Adaptive Distributed Video Coding with Correlation Estimation using Expectation Propagation

Lijuan Cui[a], **Shuang Wang**[*,b], **Xiaoqian Jiang**[b], and **Samuel Cheng**[a]

[a]School of Electrical and Computer Engineering, University of Oklahoma, Tulsa, OK

[b]Division of Biomedical Informatics, University of California, San Diego, San Diego, CA

Abstract

Distributed video coding (DVC) is rapidly increasing in popularity by the way of shifting the complexity from encoder to decoder, whereas no compression performance degrades, at least in theory. In contrast with conventional video codecs, the inter-frame correlation in DVC is explored at decoder based on the received syndromes of Wyner-Ziv (WZ) frame and side information (SI) frame generated from other frames available only at decoder. However, the ultimate decoding performances of DVC are based on the assumption that the perfect knowledge of correlation statistic between WZ and SI frames should be available at decoder. Therefore, the ability of obtaining a good statistical correlation estimate is becoming increasingly important in practical DVC implementations. Generally, the existing correlation estimation methods in DVC can be classified into two main types: pre-estimation where estimation starts before decoding and on-the-fly (OTF) estimation where estimation can be refined iteratively during decoding. As potential changes between frames might be unpredictable or dynamical, OTF estimation methods usually outperforms pre-estimation techniques with the cost of increased decoding complexity (e.g., sampling methods). In this paper, we propose a low complexity adaptive DVC scheme using expectation propagation (EP), where correlation estimation is performed OTF as it is carried out jointly with decoding of the factor graph-based DVC code. Among different approximate inference methods, EP generally offers better tradeoff between accuracy and complexity. Experimental results show that our proposed scheme outperforms the benchmark state-of-the-art DISCOVER codec and other cases without correlation tracking, and achieves comparable decoding performance but with significantly low complexity comparing with sampling method.

Keywords

Belief propagation; Expectation propagation; Distributed video coding; Adaptive decoding

1. INTRODUCTION

Nowadays, many-to-one digital video applications are becoming more and more popular, such as video surveillance with multiple tiny cameras, where each tiny camera has only limited communication bandwidth, computational power and battery life. Therefore, it is extremely crucial to restrict the encoding complexity to reduce the power consumption, as well as guarantee the encoding performance to save the communication bandwidth. Driven by these emerging applications, the industry is anxious for an entirely new coding paradigm,

which could significantly reduce the encoding complexity, even though the expense is to increase decoding complexity at the server side. Although the traditional video coding standard (e.g., H.264) is very promising in a centralized setup, it could not be easily tailored to fit such a new coding paradigm, as the encoder of H.264 standard has large computational complexity due to the existence of motion estimation. Fortunately, distributed video coding (DVC)[1, 2] provides a workaround for these difficulties, where the complexity could be efficiently shifted from the encoder side to the decoder side. The DVC technique is based on the distributed source coding (DSC) principle brought a paradigm shift from the conventional centralized video coding architecture to a totally distributed manner, where the computationally-expensive motion compensation and correlation extraction procedures will be taken at the decoder side.

From the information-theoretic perspective, DSC refers to separate compression and joint decompression of multiple correlated sources. DSC started as an information-theoretic problem in the renowned 1973 paper of Slepian and Wolf.[3] Slepian and Wolf considered the lossless compression of two physically separated sources, and demonstrated that, roughly speaking, there is no performance loss compared to joint compression as long as joint decompression is performed. In 1976, Wyner and Ziv[4] considered a lossy version (i.e., with a distortion constraint) of the asymmetric Slepian-Wolf (SW) problem known as Wyner-Ziv (WZ) coding, where one source is available at the decoder as side information (SI) (e.g, through entropy coding). Wyner and Ziv showed that for some particular correlation models (e.g. Gaussian, Laplace, etc.), there is no performance loss due to the absence of SI at the encoder.

DVC exploits WZ coding principles by performing computationally-expensive motion compensation at the decoder instead of at the encoder. The beauty of WZ coding (DSC in general) is that there is no need for the encoder to be aware of the SI, which makes it possible to accomplish predictive coding without encoder motion compensation. In a nutshell, a block of pixels/coefficients in a certain video frame (a.k.a., WZ frame) could be efficiently WZ encoded into a stream (i.e., syndromes) without any reference to any other video frames. To recover a WZ frame at the decoder side, a SI frame will be first generated based on the received key frames through motion compensation, where key frames can be decoded independent of other frames. Then the WZ decoder could decompress the WZ frame based on the received syndromes and the generated SI through DSC principle. The state-of-the-art WZ coding designs based on turbo, LDPC, and other graph-based codes have been widely used in DVC studies (see[5–7] and references therein).

Note that a key difference between conventional WZ coding and DVC is that the correlation statistics among sources in the former case is usually assumed to be known as a constant at both the encoder and decoder. In DVC, however, such assumption is normally far-fetched, as correlation statistics between WZ and SI frames would be unknown and dynamically change over time and the location of pixels/coefficients, no matter how well SI frame is generated. Indeed, due to the non-stationarity of real scenes, WZ coding in DVC has to deal with varying correlation noise statistics. Therefore, estimating correlation statistics has been identified as one of key challenges in DVC.

Most DVC designs so far (with few exceptions) usually tackle the problem by modeling correlation noise, i.e., the difference between the WZ and SI frames, as a Laplace distribution parametrized by the correlation parameter λ. The non-stationarity of between scenes can be dealt mainly by estimating correlation parameter λ (e.g., on the block or frame level) based on previously decoded frames,[8–12] which usually refers to the pre-estimation mode as the estimation starts before decoding. However, in reality, pre-estimation mode itself could not guarantee the precise estimate of correlation parameter, as

potential changes between frames (especially for high motion sequences) might be unpredictable. Therefore, a more practical approach (i.e., on-the-fly (OTF) correlation estimation mode) is to refine the pre-estimated correlation iteratively with decoding, where the currently decoded information can be used to improve the estimate and vice versa.

In this paper, we propose an efficient way for handling OTF correlation estimation between frames by augmenting the factor graph model of belief propagation (BP) based decoder to include correlation variable nodes and incorporating deterministic approximation (i.e., expectation propagation (EP)) algorithm to tackle the intractability in calculating the exact posterior probability (i.e., belief) of each correlation variable node. The proposed adaptive DVC framework with EP based OTF correlation estimator is carried out in pixel-domain with a feedback channel for rate-adaptive decoding using joint bit-plane LDPCA code.[13]

This paper's contributions can thus be summarized as:

1. We carefully constructed a single factor graph that connects BP based joint bit-plane SW decoder and EP based correlation estimators together for enabling the OTF correlation estimation between SI and WZ frame.

2. We modeled the correlation parameter λ in the context of Gamma distribution and tackled the intractability in calculating the exact posterior distribution of λ through EP based deterministic approximation method.

3. As one of deterministic approximation method, the proposed EP based OTF correlation estimation significantly reduced the computational complexity compared with sampling based methods.[14, 15]

The paper is organized as follows. In Sections 2, we outline the previous work of correlation estimation and discusses the correlation model used in DVC. The proposed adaptive DVC framework including factor graph construction and message passing rules is explained in Section 3. Moreover, in Section 4, we derived the EP based correlation estimation scheme. In Section 5, we verified the proposed framework through experimental results built on a pixel-based DVC setup. The paper concludes with Section 6.

2. RELATED WORK OF CORRELATION ESTIMATION

Since the capability of correlation parameter estimation has strong impact on the WZ video coding efficiency, many research work has been done for improving correlation estimation in the literature. At the beginning, most WZ video coding schemes assume that the correlation noise statistics are stationary along both in time and space,[5, 16, 17] where the correlation noise statistics could be obtained through training video sequences. However, the aforementioned assumption and estimation methods have many limitations, as the correlation statistics strongly depends on the video contents and may vary with both time and space. To bridge this gap, non-stationary correlation models (e.g., on the block or frame level) are studied in papers.[9–12, 18, 19]

In paper,[9] the correlation is modeled as Laplace distribution, but to capture the non-stationary nature of the scene, the correlation parameter was varied from pixel to pixel. The noise power increases if the pixel difference between motion compensated blocks in the two key frames used to generate side information is high; otherwise, it decreases. The reason behind this method is that if the difference between the two key frames is high then we have less confidence in their average and the noise variance is higher. Thus, incorporating this model within SW decoding ensures that the channel code (employed for SW decoding) assigns higher reliability to pixels that have been predicted with higher accuracy, that is, the difference between the key frames is smaller. Similarly, in paper,[10] the Laplace distribution

is also used with the parameter pre-estimated at the sequence, frame, block, and pixel level from decoded frames at the decoder. Improved pre-estimated channel estimators[11, 12] are proposed that attempt to address the difficulty in adaptive correlation in smaller spatial regions due to the difficulty in acquiring sufficient statistics. The above correlation models determine their parameters based on the noise realization in a given temporal or spatio-temporal neighborhood. In paper,[19] a side information dependent correlation noise model is proposed where the standard deviation of the Laplacian model is a function of a particular realization of the side information at each pixel position.

Note that, non-stationarity of a scene[8–12, 19] is addressed by changing correlation model in advance of decoding and supplying the SW decoder with different initial reliability estimates. However, once SW decoding starts (e.g., via BP), the correlation is fixed.

Since the SW decoding process refines starting beliefs, our prior works[14, 15] demonstrated that unifying the process of correlation estimation using sampling method and joint bit-plane decoding into a single joint process (i.e., OTF estimation mode) can provide better statistics estimate and consequently improved performance for both pixel- and transform-domain DVCs. Additionally, this unification of correlation estimation and SW decoding will also enable the correlation estimator to take into account side information statistics and any of the methods of[8–12] can be used as an initial point that will be refined during SW decoding. While OTF estimation is also discussed in our prior works,[14, 15] our prior schemes were based on the sampling method for dynamically tracking the correlation statistics, which result a large computational complexity at the decoder side. Instead of incorporating particle filtering method into BP algorithm for correlation tracking,[14, 15] an ultra-low complexity alternative (i.e. EP algorithm) is studied and proposed in this paper. Before introducing the proposed EP based framework, the Laplacian correlation model will be first described in the next subsection.

2.1 Laplacian Correlation Model

The correlation between the source and side information (SI) frames is modeled as a virtual communication channel which can be expressed in the form $X = Y + N$, where X is the source frame to be recovered, Y is the SI frame and N is the virtual channel noise. Based on experimental observations, most DVC designs so far[10, 20, 21] (with few exceptions) model the correlation noise as Laplace distribution as follows:

$$p[WZ(x, y) - SI(x, y)] = \frac{\lambda}{2}\exp[-\lambda|WZ(x, y) - SI(x, y)|], \quad (1)$$

where $WZ(x, y)$ and $SI(x, y)$ are the pixel values at the location (x, y) in WZ and SI frames, respectively, $p(\cdot)$ denotes the probability density function. Here, λ is the Laplace distribution parameter defined as

$$\lambda = \sqrt{\frac{2}{\sigma^2}} \quad (2)$$

where, σ^2 is the variance of residuals between WZ and SI frames. Moreover, the correlation parameter λ can vary along both time and space, since the residual errors are usually large, when there are high motions between frames or illumination changes within a frame.

3. SYSTEM ARCHITECTURE

To precisely catch the correlation between frames while recovering source frames, we proposed an adaptive DVC framework. In Bayesian perspective, capturing correlation

corresponds to estimating the posterior distribution of correlation parameter. Since a factor graph, as a particular type of graphical model, enables efficient computation of marginal distributions through message passing algorithm, our proposed framework is carried out on the factor graph as shown in Fig. 1. The key steps of the proposed adaptive DVC framework can be outlined as follows: 1) factor graph construction: design a factor graph with appropriately defined factor functions to capture and connect SW coding and correlation tracking (see Section 3.1); 2) message passing algorithm implementation: perform message passing algorithm on the constructed factor graph to calculate the posterior distribution of interested variables (see Section 3.2).

3.1 Factor graph construction

DVC, a video compression technology based on DSC principle, is usually implemented on a factor graph utilizing WZ coding scheme. Compared with standard DVC, the factor graph (see Fig. 1) of the proposed adaptive DVC with correlation tracking consists of two regions, where Region I refers to the correlation parameter tracking and Region II corresponds the traditional WZ coding. In Fig. 1, variable nodes (usually depicted by a circle) denote unknown variables such as coded bits, correlation parameter, and factor nodes (depicted by small squares) represent the relationship among the connected variable nodes.

3.1.1 Joint bit-plane SW coding (Region II)—WZ coding, a.k.a. the lossy version of SW coding, is usually realized by quantization followed by SW coding of the quantized indices based on channel coding.[22] Here, for WZ coding, we carry out LDPC based joint bit-plane SW coding after performing quantization, the factor graph of which is described as Region II in Fig. 1.

Note that we suppose a N-length source sample x_i, $i = 1, \cdots, N$ is quantized into $Q[x_i]$ using 2^q levels quantization, where $q = 3$ is taken as an example in Region II of Fig.1. We denote $x_i^1 x_i^2, \cdots, x_i^q$ as the binary format of the quantization index $Q[x_i]$, and denote $\mathbf{B} = x_1^1, x_1^2, \cdots, x_1^q, x_2^1, x_2^2, \cdots, x_N^q$ as the block which combines all the bit variables together. The block $\mathbf{B}$ is then encoded using LDPC-based SW codes and generates an M-length syndrome bits $\mathbf{S} = s_1, s_2, \cdots, s_M$, which results in a $qN : M$ SW compression ratio.

Similar to the standard LDPC decoding, the factor nodes $f_1, f_2, \cdots, f_M$ in Region II take into account the constraints imposed by the received syndrome bits. Thus, the factor function of factor node f_a, $a = 1,\ldots, M$ is defined as

$$f_a(\tilde{\mathbf{x}}_{\mathbf{a}}, s_a) = \begin{cases} 1, & \text{if } s_a \oplus \bigoplus \tilde{\mathbf{x}}_{\mathbf{a}} = 0, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

where $\tilde{\mathbf{x}}_{\mathbf{a}}$ denotes the set of neighbors of factor node f_a, and $\oplus \tilde{\mathbf{x}}_{\mathbf{a}}$ denotes the binary sum of all elements of the set $\tilde{\mathbf{x}}_{\mathbf{a}}$.

Let a N-length sample y_i, $i = 1, \cdots, N$, the realizations of variable nodes Y_i, be the side information available at the decoder. The factor nodes g_i are introduced in the factor graph to capture the correlation constraints as shown in (1) between source x_i and side information y_i for SW decoding. Since source samples are first passed through a quantization process, the correlation constraints (i.e., the factor function of g_i) between the quantized indices $Q[x_i]$ and the side information y_i can be expressed as:

$$g_i(Q[x_i], y_i, \lambda) = \int_{P(Q[x_i])}^{P(Q[x_i]+1)} \frac{\lambda}{2} e^{\lambda|x-y_i|} dx, \quad (4)$$

where λ is the correlation parameter of the Laplace distribution, $P(\bullet)$ denotes the lower boundary of quantization partition at index "$\bullet$", e.g. if a coefficient x_i satisfies $P(\bullet) \quad x_i < P(\bullet + 1)$), the quantization index $Q[x_i]$ of coefficient x_i is equal to "$\bullet$". Actually, given a parameter λ, the factor node g_i plays a role of providing a predetermined likelihood $p(y_i|Q[x_i], \lambda)$ to variable node $X_i^j, j = 1, \cdots, q$ for LDPC based SW decoding.

3.1.2 Correlation parameter tracking (Region I)—As described in Section 2.1, the correlation parameter, denoted by λ_l^t, can vary along both time and space, where the superscript t indicates the time dependence (i.e., frame) and the subscript l corresponds to the location of a single pixel or a group of pixels (i.e., block). In this paper, for the simplicity of notation, we drop the superscript t and denote λ_l as the correlation of the l-th pixel block in a given frame, where each pixel block possesses C number of pixels. Then, the correlation parameter λ in (4) can be replaced by λ_l for different pixel block.

Let us denote by N' the number of pixel blocks within a frame. Then, we introduce additional variable nodes $A_l, l = 1, 2,\ldots, N'$ to represent the correlation parameters λ_l in factor graph (see, Region I of Fig. 1). Since a block of C source samples (i.e., pixels) share the same correlation parameter, every C number of factor nodes g_i in Region II will be connected to the same variable node A_l, where we call C^* as the connection ratio. Moreover, to initialize a prior distribution for correlation parameter λ_l, additional factor nodes $h_l, l = 1, \cdots, N'$ are introduced, where Gamma distribution is assigned to each factor function $h_l(\lambda_l)$ for the mathematical convenience. Then, by implementing message passing rules introduced in the next subsection on the proposed factor graph, each factor node g_i will periodically update the likelihood $p(y_i|Q[x_i], \lambda_l)$ for the corresponding bit variable nodes $X_i^1 X_i^2, \cdots, X_i^q$ when a new estimate of correlation parameter λ_l is available, instead of using a predetermined likelihood $p(y_i|Q[x_i], \lambda)$.

Consequently, by introducing correlation parameter estimation in Region I, likelihood factor function in (4) will be updated as

$$g_i(Q[x_i], y_i, \lambda_l) = \int_{P(Q[x_i])}^{P(Q[x_i]+1)} \frac{\lambda_l}{2} e^{\lambda_l|x-y_i|} dx. \quad (5)$$

3.2 Message passing on the constructed factor graph

In Bayesian inference, message passing algorithm (e.g., BP) on a factor graph offers an very efficient way to calculate the marginal distributions (i.e. beliefs) of the unknown variables represented by their corresponding variable nodes. In the proposed adaptive DVC factor graph (see Fig. 1), we are interested in two unknown variables, which are represented by source variable nodes X_i^j in Region II and correlation parameter variable nodes A_l in Region I, respectively.

In Region II, without considering the connection to the Region I, the factor graph is identical to that of standard LDPC codes with discrete variables x_i^j. Hence, the posterior distribution of x_i^j can be calculated through standard BP algorithm. However, in region I, BP algorithm cannot be applied directly, as the correlation parameter λ_l represented by the variable node

---

*To estimate a stationary correlation parameter, we can set the connection ratio equal to the code length. Moreover, connection ratio provides a trade-off between complexity and spatial variation.

A_l is generally non-Gaussian continuous variable and BP algorithm only handles discrete variable with small alphabets size or continuous variable with linear Gaussian distribution.

To seek a workaround for this difficulty, let us start with the derivation of posterior distribution of the correlation parameter λ_l. According to Bayes' rule and the message passing rule, the posterior distribution of correlation parameter λ_l can be expressed as:

$$
\begin{aligned}
p(\lambda_l|\mathbf{y}_l) \\
&= \frac{1}{Z_l} \prod_{i \in \mathcal{N} \backslash h_l(A_l)} p(\lambda_l) p(y_i|\lambda_l) \\
&= \frac{1}{Z_l} \prod_{i \in \mathcal{N} \backslash h_l(A_l)} \int_{Q[x_i]} p(\lambda_l) p(Q[x_i]) p(y_i|Q[x_i];\lambda_l) \\
&= \frac{1}{Z_l} h(\lambda_l) \prod_{i \in \mathcal{N} \backslash h_l(A_l)} \sum_{\mathbf{x}_i^q} g(y_i;Q[x_i],\lambda_l) \prod_{j \in 1,2,\cdots,q} m_{x_i^j \to g_i}(x_i^j) \\
&= \frac{1}{Z_l} m_{h_l \to A_l}(\lambda_l) \prod_{i \in \mathcal{N} \backslash h_l(A_l)} m_{g_i \to A_l}(\lambda_l),
\end{aligned}
\tag{6}
$$

where Z_l is a normalization constant, $\sum_{\mathbf{x}_i^q}$ denotes a sum over all the bit variables in $\mathbf{x}_i^q$, the value of message $m_{x_i^j \to g_i}(x_i^j)$ is updated iteratively by variable node X_i^j in Region II according to BP update rule, message $m_{h_l \to A_l}(\lambda_l) = h(\lambda_l)$ comes from prior factor node in Region I, and message $m_{g_i \to A_l}(\lambda_l) = \sum_{\mathbf{x}_i^q} g(y_i;Q[x_i],\lambda_l) \prod_{j \in 1,2,\cdots,q} m_{x_i^j \to g_i}(x_i^j)$ comes from likelihood factor node in Region II according to the BP update rule.

So far, we have shown that the posterior distribution of correlation variable λ_l can be expressed as the product of all the incoming messages. In the rest of this subsections, we investigate how to efficiently compute the posterior distribution using BP and EP based approximation algorithms.

3.2.1 Belief propagation—The BP algorithm[23] is an efficient and exact inference algorithm for computing local marginals over variables on tree-structured graphs. For graphs with loops, a lot of applications (e.g. LDPC deocoding[24]) show that BP algorithm (or loopy BP algorithm) still provides a good performance. While this technique is extremely powerful in handling variables of small alphabet sizes, they cannot handle a continuous variable with arbitrary distribution or even a variable of a medium alphabet size as the computational complexities of these algorithms increase exponentially with the alphabet size.

For our problem in (6), since all the bit variables x_i^j, $j = 1, \cdots, q$, in $\mathbf{x}_i^q$ are discrete and taking values 0 or 1, the message $m_{g_i \to A_l}(\lambda_l) = \sum_{\mathbf{x}_i^q} g(y_i;Q[x_i],\lambda_l) \prod_{j \in 1,2,\cdots,q} m_{x_i^j \to g_i}(x_i^j)$ has 2^q terms and the product of all the messages $\prod_{i \in \mathcal{N} \backslash h_l(A_l)} m_{g_i \to A_l}(\lambda_l)$ is a mixture of 2^{qC} number of Laplace distributions, where $C = |\mathcal{N} \backslash h_l(A_l)|$ is the connection ratio, q is the number of bit-planes, and qC can be a large number. Thus, the direct evaluation of the posterior distribution using BP would be infeasible.

3.2.2 Expectation propagation—An approximate inference for solving the problem in (6) is to parametrize the variables through variational inference. Deterministic approximation schemes (e.g. EP[25]) provide some low complexity alternatives based on the analytical approximations to the posterior distribution. For example, suppose that posterior

distribution $p(\theta)$ of parameter θ is infeasible to be calculated directly. If the posterior can be factorized as $p(\theta) = \prod_k g_k(\theta)$, where each factor function $g_k(\theta)$ only depends a small subset of observations, EP solves this difficulty by replacing the true posterior distribution $p(\theta)$ with an approximate distribution $q(\theta) = \prod_k \tilde{g}_k(\theta)$ by sequentially computing each approximate term $\tilde{g}_k(\theta)$ for $g_k(\theta)$. The general workflow of the EP algorithm has been listed in Table 1. In particular, for our problem in (6), EP is used to sequentially compute approximate messages $\tilde{m}_{h_l \to A_l}(\lambda_l)$ and $\tilde{m}_{g_i \to A_l}(\lambda_l)$ in replace of true messages $m_{h_l \to A_l}(\lambda_l)$ and $m_{g_i \to A_l}(\lambda_l)$ in (6), then get an approximate posterior on λ_l by combining these approximations together. The details of correlation parameter estimation through EP for our problem will be discussed in the next section.

4. POSTERIOR APPROXIMATION OF CORRELATION PARAMETER USING EXPECTATION PROPAGATION

In this section, we will derive the proposed EP based correlation estimator, which can provide a fast and accurate way to approximate the posterior distribution on the factor graph as shown in Fig. 1. The procedures of the proposed EP algorithm has been detailed as follows:

1. Initialize the prior term

$$h_l(\lambda_l) = \text{Gamma}(\lambda_l, \alpha_l^0, \beta_l^0) = z_l^0 \lambda_l^{\alpha_l^0 - 1} \exp(-\beta_l^0 \lambda_l) \quad (7)$$

   with $\alpha_l^0 = 2, \beta_l^0 = \dfrac{\alpha_l^0 - 1}{\lambda^0}, z_l^0 = \dfrac{\beta_l^{0 \alpha_l^0}}{\Gamma(\alpha_l^0)}$, where λ^0 is the initial correlation parameter, and β_l^0 and α_l^0 are scale and shape parameters for Gamma distribution, respectively. The selection of the initial values for the above parameters guarantees the mode of prior distribution equals to the initial correlation λ^0.

2. Initialize the approximation term (uniform distribution)

$$\tilde{m}_{g_i \to A_l}(\lambda_l) = \text{Gamma}(\lambda_l, \alpha_{il}, \beta_{il}) = z_{il} \lambda_l^{\alpha_{il} - 1} \exp(-\beta_{il} \lambda_l) \quad (8)$$

   with $\beta_{il} = 0, \alpha_{il} = 1, z_{il} = 1$.

3. Initialize α_l^{new} and β_l^{new} for approximate posterior $q(\lambda_l) = \text{Gamma}(\lambda_l, \alpha_l^{\text{new}}, \beta_l^{\text{new}})$, where $\alpha_l^{\text{new}} = \alpha_l^0 = 2$, and $\beta_l^{\text{new}} = \beta_l^0$.

4. For each variable node λ_l

For each factor node g_i, where $g_i \in \mathcal{N}\lambda_l$)

   a. Remove $\tilde{m}_{g_i \to A_l}(\lambda_l)$ from the posterior $q(\lambda_l)$, we get $q^{\setminus g_i}(\lambda_l) = \text{Gamma}(\alpha_l^{\text{tmp}}, \beta_l^{\text{tmp}})$

$$\alpha_l^{\text{tmp}} = \alpha_l^{\text{new}} - (\alpha_{il} - 1) \quad \beta_l^{\text{tmp}} = \beta_l^{\text{new}} - \beta_{il} \quad (9)$$

   b. Update $q^{\text{new}}(\lambda_l)$ by minimizing the Kullback Leibler (KL) divergence $D(q^{\setminus g_i}(\lambda_l) m_{g_i \to A_l}(\lambda_l) \| q^{\text{new}}(\lambda_l))$ (i.e., performing moment matching (**Proj**)) (see Section 4.1 for detail).

$$q^{\text{new}}(\lambda_l) = \frac{1}{Z_l} \textbf{Proj}[\textbf{q}^{\setminus \textbf{g}_i}(\lambda_l) \textbf{m}_{\textbf{g}_i \to \textbf{A}_l}(\lambda_l)] \quad (10)$$

where $Z_l = \int_{\lambda_l} q^{\backslash g_i}(\lambda_l) m_{g_i \to A_l}(\lambda_l)$.

   c. Set approximated message

$$\alpha_{il} = \alpha_l^{\text{new}} - (\alpha_l^{\text{tmp}} - 1) \quad \beta_{il} = \beta_l^{\text{new}} - \beta_l^{\text{tmp}} \quad z_{il} = Z_l \frac{\beta_l^{\text{new}\alpha_l^{\text{new}}}}{\Gamma(\alpha_l^{\text{new}})} \left(\frac{\beta_l^{\text{tmp}\alpha_l^{\text{tmp}}}}{\Gamma(\alpha_l^{\text{tmp}})} \right)^{-1} \left(\frac{\beta_{il}^{\alpha_{il}}}{\Gamma(\alpha_{il})} \right)^{-1} \quad (11)$$

4.1 Moment matching

Through moment matching, $q(\lambda_l)$ is obtained by matching the mean and variance of $q(\lambda_l)$ to those of $q^{\backslash g_i}(\lambda_l) m_{g_i \to A_l}(\lambda_l)$. Then, we get the updated α_l^{new} and β_l^{new}, the parameters of $q(\lambda_l)$ as follows,

$$\alpha_l^{\text{new}} = m_1 \beta_l^{\text{new}} \quad \beta_l^{\text{new}} = m_1/(m_2 - m_1^2), \quad (12)$$

where m_1 and m_2 are the first and second moments of the approximate distribution as shown below.

$$m_1 = \frac{1}{Z} \sum_{Q[x_i]} (\mathscr{F}_1(z_1) - \mathscr{F}_1(z_2)) \prod_{j=1}^{j=q} m_{x_i^j \to g_i}(x_i^j) \quad m_2 = \frac{1}{Z} \sum_{Q[x_i]} (\mathscr{F}_2(z_1) - \mathscr{F}_2(z_2)) \prod_{j=1}^{j=q} m_{x_i^j \to g_i}(x_i^j) \quad Z = \sum_{Q[x_i]} (\mathscr{F}_0(z_1) - \mathscr{F}_0(z_2)) \prod_{j=1}^{j=q} m_{x_i^j \to g_i}(x_i^j) \quad (13)$$

Here, Z is the normalization term and these unknown functions in (13) can be evaluated according to (14).

$$z_1 = P(Q[x_i]+1)$$
$$z_2 = P(Q[x_i])$$
$$\mathscr{F}_0(z) = A(z) + B(z)$$
$$\mathscr{F}_1(z) = \frac{\alpha^{\text{tmp}}}{\beta^{\text{tmp}}} A(z) + \frac{\alpha^{\text{tmp}}}{\beta^{\text{tmp}} + |z - y_i|} B(z)$$
$$\mathscr{F}_2(z) = \frac{\alpha^{\text{tmp}}(\alpha^{\text{tmp}}+1)}{(\beta^{\text{tmp}})^2} A(z) + \frac{\alpha^{\text{tmp}}(\alpha^{\text{tmp}}+1)}{(\beta^{\text{tmp}} + |z - y_i|)^2} B(z) \quad (14)$$
$$A(z) = \frac{1}{2}(1 + \text{sgn}(z - y_i))$$
$$B(z) = -\text{sgn}(z - y_i) \frac{1}{2} \left(\frac{\beta^{\text{tmp}}}{\beta^{\text{tmp}} + |z - y_i|} \right)^{\alpha^{\text{tmp}}}$$

5. RESULTS

In this section, we employ a pixel-based DVC setup to demonstrate the benefit of the proposed OTF correlation tracking. As in references,[1, 10, 21] group of pictures (GOP) is equal to 2 in our study, where all even frames are treated as WZ frames and all odd frames are considered as key frames. The key frames are conventionally intra-coded, for example, using H.264 Advanced Video Coding (AVC)[26] intra coding mode. WZ frames are first quantized pixel-by-pixel and then all bit-plane of the resulting quantization indices are combined together and compressed using an LDPCA codes. At the decoder, side information frame Y is generated using motion-compensated interpolation of the forward and backward key frames.[1, 21] Spatial smoothing,[21] via vector-median filtering, is used to improve the result together with half-pixel motion search. Each WZ frame is decoded by the

proposed EP based OTF WZ decoder described in Section 3. Moreover, we incorporate LDPCA codes with a feedback channel for rate adaptive decoding.

To verify the effectiveness of correlation tracking across WZ-encoded frames in a video sequence, we tested the above set-up with three standard QCIF (i.e., 176×144) video sequences, carphone, foreman and soccer, with different scene dynamics of low, medium and high motions, respectively. All the results are based on the average of 50 WZ frames. The quantization parameters Q of H.264/AVC encoder for different video sequences with different WZ quantization bits q have been listed in Table 2. The selections of quantization parameters Q for different WZ quantization bits q make sure that both the decoded key and WZ frames have similar visual qualities in terms of PSNR. Moreover, we split each 176×144 WZ frame into 16 sub-frames with size 44×36 (i.e. $N = 1584$) for efficient coding purpose. Within each sub-frame, the block size for correlation estimation is equal to 4×6 (i.e., $C = 24$) for total $N' = 66$ number of blocks.

Results comparing the relative performance of pre-estimation in frame level DVC,[10] pre-estimation in block level DVC,[10] and the proposed EP based OTF DVC for the carphone, foreman, and soccer video sequences, respectively, are shown in Figs. 2, 3, 4, where the implementation of standard DVC codec is based on the DISCOVER framework[10] with joint bit-plane setup.

The pre-estimation methods,[10] either in frame or block levels, model the correlation as Laplace distribution, whose correlation parameter is estimated using the difference between backward and forward motion compensated frames/blocks at the decoder. Our proposed OTF estimator unifies the process of correlation estimation using EP and joint bit-plane decoding into a single joint process, where the updated decoding information can be used to improve the correlation estimation and vice versa. As expected, pre-estimation in block level has better performance than that of frame level in terms of bit rate saving, since block level correlation offers a finer granularity than the frame level correlation. More importantly, our proposed EP based OTF codec always achieves the best performance for all sequences (slow, medium and fast motions), since the proposed EP based OTF estimator can iteratively refine the correlation statistics in each block.

In particular, for the carphone sequence (slow motion), to obtain the same visual qualities (i.e., PNSRs), our proposed EP based OTF codec achieves about 10 kbps and 30 kbps saving compared to pre-estimation in block and frame levels, respectively. For foreman sequence (medium motion), the average rate decrease of EP based OTF codec is 20 kbps for pre-estimation in block level and 30 kbps for pre-estimation in frame level. Moreover, for soccer sequence with fast motion, we again observe the superiority of our proposed EP based OTF codec over the pre-estimation codecs, where the proposed EP based OTF codec offers about 38.25 kbps and 56.5 kbps saving compared to pre-estimation in block and frame levels, respectively. These results demonstrated that our proposed EP based OTF codec are more powerful for video sequences with fast motions.

A sub-frame-by-sub-frame (i.e., the first sub-frame of each WZ frame) rate variation for the soccer sequence with quantization bits equal to 3 is shown in Fig. 5. We found that the rate variation across frames is about 36.01 kbits for the proposed EP based OTF DVC codec and 40.01 kbits for DVC codec with pre-estimation in block level. Moreover, the result shows that the rate fluctuations of EP based OTF and pre-estimation in block level DVC codecs have similar trend and the proposed EP based OTF codec always has equal or lower code rate than that of the pre-estimation in block level DVC codec. The maximum difference of code rate between EP based OTF and pre-estimation block level codecs is about −5.32 kbits. Similar results are obtained for other sub-frames in all three testing sequences.

The estimation accuracy of correlation parameter is studied in Fig. 6 for the soccer sequence. Here, we use the offline estimated correlation parameter as benchmark, where the benchmark Laplacian parameter is calculated offline at the block level for each frame using the residual between the WZ frame and the side information frame. We can see that the proposed OTF correlation estimation scheme improves the estimates obtained through pre-estimation method,[10] which also explains why the proposed EP based OTF DVC outperforms the pre-estimation based DVC codec.

Finally, the proposed EP based OTF estimator offers a very low complexity overhead compared with the standard BP algorithm. The complexity of the proposed estimator lies in the evaluation of equations (12), (13) and (14) as shown in Section 4.1. Roughly speaking, the EP based OTF estimator introduces less than 10% computational overhead compared with the standard BP algorithm.

6. CONCLUSION

This paper proposes an on-the-fly (OTF) correlation estimation scheme for distributed video coding using expectation propagation (EP). Unlike previous work performing pre-estimation where estimation starts before decoding, our proposed correlation estimation technique is embedded within the WZ decoder itself, thus ensuring dynamic estimation of correlation changes in block level. This is achieved by augmenting the SW code factor graph to connect correlation parameter variable nodes together with additional factor nodes. Inference on the factor graph for continuous correlation parameter variable is achieved through EP based deterministic approximation methods, which offers better tradeoff between accuracy and complexity compared with other methods. The proposed scheme boosts coding performance together with the ease of integration with existing DVC codecs. We demonstrate the benefits of using the proposed scheme via a pixel-based DVC setup. Simulation results show significant performance improvement due to correlation tracking for multiple video sequences with the Laplacian correlation model.

Acknowledgments

REFERENCES

1. Aaron, A.; Zhang, R.; Girod, B. Wyner-Ziv coding of motion video. Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on; IEEE; 2002. p. 240-244.

2. Puri, R.; Ramchandran, K. PRISM: A new robust video coding architecture based on distributed compression principles. Proc. Annual Allerton Conference on Communication Control and Computing; Citeseer; 2002. p. 586-595.

3. Slepian D, Wolf J. Noiseless coding of correlated information sources. IEEE Trans. Inform. Theory. 1973 Jul.19:471–480.

4. Wyner A, Ziv J. The rate-distortion function for source coding with side information at the decoder. IEEE Trans. Inform. Theory. 1976 Jan.22:1–10.

5. Girod B, Aaron A, Rane S, Rebollo-Monedero D. Distributed video coding. Proceedings of the IEEE. 2005; 93(1):71–83.

6. Guillemot C, Pereira F, Torres L, Ebrahimi T, Leonardi R, Ostermann J. Distributed monoview and multiview video coding: basics, problems and recent advances. IEEE Signal Processing Magazine. 2007; 24(5):67–76.

7. Stankovic, L.; Stankovic, V.; Cheng, S. Distributed compression: Overview of current and emerging multimedia applications. ICIP-2011 IEEE International Conference on Image Processing; IEEE; 2011.

8. Meyer, P.; Westerlaken, R.; Gunnewiek, R.; Lagendijk, R. Distributed source coding of video with non-stationary side-information. Proc. SPIE; Citeseer; 2005. p. 857-866.

9. Dalai, M.; Leonardi, R.; Pereira, F. Improving turbo codec integration in pixel-domain distributed video coding. Acoustics, Speech and Signal Processing, IEEE International Conference on, ICASSP Proceedings; IEEE; 2006. II–II

10. Brites C, Pereira F. Correlation noise modeling for efficient pixel and transform domain Wyner–Ziv video coding. Circuits and Systems for Video Technology, IEEE Transactions on. 2008; 18(9): 1177–1190.

11. Fan, X.; Au, O.; Cheung, N. Adaptive correlation estimation for general Wyner-Ziv video coding. Image Processing (ICIP), 2009 16th IEEE International Conference on; IEEE; 2009. p. 1409-1412.

12. Huang, X.; Forchhammer, S. Improved virtual channel noise model for transform domain Wyner-Ziv video coding. Acoustics, Speech and Signal Processing, IEEE International Conference on, ICASSP Proceedings; IEEE; 2009. p. 921-924.

13. Varodayan, D.; Mavlankar, A.; Flierl, M.; Girod, B. Distributed grayscale stereo image coding with unsupervised learning of disparity; IEEE Data Compression Conference; 2007. p. 143-152.

14. Stankovic, L.; Stankovic, V.; Wang, S.; Cheng, S. Distributed Video Coding with Particle Filtering for Correlation Tracking; Proc. Eusipco-2010 18th European Signal Processing Conference; 2010.

15. Wang S, Cui L, Stankovic L, Stankovic V, Cheng S. Adaptive correlation estimation with particle filtering for distributed video coding. Circuits and Systems for Video Technology, IEEE Transactions on. 2011

16. Aaron A, Rane SD, Setton E, Girod B. Transform-domain Wyner-Ziv codec for video. Proceedings of SPIE. 2004; 5308:520–528.

17. Aaron A, Rane S, Girod B. Wyner-Ziv video coding with hash-based motion compensation at the receiver. IEEE ICIP'04. 2005; 5:3097–3100.

18. Meyera P, Westerlakena R, Gunnewiekb R, Lagendijka R. Distributed source coding of video with non-stationary side-information. Proceedings of SPIE. 2005; 5960:59602J.

19. Deligiannis, N.; Munteanu, A.; Clerckx, T.; Schelkens, P.; Cornelis, J. Modeling the correlation noise in spatial domain distributed video coding. 2009 Data Compression Conference; IEEE; 2009.

20. Brites, C.; Ascenso, J.; Pereira, F. Improving transform domain wyner-ziv video coding performance. Acoustics, Speech and Signal Processing, IEEE International Conference on, ICASSP Proceedings; IEEE; 2006. II–II

21. Ascenso, J.; Brites, C.; Pereira, F. Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding; 5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services; 2005.

22. Xiong Z, Liveris A, Cheng S. Distributed source coding for sensor networks. IEEE Signal Process. Magazine. 2004 Sep.21:80–94.

23. Kschischang F, Frey B, Loeliger H. Factor graphs and the sum-product algorithm. IEEE Trans. Inform. Theory. 2001 Feb.47:498–519.

24. MacKay DJC, Neal RM. Near shannon limit performance of low density parity check codes. Electronics Letters. 1996; 32(18)

25. Minka TP. Expectation propagation for approximate bayesian inference. Uncertainty in Artificial Intelligence. 2001; 17:362–369.

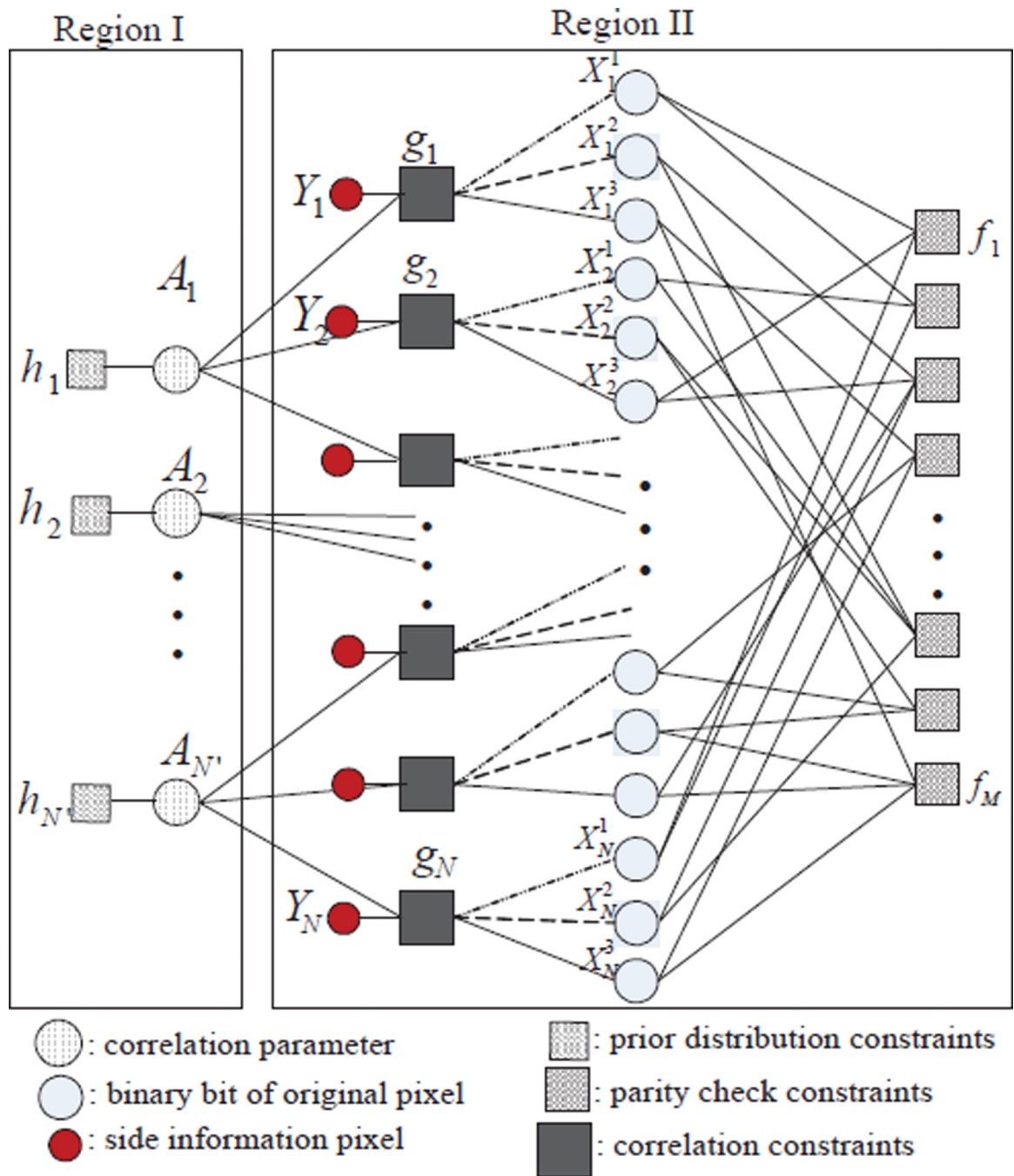26. Recommendation, I. T. U. Advanced Video Coding for Generic Audiovisual Services. ITU-T Rec. H.264. 2007

Figure 1.
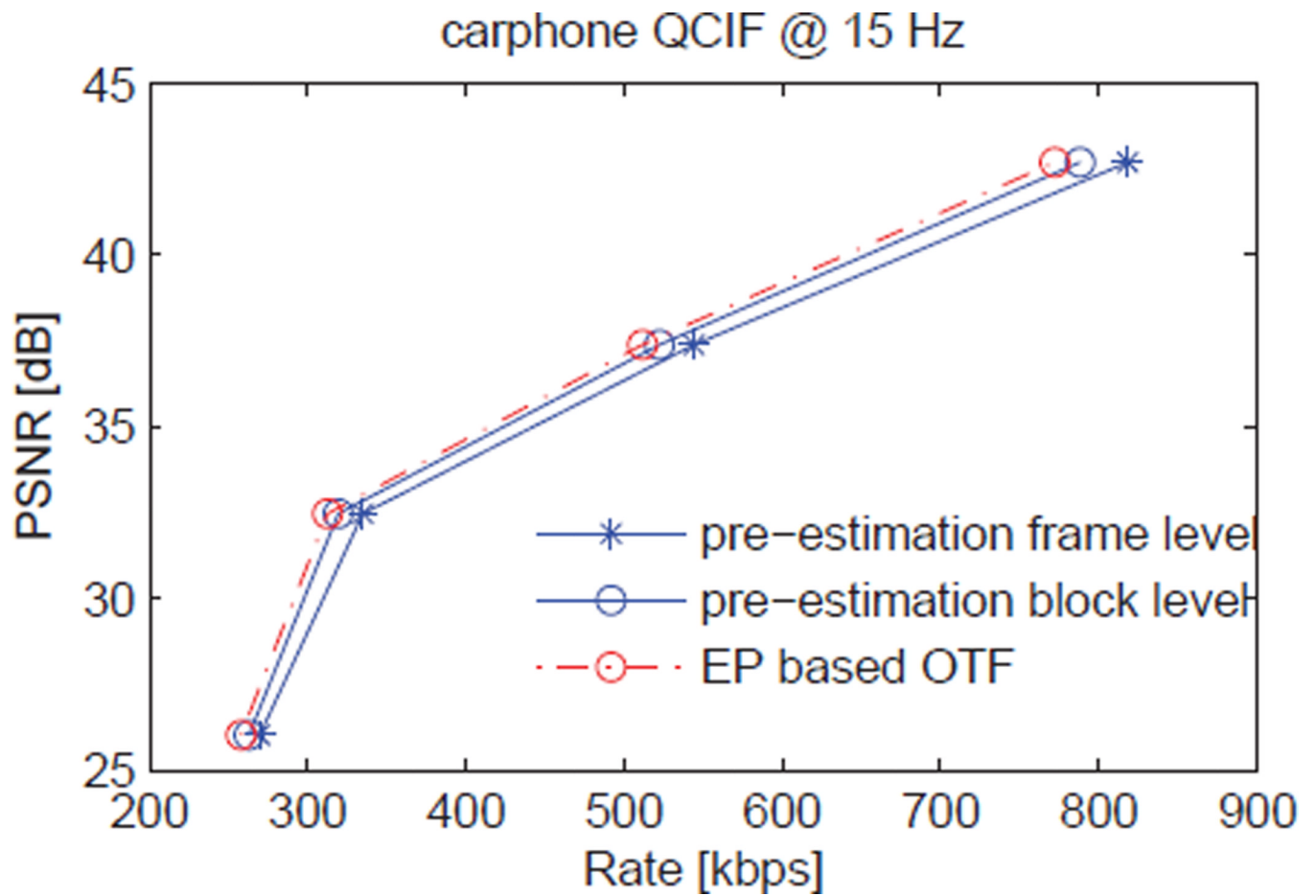Factor graph of joint bit-plane SW decoding with correlation estimation.

Figure 2.
PSNR comparison of the proposed EP based OTF and pre-estimation DVC for the QCIF carphone sequence, compressed at 15 fps.
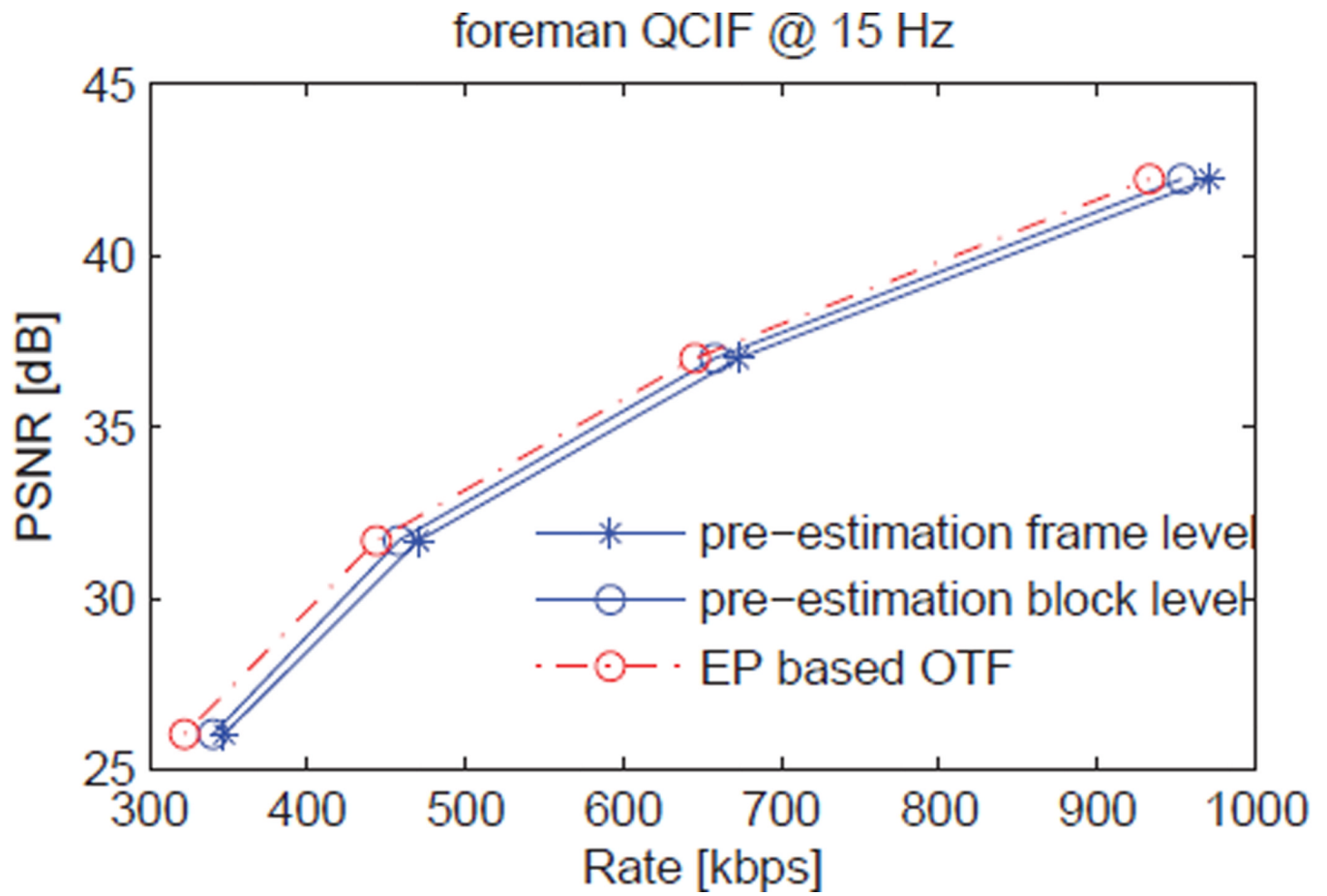
Figure 3.
PSNR comparison of the proposed EP based OTF and pre-estimation DVC for the QCIF foreman sequence, compressed at 15 fps.
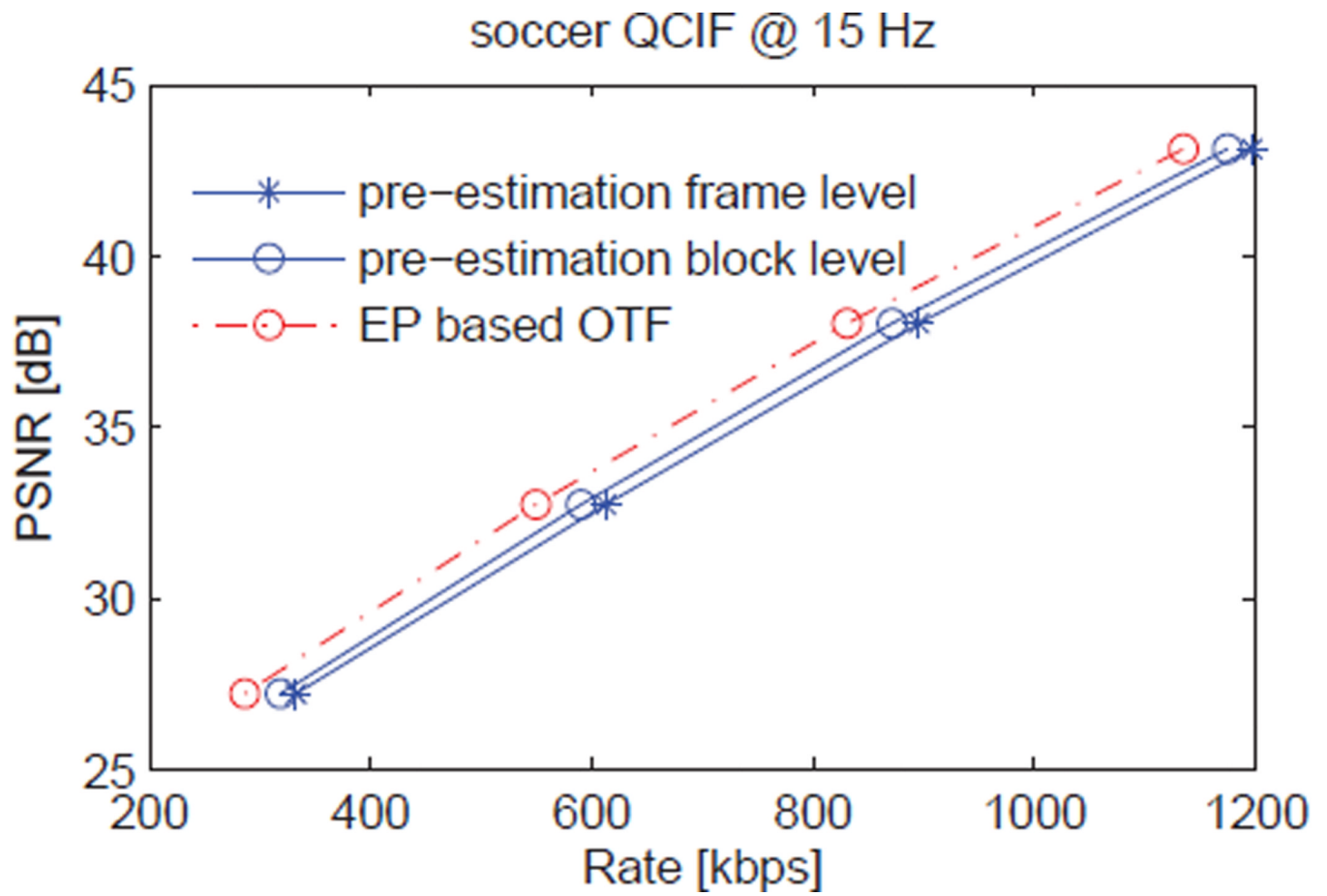
Figure 4.
PSNR comparison of the proposed EP based OTF and pre-estimation DVC for the QCIF soccer sequence, compressed at 15 fps.
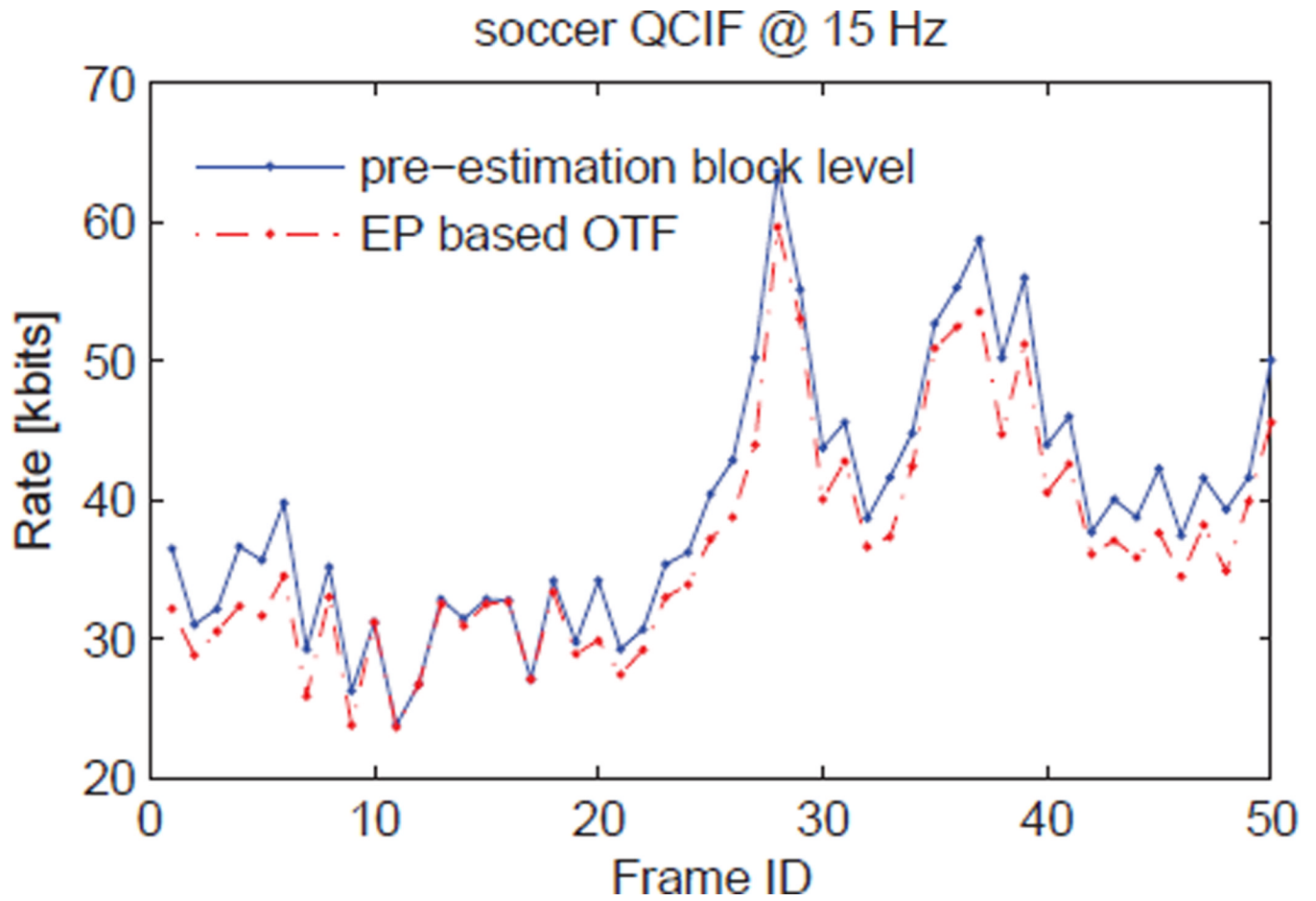
Figure 5.
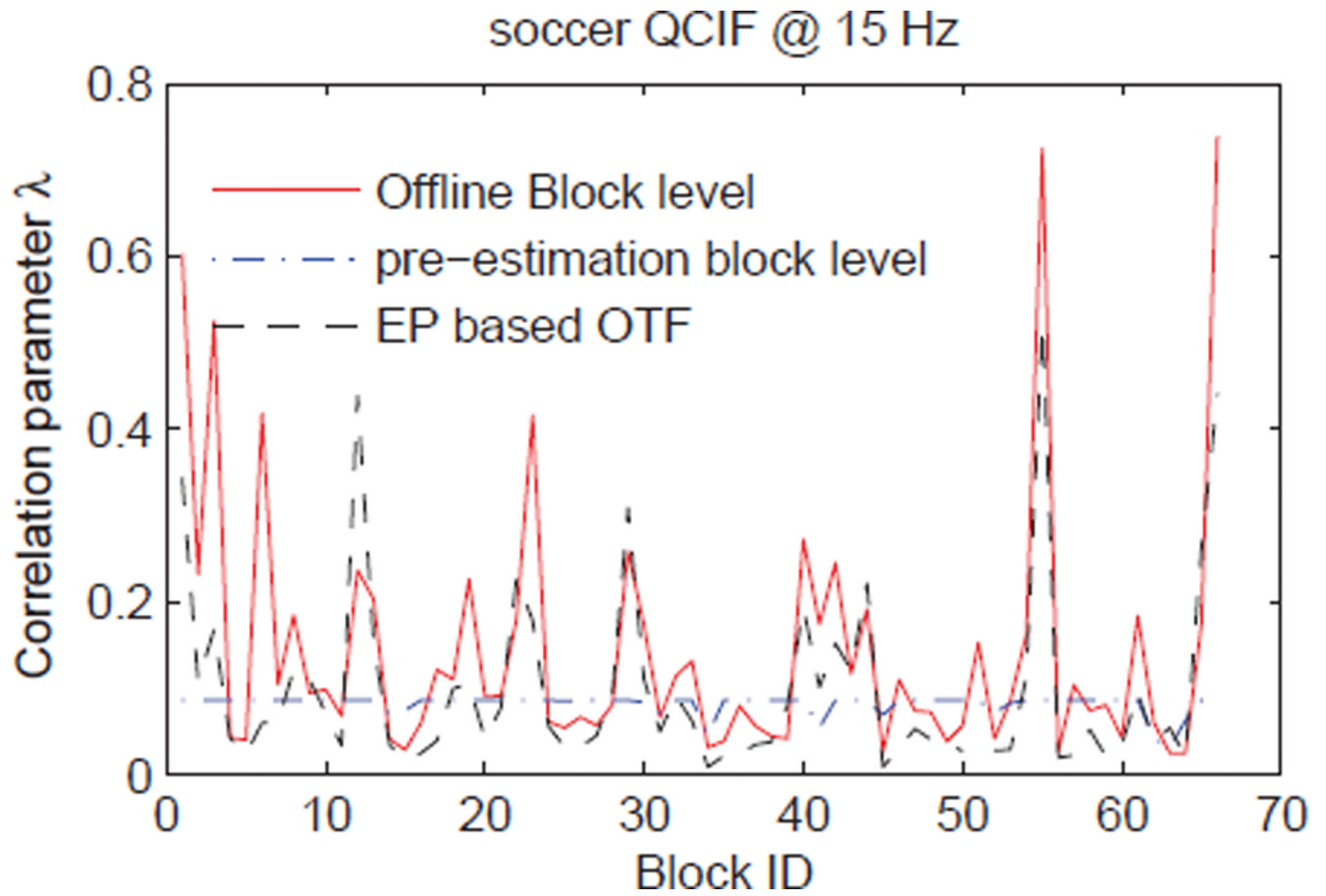Subframe-by-subframe rate variance for soccer sequence with quantization bits equal to 3.

Figure 6.
Estimation accuracy of the proposed EP based OTF DVC for the correlation parameter of the soccer sequence

Table 1

Expectation Propagation

Initialize the term approximation $\tilde{g}_k(\theta)$ and $q(\theta) = \frac{1}{Z} \prod_{k=1}^{C} \tilde{g}_k(\theta)$, where $Z = \int_\theta \prod_{k=1}^{C} \tilde{g}_k(\theta)$

repeat

  for $k = 1, \ldots, C$ **do**

    Compute $q^{\backslash k}(\theta) \propto q(\theta)/\tilde{g}_k(\theta)$

    Minimize Kullback Leibler (KL) divergence between $q(\theta)$ and $g_k(\theta)q^{\backslash k}(\theta)$ by performing moment matching

    Set approximate term $\tilde{g}_k(\theta) \propto q(\theta)/q^{\backslash k}(\theta)$

  end for

until parameters converged

Table 2

H.264/AVC quantization parameter Q for different video sequences

Quantization bits	Carphone	Foreman	Soccer
	Q	Q	Q
2	46	46	44
3	36	36	34
4	28	28	26
5	22	21	19