Dual Frame Motion Compensation with Optimal Long-Term Reference Frame Selection and Bit Allocation

Da Liu, Debin Zhao, Xiangyang Ji, and Wen Gao, *Fellow, IEEE*

***Abstract*—In dual frame motion compensation (DFMC), one short-term reference frame and one long-term reference frame (LTR) are utilized for motion compensation. The performance of DFMC is heavily influenced by the jump updating parameter and bit allocation for the reference frames. In this paper, first the rate-distortion performance analysis of motion compensated prediction in DFMC is presented. Based on this analysis, an adaptive jump updating DFMC (JU-DFMC) with optimal LTR selection and bit allocation is proposed. Subsequently, an error resilient JU-DFMC is further presented based on the error propagation analysis of the proposed adaptive JU-DFMC. The experimental results show that the proposed adaptive JU-DFMC achieves better performance over the existing JU-DFMC schemes and the normal DFMC scheme, in which the temporally most recently decoded two frames are used as the references. The performance of the adaptive JU-DFMC is significantly improved for video transmission over noisy channels when the specified error resilience functionality is introduced.**

***Index Terms*—Bit allocation, dual frame motion compensation, error propagation, error resilience, motion compensation, video coding.**

I. Introduction

MOTION-COMPENSATED prediction in inter-prediction coding plays an important role in existing hybrid video codecs such as MPEG-4 [1], H.263 [2], and H.264/AVC [3]. For each inter-block in the current frame, its prediction signal is obtained from the reference frame via motion compensation. Subsequently, the difference between the current original frame and its prediction is compressed and transmitted. Multiframe motion compensation [4]–[8] allows that more than one reference frame can be used for the motion-compensated prediction. In most cases, it improves coding performance significantly. However, with the increase of the number of reference frames, the memory storage and the motion searching complexity increase dramatically.

Dual frame motion compensation (DFMC) is the special case of multiframe motion compensation in which only two reference buffers are utilized, thus requiring a relatively modest increase in memory storage and motion searching complexity. In DFMC, as shown in Fig. 1, the first reference buffer contains the most recently decoded frame, called short-term reference frame (STR), and the second one contains a reference frame from the past that is periodically updated, called long-term reference frame (LTR).

Generally, there are two types of approaches for DFMC [9]. The first approach is jump updating DFMC (JU-DFMC), in which LTR remains static for N frames, and jumps forward to be the frame at a distance 2 back from the frame to be encoded. For example, suppose for the frames from time instant $i-N+1$ to i, the LTR is $i-N-1$. Then after encoding frame i, when the encoder moves on to encoding frame $i+1$, the STR will slide forward by one to frame i, and the LTR will jump forward by N to frame $i-1$. After that, the LTR remains fixed for N frames, and then jumps forward again. N is called the jump update parameter. The second approach is continuous updating DFMC (CU-DFMC), where the LTR for each current frame always has a fixed temporal distance D, called the continuous update parameter, to the current frame. As a result, every frame has a chance serving as an STR and as an LTR.

A number of DFMC-based approaches to improve the video coding performance have been reported in the literatures. In [10], a refreshing rule of LTR was proposed by introducing scene changing detection. In [11], the concept of the dual frame was simulated in a low bandwidth situation by the block-partitioning prediction and the utilization of two time differential reference frames. Challappa *et al*. [12] have found that using a high quality frame as a reference frame for the following frames will benefit the overall performance. Challappa *et al*. [13] and [14] have shown that peak signal-to-noise ratio (PSNR) is influenced by the different extra bandwidth and the period giving to the LTR. In [15], the update period of the LTR was set to ten frames. The PSNR of nine frames that follow the LTR frame was utilized to determine how many bits can be allocated to the LTR. In [16],

Manuscript received June 30, 2008; revised December 8, 2008 and May 11, 2009. First version published September 1, 2009; current version published March 5, 2010. This work was supported by the National Science Foundation of China under Grant 60736043, and the National Basic Research Program of China, 973 Program 2009CB320903. This paper was recommended by Associate Editor E. Steinbach.

D. Liu is with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: dliu@jdl.ac.cn).

D. Zhao is with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: dbzhao@jdl.ac.cn).

X. Ji is with the Broadband Networks and Digital Media Laboratory, Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: xyji@mail.tsinghua.edu.cn).

W. Gao is with the Key Laboratory of Machine Perception, School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: wgao@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2009.2031442

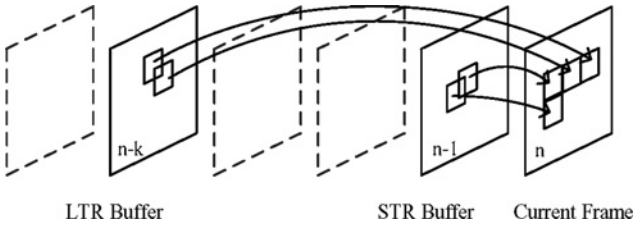

Fig. 1. Dual frame motion compensation.

simulated annealing was utilized to select the LTR; however, its computational complexity is relatively high.

For video transmission over noisy channels, in [9], [17], and [18], the recursive optimal per-pixel estimate algorithm was utilized to provide mode decision in dual frame coding. They had pointed out that the fixed jump updating parameter of the LTR was not optimal for all sequences. Feedback was also utilized in dual frame coding in [19] to control the drift errors. In [20], the uneven allocation of error protection to the LTR was examined. It showed that assigning higher error protection for the LTRs was better than assigning equal error protection for all frames. In [21], a binary decision tree designed by the classification and regression trees algorithm was utilized to choose among various error concealment choices in the dual frame coding. The trade-off of end-to-end delay and compression efficiency in dual frame coding motion compensation was investigated in [22] and [23].

Multihypothesis motion compensated prediction (MHMCP) was also utilized to enhance error resilience. In [24], each block is predicted from two reference blocks using two motion vectors. In [25], the error propagation model of MHMCP jointly considered the coding efficiency and error resilience in predictor selection. Furthermore, the reference picture interleaving and data partitioning was utilized in [26] to make MHMCP more resilient to channel errors. In [27], the error propagation impact in MHMCP was examined and the rate-distortion performance considering the hypothesis number and coefficients were analyzed. In [28], two-hypothesis prediction and one-hypothesis prediction were adaptively used to decrease error propagation. In [29], all frames were divided into period frame and nonperiod frame. The period frame has fixed distance between each other. For all nonperiod frames, only a previous period frame was utilized as the reference frame. However, the adaptive period frame selection and bit allocation for different packet loss rates was not reported.

For different video sequences, adaptive LTR selection and bit allocation to improve coding efficiency have not yet been fully studied. In this paper, the rate distortion (R-D) performance for motion compensated prediction (MCP) in DFMC is analyzed. Based on the analysis, an adaptive JU-DFMC with optimal LTR selection and bit allocation is provided. Furthermore, for video transmission over noisy channels, the error propagation in the proposed adaptive JU-DFMC is analyzed first, then an error resilient JU-DFMC is presented.

The rest of the paper is organized as follows. In Section II, the R-D performance analysis for MCP in DFMC is given. Based on the analysis, optimal LTR selection and bit allocation in the adaptive JU-DFMC are separately presented in Sections III and IV. In Section V, based on the error propagation analysis for the proposed adaptive JU-DFMC, an error resilient JU-DFMC is presented for video transmission over noisy channels. The experimental results and discussions are provided in Section VI. Finally, Section VII concludes this paper.

II. Rate Distortion Performance Analysis for MCP in DFMC

In this section, the R-D performance analysis for MCP is first presented. Second, the prediction error variances in both JU-DFMC and CU-DFMC are formulated.

A. Power Spectral Density

Power spectral density (PSD) $\Phi(\omega_x, \omega_y)$ describes the power of a signal as a function of frequency and is achieved as

$$\Phi(\omega_x, \omega_y) = \int_{-\infty}^{\infty} R(\tau) e^{-2\pi i(\omega_x, \omega_y)\tau} d\tau \tag{1}$$

where (ω_x, ω_y) is a vector representing the 2-D spatial frequency, $R(\tau)$ is an autocorrelation function that describes the correlation between different time points.

The rate-distortion analysis of MCP [30] relates the PSD of the prediction error to the accuracy of motion compensation captured by the probability density function (pdf) of displacement error. The rate-distortion analysis was extended to the multihypothesis prediction in [31]. Especially, as described in [23], the PSD for two hypotheses prediction can be simplified to

$$\Phi_{ee}(\Lambda) = \Phi_{ss}(\Lambda)\left(\frac{6+\alpha_1+\alpha_2+2P_1(\Lambda)P_2(\Lambda)}{4} - P_1(\Lambda) - P_2(\Lambda)\right) \tag{2}$$

where $\Phi_{ee}(\Lambda)$ is the PSD of the prediction error, $\Phi_{ss}(\Lambda)$ is the signal power spectrum of the input video signal and is non-negative, $\Lambda = (\omega_x, \omega_y)$, and

$$P(\omega_x, \omega_y) = e^{-2\pi\sigma_\Delta^2(\omega_x^2+\omega_y^2)}. \tag{3}$$

$P_1(\Lambda)$ and $P_2(\Lambda)$ are separately the displacement error pdfs from the first and the second hypotheses. σ_Δ^2 is the displacement error variance (DEV) and it reflects the inaccuracy of the displacement vector used for the motion compensation [30]. α_i represents the spectral noise-to-signal power ratio in the ith hypothesis and it has been given in [31] as $\alpha_i = \Phi_{nn\_i}(\Lambda)/\Phi_{ss}(\Lambda)$. $\Phi_{nn\_i}(\Lambda)$ represents the PSD of residual noise in the ith hypothesis and has been given in [30] as $\Phi_{nn\_i}(\Lambda) = \max[0, \theta(1-\theta/\Phi_{ee\_i}(\Lambda))]$. $\Phi_{ee\_i}(\Lambda)$ is the PSD of the prediction error in the ith hypothesis. θ is a parameter that generates the rate-distortion function by taking on all positive real values [30]. If $\Phi_{nn\_i}(\Lambda) = 0$, it has no influence on $\Phi_{ee}(\Lambda)$. If $\Phi_{nn\_i}(\Lambda) = \theta(1-\theta/\Phi_{ee\_i}(\Lambda))$, since $\Phi_{nn\_i}(\Lambda)$ is nearly linear proportional to $\Phi_{ee\_i}(\Lambda)$ in the short term, it can be simplified as $\Phi_{nn\_i}(\Lambda) = h(\Phi_{ee\_i}(\Lambda))$. $h(\,)$ is the linear function that represents the relationship between

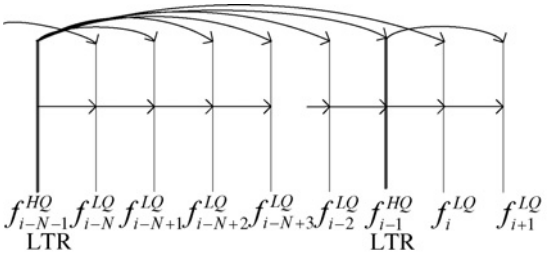

Fig. 2. JU-DFMC structure.

$\Phi_{nn\_i}(\Lambda)$ and $\Phi_{ee\_i}(\Lambda)$. So considering both cases, (2) can be re-written as

$$\Phi_{ee}(\Lambda) = \frac{h(\Phi_{ee\_1}) + h(\Phi_{ee\_2})}{4} + \Phi_{ss}(\Lambda)\frac{3 + P_1(\Lambda)P_2(\Lambda) - 2P_1(\Lambda) - 2P_2(\Lambda)}{2}. \quad (4)$$

The PSD $\Phi_{ee}(\omega_x, \omega_y)$ is utilized as the performance measurement. From [31]

$$\Delta R = \frac{1}{8\pi^2}\int_{-\pi}^{+\pi}\int_{-\pi}^{+\pi}\log_2\left(\frac{\Phi_{ee}(\omega_x, \omega_y)}{\Phi_{ss}(\omega_x, \omega_y)}\right)d\omega_x d\omega_y \quad (5)$$

where ΔR represents the maximum bit-rate reduction (in bits/sample) by optimum encoding of the prediction error, compared to the optimum intra frame encoding of the signals for the same mean squared error (MSE) [32]. A negative ΔR corresponds to a reduced bit-rate compared to the optimum intra frame coding. From (5), we can conclude that when the reconstructed frames from two prediction structures have the same MSE, if the PSD $\Phi_{ee}(\omega_x, \omega_y)$ in the frame from one prediction structure is smaller than the other, then its ΔR will be smaller, and its coding efficiency will be better.

B. Prediction Error Variance

The PSD $\Phi_{ee}(\Lambda)$ is related to the displacement error pdf. From (3), the displacement error pdf is determined by the DEV $\sigma_\triangle^2$. In video coding, the displacement error is obtained from the distance between the actual motion vector and its estimated one. The DEV reflects the inaccuracy of the motion compensation [30]. The prediction error variance (PEV) is the variance of the motion compensated error between the original value and the reference value, it also reflects the inaccuracy of the motion compensation. From [23], in the short term, the DEV $\sigma_\triangle^2$ is nearly directly proportional to the PEV σ_e^2. So σ_e^2 plays a key role in determining $\Phi_{ee}(\Lambda)$. In the following, we will give the analysis of σ_e^2.

In JU-DFMC, there are two kinds of frames. As shown in Fig. 2, one kind of frame is called high quality frame (HQF) with relatively more bits allocated, such as the $(i-1)$th frame f_{i-1}^{HQ} and the $(i-N-1)$th frame f_{i-N-1}^{HQ}. The other kind of frame has relatively lower quality (LQF), such as the $(i+k)$th frame f_{i+k}^{LQ} $(k = -N, -N+1, \ldots, -2, 0, 1)$. To simplify our discussion, all these LQFs are coded to have similar MSEs. Any one frame (HQF or LQF) can be utilized as STR for the next frame. Meanwhile, the HQF is utilized as LTR for the following several frames. One example of the PEV σ_e^2 in JU-DFMC is shown in Fig. 3. The frame at time instant 1 is encoded as an HQF, and then utilized as the LTR for the following several frames. The frames at other time instants are encoded as LQFs (in the figure, the bit-rate in LQF is 420.62 kb/s, bit allocation in HQF is three times that in LQF). In every encoded LQF, the prediction performance from STR is similar, so σ_e^2 from STR (dashed line) is similar. For the second LQF (at time instant 3) following the LTR, the prediction performance from the LTR is better, thus σ_e^2 is smaller. With the coding of the following frames, the prediction performance from the LTR degrades while σ_e^2 from the LTR increase (solid line).

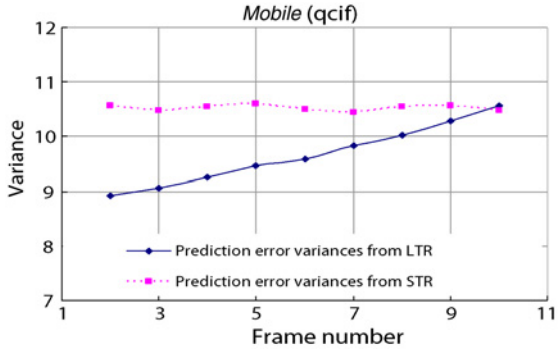

Fig. 3. Prediction error variance σ_e^2 in JU-DFMC.

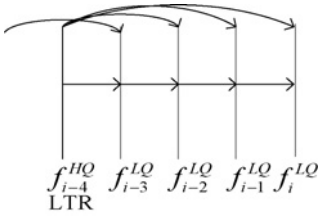

Fig. 4. LQF i in JU-DFMC.

In CU-DFMC of this paper, every frame is allocated approximately the same quality. The continuous update parameter D is generalized as 2. For every frame, the most recently decoded two frames are separately STR and LTR. Therefore for every frame in CU-DFMC, the prediction performance from LTR is nearly the same and thus σ_e^2 from LTR is nearly the same. The same conclusion can also be obtained for STR prediction.

Generally, if the prediction performance is better, the PEV σ_e^2 will be smaller. Since σ_e^2 is nearly directly proportional to $\sigma_\triangle^2$ [23], $\sigma_\triangle^2$ is smaller as well.

III. Optimal LTR Selection in JU-DFMC

In this section, the optimal LTR selection in JU-DFMC is presented. In the first section, the coding performance of the same LQF in CU-DFMC and JU-DFMC is compared. In the second section, the coding performance of the same HQF in CU-DFMC and JU-DFMC is compared. The optimal LTR selection in JU-DFMC is provided in the last section.

A. Coding Performance Comparison of LQF

The frame at instant i can be encoded as an LQF (denoted as f_i^{LQ}) in JU-DFMC or in CU-DFMC (as shown in Figs. 4

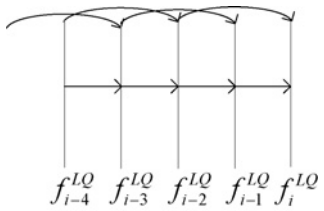

Fig. 5. LQF i in CU-DFMC.

and 5), its power spectral densities are separately represented as $\Phi_{ee}^{J}(\Lambda)$ and $\Phi_{ee}^{CL}(\Lambda)$. For giving the performance comparison when f_i^{LQ} is encoded in the two DFMC coding structures, the reconstructed f_i^{LQ}s in both structures are assumed to have the same MSE. Then the difference of power spectral densities $\widehat{\Phi}_{ee}(\Lambda)$ between the two f_i^{LQ}s in the two structures can be derived from (4) as follows:

$$\begin{aligned}\widehat{\Phi}_{ee}(\Lambda) &= \Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda) \\ &= \frac{1}{4}(h(\Phi_{ee\_1}^{J}(\Lambda) - \Phi_{ee\_1}^{CL}(\Lambda)) \\ &\quad + h(\Phi_{ee\_2}^{J}(\Lambda) - \Phi_{ee\_2}^{CL}(\Lambda))) + \Phi_{ss}(\Lambda)P^{LQ} \end{aligned} \quad (6)$$

where $\Phi_{ee\_k}^{J}(\Lambda)$ and $\Phi_{ee\_k}^{CL}(\Lambda)$ are separately the PSD in the kth hypothesis (k = 1 or 2 represent taking STR or LTR as hypothesis here and after) in JU-DFMC and CU-DFMC, and

$$\begin{aligned}P^{LQ} &= P_1^{J}(\Lambda) \times (\frac{1}{2}P_2^{J}(\Lambda) - 1) - P_2^{J}(\Lambda) \\ &\quad -(P_1^{CL}(\Lambda) \times (\frac{1}{2}P_2^{CL}(\Lambda) - 1) - P_2^{CL}(\Lambda)). \end{aligned} \quad (7)$$

In (7), $P_1^{J}(\Lambda)$ and $P_2^{J}(\Lambda)$ separately represent the displacement error pdfs from STR and LTR in JU-DFMC, $P_1^{CL}(\Lambda)$ and $P_2^{CL}(\Lambda)$ separately represent the displacement error pdfs from STR and LTR in CU-DFMC.

From (6), $\Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda)$ depends on $\Phi_{ss}(\Lambda)P^{LQ}$ and $h(\Phi_{ee\_k}^{J}(\Lambda) - \Phi_{ee\_k}^{CL}(\Lambda))$. The calculation of $\Phi_{ee\_k}^{J}(\Lambda) - \Phi_{ee\_k}^{CL}(\Lambda)$ is the same as $\Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda)$. Therefore in the iterative formula (6), P^{LQ} plays a dominant role in $\Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda)$.

In JU-DFMC and CU-DFMC, the reconstructed f_i^{LQ}s are assumed to have the same MSE. STR in both structures has nearly the same MSE as the current reconstructed f_i^{LQ}, and the temporal distance from STR to the current f_i^{LQ} is the same (one frame distance), then the prediction performance from STRs in both structures is nearly the same. From Section II-B, the prediction performance influences the DEV $\sigma_{\triangle}^2$, so in the two DFMC coding structures, $\sigma_{\triangle}^2$ from STR is nearly the same. According to (3), $P_1^{J}(\Lambda)$ is nearly the same as $P_1^{CL}(\Lambda)$. Then (7) can be further represented as

$$P^{LQ} = (1 - \frac{1}{2}P_1^{J}(\Lambda))(P_2^{CL}(\Lambda) - P_2^{J}(\Lambda)). \quad (8)$$

In JU-DFMC (in Fig. 4), the MSE of LTR is smaller than that of the current reconstructed f_i^{LQ}. In CU-DFMC (in Fig. 5), the MSE of LTR is nearly the same as that of the current reconstructed f_i^{LQ}. When f_i^{LQ} is located in the several previous LQFs following LTR in JU-DFMC, the prediction performance from LTR is better than that when f_i^{LQ} is encoded in CU-DFMC. From the analysis in Section II-B, if the prediction performance is better, the DEV $\sigma_{\triangle}^2$ will be smaller, so $\sigma_{\triangle}^2$ from LTR in JU-DFMC is smaller than that in CU-DFMC. According to (3), $P_2^{CL}(\Lambda)$ is smaller than $P_2^{J}(\Lambda)$. And the $\sigma_{\triangle}^2$ is always larger than 0, then $P_1^{J}(\Lambda) < 1$. According to the above analysis and (8), $P^{LQ} = (1 - 1/2P_1^{J}(\Lambda))(P_2^{CL}(\Lambda) - P_2^{J}(\Lambda)) < 0$. Since P^{LQ} plays a dominant role in the determination of $\Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda)$, we can get $\Phi_{ee}^{J}(\Lambda) - \Phi_{ee}^{CL}(\Lambda) < 0$. This means that compared to encoding f_i^{LQ} using CU-DFMC, the coding of f_i^{LQ} using JU-DFMC has better R-D performance (represented as bits saving in the same reconstructed MSE) if f_i^{LQ} is located in the several previous LQFs following LTR. The R-D performance gain is represented as $\widehat{\Phi}_{ee}(\Lambda)$.

B. Coding Performance Comparison of HQF

The frame at instant i can also be encoded as an HQF (denoted as f_i^{HQ}) in JU-DFMC or in CU-DFMC (as shown in Figs. 6 and 7); its PSDs are separately represented as $\Phi_{ee}^{J}(\Lambda)$ and $\Phi_{ee}^{CH}(\Lambda)$. For giving the performance comparison when f_i^{HQ} is encoded in the two DFMC coding structures, the reconstructed f_i^{HQ}s in both structures are assumed to have the same MSE. Then, the difference of power spectral densities $\overline{\Phi}_{ee}(\Lambda)$ between the two f_i^{HQ}s in the two structures can be derived from (4) as follows:

$$\begin{aligned}\overline{\Phi}_{ee}(\Lambda) &= \Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda) \\ &= \frac{1}{4}(h(\Phi_{ee\_1}^{CH}(\Lambda) - \Phi_{ee\_1}^{J}(\Lambda)) \\ &\quad + h(\Phi_{ee\_2}^{CH}(\Lambda) - \Phi_{ee\_2}^{J}(\Lambda))) + \Phi_{ss}(\Lambda)P^{HQ} \end{aligned} \quad (9)$$

where $\Phi_{ee\_k}^{CH}(\Lambda)$ and $\Phi_{ee\_k}^{J}(\Lambda)$ (k = 1 or 2) are separately the PSD in the kth hypothesis in CU-DFMC and JU-DFMC, and

$$\begin{aligned}P^{HQ} &= ((P_1^{CH}(\Lambda) \times (\frac{1}{2}P_2^{CH}(\Lambda) - 1) - P_1^{J}(\Lambda) \\ &\quad \times (\frac{1}{2}P_2^{J}(\Lambda) - 1)) + (P_2^{J}(\Lambda) - P_2^{CH}(\Lambda)). \end{aligned} \quad (10)$$

In (10), $P_1^{J}(\Lambda)$ and $P_2^{J}(\Lambda)$ separately represent the displacement error pdfs from STR and LTR in JU-DFMC, $P_1^{CH}(\Lambda)$ and $P_2^{CH}(\Lambda)$ separately represent the displacement error pdfs from STR and LTR in CU-DFMC.

$\Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda)$ depends on $\Phi_{ss}(\Lambda)P^{HQ}$ and $h(\Phi_{ee\_k}^{CH}(\Lambda) - \Phi_{ee\_k}^{J}(\Lambda))$ (k = 1 or 2). The calculation of $\Phi_{ee\_k}^{CH}(\Lambda) - \Phi_{ee\_k}^{J}(\Lambda)$ is the same as $\Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda)$. Therefore in the iterative formula (9), P^{HQ} plays a dominant role in $\Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda)$.

The reconstructed f_i^{HQ}s in both structures are assumed to have the same MSE. In CU-DFMC (in Fig. 7), STR and LTR have nearly the same MSE as the current reconstructed f_i^{HQ}. In JU-DFMC (in Fig. 6), STR has larger MSE than the current reconstructed f_i^{HQ}; LTR has nearly the same MSE as the current reconstructed f_i^{HQ}, but the long temporal distance weakens its influence. So the prediction performance from LTR and STR in CU-DFMC is separately better than those in JU-DFMC. From the analysis in Section II-B, if the prediction performance is better, the DEV $\sigma_{\triangle}^2$ will be smaller, and according to (3), we can have $0 < P_1^{J}(\Lambda) < P_1^{CH}(\Lambda)$, $0 < P_2^{J}(\Lambda) < P_2^{CH}(\Lambda)$, and $P_1^{J}(\Lambda) < 1$. From the above

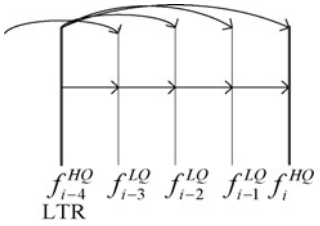

Fig. 6. HQF i in JU-DFMC.

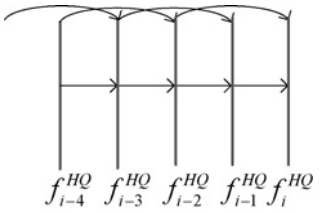

Fig. 7. HQF i in CU-DFMC.

TABLE I
NOTATIONS

Variable	Definition
$\sigma^2_{\triangle\_JL}$	DEV from LTR in JU-DFMC
$\sigma^2_{\triangle\_CLL}$	DEV from LTR in CU-DFMC, in which every frame is LQF
$\sigma^2_{\triangle\_CHL}$	DEV from LTR in CU-DFMC, in which every frame is HQF
$\sigma^2_{e\_CLL}$	PEV from LTR in CU-DFMC, in which every frame is LQF
$\sigma^2_{e\_CHL}$	PEV from LTR in CU-DFMC, in which every frame is HQF
$\sigma^2_{e\_JL2}$	PEV in the second LQF following LTR (HQF) in JU-DFMC when the reference frame is LTR
$\sigma^2_{e\_JSA}$	Average PEV in LQFs when the reference frame is STR

analysis and (10), we can get (see Appendix A)

$$P^{HQ} \approx (1 - \frac{1}{2}P_1^J(\Lambda)) \times (P_2^J(\Lambda) - P_2^{CH}(\Lambda)) < 0. \quad (11)$$

Since P^{HQ} plays a key role in determining $\Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda)$, we can get $\Phi_{ee}^{CH}(\Lambda) - \Phi_{ee}^{J}(\Lambda) < 0$. This means that compared with the coding of f_i^{HQ} using CU-DFMC, the coding of f_i^{HQ} in JU-DFMC results in R-D performance loss (represented as more bits consumption in the same reconstructed MSE), denoted as $\overline{\Phi}_{ee}(\Lambda)$.

C. Optimal LTR Selection

Before describing the proposed LTR selection method, some frequently used notations in the section are given in Table I.

1) *LTR Selection Strategy:* In this section, based on the previous analysis of the performance comparison, the optimal LTR selection in JU-DFMC is presented. In CU-DFMC, the bit allocation in every frame can be changed, but the R-D performance is fixed. For example, although the quality of every LQF in Fig. 5 is lower than that of every HQF in Fig. 7, the R-D performance is stable. So the R-D performance in CU-DFMC is used to compare with the R-D performance in JU-DFMC. In JU-DFMC, the R-D performance in LQF and HQF is different. From Section III-A, compared with the coding of f_i^{LQ} in CU-DFMC, the encoded f_i^{LQ} in JU-DFMC has better R-D performance if f_i^{LQ} is located in the several previous LQFs following LTR. From Section III-B, compared with the coding of f_i^{HQ} in CU-DFMC, the encoded f_i^{HQ} in JU-DFMC has lower R-D performance. In JU-DFMC, with the coding of frames following LTR, the R-D performance gain when the frame is encoded as an LQF and the R-D performance loss when the frame is encoded as an HQF are utilized to determine the end of LQF coding and beginning the next HQF (LTR).

When a frame is separately encoded as an LQF and an HQF in JU-DFMC, the R-D performance gain and loss are separately denoted as $\widehat{\Phi}_{ee}(\Lambda)$ and $\overline{\Phi}_{ee}(\Lambda)$. The difference of R-D performance gain and loss can be obtained as (see Appendix B)

$$\widehat{\Phi}_{ee}(\Lambda) - \overline{\Phi}_{ee}(\Lambda) \approx \frac{1}{2}h(\widehat{\Phi}_{ee\_1}(\Lambda) - \overline{\Phi}_{ee\_1}(\Lambda)) + \Phi_{ss}(P^{LQ} - P^{HQ}). \quad (12)$$

In (12), $\widehat{\Phi}_{ee\_1}(\Lambda)$ and $\overline{\Phi}_{ee\_1}(\Lambda)$ are the R-D performance gain and loss when STR is separately encoded as an LQF or an HQF in JU-DFMC, the calculation of $\widehat{\Phi}_{ee\_1}(\Lambda) - \overline{\Phi}_{ee\_1}(\Lambda)$ is the same as $\widehat{\Phi}_{ee}(\Lambda) - \overline{\Phi}_{ee}(\Lambda)$, therefore in the iterative formula (12), $(P^{LQ} - P^{HQ})$ plays a dominant role in $\widehat{\Phi}_{ee}(\Lambda) - \overline{\Phi}_{ee}(\Lambda)$.

P^{LQ} and P^{HQ} are determined by the DEV $\sigma^2_{\triangle}$. In the short term, $P_2(\Lambda)$ has nearly a linear relationship with $\sigma^2_{\triangle}$. For the simplicity of calculation, we suppose $P_2^J(\Lambda) \approx 1 - k \times \sigma^2_{\triangle\_JL}$, consequently, $P_2^{CL}(\Lambda) \approx 1 - k \times \sigma^2_{\triangle\_CLL}$ and $P_2^{CH}(\Lambda) \approx 1 - k \times \sigma^2_{\triangle\_CHL}$, where $\sigma^2_{\triangle\_JL}$ is the DEV from LTR in JU-DFMC; $\sigma^2_{\triangle\_CLL}$ is the DEV from LTR in CU-DFMC, in which every frame is an LQF; and $\sigma^2_{\triangle\_CHL}$ is also the DEV from LTR in CU-DFMC, but in which every fame is an HQF. Then P^{LQ} and P^{HQ} can be separately written as

$$\begin{aligned} P^{LQ} &= (1 - \frac{1}{2}P_1^J(\Lambda)) \times (P_2^{CL}(\Lambda) - P_2^J(\Lambda)) \\ &\approx -(1 - \frac{1}{2}P_1^J(\Lambda)) \times k \times (\sigma^2_{\triangle\_CLL} - \sigma^2_{\triangle\_JL}) \quad (13) \end{aligned}$$

$$\begin{aligned} P^{HQ} &\approx (1 - \frac{1}{2}P_1^J(\Lambda)) \times (P_2^J(\Lambda) - P_2^{CH}(\Lambda)) \\ &\approx -(1 - \frac{1}{2}P_1^J(\Lambda)) \times k \times (\sigma^2_{\triangle\_JL} - \sigma^2_{\triangle\_CHL}). \quad (14) \end{aligned}$$

The DEV $\sigma^2_{\triangle}$ can be used to compare the performance and determine the next LTR. In JU-DFMC, with the encoding of the frame j ($j = i+1, i+2, i+3, \ldots$) following LTR, the prediction performance from the LTR decreases, and the DEV $\sigma^2_{\triangle\_JL}$ from the LTR increases. When $\sigma^2_{\triangle\_JL}$ from LTR is less than $(\sigma^{/2}_{\triangle\_CLL} + \sigma^2_{\triangle\_CHL})/2$, $(\sigma^2_{\triangle} - \sigma^2_{\triangle\_CHL}) < (\sigma^2_{\triangle\_CLL} - \sigma^2_{\triangle})$. Then P^{HQ} is larger than P^{LQ} and $\overline{\Phi}_{ee}(\Lambda)$ is larger than $\widehat{\Phi}_{ee}(\Lambda)$. The R-D performance gain when frame j is encoded as an LQF is larger than the R-D performance loss when frame j is encoded as an HQF. At this point, if the frame j is encoded as an LQF, JU-DFMC will have better performance. Otherwise, when $\sigma^2_{\triangle\_JL}$ from the LTR is larger than $(\sigma^2_{\triangle\_CLL} + \sigma^2_{\triangle\_CHL})/2$, the R-D performance gain will be smaller than the performance loss. At this point, if the frame j is encoded as an LQF, the coding of the next HQF will result

in more quality loss, and thus the performance of JU-DFMC will decrease.

From the above analysis, we select the DEV

$$\sigma^2_{\triangle\_T} = (\sigma^2_{\triangle\_CLL} + \sigma^2_{\triangle\_CHL})^{/2} \tag{15}$$

as the point to terminate the LQF coding. When $\sigma^2_{\triangle\_JL}$ from LTR is larger than $\sigma^2_{\triangle\_T}$, the next frame is encoded as an HQF.

2) *Implementation in Video Coding:* In the short term, the DEV $\sigma^2_{\triangle}$ is nearly directly proportional to the PEV σ^2_e [23]. Then in JU-DFMC, $\sigma^2_{e\_T} = (\sigma^2_{e\_CLL} + \sigma^2_{e\_CHL})^{/2}$ can be utilized instead of $\sigma^2_{\triangle\_T}$ as the point to terminate the coding of LQF, where $\sigma^2_{e\_T}$ is the PEV from LTR in JU-DFMC; $\sigma^2_{e\_CLL}$ is the PEV from LTR in CU-DFMC, in which every frame is an LQF; $\sigma^2_{e\_CHL}$ is also the PEV from LTR in CU-DFMC, in which every frame is an HQF.

$\sigma^2_{e\_CLL}$ and $\sigma^2_{e\_CHL}$ cannot be obtained in JU-DFMC. HQFs in CU-DFMC and JU-DFMC are assumed to have the same MSE, then $\sigma^2_{e\_CHL}$ can be replaced by $\sigma^2_{e\_JL2}$, which is the PEV in the second LQF following LTR (HQF) in JU-DFMC when the reference frame is LTR. LQFs in CU-DFMC and JU-DFMC are assumed to have the same MSE. If the quality of the reference frame is the same, and the temporal distance between the two reference frames is only one frame, the PEVs from the two reference frames are similar, so $\sigma^2_{e\_CLL}$ can be replaced by the PEV in LQF in JU-DFMC when the reference frame is an STR (LQF). For safety, $\sigma^2_{e\_CLL}$ is replaced by $\sigma^2_{e\_JSA}$, which is the average PEV in LQFs (from the second LQF to the last encoded LQF) in the GOP when the reference frame is STR.

So finally in JU-DFMC, $\sigma^2_{e\_T} = (\sigma^2_{e\_JL2} + \sigma^2_{e\_JSA})^{/2}$ is utilized instead of $\sigma^2_{e\_T} = (\sigma^2_{e\_CLL} + \sigma^2_{e\_CHL})^{/2}$ as the point to terminate the coding of LQF.

IV. Bit Allocation for JU-DFMC

A. Performance Measurement of Bit Allocation

In determining bit allocation, PSD $\Phi_{ee}(\omega_x, \omega_y)$ is compared under the same bit-rate. According to Parseval's relation

$$\sigma^2_e = \frac{1}{4\pi^2}\int_{-\pi f_{sx}}^{+\pi f_{sx}}\int_{-\pi f_{sy}}^{+\pi f_{sy}} \Phi_{ee}(\omega_x, \omega_y)d\omega_x d\omega_y \tag{16}$$

where terms f_{sx} and f_{sy} are the spatial sampling frequencies in horizontal and vertical directions. From (16), if $\Phi_{ee}(\omega_x, \omega_y)$ is smaller, σ^2_e will be smaller. Also, from [33]

$$R(D) = \frac{1}{2}\log_2(\frac{\sigma^2_e}{D}). \tag{17}$$

Under the same bit-rate, if σ^2_e is smaller, D (MSE) will be smaller, and the coding efficiency will be better. Therefore, given the same bit-rate, if PSD $\Phi_{ee}(\omega_x, \omega_y)$ is smaller, the coding efficiency will be better.

In JU-DFMC, one HQF and the following LQFs comprise a group of pictures (GOP). The total PSD of all frames in a GOP is utilized as performance measure of bit allocation. Assuming the overall bit-rate of a GOP is fixed, the bit allocation between HQF and LQFs can be changed. The changed bits in LQFs are assumed to be averagely allocated to every LQF, and thus the changed MSE in every LQF is nearly the same. With the change of bit allocation between HQF and LQFs under the overall bit-rate of the GOP, the PSD in HQF and LQFs will change, when the total PSD is the smallest, the coding efficiency is the best, and then the bit allocation ratio between HQF bits and average LQF bits is the best.

The total PSD can be simplified to a concise performance measure. Suppose before the change of bit allocation, the PSD in frame i and the total PSD in the GOP are separately denoted as $\widehat{\Phi}_{eei}(\Lambda)$ and $\widehat{\Phi}_{eeT}(\Lambda)$. After the change of bit allocation, the PSD in frame i and the total PSD in the GOP are separately denoted as $\breve{\Phi}_{eei}(\Lambda)$ and $\breve{\Phi}_{eeT}(\Lambda)$. Assume a GOP has n frames. Then we have

$$\widehat{\Phi}_{eeT}(\Lambda) = \sum_{i=1}^{n}\widehat{\Phi}_{eei}(\Lambda) \tag{18}$$

$$\breve{\Phi}_{eeT}(\Lambda) = \sum_{i=1}^{n}\breve{\Phi}_{eei}(\Lambda). \tag{19}$$

With the change of bit allocation, the change of total PSD is

$$\breve{\Phi}_{eeT}(\Lambda) - \widehat{\Phi}_{eeT}(\Lambda) = \sum_{i=1}^{n}(\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda)). \tag{20}$$

In (20), $\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda)$ ($i = 1, 2, \ldots, n$) is the change of PSD in every frame and can be calculated by (see Appendix C)

$$\begin{aligned}&\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda)\\ &= \tfrac{1}{4}h(\breve{\Phi}_{eei\_1}(\Lambda) - \widehat{\Phi}_{eei\_1}(\Lambda))\\ &\quad + \tfrac{1}{4}h(\breve{\Phi}_{eei\_2}(\Lambda) - \widehat{\Phi}_{eei\_2}(\Lambda))\\ &\quad + \tfrac{1}{2}\Phi_{ss}(\Lambda)((2 - \breve{P}_{i_2}(\Lambda))(2 - \breve{P}_{i_1}(\Lambda))\\ &\quad - (2 - \widehat{P}_{i_2}(\Lambda))(2 - \widehat{P}_{i_1}(\Lambda))).\end{aligned} \tag{21}$$

In (21), $\breve{\Phi}_{eei\_k}(\Lambda)$ and $\widehat{\Phi}_{eei\_k}(\Lambda)$ ($k = 1$ or 2) represent the PSD in the kth reference frame before and after the change of bit allocation, respectively. $(2 - \breve{P}_{i_2}(\Lambda))(2 - \breve{P}_{i_1}(\Lambda))$ and $(2 - \widehat{P}_{i_2}(\Lambda))(2 - \widehat{P}_{i_1}(\Lambda))$ represent the values of $(2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda))$ before and after the change of bit allocation, respectively. $P_{i_2}(\Lambda)$ and $P_{i_1}(\Lambda)$ are the displacement error pdfs in frame i when the reference is separately an LTR and an STR in JU-DFMC. The calculation of $\breve{\Phi}_{eei\_k}(\Lambda) - \widehat{\Phi}_{eei\_k}(\Lambda)$ is the same as $\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda)$, therefore in the iterative formula (21), $(2 - \breve{P}_{i_2}(\Lambda))(2 - \breve{P}_{i_1}(\Lambda)) - (2 - \widehat{P}_{i_2}(\Lambda))(2 - \widehat{P}_{i_1}(\Lambda))$ plays a dominant role in $\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda)$. From (20), $\sum_{i=1}^{n}((2 - \breve{P}_{i_2}(\Lambda))(2 - \breve{P}_{i_1}(\Lambda)) - (2 - \widehat{P}_{i_2}(\Lambda))(2 - \widehat{P}_{i_1}(\Lambda)))$ plays a dominant role in $\breve{\Phi}_{eeT}(\Lambda) - \widehat{\Phi}_{eeT}(\Lambda)$. This means that the change of $(2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda))$ plays a dominant role in the change of PSD in a frame, and the change of $\sum_{i=1}^{n}((2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda)))$ plays a key role in the change of total PSD in the GOP.

With the change of bit allocation between HQF and LQFs under the same overall bit-rate of the GOP, if $\sum_{i=1}^{n}((2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda)))$ increases (or decreases), the value of total PSD will also increase (or decrease). When $\sum_{i=1}^{n}((2 -$

$P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda)))$ is the smallest, the total PSD is the smallest, the coding efficiency is the best. Therefore, the value of $\sum_{i=1}^{n}((2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda)))$ is utilized instead of total PSD as the performance measure of bit allocation.

$(2-P_{i_2}(\Lambda))(2-P_{i_1}(\Lambda))$ can be further simplified. According to (3), $P_{i_2}(\Lambda)$ and $P_{i_1}(\Lambda)$ are within (0, 1). If $P_{i_2}(\Lambda)$ or $P_{i_1}(\Lambda)$ increases, the value of $(2 - P_{i_2}(\Lambda))$ or $(2 - P_{i_1}(\Lambda))$ will decrease. So $(2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda))$ has an inverse relationship with $P_{i_2}(\Lambda)P_{i_1}(\Lambda)$, which can be calculated from (3) as follows:

$$\begin{aligned} P_{i_2}(\Lambda)P_{i_1}(\Lambda) &= e^{-2\pi\sigma^2_{\triangle\_JLi}(\omega_x^2+\omega_y^2)} \times e^{-2\pi\sigma^2_{\triangle\_JSi}(\omega_x^2+\omega_y^2)} \\ &= e^{-2\pi(\sigma^2_{\triangle\_JLi}+\sigma^2_{\triangle\_JSi})(\omega_x^2+\omega_y^2)} = e^{-2\pi(\sigma^2_{\triangle\_Pi})(\omega_x^2+\omega_y^2)} \end{aligned} \tag{22}$$

where

$$\sigma^2_{\triangle\_Pi} = \sigma^2_{\triangle\_JSi} + \sigma^2_{\triangle\_JLi}. \tag{23}$$

$\sigma^2_{\triangle\_JLi}$ is the DEV in frame i when the reference frame is a LTR in JU-DFMC, $\sigma^2_{\triangle\_JSi}$ is the DEV in frame i when the reference frame is an STR in JU-DFMC.

From (22), $\sigma^2_{\triangle\_Pi}$ has an inverse relationship with $P_{i_2}(\Lambda)P_{i_1}(\Lambda)$, then it has a direct relationship with $(2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda))$. Therefore, $\sigma^2_{\triangle\_Pi}$ is utilized instead of $(2 - P_{i_2}(\Lambda))(2 - P_{i_1}(\Lambda))$ as the performance measure of bit allocation for frame i and thus $\sum_{i=1}^{n}\sigma^2_{\triangle\_Pi}$ is utilized as the performance measure of bit allocation for the GOP. With the change of bit allocation between HQF and LQFs, when $\sum_{i=1}^{n}\sigma^2_{\triangle\_Pi}$ is the smallest, the overall power special density is the smallest, the bit allocation ratio between HQF bits and average LQF bits a GOP is the best.

In the short term, the DEV $\sigma^2_{\triangle}$ is nearly directly proportional to the PEV σ^2_e [23]. Then in every GOP, $\sigma^2_{\triangle\_JLi}$ and $\sigma^2_{\triangle\_JSi}$ are nearly directly proportional to $\sigma^2_{e\_JLi}$ and $\sigma^2_{e\_JSi}$, which are the PEVs in frame i when the reference frame is LTR and STR, respectively. So $\sigma^2_{e\_Pi} = \sigma^2_{e\_JLi} + \sigma^2_{e\_JSi}$ and $\sum_{i=1}^{n}\sigma^2_{e\_Pi}$ can be used instead of $\sigma^2_{\triangle\_Pi}$ and $\sum_{i=1}^{n}\sigma^2_{\triangle\_Pi}$ as the performance measure of bit allocation for frame i and for a GOP, respectively.

If MSE in the LTR changes, the change of PEV $\sigma^2_{e\_JLi}$ in each frame i is nearly the same. If the change of MSE in different STRs is the same, the change of PEV $\sigma^2_{e\_JSi}$ in each frame i is nearly the same as well. Then with the change of MSE in LTR and STRs, the change of $\sigma^2_{e\_Pi}$ in each frame is nearly the same. Therefore, $\sigma^2_{e\_Pi}$ can be used instead of $\sum_{i=1}^{n}\sigma^2_{e\_Pi}$ as the performance measure of bit allocation for the GOP.

Finally, $\sigma^2_{e\_P} = \sigma^2_{e\_JL2} + \sigma^2_{e\_JSA}$ is utilized as the performance measurement of bit allocation for the GOP in accordance with that in the optimal LTR selection.

B. Bit Allocation

1) *Step 1—GOP Level Bit Allocation:* In the jth GOP, GOP length is initialized as N for performing GOP level bit allocation. If j is equal to 1, N is set to 10. Otherwise, N is set to the actual GOP length in the previous GOP. Suppose the overall remaining frames waiting to be encoded are M, while the remaining bits are R_r. The target bit allocation $T(j)$ for the jth GOP is calculated as

$$T(j) = \frac{N}{M} \times R_r. \tag{24}$$

2) *Step 2—Obtaining Bit Allocation Ratio:* After we obtained the GOP level bit allocation, the bit allocation ratio Ra between HQF bits and average LQF bits in the jth GOP is calculated. If j is equal to 1, Ra is initialized as 4. Otherwise, Ra is updated from the previous GOP as follows.

After encoding the $(j-1)$th GOP, the bit allocation and MSEs in HQF and LQFs is obtained and fixed. Assuming the overall bit-rate of the $(j-1)$th GOP is fixed, if the bit allocation between HQF and LQFs in the $(j-1)$th GOP changes (suppose the changed bit allocation in every LQF is the same) based on the fixed bit allocation, the changed MSE in HQF and LQFs corresponding to the changed bit-rate can be approximately calculated from the derivation of function $R(D)$ in (17)

$$\frac{dR}{dD} = \frac{1}{2\ln 2} \times \frac{D}{\sigma_e^2} \times (-\frac{\sigma_e^2}{D^2}) = -\frac{1}{2\ln 2} \times \frac{1}{D} \tag{25}$$

then

$$\Delta D = -2\ln 2 \times D \times \Delta R \tag{26}$$

where ΔD is the change of MSE, ΔR is the change of bit-rate, and D is the MSE in the frame.

After adding the changed MSE to the fixed MSE, and adding the changed bit-rate to the fixed bit-rate, the different MSE corresponding to different bit allocation in HQF and LQFs of the $(j-1)$th GOP can be calculated.

The MSE of the reference frame is nearly directly proportional to the PEV σ_e^2 from the reference frame, thus if the MSEs in HQF and LQFs are known, $\sigma^2_{e\_JL2}$ from LTR and the average $\sigma^2_{e\_JSA}$ from STRs can be calculated.

In the strategy of bit allocation change in the work, the bit allocation change step in HQF is $R_H \times 1\%$, and the changed bit allocation in HQF is within the range $(-R_H \times 10\%, +R_H \times 10\%)$, where R_H is the actual bit allocation in HQF after encoding the $(j-1)$th GOP. The bit allocation change in HQF is averagely compensated from LQFs. For each changed bit allocation case, the $\sigma^2_{e\_P}$ $(\sigma^2_{e\_JSA} + \sigma^2_{e\_JL2})$ is calculated, respectively. The bit allocation ratio which brings the smallest $\sigma^2_{e\_P}$ is reserved.

As in adjacent GOPs, the bit allocation ratios between HQF bits and average LQF bits have little difference, so the reserved bit allocation ratio in the $(j-1)$th GOP is selected as the bit allocation ratio in the jth GOP.

3) *Step 3—Frame Level Bit Allocation:* The bit allocation in HQF T_H and the average bit allocation in LQFs T_{L_ave} in the jth GOP are separately calculated using the reserved bit allocation ratio and GOP level bit allocation

$$T_H = \frac{Ra}{Ra + (N-1) \times 1} \times T(j) \tag{27}$$

$$T_{L\_ave} = \frac{1}{Ra + (N-1) \times 1} \times T(j). \tag{28}$$

4) *Step 4—Quantization Parameter Determination:* For determining quantization parameter (QP) in HQF and LQFs, the rate and quantization parameter models are similar to those in [34] with some modifications. For LQFs and HQFs, the quadratic rate control model is updated and stored, respectively. In every GOP, to maintain the smoothness of LQFs quality, the average bit allocation in LQFs is utilized to calculate the average QP (Q) in LQFs, and then in different LQFs, the QP is slightly adjusted based on the average QP. In the work, for the first LQF in the GOP, the QP is set to $Q+1$. For the middle LQFs (from the second LQF to the $(2N/3)th$ LQF, where N is the actual GOP length of previous GOP), the QP is set to Q. For the following LQFs, the QP is set to $Q-1$.

V. Extension to Noisy Channels

Based on the proposed adaptive JU-DFMC, an error resilient JU-DFMC prediction structure is first given. Then, adaptive LTR selection and bit allocation with respect to different packet loss rates are presented.

A. Error Resilient JU-DFMC

For the video transmission over error prone channels, the end-to-end distortion model [35], [36] is extended in this paper to analyze the error propagation. In the end-to-end distortion model, transmission error rates of two data partitions A (the header information) and B (transformed coefficients of the inter-coded blocks) are represented by p_A and p_B, respectively. Let f_n^i be the original value of pixel i in frame n, and let $\hat{f}_n^i$ and $\tilde{f}_n^i$ be the reconstructed values in the encoder and decoder, respectively. And suppose it references pixel k in frame *ref*. Then, the expected inter-mode end-to-end distortion in decoder is represented in [35] as

$$\begin{aligned} d(n,i) &= E\{(f_n^i - \tilde{f}_n^i)^2\} \\ &= (1-p_A)E\{(\hat{f}_{ref}^k - \tilde{f}_{ref}^k)^2\} \\ &\quad +(1-p_A)(1-p_C)E\{(f_n^i - \hat{f}_n^i)^2\} \\ &\quad +(1-p_A)p_C E\{(f_n^i - \hat{f}_{ref}^k)^2\} + p_A(E\{(f_n^i - \hat{f}_{n-1}^i)^2\} \\ &\quad +E\{(\hat{f}_{n-1}^i - \tilde{f}_{n-1}^i)^2\}) \\ &= (1-p_A)d_{ep\_ref} + (1-p_A)(1-p_C)d_s \\ &\quad +(1-p_A)p_C d_{ec\_ref\_o} + p_A(d_{ec\_prev\_o} + d_{ep\_prev}). \end{aligned} \tag{29}$$

In (29), $d(n,i)$ is the expected end-to-end distortion in decoder, d_{ep_ref} is the error propagated distortion from the reference frame, d_s denotes the source distortion, $d_{ec\_ref\_o}$ indicates the original referenced error-concealment distortion, $d_{ec\_prev\_o}$ denotes the original previous error-concealment distortion, d_{ep_prev} denotes the error-propagated distortion from the previous frame.

By the observation of (29), in the expected end-to-end distortion $d(n,i)$, the percentage of error propagated distortion d_{ep}(including d_{ep_ref} and d_{ep_prev}) is larger than the source distortion d_s, the error-concealment distortions $d_{ec\_ref\_o}$, and $d_{ec\_prev\_o}$. Furthermore, the error propagated distortion d_{ep} increases frame by frame in video decoding. Therefore, if there is an error in the current frame, the image quality in the following frames will be heavily affected.

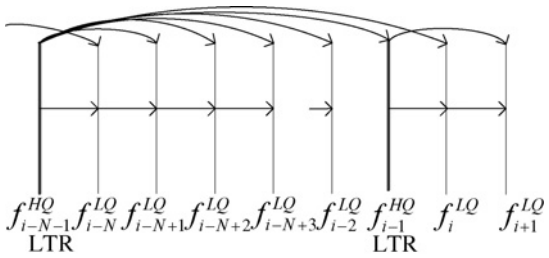

Fig. 8. Error resilient JU-DFMC structure.

To reduce the error propagation, an error resilient JU-DFMC structure is proposed. As shown in Fig. 8, for the HQF in each GOP, only the previous LTR (HQF) is utilized for prediction, and for LQFs, the prediction structure is unchanged. Then, when transmission errors occur in LQFs of the current GOP, the error will not be propagated to the following HQFs and GOPs.

B. Optimal LTR Selection and Bit Allocation in Error Resilient JU-DFMC

To deduce the average increase of error propagation in a GOP, we adopt [35]

$$\begin{aligned} d_{ep} &= (1-p_A)(1-p_C)d_{ep\_ref} + (1-p_A)p_C(d_{ec\_ref\_r} + d_{ep\_ref}) \\ &\quad + p_A(d_{ec\_prev\_r} + d_{ep\_prev}) \end{aligned} \tag{30}$$

where d_{ep}, d_{ep_ref}, and d_{ep_prev} are the error propagated distortions in the current HQF, the previous LTR, and the previous LQF, respectively. Since d_{ep_prev} only accounts for a small percentage p_A in d_{ep}, we adopt $p_A \times d_{ep\_prev} \approx p_A \times d_{ep\_ref}$. So (30) can be rewritten as

$$d_{ep} \approx d_{ep\_ref} + (1-p_A)p_C d_{ec\_ref\_r} + p_A d_{ec\_prev\_r}. \tag{31}$$

If the GOP length is N, the average increase of the error propagated distortion d_{ep_ave} in every frame of the GOP is

$$\begin{aligned} d_{ep\_ave} &= \frac{d_{ep} - d_{ep\_ref}}{N} \\ &\approx \frac{(1-p_A)p_C d_{ec\_ref\_r} + p_A d_{ec\_prev\_r}}{N}. \end{aligned} \tag{32}$$

From (32), we can see that to reduce the average increase of error propagated distortion, two factors can be adjusted. The first is to increase of GOP length N. The second is to decrease error concealment distortion values $d_{ec\_ref\_r}$ and $d_{ec\_prev\_r}$, which needs to increase the bit allocation in the HQF. If more bits are allocated to HQF, the prediction performance will be better and the error concealment distortion will be smaller.

In determining the LTR (HQF) selection and bit allocation in error resilient JU-DFMC, the method is the same as that in the proposed adaptive JU-DFMC. But the PEV utilized in calculating performance measure is not computed from source distortion but the expected end-to-end distortion $d(n,i)$ in (29). Since the distortion of the reference frame is nearly directly

proportional to the PEV σ_e^2 from the reference frame, and the expected end-to-end distortion $d(n,i)_{LQ}$ in LQF and $d(n,i)_{HQ}$ in HQF are separately calculated by (29), the virtual PEV $\tilde{\sigma}_{e\_JS}^2$ from $d(n,i)_{LQ}$ and $\tilde{\sigma}_{e\_JL}^2$ from $d(n,i)_{HQ}$ are separately calculated as

$$\tilde{\sigma}_{e\_JS}^2 = \frac{d(n,i)_{LQ}}{d_{S\_LQ}} \times \sigma_{e\_JS}^2 \tag{33}$$

$$\tilde{\sigma}_{e\_JL}^2 = \frac{d(n,i)_{HQ}}{d_{S\_HQ}} \times \sigma_{e\_JL}^2 \tag{34}$$

where d_{S_LQ} and d_{S_HQ} are the source distortion d_S in LQF and HQF, $\sigma_{e\_JS}^2$ and $\sigma_{e\_JL}^2$ are the PEV from the source distortion d_{S_LQ} and the source distortion d_{S_HQ}. With the same method as in the proposed adaptive JU-DFMC, the performance measure $\sigma_{e\_T}^2$ of LTR selection and performance measure $\sigma_{e\_P}^2$ of bit allocation in error resilient JU-DFMC are

$$\sigma_{e\_T}^2 = (\tilde{\sigma}_{e\_JL2}^2 + \tilde{\sigma}_{e\_JSA}^2)^{/2} \tag{35}$$

$$\sigma_{e\_P}^2 = \tilde{\sigma}_{e\_JL2}^2 + \tilde{\sigma}_{e\_JSA}^2. \tag{36}$$

In (35) and (36), $\tilde{\sigma}_{e\_JL2}^2$ is the virtual PEV in the second LQF of the GOP when the reference frame is LTR; $\tilde{\sigma}_{e\_JSA}^2$ is the average virtual PEV in LQFs (from the second LQF to the last encoded LQF) of the GOP when the reference frame is STR.

However, in the rate-distortion mode decision, the distortion is not the overall end-to-end distortion $d(n,i)$ but the source distortion d_S.

VI. Experimental Results and Discussions

To evaluate the general performance of the proposed adaptive JU-DFMC, we integrated the proposed methods into the H.264/AVC reference software JM10.2. In the proposed adaptive JU-DFMC, the first P frame is set as the first LTR, its allocated bits are initialized as four times the average bits of the following STRs. For the selection of the following LTR and the corresponding bit allocation, the proposed methods in Sections III and IV are adopted. In the CU-DFMC, two most recently decoded frames are used for motion compensation, LTRs are continuously updated and not allocated any extra rate. The test sequences are all encoded at 30 frames/s with 120 frames in total for each sequence. In motion estimation, the search range is ± 16. The entropy coder is context-adaptive binary arithmetic coding. Each row of macroblocks comprises a slice and is transmitted in a separate packet.

In Table II, the PSNRs of CU-DFMC (fixed QP in every frame), the JU-DFMC [16], and the proposed adaptive JU-DFMC are given respectively. The performance gain of the proposed adaptive JU-DFMC under different bit-rates is also presented. Compared with the JU-DFMC [16], the average PSNR gains obtained by the proposed adaptive JU-DFMC are 0.72, 0.53, 0.86, 0.44, 0.32, 0.35, 0.30, 0.48, and 0.21 dB in sequences *Mobile*, *Tempete*, *Waterfall*, *Container*, *News*, *Paris*, *Foreman*, *Silent*, and *Hall*, respectively.

Table III shows the average percentage of blocks in every frame that utilizes LTR or STR as reference frames. Intra mode

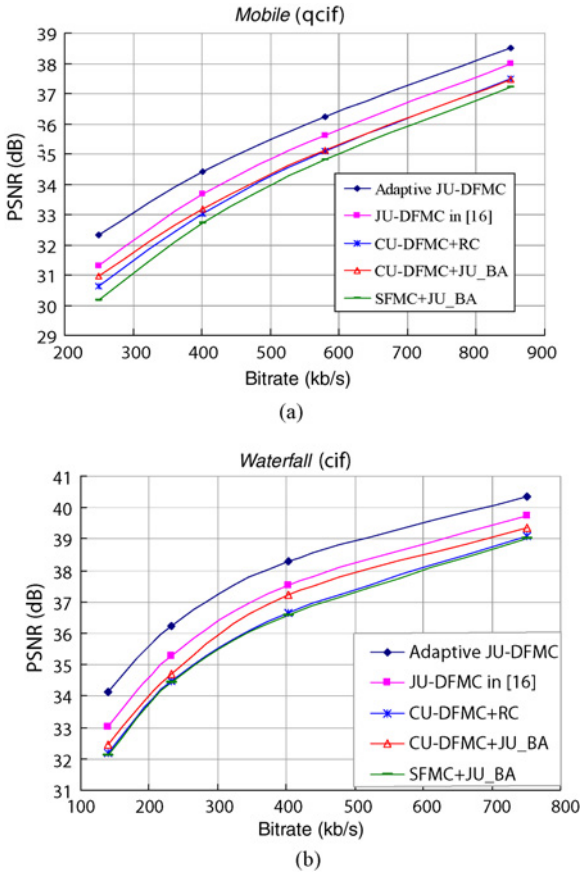

Fig. 9. Rate distortion curves for some sequences. (a) *Mobile* (qcif). (b) *Waterfall* (cif).

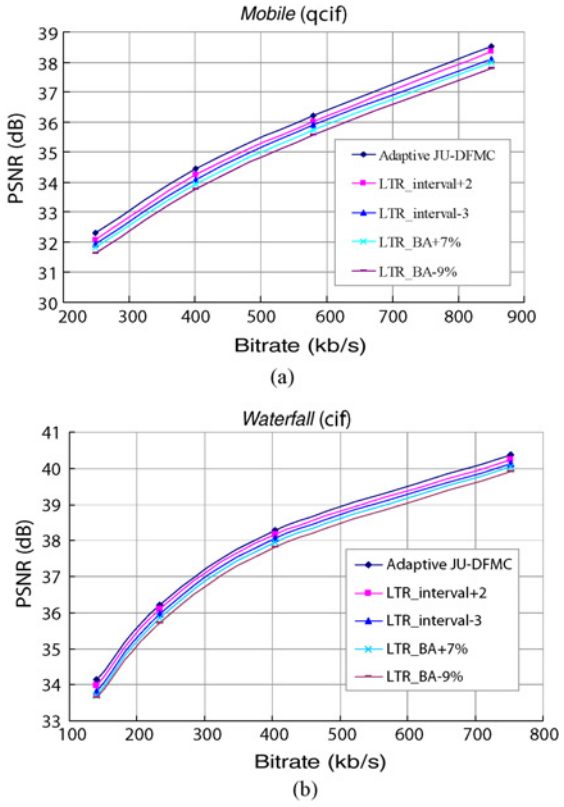

Fig. 10. Rate distortion curves under different LTR intervals and different bit allocations. (a) *Mobile* (qcif). (b) *Waterfall* (cif).

TABLE II

PERFORMANCE COMPARISON OF THE PROPOSED ADAPTIVE JU-DFMC WITH EXISTING DFMC SCHEMES

Sequence	Bit-rate (kb/s)	PSNR in CU-DFMC (dB)	PSNR in JU-DFMC [16] (dB)	PSNR in Proposed Adaptive JUDFMC (dB)	Gain Over CU-DFMC		Gain Over JU-DFMC [16]	
					PSNR	Average Gain (dB)	PSNR	Average Gain (dB)
Mobile	249.12	30.63	31.32	32.32	1.69	1.31	1.00	0.72
(qcif)	401.23	33.03	33.68	34.43	1.40		0.75	
	580.10	35.08	35.62	36.22	1.14		0.60	
	850.12	37.49	37.99	38.51	1.02		0.52	
Tempete	149.53	30.36	31.11	31.87	1.51	1.14	0.76	0.53
(qcif)	310.27	33.83	34.43	34.99	1.16		0.56	
	507.21	36.56	37.10	37.48	0.92		0.38	
	673.68	38.14	38.68	39.09	0.95		0.41	
Waterfall	139.92	32.11	33.04	34.14	2.03	1.72	1.10	0.86
(cif)	232.82	34.42	35.27	36.21	1.79		0.94	
	403.63	36.59	37.54	38.29	1.70		0.75	
	751.94	39.03	39.73	40.37	1.34		0.64	
Container	17.66	32.88	33.97	34.56	1.68	1.39	0.59	0.44
(qcif)	27.69	34.81	35.79	36.19	1.38		0.40	
	46.47	36.86	37.88	38.24	1.38		0.36	
	80.19	38.91	39.64	40.04	1.13		0.40	
News	100.36	34.48	35.12	35.44	0.96	0.98	0.32	0.32
(cif)	149.87	36.68	37.32	37.67	0.99		0.35	
	200.55	38.23	38.91	39.26	1.03		0.35	
	292.76	40.05	40.72	40.97	0.92		0.25	
Paris	135.22	32.02	32.91	33.23	1.21	1.22	0.32	0.35
(qcif)	200.83	34.61	35.52	35.82	1.21		0.30	
	270.02	36.54	37.43	37.88	1.34		0.45	
	317.80	37.86	38.65	38.99	1.13		0.34	
Foreman	70.47	33.92	34.28	34.54	0.62	0.73	0.26	0.30
(qcif)	141.12	36.73	37.27	37.64	0.91		0.37	
	194.78	38.02	38.48	38.73	0.71		0.25	
	255.54	39.33	39.67	39.99	0.66		0.32	
Silent	225.45	35.85	36.55	36.99	1.14	1.25	0.44	0.48
(cif)	349.76	38.02	38.93	39.23	1.21		0.30	
	539.24	40.08	40.82	41.60	1.52		0.78	
	708.35	41.85	42.59	42.99	1.14		0.40	
Hall	34.25	34.54	35.13	35.29	0.75	0.75	0.16	0.21
(qcif)	45.12	36.41	36.82	37.03	0.62		0.21	
	68.82	38.32	39.13	39.36	1.04		0.23	
	94.18	39.97	40.34	40.57	0.60		0.23	

TABLE III

AVERAGE PERCENTAGE OF BLOCKS IN EACH FRAME WHOSE REFERENCE FRAME IS LTR OR STR

Sequence	Adaptive JU-DFMC		CU-DFMC	
	Reference Frame is LTR (%)	Reference Frame is STR (%)	Reference Frame is LTR (%)	Reference Frame is STR (%)
Mobile (qcif)	63.91	36.09	24.61	75.39
Tempete (qcif)	49.85	50.15	8.20	91.80
Waterfall (cif)	68.22	31.78	8.50	91.50
Container (qcif)	52.87	47.13	1.12	98.88
News (cif)	68.93	31.07	1.08	98.92
Paris (qcif)	42.35	57.65	5.20	94.80
Foreman (qcif)	37.41	62.52	14.20	85.80
Silent (cif)	45.62	54.36	1.23	98.77
Hall (qcif)	64.35	35.65	0.80	99.20

TABLE IV
PERFORMANCE COMPARISON OF THE ERROR RESILIENT JU-DFMC WITH OTHER SCHEMES

Sequence	Schemes	PSNR of Different Packet Loss Rates (dB)				Original PSNR (dB)			
		3%	5%	10%	20%	3%	5%	10%	20%
Mobile (qcif) 650.35 kb/s	2HMC [26]	31.49	30.21	28.23	26.14	35.32	35.32	35.32	35.32
	CU-DFMC +DPRD [36]	31.51	30.26	28.34	26.22	33.01	31.94	30.22	28.31
	Adaptive JU-DFMC	33.56	30.32	28.21	26.08	36.79	36.79	36.79	36.79
	Error resilient JU-DFMC	34.42	32.67	31.85	30.68	36.45	36.33	36.21	36.10
Tempete (qcif) 500.26 kb/s	2HMC [26]	32.38	31.12	29.25	27.16	35.92	35.92	35.92	35.92
	CU-DFMC +DPRD [36]	32.41	30.95	29.46	27.52	33.51	32.46	31.16	29.62
	Adaptive JU-DFMC	32.47	31.14	29.21	27.09	37.33	37.33	37.33	37.33
	Error resilient JU-DFMC	33.56	32.86	31.76	30.58	37.03	36.92	36.79	36.67
Waterfall (cif) 480.78 kb/s	2HMC [26]	33.04	32.08	30.46	28.42	36.75	36.75	36.75	36.75
	CU-DFMC +DPRD [36]	33.06	32.12	30.53	28.65	34.58	33.79	32.43	30.78
	Adaptive JU-DFMC	33.12	32.13	30.42	28.38	38.79	38.79	38.79	38.79
	Error resilient JU-DFMC	34.96	34.25	33.67	32.43	38.50	38.41	38.31	38.22
Container (qcif) 75.24 kb/s	2HMC [26]	35.55	34.09	32.54	30.71	38.18	38.18	38.18	38.18
	CU-DFMC +DPRD [36]	35.56	34.12	32.57	30.87	36.88	35.56	34.12	32.68
	Adaptive JU-DFMC	35.63	34.14	32.52	30.62	39.84	39.84	39.84	39.84
	Error resilient JU-DFMC	36.82	35.92	34.69	33.87	39.53	39.43	39.32	39.20
News (cif) 300.57 kb/s	2HMC [26]	36.31	34.96	33.50	31.51	39.98	39.98	39.98	39.98
	CU-DFMC +DPRD [36]	36.43	35.01	33.56	31.72	37.82	36.61	35.12	33.52
	Adaptive JU-DFMC	36.52	35.03	33.41	31.42	41.18	41.18	41.18	41.18
	Error resilient JU-DFMC	37.23	36.58	35.92	35.23	40.89	40.81	40.72	40.65
Paris (qcif) 200.83 kb/s	2HMC [26]	31.94	30.78	29.17	27.84	33.94	33.94	33.94	33.94
	CU-DFMC +DPRD [36]	32.01	30.92	29.26	27.94	32.92	32.02	30.56	29.04
	Adaptive JU-DFMC	32.13	30.92	29.08	27.73	35.82	35.82	35.82	35.82
	Error resilient JU-DFMC	33.32	32.78	32.36	31.82	35.38	35.30	35.19	35.09
Foreman (qcif) 150.52 kb/s	2HMC [26]	34.26	33.13	30.67	28.01	36.55	36.55	36.55	36.55
	CU-DFMC +DPRD [36]	34.45	33.32	30.97	28.27	35.82	34.87	32.99	30.98
	Adaptive JU-DFMC	33.82	33.29	30.62	27.98	37.87	37.87	37.87	37.87
	Error resilient JU-DFMC	35.57	35.37	33.89	31.87	37.53	37.41	37.26	37.14
Silent (cif) 404.26 kb/s	2HMC [26]	36.94	35.97	33.40	31.70	38.17	38.17	38.17	38.17
	CU-DFMC +DPRD [36]	37.01	36.02	33.56	31.92	37.12	36.34	35.12	33.72
	Adaptive JU-DFMC	37.19	36.04	33.32	31.67	40.02	40.02	40.02	40.02
	Error resilient JU-DFMC	37.84	37.32	36.27	35.12	39.74	39.66	39.54	39.45
Hall (qcif) 41.28 kb/s	2HMC [26]	35.16	35.01	33.78	32.64	34.62	34.62	34.62	34.62
	CU-DFMC +DPRD [36]	35.18	35.05	33.89	32.78	35.52	35.54	34.57	33.69
	Adaptive JU-DFMC	35.23	35.09	33.73	32.61	36.49	36.49	36.49	36.49
	Error resilient JU-DFMC	35.87	35.67	34.83	33.68	36.22	36.12	36.04	35.94

blocks are not included in the experimental results. It can be seen that in the adaptive JU-DFMC, the percentage of blocks which utilize LTR as reference frame is higher than that in CU-DFMC.

The R-D curves in some sequences achieved from the proposed adaptive JU-DFMC (Adaptive JU-DFMC), JU-DFMC in [16], CU-DFMC with rate control [34] (CU-DFMC + RC) are shown in Fig. 9. Furthermore, CU-DFMC structure with proposed JU-DFMC bit allocation profile (CU-DFMC + JU_BA), single reference frame motion compensation with proposed JU-DFMC bit allocation profile (SFMC + JU_BA) are also presented in the figure. In CU-DFMC + JU_BA and SFMC + JU_BA, the bit allocation in every frame is the same as that in the proposed adaptive JU-DFMC scheme. But for every frame in CU-DFMC + JU_BA, only the two recently decoded frames are utilized for motion compensation. For every frame in SFMC + JU_BA, only one recently decoded frame is utilized for motion compensation.

The performance of CU-DFMC + JU_BA is worse than the proposed adaptive JU-DFMC. This is because in CU-DFMC + JU_BA, some frames following HQF do not utilize the HQF as references to improve coding performance. SFMC + JU_BA also has worse performance than the proposed

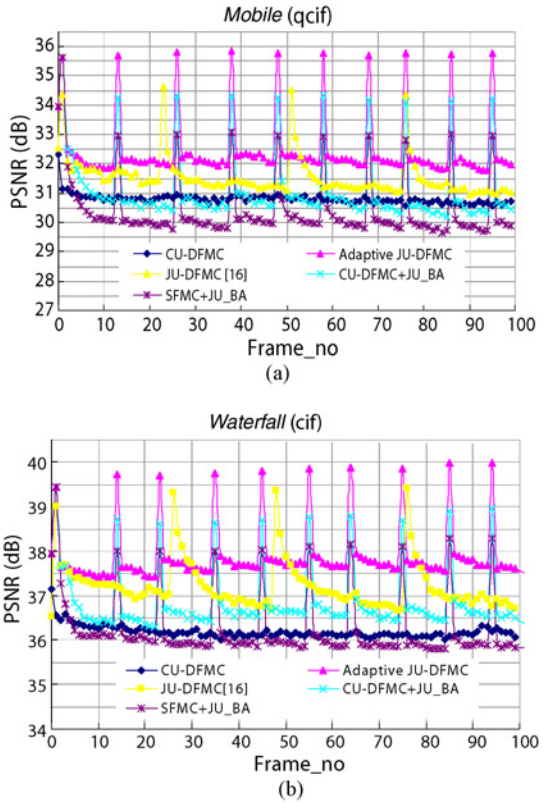

Fig. 11. Frame-by-frame PSNR in test sequences *Mobile* and *Waterfall* under the same bit-rate. (a) *Mobile* (qcif). (b) *Waterfall* (cif).

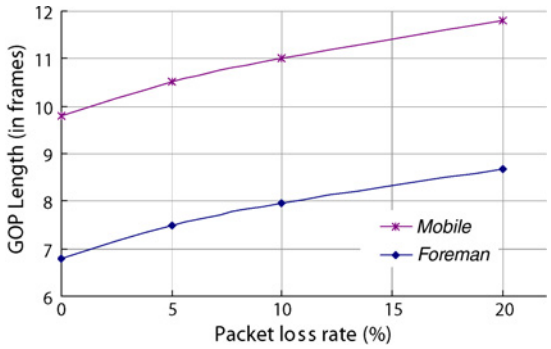

Fig. 12. Average jump updating parameter of LTR.

adaptive JU-DFMC. The reason is that only one frame is utilized for motion compensation.

If the proposed bit allocation method does not change, but the LTR interval is two frames larger or three frames less than the actual LTR interval proposed in adaptive JU-DFMC (separately named as LTR_interval +2 and LTR_interval −3), the performance is given in Fig. 10. If the proposed LTR interval does not change, but LTR bit allocation is 7% larger or 9% less than the actual LTR bit allocation proposed in adaptive JU-DFMC (separately named as LTR_BA + 7% and LTR_BA−9%), the performance is given in Fig. 10 as well. The performance of LTR_interval +2, LTR_interval −3, LTR_BA + 7%, and LTR_BA − 9% is slightly lower than that in the proposed adaptive JU-DFMC scheme. It shows that the proposed adaptive LTR selection and bit allocation are effective.

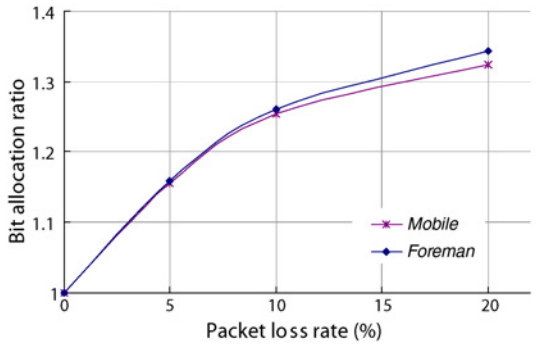

Fig. 13. Ratio of bit allocation in LTR.

The frame-by-frame PSNR comparison of proposed adaptive JU-DFMC (Adaptive JU-DFMC), JU-DFMC in [16], CU-DFMC + JU_BA, SFMC + JU_BA and CU-DFMC (fixed QP in every frame) under the same bit-rate in sequences *Mobile* (255.71 kb/s), *Tempete* (220.62 kb/s), *Waterfall* (362.82 kb/s) and *Container* (71.64 kb/s) is shown in Fig. 11. It can be seen that the PSNR deterioration after the HQF in [16] is much more graceful than the proposed scheme, but the PSNR in majority of LQFs in the proposed scheme is better than that in [16] and CU-DFMC. The PSNR in HQFs in the proposed adaptive JU-DFMC is much larger than that of LQFs. This will introduce objectionable pulses in quality over time. But as mentioned in [14], when the PSNR of the sequence is higher than 30 dB, the pulsing is not perceptible. Furthermore, the pulsing can benefit some applications, such as video surveillance, the HQF can create higher image quality of video surveillance content. In the video communication or multimedia, the HQF can be quantized by a larger QP when it is playbacked in the decoder to maintain similar image quality as that in LQFs; then the overall quality fluctuation in the proposed scheme is less than that in [16]. The image quality in LQFs can likewise be enhanced using the previous and the following HQFs in decoder.

The experimental results of the error resilient JU-DFMC in decoder are repeated 100 times using the bit error sequences which are transmitted from the encoder via the error-prone channels [38]. Although the method in [26] is used without the need of live encoding, whereas the proposed error resilient JU-DFMC needs to adaptively adjust the coding parameters, the performance of 2HMC [26] is listed to provide useful information. CU-DFMC + Dewan Perwakilan Rakyat Daerah (DPRD) [36] is also utilized for performance comparison here. In Table IV, the experimental results of the proposed error resilient JU-DFMC (error resilient JU-DFMC), CU-DFMC + DPRD [36], proposed adaptive JU-DFMC (Adaptive JU-DFMC), 2HMC [26] are listed and compared.

Under the given bit-rate, the original PSNR of proposed sequences without error is shown on the right-hand side of Table IV. The PSNRs of the error resilient JU-DFMC and the other coding schemes measured after packet loss rate are listed on the left side of Table IV. 2HMC [26] has slightly lower performance than CU-DFMC + DPRD [36], it is because error propagation in [26] is alleviated, but not terminated. Adaptive JU-DFMC is slightly better than CU-DFMC + DPRD [36] in

low packet loss rate. This is because lots of blocks in LQF utilize LTR as their reference frame, if some errors occur in LQF while the error occurred areas are not utilized as reference for the next LQF, the error will not be propagated to the following frames. And the other reason is that the original PNSR of the adaptive JU-DFMC is better than CU-DFMC + DPRD [36]. However, at a high packet loss rate, the performance of the adaptive JU-DFMC is less than that in DFMC + DPRD [36]. This is because the adaptive JU-DFMC cannot terminate error propagation.

Compared to CU-DFMC + DPRD [36], the proposed error resilient JU-DFMC can achieve a maximum gain of 4.46 dB on *Mobile* sequence at a packet loss rate of 20% and an average gain of 2.23 dB on all the sequences under the different packet loss rates. This is because in CU-DFMC + DPRD [36], the error propagation is terminated by inserting intra mode blocks, but the R-D cost is huge, and sometimes the error cannot be accurately predicted in encoder. Compared with the adaptive JU-DFMC, the proposed error resilient JU-DFMC can achieve a maximum gain of 4.60 dB on *Mobile* sequence at packet loss rate at 20% and an average gain of 2.27 dB on the all sequences under the different packet loss rates. This shows the proposed error resilient JU-DFMC is effective in eliminating error propagation. The elimination of error propagation comes from two factors, the first factor is that if an error occurred in LQFs in the error resilient JU-DFMC (this kind of error takes the largest percentage), the error propagation will be terminated in the last LQF of the GOP. The second factor is that even error occurs in HQF, the average error propagation speed is smaller than the other schemes since the HQF interval is large. However, HQF is more important than LQFs. More protection to HQFs is needed as indicated in [20].

In addition, Fig. 12 shows the average jump updating parameter of LTR at different packet loss rates. Fig. 13 shows the ratio of bit allocation for LTR under different packet loss rates compared with that without packet loss rate. It can be seen that with the increase of packet loss rate, the jump updating parameter and bits in HQF (LTR) increase adaptively.

VII. Conclusion

In this paper, an optimal LTR selection and bit allocation for JU-DFMC has been presented. First, the rate-distortion performance of MCP for DFMC was analyzed. Then based on the analysis, an adaptive JU-DFMC with the optimal LTR selection and the bit allocation was given. The proposed adaptive JU-DFMC can lead to better performance than previous JU-DFMC schemes. For video transmission over noisy channels, the error propagation of the proposed adaptive JU-DFMC was analyzed. Furthermore, an error resilient JU-DFMC considering the LTR selection and the bit allocation was presented. The error resilient DFMC can obtain a much better performance in video transmission over noisy channels. The proposed schemes can be used in multimedia applications and video surveillance systems. It can also be utilized to instruct the rental of extra bandwidth for the video transmission over cognitive radio. In the future, the rate control for the proposed scheme will be further exploited.

Appendix A

We know $0 < P_1^J(\Lambda) < P_1^{CH}(\Lambda)$ and $0 < P_2^J(\Lambda) < P_2^{CH}(\Lambda)$, $P_1^J(\Lambda) < 1$. Also, the difference between DEVs from STRs in JU-DFMC and CU-DFMC is much smaller than the DEV from STR in JU-DFMC; then $P_1^{CH}(\Lambda) - P_1^J(\Lambda) \ll P_1^J(\Lambda)$. From (10), we have

$$\begin{aligned} P^{HQ} &\approx (P_1^J(\Lambda) \times (\tfrac{1}{2}P_2^{CH}(\Lambda) - 1) - P_1^J(\Lambda) \\ &\quad \times (\tfrac{1}{2}P_2^J(\Lambda) - 1)) + (P_2^J(\Lambda) - P_2^{CH}(\Lambda)) \\ &= (P_1^J(\Lambda) \times \tfrac{1}{2} \times (P_2^{CH}(\Lambda) - P_2^J(\Lambda)) + (P_2^J(\Lambda) - P_2^{CH}(\Lambda)) \\ &= (1 - \tfrac{1}{2}P_1^J(\Lambda)) \times (P_2^J(\Lambda) - P_2^{CH}(\Lambda)) < 0. \end{aligned} \tag{A1}$$

Appendix B

Suppose $\widehat{\Phi}_{ee}(\Lambda)$ and $\overline{\Phi}_{ee}(\Lambda)$ are the R-D performance gain and loss when frame j is separately encoded as LQF and HQF in JU-DFMC. The difference of R-D performance gain $\widehat{\Phi}_{ee}(\Lambda)$ and R-D performance loss $\overline{\Phi}_{ee}(\Lambda)$ can be obtained as

$$\begin{aligned} &\widehat{\Phi}_{ee}(\Lambda) - \overline{\Phi}_{ee}(\Lambda) \\ &= \tfrac{1}{4}(h(\Phi^J_{ee\_1}(\Lambda) - \Phi^{CL}_{ee\_1}(\Lambda)) + h(\Phi^J_{ee\_2}(\Lambda) - \Phi^{CL}_{ee\_2}(\Lambda)) \\ &\quad - h(\Phi^{CH}_{ee\_1}(\Lambda) - \Phi^J_{ee\_1}(\Lambda)) - h(\Phi^{CH}_{ee\_2}(\Lambda) - \Phi^J_{ee\_2}(\Lambda))) \\ &\quad + \Phi_{ss}(P^{LQ} - P^{HQ}) \\ &= \tfrac{1}{2}h(\Phi^J_{ee\_1}(\Lambda) - \tfrac{1}{2}(\Phi^{CL}_{ee\_1}(\Lambda) + \Phi^{CL}_{ee\_2}(\Lambda))) \\ &\quad - \tfrac{1}{2}h(\tfrac{1}{2}(\Phi^{CH}_{ee\_1}(\Lambda) + \Phi^{CH}_{ee\_2}(\Lambda)) - \Phi^J_{ee\_2}(\Lambda)) \\ &\quad + \Phi_{ss}(P^{LQ} - P^{HQ}). \end{aligned} \tag{B1}$$

For every frame in CU-DFMC, the prediction performance from reference frames (LTR and STR) is roughly similar, then the PSD in every frame is roughly similar, $\Phi^{CL}_{ee\_1}(\Lambda) \approx \Phi^{CL}_{ee\_2}(\Lambda)$ and $\Phi^{CH}_{ee\_1}(\Lambda) \approx \Phi^{CH}_{ee\_2}(\Lambda)$. In JU-DFMC, the GOP length of the current GOP is nearly the same as that in the previous GOP. For the last several frames in the current GOP, in the coding of its STR, the prediction performance in the STR from its reference frames (its STR and LTR) is nearly the same as that in the coding of LTR (HQF). So whenever STR is encoded as an LQF or an HQF, the PSD $\Phi^J_{ee\_1}(\Lambda)$ in the STR is nearly the same as the PSD $\Phi^J_{ee\_2}(\Lambda)$ in LTR, $\Phi^J_{ee\_1}(\Lambda) \approx \Phi^J_{ee\_2}(\Lambda)$. So (B1) can be further rewritten as

$$\begin{aligned} &\widehat{\Phi}_{ee}(\Lambda) - \overline{\Phi}_{ee}(\Lambda) \\ &\approx \tfrac{1}{2}h(\Phi^J_{ee\_1}(\Lambda) - \Phi^{CL}_{ee\_1}(\Lambda)) - \tfrac{1}{2}h(\Phi^{CH}_{ee\_1}(\Lambda) \\ &\quad - \Phi^J_{ee\_1}(\Lambda)) + \Phi_{ss}(P^{LQ} - P^{HQ}) \\ &= \tfrac{1}{2}h(\widehat{\Phi}_{ee\_1}(\Lambda) - \overline{\Phi}_{ee\_1}(\Lambda)) + \Phi_{ss}(P^{LQ} - P^{HQ}). \end{aligned} \tag{B2}$$

In (B2), $\widehat{\Phi}_{ee\_1}(\Lambda)$ represents the performance gain when the STR is encoded as LQF in JU-DFMC; $\overline{\Phi}_{ee\_1}(\Lambda)$ represents the performance loss when the STR is encoded as HQF in JU-DFMC.

Appendix C

Before and after the change of bit allocation, the PSD in frame i is separately denoted as $\widehat{\Phi}_{eei}(\Lambda)$ and $\breve{\Phi}_{eei}(\Lambda)$. From

(4), we have

$$\begin{aligned}
&\breve{\Phi}_{eei}(\Lambda) - \widehat{\Phi}_{eei}(\Lambda) \\
&= \tfrac{1}{4}h(\breve{\Phi}_{eei_1}(\Lambda) - \widehat{\Phi}_{eei_1}(\Lambda)) + \tfrac{1}{4}h(\breve{\Phi}(\Lambda)_{eei_2} \\
&\quad - \widehat{\Phi}_{eei_2}(\Lambda)) + \Phi_{ss}(\Lambda) \\
&\quad \times(\tfrac{1}{2}\breve{P}_{i_1}(\Lambda)\breve{P}_{i_2}(\Lambda) - \breve{P}_{i_1}(\Lambda) - \breve{P}_{i_2}(\Lambda)) \\
&\quad -(\tfrac{1}{2}\widehat{P}_{i_1}(\Lambda)\widehat{P}_{i_2}(\Lambda) - \widehat{P}_{i_1}(\Lambda) - \widehat{P}_{i_2}(\Lambda))) \\
&= \tfrac{1}{4}h(\breve{\Phi}_{eei_1}(\Lambda) - \widehat{\Phi}_{eei_1}(\Lambda)) + \tfrac{1}{4}h(\breve{\Phi}(\Lambda)_{eei_2} - \widehat{\Phi}_{eei_2}(\Lambda)) \\
&\quad + \tfrac{1}{2}\Phi_{ss}(\Lambda)((2 - \breve{P}_{i_2}(\Lambda))(2 - \breve{P}_{i_1}(\Lambda)) \\
&\quad -(2 - \widehat{P}_{i_2}(\Lambda))(2 - \widehat{P}_{i_1}(\Lambda))).
\end{aligned}$$

ACKNOWLEDGMENT

The authors would like to thank Associate Editor E. Steinbach, as well as the anonymous reviewers whose invaluable comments and suggestions led to a greatly improved manuscript.

REFERENCES

[1] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 19–31, Feb. 1997.

[2] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: Video coding at low bit-rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 7, pp. 849–865, Nov. 1998.

[3] T. Wiegand, *Joint Final Committee Draft for Joint Video Specification H.264*, document JVT-D157.doc, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Jul. 2002.

[4] D. Hepper, "Efficiency analysis and application of uncovered background prediction in a low bit-rate image coder," *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1578–1584, Sep. 1990.

[5] F. Dufaux and F. Moscheni, "Background mosaicking for low bit-rate video coding," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1. Lausanne, Switzerland, Sep. 1996, pp. 673–676.

[6] *Core Experiment on Sprites and GMC*, document MPEG96/N1648.doc, ISO/IEC JTC1/SC29/WG11, Apr. 1997.

[7] M. Budagavi and J. D. Gibson, "Multiframe video coding for improved performance over wireless channels," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 252–265, Feb. 2001.

[8] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 70–84, Feb. 1999.

[9] A. Leontaris and P. C. Cosman, "Video compression for lossy packet networks with mode switching and a dual-frame buffer," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 885–897, Jul. 2004.

[10] *Core Experiment of Video Coding with Block-Partitioning and Adaptive Selection of Two Frame Memories (STFM/LTFM)*, document MPEG96/M0654, ISO/IEC JTC1/SC29/WG11, Dec. 1996.

[11] T. Fukuhara, K. Asai, and T. Murakami, "Very low bit-rate video coding with block partitioning and adaptive selection of two time-differential frame memories," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 212–220, Feb. 1997.

[12] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Dual frame motion compensation for a rate switching network," in *Proc. Asilomar Conf. Signals Syst. Comp.*, vol. 2. Nov. 2003, pp. 1539–1543.

[13] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Dual frame motion compensation with uneven quality assignment," in *Proc. IEEE Data Compression Conf.*, Mar. 2004, pp. 262–271.

[14] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Dual frame motion compensation with uneven quality assignment," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 2, pp. 249–256, Feb. 2008.

[15] M. Tiwari and P. C. Cosman, "Dual frame video coding with pulsed quality and a lookahead window," in *Proc. IEEE Data Compression Conf.*, 2006, pp. 372–381.

[16] M. Tiwari and P. C. Cosman, "Selection of long-term reference frames in dual-frame video coding using simulated annealing," *IEEE Signal Process. Lett.*, vol. 15, no. 1, pp. 249–252, 2008.

[17] A. Leontaris and P. C. Cosman, "Video compression with intra/inter mode switching and a dual frame buffer," in *Proc. IEEE Data Compression Conf.*, 2003, pp. 63–72.

[18] A. Leontaris, V. Chellappa, and P. Cosman, "Optimal mode selection for a pulsed-quality dual frame video coder," *IEEE Signal Process. Lett.*, vol. 11, no. 12, pp. 952–955, Dec. 2004.

[19] A. Leontaris and P. C. Cosman, "Dual frame video encoding with feedback," in *Proc. Asilomar Conf. Signals Syst. Comp.*, Nov. 2003, pp. 1514–1518.

[20] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Source and channel coding trade-offs for a pulsed quality video encoder," in *Proc. 40th Asilomar Conf. Signals Syst. Comput.*, Nov. 2006, pp. 1099–1102.

[21] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Error concealment for dual frame video coding with uneven quality assignment," in *Proc. IEEE Data Compression Conf.*, Mar. 2005, pp. 319–328.

[22] A. Leontaris and P. C. Cosman, "End-to-end delay for hierarchical B-pictures and pulsed quality dual frame video coders," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 3133–3136.

[23] A. Leontaris and P. C. Cosman, "Compression efficiency and delay tradeoffs for hierarchical B-picture frames and pulsed-quality frames," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1726–1740, Jul. 2007.

[24] C.-S. Kim, R.-C. Kim, and S.-U. Lee, "Robust transmission of video sequence using double-vector motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 9, pp. 1011–1021, Sep. 2001.

[25] S. Lin and Y. Wang, "Error resilience property of multihypothesis motion compensated prediction," in *Proc. IEEE Int. Conf. Image Process.*, 2002, pp. 545–548.

[26] Y.-C. Tsai, C.-W. Lin, and C.-M. Tsai, "H.264 error resilience coding based on multihypothesis motion-compensated prediction," *Signal Process. Image Commun.*, vol. 22, no. 9, pp. 734–751, Oct. 2007.

[27] W.-Y. Kung, C.-S. Kim, and C.-C. Kuo, "Analysis of multihypothesis motion compensated prediction (MHMCP) for robust visual communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 146–153, Jan. 2006.

[28] M. Ma, O. C. Au, L. Guo, S.-H. G. Chan, X. Fan, and L. Hou, "Alternate motion-compensated prediction for error resilient video coding," *J. Visual Commun. Image Representation*, vol. 19, no. 7, pp. 437–449, Oct. 2008.

[29] I. Rhee and S. Joshi, "Error recovery for interactive video transmission over the internet," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1033–1049, Jun. 2000.

[30] B. Girod, "The efficiency of motion-compensating prediction for hybrid coding of video sequences," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 7, pp. 1140–1154, Aug. 1987.

[31] B. Girod, "Efficiency analysis of multihypothesis motion-compensated prediction for video coding," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 173–183, Feb. 2000.

[32] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice-Hall, 1984, pp. 223–435.

[33] T. M. Cover and J. A. Thomas. *Elements of information theory*, ch. 13 [Online]. Available: http://www.matf.bg.ac.yu/nastavno/viktor/Rate_Distortion_Theory.pdf

[34] Z. G. Li, F. Pan, K. P. Lim, G. Feng, X. Lin, and S. Rahardja, "Adaptive basic unit layer rate control for JVT," presented at the 7th Joint Video Team Meeting, Thailand, Mar. 2003, Paper JVT-G012-rl.

[35] Y. Zhang, W. Gao, and D. Zhao, "Joint data partition and rate-distortion optimized mode selection for H.264 error-resilient coding," in *Proc. IEEE 8th Workshop Multimedia Signal Process.*, Oct. 2006, pp. 248–251.

[36] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao, "Joint source-channel rate-distortion optimization for H.264 video coding over error-prone networks," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 445–454, Apr. 2007.

[37] *Error Patterns for Internet Experiments*, document Q15-I-16r1.doc, ITU-T SG16, 1999.

Da Liu received the B.S. degree in computer science from the Hebei University of Science and Technology, Shijiazhuang, China, in 2002, and the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2004. Currently, he is working toward the Ph.D. degree in computer science from the Department of Computer Science, Harbin Institute of Technology.

His research interests include image/video coding, image processing, and computer vision.

Debin Zhao received the B.S., M.S., and Ph.D. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 1985, 1988, and 1998, respectively, all in computer science.

He joined the Department of Computer Science, HIT, as an Associate Professor in 1993. He is currently a Professor with the Department of Computer Science, HIT, and also with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He has authored or coauthored over 200 publications. His research interests include video coding and transmission, multimedia processing, and pattern recognition.

Dr. Zhao received three National Science and Technology Progress Awards of China (Second Prize), as well as the Excellent Teaching Award from the Baogang Foundation.

Xiangyang Ji received the B.S. and M.S. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively. He received the Ph.D. degree in computer science from the Institute of Computing Technology at the Graduate School of the Chinese Academy of Science, Beijing, China, in 2008.

Currently, he is with the Broadband Networks and Digital Media Laboratory, Department of Automation, Tsinghua University, Beijing, China. He has authored or co-authored over 40 conference and journal papers. His research interests include video/image coding, video streaming, and multimedia processing.

Wen Gao (M'92–SM'05–F'09) received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1985, and the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He is currently a Professor of Computer Science at the Key Laboratory of Machine Perception, School of Electronic Engineering and Computer Science, Peking University, Beijing, China. Before joining Peking University, he was a Full Professor of Computer Science at the Harbin Institute of Technology, Harbin, China from 1991 to 1995. From 1996 to 2005, he was with the Chinese Academy of Sciences (CAS), Beijing, China. During his time at CAS, he held the positions of Professor, Managing Director of the Institute of Computing Technology, Executive Vice President of the Graduate School of CAS, and Vice President of the University of Science and Technology of China, Hefei China. He has published extensively, including four books and over 500 technical articles in refereed journals and conference proceedings in the areas of image processing, video coding and communication, pattern recognition, multimedia information retrieval, multimodal interface, and bioinformatics.

Dr. Gao is the Editor-in-Chief of the *Journal of Computers* (a journal of the Chinese Computer Federation), and an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT. He is also an Area Editor of *EURASIP Journal of Image Communications*, and an Editor of the *Journal of Visual Communication and Image Representation*. He has chaired a number of prestigious international conferences on multimedia and video signal processing, and he has also served on the advisory and technical committees of numerous professional organizations.