side information frame that is used for decoding the first DCT band is presented. In Section 4.3, this initial side information frame and the resulting first decoded DCT band are then used to develop an algorithm to refine successively the side information frame and to use it for decoding the succeeding DCT bands. In Section 4.4, the performance of the proposed refinement method is studied in the framework of a DVC codec using several video sequences. Finally, in Section 4.5, a study on improving the rate-distortion performance of a DVC codec is conducted by incorporating the proposed algorithm into the CNPR decoder developed in Chapter 3.

4.2 Generation of an Initial Side Information Frame

The side information frame Y_{SI} corresponding to a WZ frame is generated at the decoder using the previously decoded past and future WZ frame(s) or Key frame(s) closest to the

WZ frame in question. A hierarchical motion compensation interpolation procedure, which is the same as the one used in the DISCOVER codec (see Section 2.2), is carried out in order to generate an initial side information frame. By considering the past and future decoded frames, X_P and X_F, first a bidirectional motion estimation between these two frames is carried out to find a pair of symmetric motion vectors, $\left(\frac{v_x}{2},\frac{v_y}{2}\right)$ and $\left(-\frac{v_x}{2},-\frac{v_y}{2}\right)$, between each block in the intermediate frame and the two matched blocks in the frames X_P and X_F. A bidirectional motion compensation is then performed between X_P and X_F using the symmetric motion vectors to generate the interpolated frame as an initial side information frame given by

$$Y_{SI}^{(0)}(x,y)=\frac{X_P\left(x-R_x^{(0)},y-S_y^{(0)}\right)+X_F\left(x+U_x^{(0)},y+V_y^{(0)}\right)}{2} \tag{4-1}$$

where

$$R_x^{(0)}=U_x^{(0)}=\frac{v_x}{2}$$
$$S_y^{(0)}=V_y^{(0)}=\frac{v_y}{2},$$

(x,y) represents the position of pixel in the interpolated frame, and $X_P\left(x-R_x^{(0)},y-S_y^{(0)}\right)$ and $X_F\left(x+U_x^{(0)},y+V_y^{(0)}\right)$ are, respectively, the backward and forward motion compensated frames. This initial side information frame $Y_{SI}^{(0)}$ is used to decode only the first DCT band of the Wyner-Ziv frame in question.

4.3 Proposed Method for Successive Refinement of Side Information

We now propose a scheme for successively refining the side information frame in order to improve its quality after the decoding of each DCT band of a Wyner-Ziv frame. We will call this scheme as successive refinement side information frame (SRSIF) based scheme. Given $Y_{SI}^{(i-1)}(x,y)$, $i = 1,2,...,\eta$, (η being the number of non-zero elements in the quantization matrix) , as the side information frame refined after decoding the $(i-1)^{st}$ DCT band, the i^{th} refined side information $Y_{SI}^{(i)}(x,y)$ is obtained by following the steps given below.

Step 1 - The i^{th} DCT band is decoded by a Wyner-Ziv decoder by employing the side information frame $Y_{SI}^{(i-1)}(x,y)$.

Step 2 - An optimal MMSE reconstruction is carried out for each DCT symbol $x_n^{(i)}$ in the decoded DCT band $X_{DCT}^{(i)}$ $(n = 1,...,N)$, N being the length of the DCT band, to obtain the reconstructed DCT coefficient $\hat{x}_n^{(i)} = E\left(x_n^{(i)} \mid y_n\right)$ [60].

Step 3 – A discrete cosine transform is applied to each non-overlapped 4x4 blocks in the backward and forward motion compensated frames, $X_P\left(x - R_x^{(i-1)}, y - S_y^{(i-1)}\right)$ and $X_F\left(x + U_x^{(i-1)}, y + V_y^{(i-1)}\right)$, used to obtain $Y_{SI}^{(i-1)}(x,y)$, yielding

$$X_{DCT}^{P\ (i)} = DCT\left(X_P\left(x - R_x^{(i-1)}, y - S_y^{(i-1)}\right)\right) \quad (4\text{-}2)$$

$$X_{DCT}^{F\ (i)} = DCT\left(X_F\left(x + U_x^{(i-1)}, y + V_y^{(i-1)}\right)\right) \quad (4\text{-}3)$$

Step 4 - The corresponding DCT coefficients in each 4x4 block given by (4-2) and (4-3) are replaced by the reconstructed DCT coefficients belonging to all the i DCT bands decoded so far, giving modified DCT coefficients denote by $\widehat{X}_{DCT}^{P\ (i)}$ and $\widehat{X}_{DCT}^{F\ (i)}$, respectively.

Step 5 – The inverse DCT is applied to each 4x4 non-overlapped block in $\widehat{X}_{DCT}^{P}$ and $\hat{X}_{DCT}^{F}$ to obtain $\widehat{X}_{P}^{(i)}$ and $\widehat{X}_{F}^{(i)}$, respectively.

Step 6- The motion vector $(R_x^{(i)}, S_y^{(i)})$ from each block in the frame $\widehat{X}_{P}^{(i)}$ to the matched block in the frame X_P and motion vector $(U_x^{(i)}, V_y^{(i)})$ from each block in the frame $\widehat{X}_{F}^{(i)}$ to the matched block in the frame X_F are estimated.

Step 7 – The motion vectors $(R_x^{(i)}, S_y^{(i)})$ and $(U_x^{(i)}, V_y^{(i)})$ are used to obtain i^{th} refined side information as

$$Y_{SI}^{(i)}(x,y) = \frac{X_P\left(x - R_x^{(i)}, y - S_y^{(i)}\right) + X_F\left(x + U_x^{(i)}, y + V_y^{(i)}\right)}{2} \qquad (4\text{-}4)$$

The above algorithm is applied repeatedly for $i = 1, 2, ..., \eta$. Then, the discrete cosine transform is applied to each of the 4x4 blocks of the last refined side information frame, namely $Y_{SI}^{(\eta)}(x,y)$, and the first η DCT coefficients, going in a zig-zag order (see Figure 1.7), in its 4x4 blocks are replaced by the corresponding reconstructed DCT coefficients in all the η decoded DCT bands. Finally, a 4x4 inverse DCT is applied to the resulting frame to obtain the decoded Wyner-Ziv frame.

4.4 Simulation Results

In this section, we study the rate-distortion performance of the DISCOVER codec without as well as with the refinement of the side information frame by the SIR method given in [50] and the scheme proposed in Section 4.3. For the simulations, the *Hall*, *Coastguard*, *Foreman* and *Soccer* video sequences, each having 150 frames in the QCIF format with *15-Hz* frame rate, are used. As in our experiments in Section 3.8, the key frames are encoded using the intra coding mode of the H.264/AVC codec. The QP values in this codec are set to be the same as the ones used for the key frames in the original DISCOVER codec. Also, only the luminance component (Y) of the video frames is considered in our simulation.

To examine the performance of the proposed refinement scheme of a side information frame, some screenshots demonstrating the successive refinement of the side information frame corresponding to the first frame of the *Foreman* and *Soccer* sequences are shown in Figures 4.1 and 4.2, respectively. It is seen from these screenshots that the proposed scheme of refinement successively improves the subjective quality of the side information frame. It is also seen from associated PSNR values specified in the captions

(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.1 Successive refinements of the side information frame corresponding to the first WZ frame of the *Foreman* sequence, Quantization matrix: Q_6. (a) Initial side information, PSNR: 28.66 dB. (b) First refinement, PSNR: 30.86 dB. (c) Second refinement, PSNR: 31.26 dB. (d) Third refinement, PSNR: 32.50 dB. (e) Forth refinement, PSNR: 32.65 dB. (f) Fifth refinement, PSNR: 33.26.

(a)

(b)

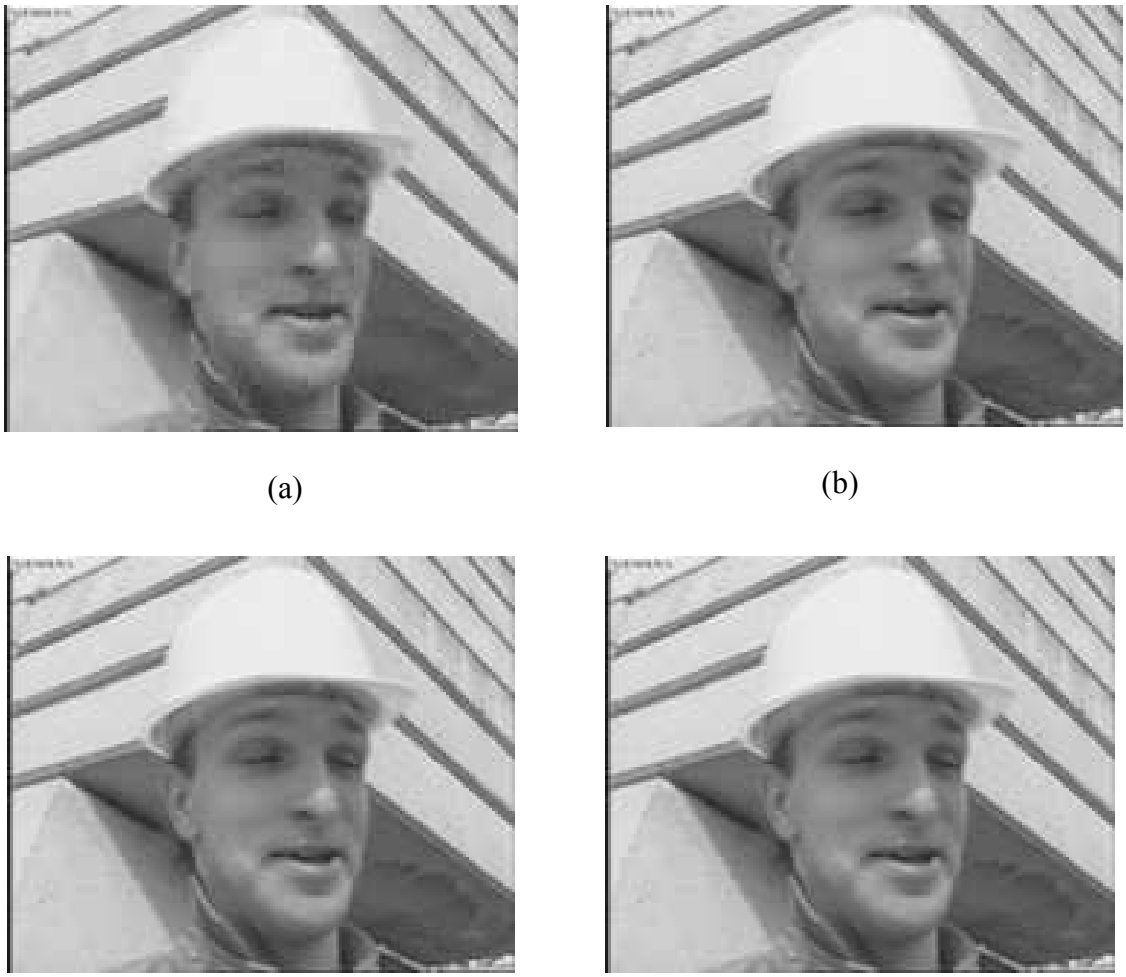
(c)

(d)

(e)

(f)

Figure 4.2 Successive refinements of the side information frame corresponding to the first WZ frame of the *Soccer* sequence, Quantization matrix: Q_6. (a) Initial side information, PSNR: 21.86 dB, (b) First refinement, PSNR: 26.15 dB. (c) Second refinement, PSNR: 28.29 dB. (d) Third refinement, PSNR: 29.29 dB. (e) Forth refinement, PSNR: 29.46 dB. (f) Fifth refinement, PSNR: 29.79.



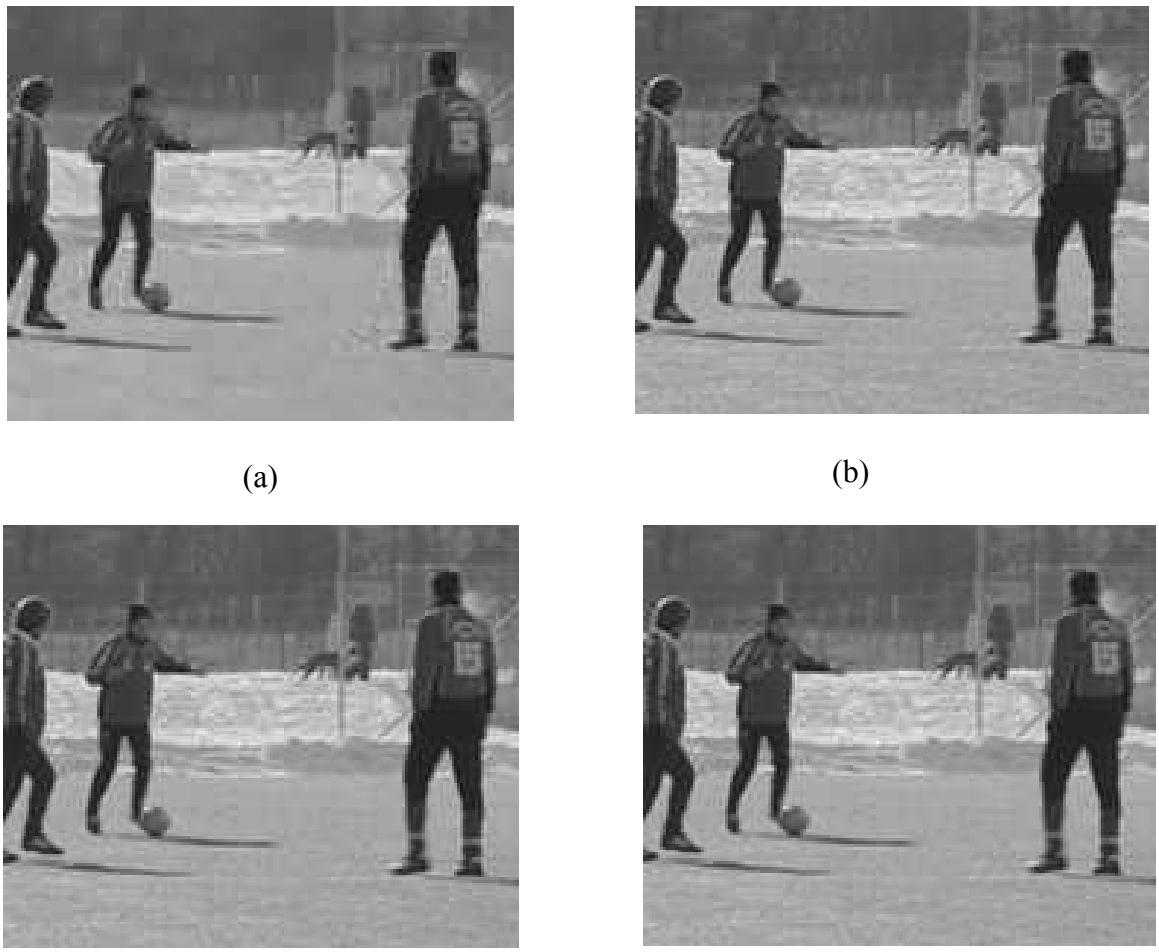
(a) (b)

(c) (d)

Figure 4.3. The decoded frame corresponding to the first WZ frame of the *Foreman* sequence. (a) The Wyner-Ziv frame at the encoder. The decoded WZ frame using (b) the original DISCOVER codec, PSNR:33.9 dB. (c) The DVC codec with the SIR method in [50], PSNR: 34.02 dB, and (d) the DVC codec with the proposed scheme, PSNR: 34.44 dB.

of the figures that each refinement improves the objective quality of the side information frame as well.

Figures 4.3 and 4.4 give, respectively, the examples of the decoded frames of the first Wyner-Ziv frames of the *Foreman* and *Soccer* sequences. Parts (a) of these figures show



(a) (b)

(c) (d)

Figure 4.4. Decoded frame corresponding to the first WZ frame of the *Soccer* sequence. (a) The Wyner-Ziv frame at the encoder. The decoded WZ frame using (b) the original DISCOVER codec, PSNR:32.03 dB. (c) The DVC codec with the SIR method in [50], PSNR: 32.15 dB, and (d) the DVC codec with the proposed scheme, PSNR: 32.46 dB.

the Wyner-Ziv frame at the encoder, whereas parts (b), (c), (d) of these figures show, respectively, the frames decoded by the original DISCOVER codec, and by the DVC codecs with the SIR and the proposed schemes for the refinement of the side information frames. It can be seen from these figures that using the SIR method for the refining the

side information frame results in improving the quality of the decoded WZ frame. However, this improvement is even more when the proposed method of refinement is used.

Tables 4.1 to 4.4 give the total number of bits needed to decode each DCT band of all the WZ frames in the *Hall, Coastguard, Foreman* and *Soccer* sequences, respectively, when they are encoded with the quantization matrix Q_6. It is seen from these tables that the proposed refinement method results in the reduction in the number of bits needed for decoding almost all of the DCT bands with the exception of the 12^{th} to 15^{th} DCT bands of the *Hall* sequence. It is also noted that these reductions are more substantial for the video sequences with the fast and complex motions such as the *Soccer* and *Foreman* sequences.

The rate-distortion curves fitting the eight RD points corresponding to the eight 4x4 quantization matrices are obtained in order to demonstrate the improvements in the RD performance in the DVC codecs with the proposed method of refining the side information frames. Figures 4.5 to 4.8 show the overall RD performance for the *Hall*, *Foreman*, *Coastguard* and *Soccer* sequences, respectively, using the original DISCOVER codec, and the DISCOVER codec with the side information refinement scheme of [50] and that with the proposed scheme, for the GOP size of 2. It is seen from these figures that the DISCOVER codecs modified by our proposed side information refinement scheme exhibits the best RD performance. Further, it is also noted that the improvements in the RD performance is even more in the case of the sequences with fast and complex motions such as *Foreman* and *Soccer* sequences for which the quality of initial side information frames is poor, and hence, these frames benefit more from the proposed scheme of refinement.

Table 4.1. Total number of bits for decoding each DCT band in all the WZ frames of the *Hall* sequence, without and with using the proposed scheme for refining the side information frame.

DCT band	Without refinement	With refinement	Percentage reduction
1st DCT band	173.993 kbits	173.993 kbits	0
2nd DCT band	33.984 kbits	32.811 kbits	3.4
3rd DCT band	22.265 kbits	21.972 kbits	1.3
4th DCT band	12.304 kbits	12.304 kbits	0
5th DCT band	14.282 kbits	13.769 kbits	3.6
6th DCT band	19.116 kbits	18.310 kbits	4.2
7th DCT band	24.026 kbits	22.440 kbits	6.6
8th DCT band	17.651 kbits	16.845 kbits	4.5
9th DCT band	19.116 kbits	18.823 kbits	1.4
10th DCT band	13.476 kbits	13.403 kbits	0.5
11th DCT band	8.203 kbits	8.203 kbits	0
12th DCT band	8.203 kbits	8.203 kbits	0
13th DCT band	8.203 kbits	8.203 kbits	0
14th DCT band	8.203 kbits	8.203 kbits	0
15th DCT band	8.203 kbits	8.203 kbits	0
Aggregate	391.228 kbits	385.685 kbits	1.4

Table 4.2. Total number of bits for decoding each DCT band in all the WZ frames of the *Coastguard* sequence, without and with using the proposed scheme for refining the side information frame.

DCT band	Without refinement	With refinement	Percentage reduction
1st DCT band	218.554 kbits	218.554 kbits	0
2nd DCT band	41.894 kbits	39.843 kbits	4.9
3rd DCT band	61.523 kbits	60.717 kbits	1.3
4th DCT band	21.972 kbits	21.093 kbits	4.0
5th DCT band	28.125 kbits	25.488 kbits	9.3
6th DCT band	29.370 kbits	25.122 kbits	14.4
7th DCT band	42.919 kbits	38.643 kbits	9.9
8th DCT band	30.981 kbits	25.488 kbits	17.7
9th DCT band	38.745 kbits	34.277 kbits	11.6
10th DCT band	35.009 kbits	32.812 kbits	6.2
11th DCT band	11.132 kbits	10.253 kbits	7.9
12th DCT band	9.301 kbits	8.715 kbits	6.3
13th DCT band	8.862 kbits	8.203 kbits	6.8
14th DCT band	9.878 kbits	8.935 kbits	9.5
15th DCT band	13.03 kbits	11.791 kbits	9.5
Aggregate	601.297 kbits	569.934 kbits	5.2

Table 4.3. Total number of bits for decoding each DCT band in all the WZ frames of the *Foreman* sequence, without and with using the proposed scheme for refining the side information frame.

DCT band	Without refinement	With refinement	Percentage reduction
1st DCT band	317.126 kbits	317.126 kbits	0
2nd DCT band	62.676 kbits	60.504 kbits	3.4
3rd DCT band	107.735 kbits	104.788 kbits	2.7
4th DCT band	40.007 kbits	34.978 kbits	12.5
5th DCT band	54.980 kbits	51.639 kbits	6.1
6th DCT band	15.867 kbits	14.054 kbits	11.4
7th DCT band	17.564 kbits	16.005 kbits	8.8
8th DCT band	53.745 kbits	44.573 kbits	17.0
9th DCT band	56.432 kbits	44.566 kbits	21.0
10th DCT band	32.324 kbits	27.992 kbits	13.3
11th DCT band	14.346 kbits	12.446 kbits	13.2
12th DCT band	15.678 kbits	12.432 kbits	20.1
13th DCT band	14.734 kbits	12.634 kbits	14.2
14th DCT band	16.006 kbits	11.834 kbits	26.0
15th DCT band	19.303 kbits	16.011 kbits	17.0
Aggregate	838.523 kbits	781.582 kbits	6.8

Table 4.4. Total number of bits for decoding each DCT band in all the WZ frames of the *Soccer* sequence, without and with using the proposed scheme for refining the side information frame.

DCT band	Without refinement	With refinement	Percentage reduction
1st DCT band	489.343 kbits	489.343 kbits	0
2nd DCT band	157.112 kbits	131.234 kbits	16.4
3rd DCT band	141.466 kbits	126.345 kbits	10.7
4th DCT band	60.891 kbits	52.111 kbits	14.42
5th DCT band	53.422 kbits	37.666 kbits	29.5
6th DCT band	56.145 kbits	40.776 kbits	27.34
7th DCT band	57.112 kbits	36.677 kbits	35.8
8th DCT band	44.156 kbits	31.678 kbits	28.2
9th DCT band	47.316 kbits	34.539 kbits	27.0
10th DCT band	70.448 kbits	59.1156 kbits	16.0
11th DCT band	12.357 kbits	11.911 kbits	3.6
12th DCT band	9.322 kbits	8.124 kbits	12.8
13th DCT band	11.567 kbits	10.654 kbits	7.9
14th DCT band	11.173 kbits	10.566 kbits	5.4
15th DCT band	11.678 kbits	10.112 kbits	13.4
Aggregate	1233.2 kbits	1090.9 kbits	11.5

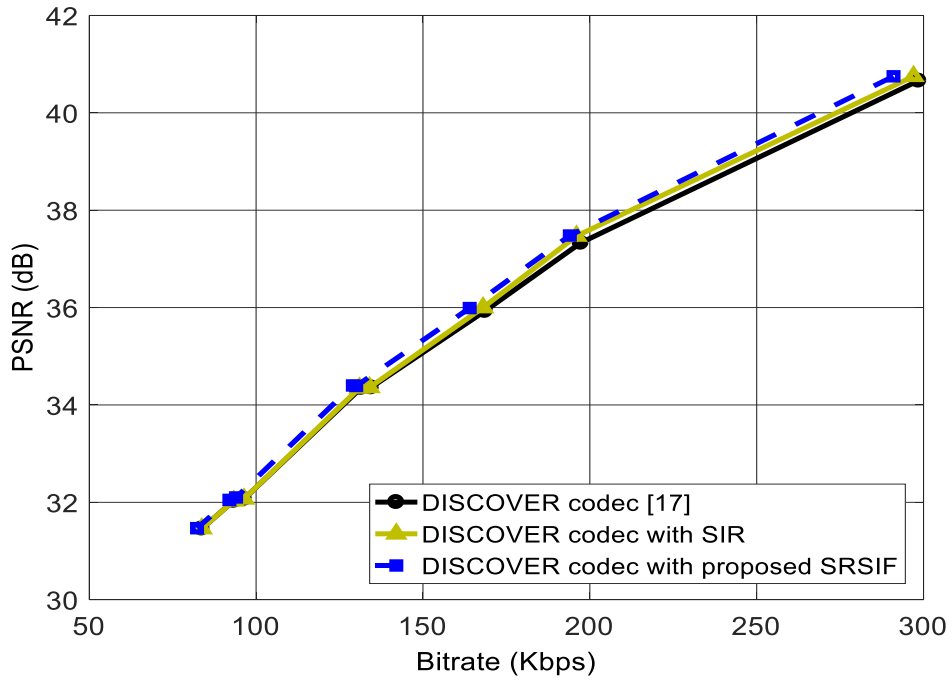


Figure 4.5 RD performance of the DVC codecs for the *Hall* sequence (GOP size: 2)

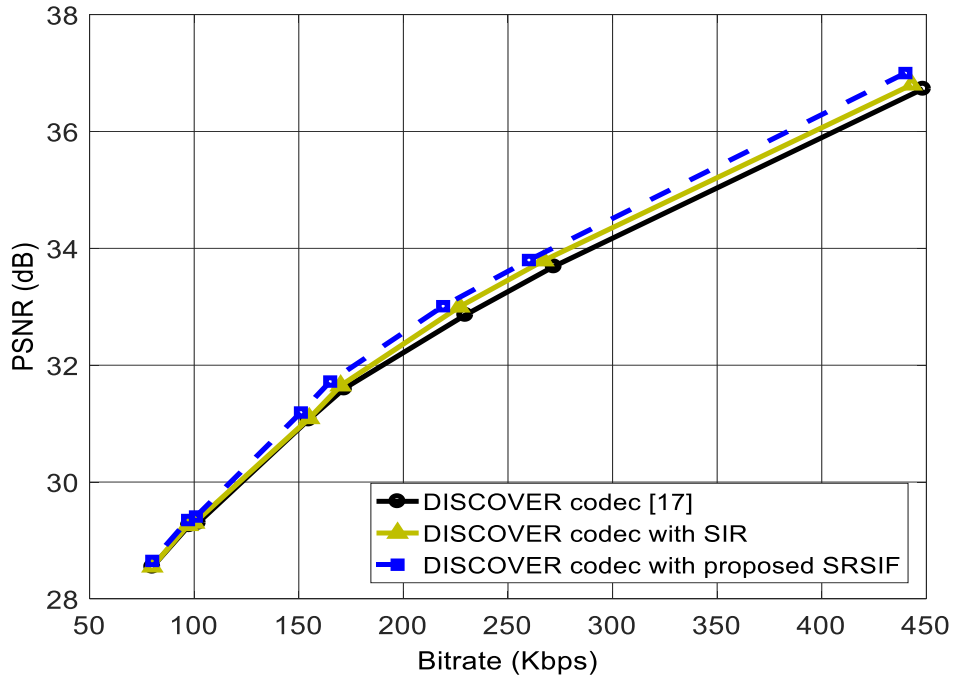


Figure 4.6 RD performance of the DVC codecs for the *Coastguard* sequence (GOP size: 2)

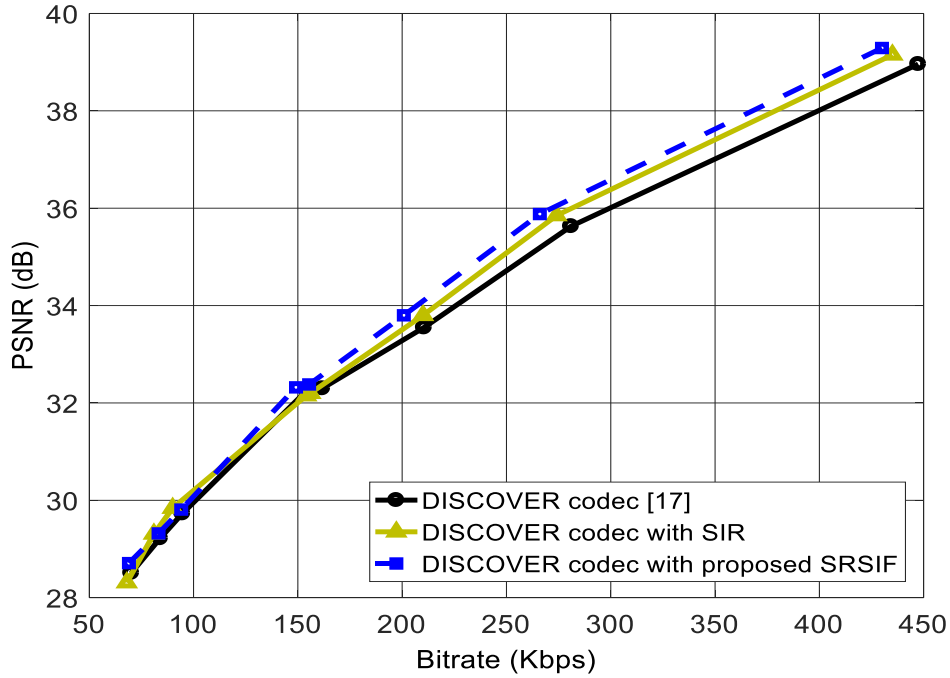


Figure 4.7 RD performance of the DVC codecs for the *Foreman* sequence (GOP size: 2)

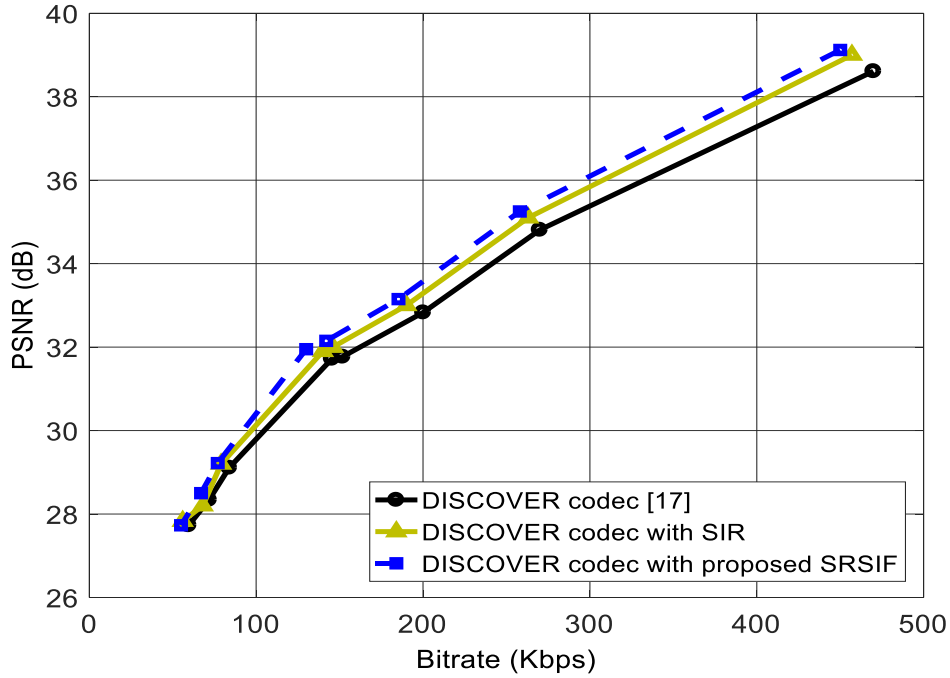


Figure 4.8 RD performance of the DVC codecs for the *Soccer* sequence (GOP size: 2)

4.5 Incorporation of the Proposed Scheme for Successive Refinement of Side Information Frame into the CNPR Decoder

In Chapter 3, a new decoder, called the *CNPR* decoder, was proposed for decoding each DCT band followed by refinement of a correlation noise parameter in a DVC codec in order to improve its rate-distortion performance. In this chapter, we have proposed a scheme for successive refinements of the side information frame (*SRSIF*). We now incorporate this refinement scheme into the *CNPR* decoder and call the resulting decoder as the *CNPR-SRSIF* decoder. This decoder is shown in Figure 4.9.

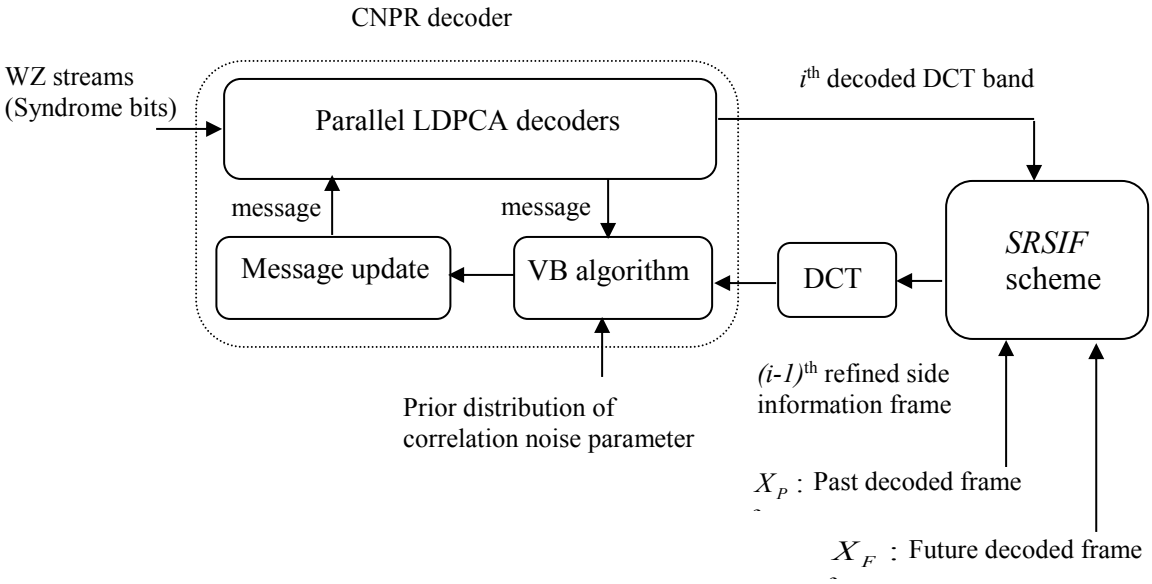


Figure 4.9. CNPR-SRSIF decoder.

Given $Y_{SI}^{(i-1)}(x,y)$, $i = 1,2,...,\eta$, as the side information frame refined after decoding the $(i-1)^{st}$ DCT band, the CNPR-SRSIF decoder works by carrying out following two steps:

1- i^{th} DCT band of a WZ frame is decoded by the CNPR decoder by applying the joint decoding and correlation noise parameter refinement algorithm proposed in Section 3.5, using the i^{th} DCT band in the side information frame, $Y_{SI}^{(i-1)}(x,y)$

2- The resulting i^{th} decoded DCT band obtained in Step 1 is then used by the *SRSIF* scheme of Section.4.3 to obtain the i^{th} refined side information frame, $Y_{SI}^{(i)}(x,y)$.

The above two steps are applied successively by the *CNPR-SRSIF* decoder for $i = 1, 2, ..., \eta$. Then, the discrete cosine transform is applied to each of the 4x4 blocks of the last refined side information frame, namely $Y_{SI}^{(\eta)}(x,y)$, and the first η DCT coefficients, going in a zig-zag order in its 4x4 blocks are replaced by the corresponding reconstructed DCT coefficients in all the η decoded DCT bands. Finally, a 4x4 inverse DCT is applied to resulting frame to obtain the decoded Wyner-Ziv frame.

A DVC codec using this new decoder will be referred to as the *CNPR-SRSIF* DVC codec. This new codec is shown in Figure 4.10.

We now examine the rate-distortion performance of the new CNRP-SRSIF DVC codec and compare its performance results with those of the original DISCOVER codec, the CNPR DVC codec and the SRSIF incorporated DISCOVER codec. For the simulations, the *Hall*, *Coastguard*, *Foreman*, and *Soccer* video sequences, each having 150 frames in QCIF format with *15-Hz* frame rate, are used. The key frames are encoded

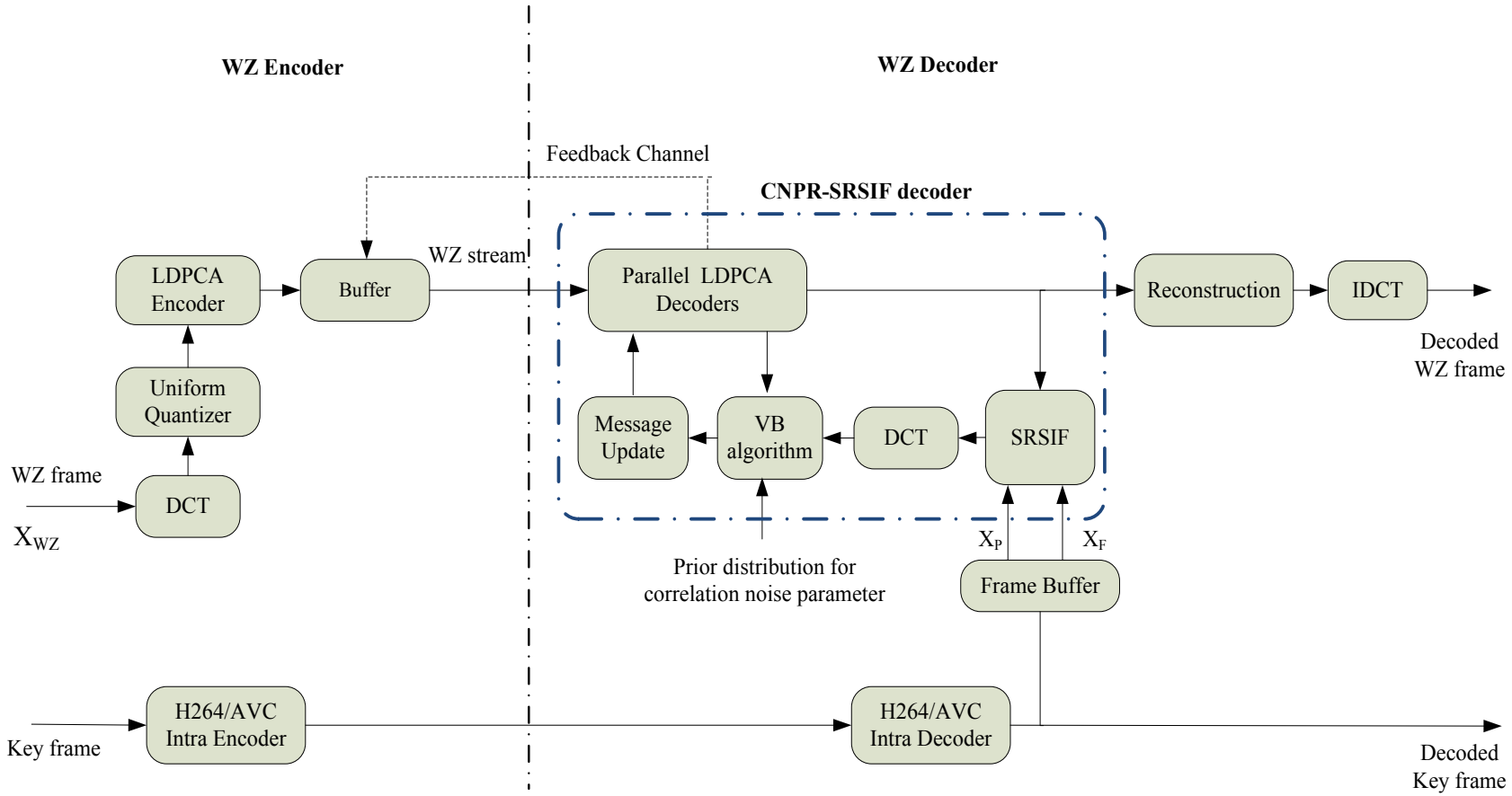


Figure 4.10 CNPR-SRSIF DVC codec

using the intra coding mode of the H.264/AVC codec. The QP values in this codec are set to have the same values as the ones used for the key frames of video sequences in the DISCOVER codec. Only the luminance component (Y) of the video frames is considered in this simulation.

Table 4.5 gives, for the CNPR-SRSIF DVC codec, the relative average savings (in %) in the bitrate and improvement in PSNR (in dB) (computed using Bjøntegaard metric[70]) over that of the DISCOVER codec corresponding to the WZ frame as well as for all the frames. By comparing the values in this table with the corresponding values in Table 3.1, it is seen that incorporating the SRSIF scheme into the CNPR decoder leads to higher improvements in PSNR and more reduction in bitrate in the DVC codec.

Table 4.5 - The relative bitrate saving (%) and improvement in PSNR(dB) CNPR-SRSIF DVC codec over that of DISCOVER codec, computed using the BJØNTEGAARD metric

	GOP=2				GOP=4			
	WZ frames		All frames		WZ frames		All frames	
	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)	ΔR %	$\Delta PSNR$ (in dB)
Foreman	14.47	0.39	7.53	0.22	13.18	0.36	10.67	0.31
Coastguard	11.63	0.34	5.81	0.19	10.61	0.35	8.67	0.25
Hall	8.13	0.31	3.9	0.16	7.88	0. 31	5.46	0.23
Soccer	18.71	0.79	10.23	0.38	15.2	0.61	13.16	0.44
Average	13.23	0.46	6.86	0.23	11.71	0.40	9.49	0.30

Figures 4.11 - 4.14 show the overall RD performance of the four DVC codecs for the *Hall*, *Coastguard*, *Foreman* and *Soccer* sequences, respectively for GOP size of 2. The corresponding results for GOP size of 4 are shown in Figures 4.15 - 4.18. It is seen from figures 4.11 - 4.18 that by incorporating the proposed refinement scheme for the side information into the CNPR decoder, the RD performance of the resulting DVC codec

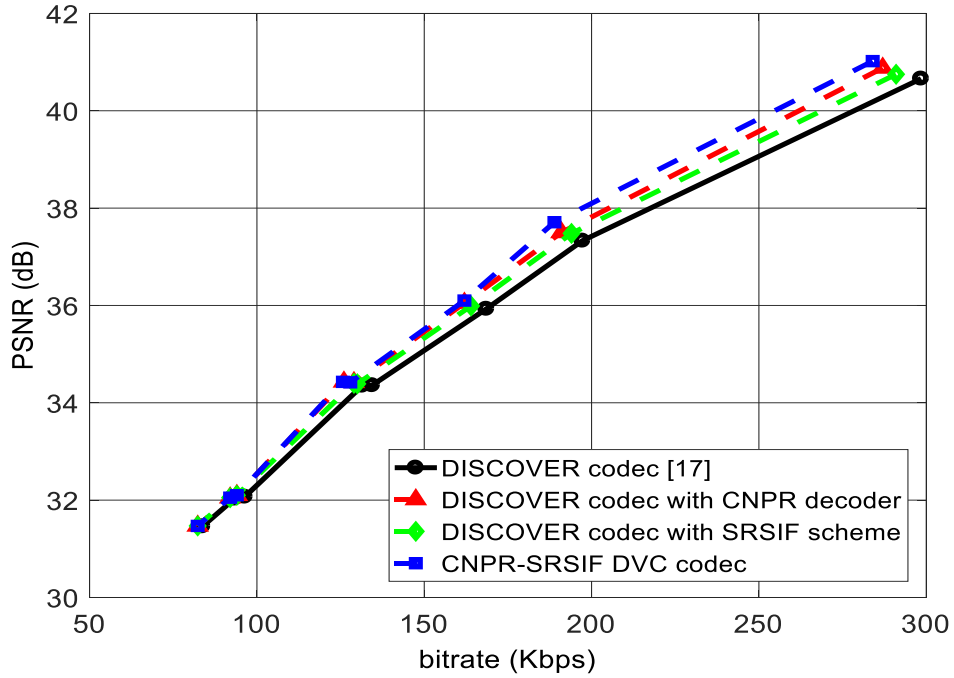


Figure 4.11 RD performance of the DVC codecs for the *Hall* sequence (GOP size: 2)

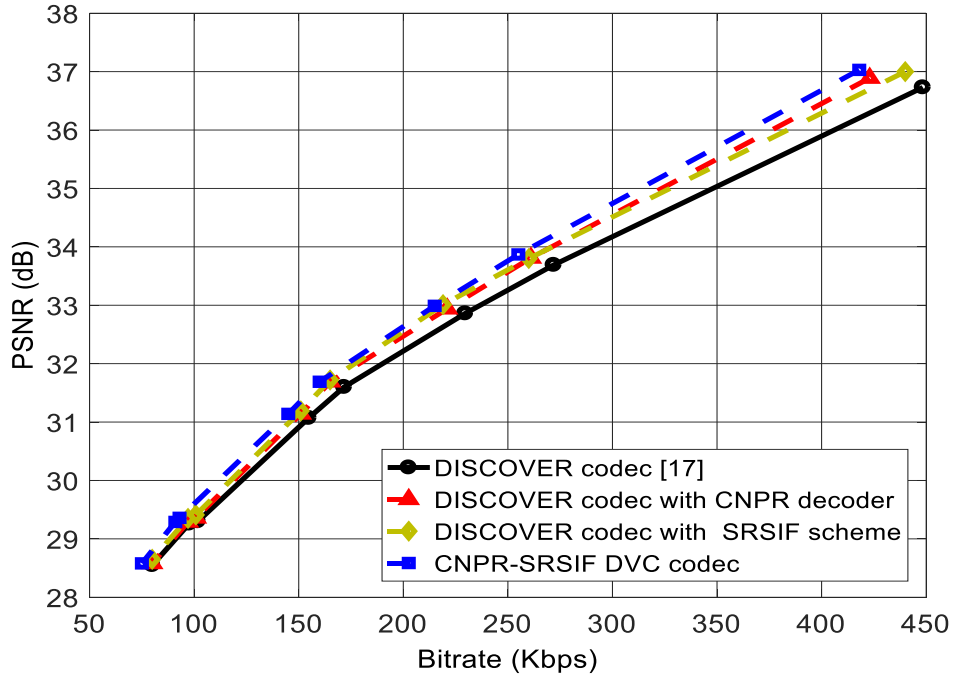


Figure 4.12 RD performance of the DVC codecs for the *Coastguard* sequence (GOP size: 2)

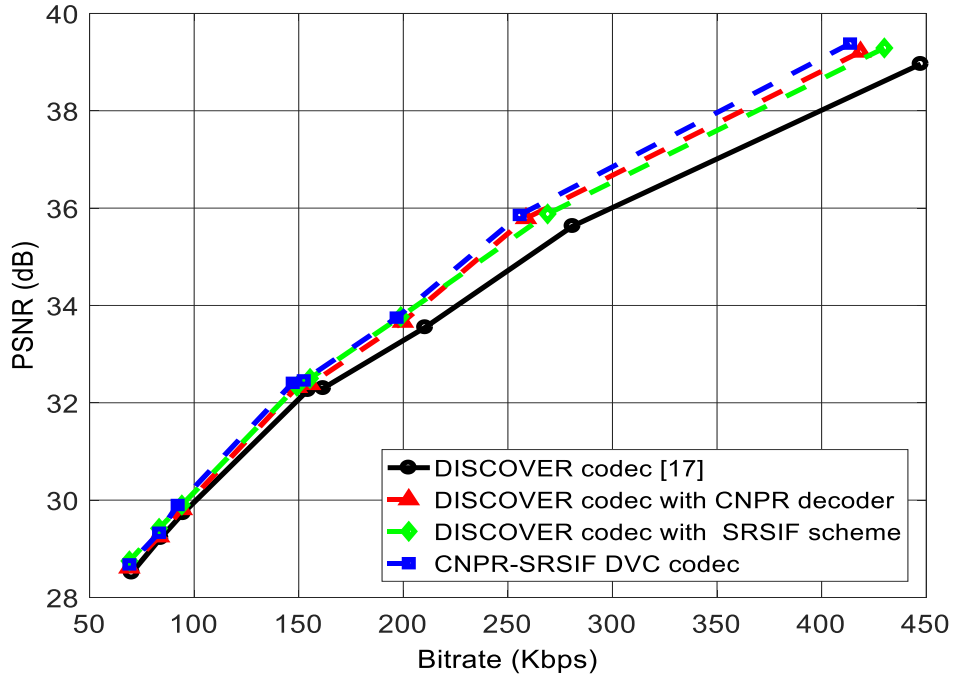


Figure 4.13 RD performance of the DVC codecs for the *Foreman* sequence (GOP size: 2)

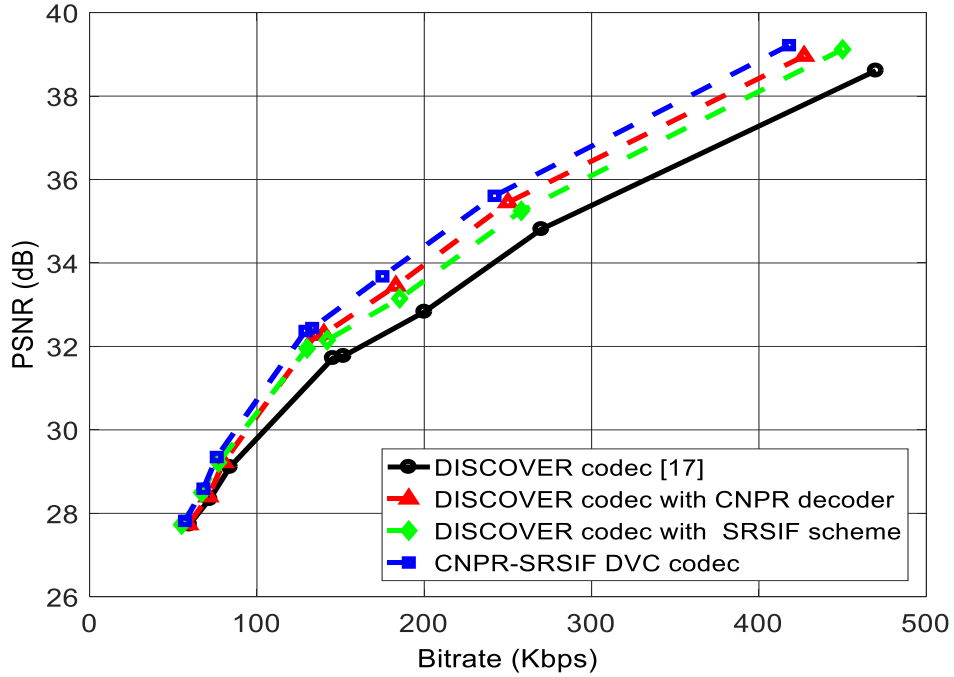


Figure 4.14 RD performance of the DVC codecs for the *Soccer* sequence (GOP size: 2)

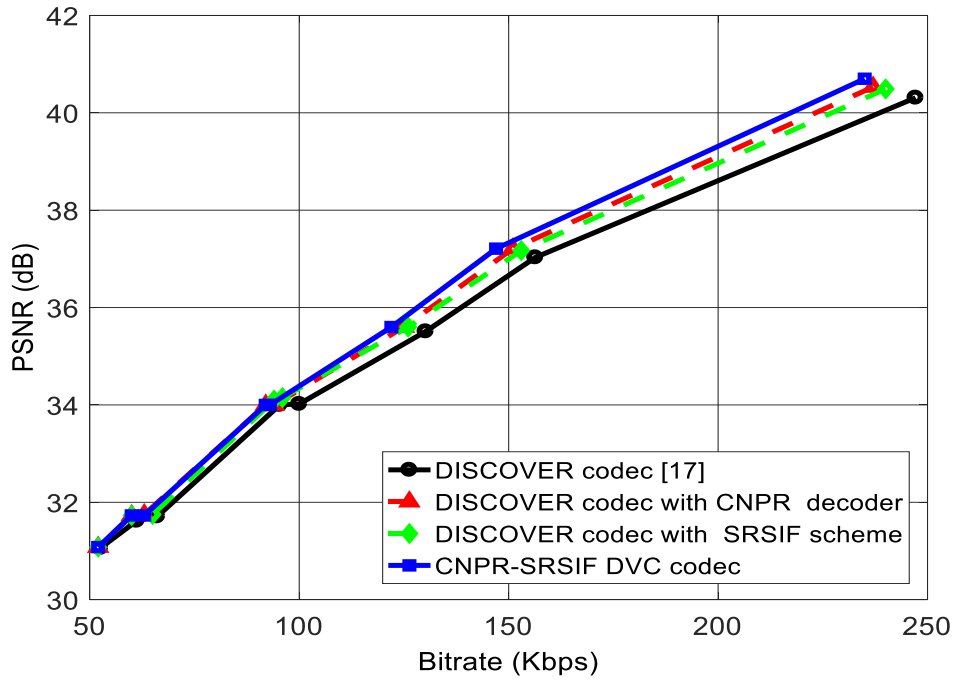


Figure 4.15 RD performance of the DVC codecs for the *Hall* sequence (GOP size: 4)

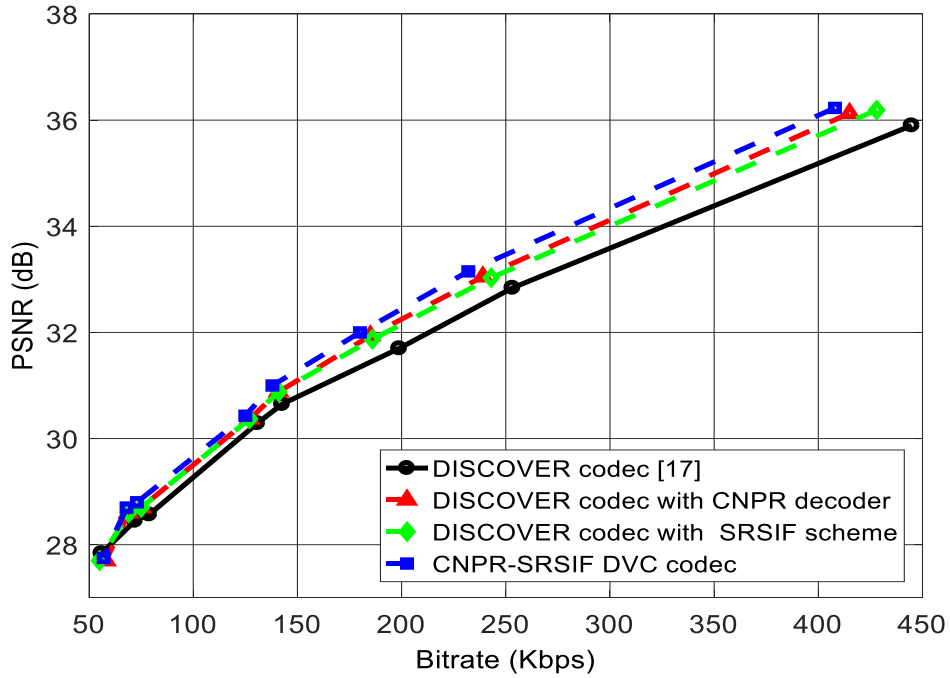


Figure 4.16 RD performance of the DVC codecs for the *Coastguard* sequence (GOP size: 4)

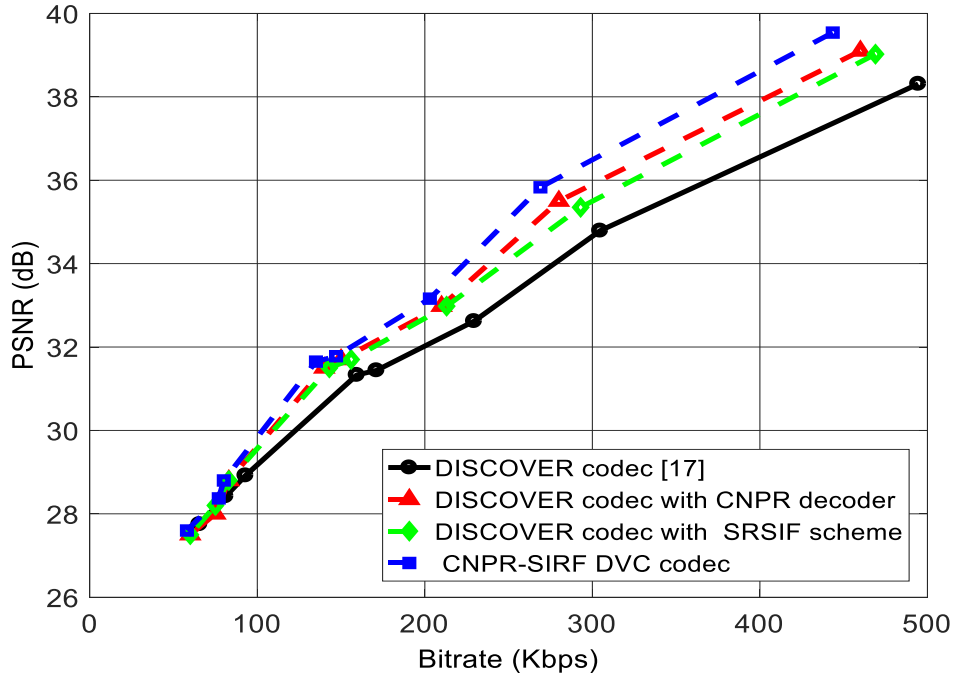


Figure 4.17 RD performance of the DVC codecs for the *Foreman* sequence (GOP size: 4)

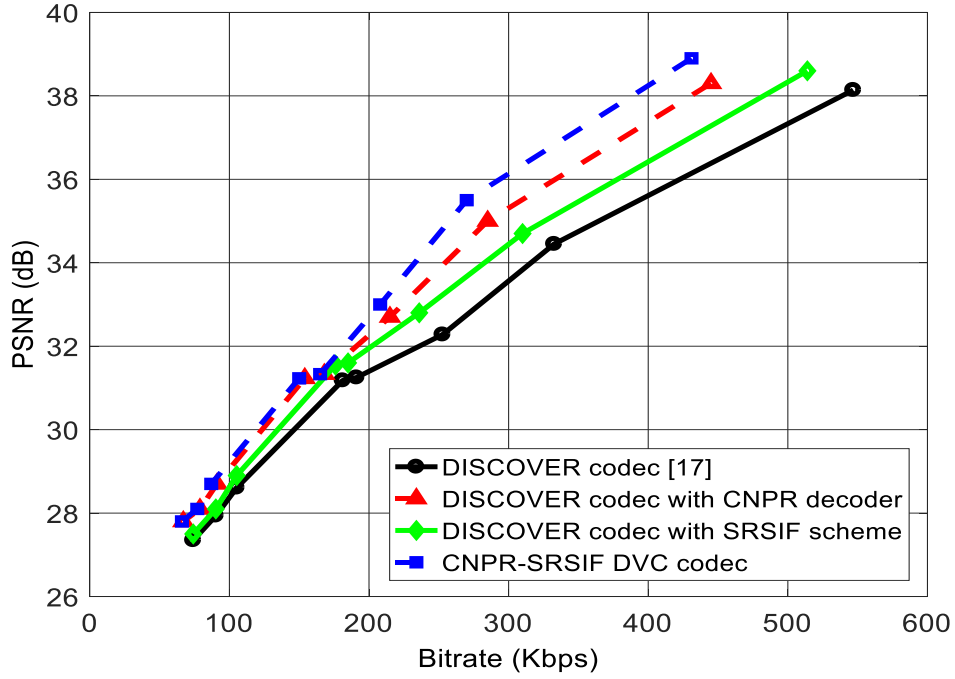


Figure 4.18 RD performance of the DVC codecs for the *Soccer* sequence (GOP size: 4)

(CNPR-SRSIF DVC codec) is improved over that of a DVC codec modified by the inclusion of the CNPR decoder or the SRSIF scheme alone, particularly in video sequences with complex motions.

4.6 Summary

In this chapter, the problem of enhancing the quality of the side information frames in a DVC codec has been investigated with a view to improve its rate-distortion performance and coding efficiency. To this end, a scheme for successive refinements of a side information frame, called the successive refinements of side information frame (SRSIF) has been proposed. In this scheme, the side information frame is refined after decoding each DCT band of a Wyner-Ziv frame. Then, the refined side information frame is used to decode the next DCT band. The performance of the DVC codec with the proposed algorithm for refining the side information has been experimentally studied using the *Hall*, *Coastguard, Foreman* and *Soccer* sequences. It has been shown that the DISCOVER codec with the proposed side information refinement scheme exhibits a rate-distortion performance that is superior to that of the original DISCOVER codec and DVC codec modified by the SIR refinement scheme, particularly for sequences having fast and complex motions. Finally, the proposed side information refinement scheme has been incorporated into the correlation noise parameter refinement (*CNPR)* based decoder proposed in Chapter 3 to obtain the *CNPR-SRSIF* decoder. It has been shown that a DVC codec using this new decoder exhibits an RD performance that is superior to that of a DVC codec using the CNPR decoder or the one using the SRSIF scheme.

CHAPTER 5

Conclusion

5.1 Concluding Remarks

In conventional video codecs, such as MPEG-4 and H264/AVC codecs, the computational complexity of the encoder is 5 to 10 times higher than that of the decoder, since a computationally exhaustive motion estimation and compensation algorithm is employed in the former. In some emerging applications, it is preferred to have a lightweight encoding even at the expense of a highly complex decoder. Distributed video coding or Wyner-Ziv video coding is a video coding paradigm aimed to meet this requirement. In this video coding scheme, the complexity is shifted from the encoder to the decoder by avoiding the computationally intensive temporal prediction and estimation at the encoder and exploiting the temporal redundancy only at the decoder.

There are some issues in the present distributed video coding schemes that need to be addressed before making the idea of distributed video coding a viable practical solution. One of the most important issues with distributed video coding is that the rate-distortion performance and coding efficiency is not on par with that of the conventional video coding.

The estimation of the correlation noise distribution parameter and the quality of the side information frames are known to have a significant influence on the rate-distortion

performance and coding efficiency of distributed video coding. This thesis has been concerned with a study of accurate estimation of the correlation noise model parameters and enhancement of the quality of the side information from the view point of improving the rate-distortion performance of distributed video coding.

Since the correlation noise is not stationary, and also the decoder does not have access to the original WZ frames from the encoder, estimation of the correlation noise model parameter is a challenging task. Moreover, since the motions between the successive key frames are nonlinear, generating a high-quality side information frame is also a difficult task. In view of these challenges, new schemes for improving the rate distortion performance need to be devised which take advantage of the higher degree of information on a WZ frame that progressively becomes available as the decoding of the various DCT bands continues. The work of this thesis is aimed to focus on this philosophy.

In the first part of this thesis, a new correlation noise parameter refinement (CNPR) based decoder has been developed. In the proposed decoder, successive refinements of the correlation noise parameters and simultaneous decoding of all the bitplanes of a DCT band in a Wyner-Ziv frame are carried out in a recursive manner. This process is performed on an augmented factor graph using a new recursive message passing algorithm. Unlike most of the DVC schemes, in which the parameter of the correlation noise distribution is obtained before decoding each DCT band of the WZ frame, in our proposed decoder, the estimation of the correlation noise parameter has been refined during the decoding of each DCT coefficient band. It has been shown through experimentation that the DVC codecs modified by the proposed CNPR decoder exhibit a

rate-distortion performance superior to that of the original DVC codecs, particularly on sequences with fast and complex motions.

In the second part of this thesis, a scheme for successive refinements of a side information frames (SRSIF) corresponding to the WZ frame has been proposed. In this scheme, first, the reconstructed DCT coefficients in all the previously decoded DCT bands of the WZ frame are used to modify the backward and forward motion compensated frames and then, a motion estimation based on block matching is performed to generate a refined side information. This procedure of refinement is carried out successively after decoding each DCT band of the WZ frame. Extensive simulations have been carried showing that a DVC codec using the proposed scheme of refinement of the side information frames exhibits a rate-distortion performance that is superior to that of the DISCOVER codec and the DVC codec modified by the SIR refinement scheme. Finally, the proposed refinement scheme has been incorporated into the *CNPR* decoder to obtain a decoder designated as the CNPR-SRSIF decoder. It has been shown that a DVC codec using this decoder exhibits an RD performance better than that of a DVC codec using the CNPR decoder or SRSIF scheme alone.

In this thesis, efficient schemes have been developed for the refinements of the correlation noise parameters and the side information frames corresponding to the WZ frames of a video sequence with a view of improving the rate-distortion performance of the transform-domain Stanford-based mono-view DVC codes. These schemes, however, are general, in that they can be used for improving the rate-distortion performance of other DVC codecs such as multi-view DVC codec.

Improvements in the rate-distortion performance could also be investigated by focusing on the channel codes that could be specifically designed for DVC codecs. Also, this channel codes may be designed to work well under high compression ratios with their performance close to the channel capacity bound i.e. Shannon limit.

5.2 Scope for Further Investigation

The work of this thesis has been concerned on improving the rate-distortion performance of DVC codec by focusing on the estimation of the noise correlation parameter and refinements of the side information frames. There are number of ways these ideas can be further refined or applied to other related problems.

In the proposed scheme for the side information frame refinements to improve the rate-distortion performance, the refinements begin with that of the initial side information frame. First DCT band of a WZ frame requires the largest number of bits to be decoded and the initial side information frame has an immediate impact on this decoding. Hence, the quality of the side information frame greatly influences the rate-distortion performance of a DVC codec. Moreover, the quality of the refined side information is very much dependents on the quality of initial side information frame. Hence, more studies should be undertaken to generate an initial side information frame of an optimum quality so as have the greatest impact on the rate-distortion performance of a DVC codec.

The proposed method for refinement of correlation noise parameters during decoding of a WZ frame in DVC could be adapted for the general channel coding problem. In this

problem, the proposed scheme can be employed for the estimation and refinement of the parameter of noise distribution, which varies with time in non-stationary channels, in order to improve the decoding efficiency of LDPC decoder and consequently to reduce the bit error rate.

The ideas of accurate estimation of the correlation noise parameters and the generation of the high-quality side information for improving the rate-distortion performance have been investigated in this thesis in the context of mono-view distributed video coding. These ideas can be further explored in the case of multi-view distributed video coding problems.

REFERECES

[1] O. Avaro, A. Eleftheriadis, C. Herpel, G. Rajan and L. Ward, "MPEG-4 systems: overview," *Signal Processing: Image Communication*, *15*(4), pp.281-298, Jan 2000.

[2] T. Wiegand, G.J. Sullivan, G. Bjontegaard and A. Luthra, "Overview of the H. 264/AVC video coding standard,". *IEEE Transactions on circuits and systems for video technology*, vol. *13,* no. 7, pp.560-576, July 2003.

[3] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," in *Proc. of the IEEE*, vol. *93,* no. 1, pp.71-83, Jan 2005.

[4] F. Dufaux, W. Gao, S. Tubaro, and A. Vetro, "Distributed video coding: trends and perspectives," *EURASIP Journal on Image and Video Processing*, *2009*(1), p.508167.

[5] F. Pereira, L. Torres, C. Guillemot, T. Ebrahimi, R. Leonardi and S. Klomp, "Distributed video coding: selecting the most promising application scenarios," *Signal Processing: Image Communication*, vol. *23*, no. 5, pp.339-352, Jun 2008.

[6] [Online] available at : " https://www.lorextechnology.com/hd-dvr-security-system/surveillance-system-with-wired-and-wireless-security-cameras/LW422W-1-p"

[7] N. Imran, B.-C. Seet, A. C. M. Fong, "Distributed video coding for wireless video sensor networks: A review of the state-of-the-art architectures", *SpringerPlus*, vol. 4, pp. 513, Dec. 2015.

[8] D. G. Costa, I. Silva, L. A. Guedes, F. Vasques, P. Portugal, "Availability issues in wireless visual sensor networks", *Sensors*, vol. 14, no. 2, pp. 2795-2821, February 2014.

[9] S.S Pradhan and K. Ramchandran, "Distributed source coding: Symmetric rates and applications to sensor networks," in *proc. of IEEE Data Compression Conference. DCC 2000*, March 2000, pp. 363-372.

[10] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources". *IEEE Transactions on information Theory*, vol. 19, no.4, pp.471-480. July 1973

[11] 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,* no. 1, pp.1-10, Jan 1976

[12] D. Varodayan, Y.C. Lin, and B. Girod, “Adaptive distributed source coding,” *IEEE Transactions on Image Processing*, vol. *21*(5), pp.2630-2640, May 2012

[13] C. Brites and F. Pereira, “Distributed video coding: bringing new applications to life,” In *proc. of 5th Conference on Telecommunications-ConfTele*. April 2005

[14] R. Puri and K. Ramchandran, “PRISM: A new robust video coding architecture based on distributed compression principles,” in *Proc. of the annual allerton conference on communication , control and computing*, Vol. 40, No. 1, October 2002, pp. 586-595.

[15] A. Aaron and B. Girod, “Compression with side information using turbo codes”, In *proc. of* IEEE *Data Compression Conference,*, 2002, pp. 252-261

[16] A. Aaron, S. Rane, and B. Girod, “Transform domain Wyner-Ziv codec for video,” in *Proc. of* SPIE Visual Commun. Image Process, Jan 2004, pp. 520–528, Santa Clara, CA

[17] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: architecture, techniques and evaluation," in *proc. of Picture Coding Symposium (PCS),* pp. 1-4 , Lisbon, Portugal, November 2007.

[18] Vijay Kumar Kodavalla, and P.G. Krishna Mohan, "Distributed Video Coding: Codec Architecture and Implementation," *An International Journal in Signal & Image Processing*, vol. 2, no. 1, pp. 151-163, March 2011

[19] D. Varodayan, A. Aaron and B. Girod, “Rate-adaptive codes for distributed source coding”, *Signal Processing*, *86*(11), pp.3123-3130, November 2006

[20] Z. Yang, S. Li, H. Feng, T. Honold and G. Yu, “Cross-layer iterative decoding of irregular ldpc codes using cyclic redundancy check codes,” in *proc*. of IEEE *Wireless Communications and Networking Conference,* April 2009, pp. 1-6.

[21] D. Kubasov, J. Nayak, and C. Guillemot, “Optimal reconstruction in Wyner-Ziv video coding with multiple side information,” in *Proc*. *9th Workshop on Multimedia Signal Processing MMSP,* October 2007, pp. 183-186,

[22] C. Brites, “Exploiting Correlation Noise Modeling in Wyner-Ziv Video Coding,” Instituto Superior Técnico (IST), 2011

[23] Y.M.Taheri, M.O. Ahmad and M.N.S. Swamy, “A study on compression rate bounds in distributed video coding based on correlation noise models,”, *in Proc. of IEEE International Symposium on Circuits and Systems (ISCAS),* , May 2016, pp. 2691-2694.

[24] V. Toto-Zarasoa, A. Roumy, and C. Guillemot, “Source modeling for distributed video coding,” *IEEE Trans. Circuits Syst. Video Techn*., vol. 22, no. 2, pp. 174–187, Feb. 2012

[25] J. Park, B. Jeon, D. Wang, A. Vincent, "Wyner–Ziv video coding with region adaptive quantization and progressive channel noise modeling", in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)* , May 2009, pp. 1-6.

[26] J. Škorupa, J. Slowack, S.Mys, , N. Deligiannis, , J. De Cock, , P.Lambert, A. Munteanu, and R. Van de Walle, “Exploiting quantization and spatial correlation in virtual-noise modeling for distributed video coding”. *Signal Processing: Image Communication*, vol *25*, no. 9, pp.674-686. October 2010

[27] A. Trapanese, M. Tagliasacchi, S. Tubaro, J. Ascenso, C. Brites, and F. Pereira, “Improved correlation noise statistics modeling in frame-based pixel domain Wyner-Ziv video coding,” in *Proc*. *of* VLBV, Sep. 2005, pp. 1–4, Sardinia, Italy,

[28] A. Aaron, S. Rane, and B. Girod, “Wyner-Ziv video coding with hashbased motion compensation at the receiver,”in *Proc. of IEEE International Conference on Image Processing*, Singapore, vol. 5, Oct. 2004, pp. 3097–3100,

[29] C. Brites, J. Ascenso and F. Pereira, “Studying temporal correlation noise modeling for pixel based Wyner-Ziv video coding,” in *Proc*. of *IEEE International Conference on Image Processing (ICIP2006)*, October 2006 , pp. 273-276.

[30] 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,* no. 9, pp. 1177-1190, September 2008.

[31] X. Huang and S. Forchhammer, “Improved virtual channel noise model for transform domain Wyner-Ziv video coding,”, in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2009.* April 2009, pp. 921-924.

[32] G.R. Esmaili and P.C. Cosman, , “Correlation noise classification based on matching success for transform domain Wyner-Ziv video coding,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP2009,* April 2009 pp. 801-804.

[33] G.R. Esmaili and P.C. Cosman, “Wyner–Ziv video coding with classified correlation noise estimation and key frame coding mode selection,”, *IEEE Transactions on Image Processing*, vol. *20,* no.9, , pp.2463-2474, September 2011.

[34] X. Fan, O.C. Au and N.M. Cheung, "Adaptive correlation estimation for general Wyner-Ziv video coding," in *Proc. of 16th IEEE International Conference on Image Processing (ICIP)*, November 2009, pp. 1409-1412,

[35] X. Huang and S. Forchhammer, "Cross-band noise model refinement for transform domain Wyner–Ziv video coding,", *Signal Processing: Image Communication*, vol. *27,* no.1, pp.16-30, Jan 2012.

[36] H. Van Luong and X. Huang, X. "Parallel iterative decoding of transform domain Wyner-Ziv video using cross bitplane correlation". in *proc. of 18th IEEE International Conference on Image Processing (ICIP2011)*, September 2011, pp. 2633-2636.,

[37] L. Stankovic, V. Stankovic, S. Wang and S. Cheng, "Correlation estimation with particle-based belief propagation for distributed video coding,"*IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP2011),* May 2011*,* pp. 1505-1508.

[38] S. Wang, L. Cui, L. Stankovic, V. Stankovic and S. Cheng, "Adaptive correlation estimation with particle filtering for distributed video coding,"*IEEE Transactions on Circuits and Systems for Video Technology*, *22*(5), pp. 649-658, May 2012

[39] J. Ascenso, C. Brites and F. Pereira, "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding," in *proc. of 5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services*, pp. 1-6., July 2005

[40] T.N. Dinh, G.Lee, , J.Y. Chang, and H.J.Cho, "A novel motion compensated frame interpolation method for improving side information in distributed video coding,". in *proc. of International Symposium on Information Technology Convergence, ISITC 2007.*, November 2007, pp. 179-183.

[41] R. Liu, Z. Yue, C. Chen, "Side information generation based on hierarchical motion estimation in distributed video coding", *Chin. J. Aeronautics*, vol. 22, no. 2, pp. 167-173, April 2009.

[42] S. Argyropoulos, , N.Thomos, N.V. Boulgouris, and M.G. Strintzis,. "Adaptive frame interpolation for Wyner-Ziv video coding," in *proc. of IEEE 9th Workshop on Multimedia Signal Processing, 2007.* October 2007, pp. 159-162.

[43] J. Ascenso and F. Pereira, "Advanced side information creation techniques and framework for Wyner–Ziv video coding,". *Journal of Visual Communication and Image Representation*, vol. *19,* no. 1, pp.600-613, Dec 2008.

[44] H. V. Luong, L. L. Raket, X. Huang, S. Forchhammer, "Side information and noise learning for distributed video coding using optical flow and clustering," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4782-4796, Dec. 2012.

[45] Y.M. Taheri, , M.O. Ahmad and M.N.S. Swamy, "Side information generation using optical flow and block matching in Wyner-Ziv video coding," in *proc. of 21st IEEE International Conference on Electronics, Circuits and Systems (ICECS),* Dec 2014, pp. 722-725.

[46] A. Aaron, S.Rane, and B. Girod. "Wyner-Ziv video coding with hash-based motion compensation at the receiver." in *Proc. of IEEE International Conference on Image Processing, ICIP'04*, vol. 5*,* October *2004,* pp. 3097-3100,

[47] J. Ascenso and F. Pereira, "Adaptive hash-based side information exploitation for efficient Wyner-Ziv video coding". in *Proc. of IEEE International Conference on Image Processing, (ICIP 2007)*, Vol. 3, September 2007, pp. III-29.

[48] C. Yaacoub, J. Farah, J. and B. Pesquet-Popescu, "Improving hash-based Wyner-Ziv video coding using genetic algorithms," in *Proc. of the 5th International ICST Mobile Multimedia Communications Conference*, September 2009, p. 30.

[49] D. Varodayan, D. Chen, M. Flierl and B. Girod, "Wyner–Ziv coding of video with unsupervised motion vector learning,". *Signal Processing: Image Communication*, vol. *23,* no. 5, pp.369-378, Jun 2008

[50] R. Martins, C. Brites, J. Ascenso and F. Pereira, "Refining side information for improved transform domain Wyner-Ziv video coding," *IEEE Transactions on circuits and systems for video technology*, vol. 19, no. 9 , pp.1327-1341, September 2009

[51] A. Aaron and B. Girod, "Compression with side information using turbo codes" in *Proc* of *IEEE Data Compression Conference*, April 2002, pp. 252–261.

[52] A.D. Liveris, Zixiang Xiong, and C.N. Georghiades, "Compression of binary sources with side information using low-density parity-check codes," in *Proc. of* Global Telecommunications Conference, vol. 2, November 2002, pp. 1300–1304,.

[53] R. G. Gallager, "Low-Density Parity-Check Codes". Cambridge, MA: MIT Press, 1963.

[54] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low-density parity-check codes," Electron. Lett., vol. 32, pp. 1645–1646, Aug. 1996.

[55] H. Wang, Y. Zhao and A. Wang, "Performance Comparisons of Different Channel Codes in Distributed Video Coding," in *Proc. of International Conference*

on Innovative Computing, Information and Control. ICICIC'06. Vol. 2, August 2006, pp. 225-228.

[56] B.Li, Y.Wang, , Q., Huang, and Y. Liu, "An LDPC-based improved decoding scheme for distributed video codec," in *Proc. of 18th International Conference on Telecommunications (ICT), 2011*, May 2011 pp. 298-303.

[57] T. Murayama, "Statistical mechanics of linear compression codes in network communication." *Europhys. Lett,* 2001

[58] A. Liveris, Z. Xiong, and C. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes", IEEE Commun. Lett., vol. 6, no. 10, pp.440 -442 , 2002

[59] D.Varodayan, A. Aaron, and B.Girod, "Rate-adaptive codes for distributed source coding". *Signal Processing*, vol *86*, 11, pp.3123-3130, November 2006.

[60] Y.M .Taheri, M.O. Ahmad, and M.N.S.Swamy, "Joint noise distribution parameter estimation and LDPC decoding using variational Bayes," in *Proc. of IEEE 57th International Midwest Symposium on Circuits and Systems (MWSCAS),* August 2014, pp. 809-812.

[61] Y.M. Taheri, M.O. Ahmad, and M.N.S. Swamy, "A joint correlation noise estimation and decoding algorithm for distributed video coding". Springer journal of *Multimedia Tools and Applications*, pp.1-29. April 2017

[62] T-Z. Velotiaray, A. Roumy, C. Guillemot, "Maximum likelihood BSC parameter estimation for the Slepian-Wolf problem" . *IEEE Commun Lett*, pp 232–234, February 2011

[63] A. Zia A, JP. Reilly, S. Shahram "Distributed parameter estimation with side information: a factor graph approach" in *Proc. of* IEEE International Symposium of Information Theory, Jun 2007, pp 2556–2560

[64] N. Deligiannis, A. Munteanu, S. Wang, S. Cheng and P. Schelkens,. "Maximum likelihood Laplacian correlation channel estimation in layered Wyner-Ziv coding," *IEEE Transactions on Signal Processing*, vol. *62*, no.4, pp.892-904, February 2014

[65] AJ. Haug, "A tutorial on Bayesian estimation and tracking techniques applicable to nonlinear and non-Gaussian Processes". *The Mitre Corporation*, McLean, Virginia, pp 1–52, 2005

[66] P. Carbonetto, M. King, and F. Hamze. A stochastic approximation method for inference in probabilistic graphical models. In NIPS, volume 22, pages 216–224. Citeseer, 2009.

[67] C.Bishop "Pattern recognition and machine learning", 2006

[68] C. Fox, S. Roberts, "A tutorial on variational Bayesian inference". Artif Intell Rev:1–13, 2011

[69] Joint Video Team (JVT) reference software. [Online]. Available at: "http://iphome.hhi.de/suehring/tml/ index.htm"

[70] Bjøntegaard G, "Calculation of average PSNR differences between RD curves", Tech. Rep., 13th VCEGM33 Meeting. Austin, 2001

[71] Y.M .Taheri, M.O. Ahmad, and M.N.S.Swamy , "Successive refinements of side information frames in distributed video coding," Springer journal of *Multimedia Tools and Applications*.(Submitted)