Laplace Distribution Based Lagrangian Rate Distortion Optimization for Hybrid Video Coding

Xiang Li, *Student Member, IEEE*, Norbert Oertel, *Member, IEEE*, Andreas Hutter, *Member, IEEE*, and André Kaup, *Senior Member, IEEE*

***Abstract*—In today's hybrid video coding, Rate-Distortion Optimization (RDO) plays a critical role. It aims at minimizing the distortion under a constraint on the rate. Currently, the most popular RDO algorithm for one-pass coding is the one recommended in the H.264/AVC reference software. It, or HR-λ for convenience, is actually a kind of universal method which performs the optimization only according to the quantization process while ignoring the properties of input sequences. Intuitively, it is not efficient all the time and an adaptive scheme should be better. Therefore, a new algorithm Lap-λ is presented in this paper. Based on the Laplace distribution of transformed residuals, the proposed Lap-λ is able to adaptively optimize the input sequences so that the overall coding efficiency is improved. Cases which cannot be well captured by the proposed models are considered via escape methods. Comprehensive simulations verify that compared with HR-λ, Lap-λ shows a much better or similar performance in all scenarios. Particularly, significant gains of 1.79 dB and 1.60 dB in PSNR are obtained for slow sequences and B-frames, respectively.**

***Index Terms*—Hybrid video coding, Lagrange multiplier selection, rate-distortion optimization (RDO).**

I. Introduction

IN MODERN hybrid video coding, more and more coding modes are developed to improve the coding efficiency. For example, in the state-of-art video coding standard H.264/AVC, up to seven block types and 16 reference frames are allowed [1]. Consequently, how to select a best mode from so many candidates is a critical problem during the encoding. Previously, the mode with minimal distortion was widely accepted as the best mode. However, in recent years, it is realized that such a selection is not efficient all the time since the minimal distortion may result in a high rate and finally degrades the overall performance. To overcome this difficulty, the mode selection in the current video codec was normally determined by the so called rate-distortion optimization (RDO).

Manuscript received February 18, 2008; revised April 18, 2008. First published December 09, 2008; current version published January 30, 2009. This paper was recommended by Associate Editor O. C. Au.

X. Li is with Siemens Corporate Technology, Munich 81739, Germany, and also with the Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, Erlangen 91058, Germany (e-mail: xiang@lnt.de).

A. Kaup is with the Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, Erlangen 91058, Germany.

N. Oertel and A. Hutter are with the Siemens Corporate Technology, Munich 81739, Germany.

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.2008.2009255

Generally, the target of the RDO is to minimize the distortion D for a given rate R_c by appropriate selections of coding parameters, namely

$$\min\{D\} \text{ subject to } R \leq R_c \tag{1}$$

where R and D represent the rate and distortion for a coding unit which may be a macroblock, a frame, or even a group of frames. To solve such a constrained problem, there are two popular approaches : Lagrangian optimization and dynamic programing [2].

A. Lagrangian Optimization

Lagrangian optimization, or more precisely the discrete version of Lagrangian optimization, is a classical solution to (1). It was first introduced by Everett in 1963 [3]. The basic idea of this technique is to convert the constrained problem (1) to an unconstrained form (2) by the Lagrange multiplier method.

$$\min\{J\} \text{ where } J = D + \lambda \cdot R. \tag{2}$$

Here J is the Lagrangian cost function and λ is the so-called Lagrange multiplier. When the rate-distortion (R-D) curve is convex, and both R and D are differentiable everywhere, the minimal J for a coding unit is given by setting its derivative to zero, i.e.,

$$\frac{\mathrm{d}J}{\mathrm{d}R} = \frac{\mathrm{d}D}{\mathrm{d}R} + \lambda = 0. \tag{3}$$

As the solution to (3) is the solution to (1) as well, when the coding unit is operated at the point where the line with negative slope λ is the tangent to the convex hull of the R-D curve, as indicated by (3), the target of RDO (1) can be easily achieved at the same time.

However, the Lagrange multiplier method itself does not point out how to determine λ. In the literature, there are two kinds of methods to get λ. The first one is by a heuristic way. In such methods, λ is determined by an iterative process or an empirical expression, such as the bisection search algorithm [4], rate based λ[5], buffer state based λ[6] and λ local adjustment according to R-D complexity [7]–[9]. Generally, the performance of such methods is not satisfiable due to high computational complexity and no concrete theoretical foundation. Therefore, a second approach is proposed. Compared with the first one, it is an analytical way. In fact, if (3) is rewritten as

$$\lambda = -\frac{\mathrm{d}D}{\mathrm{d}R}. \tag{4}$$

λ can be determined by R-D models. As no iterative process is necessary in such a method, it is quite appealing in terms of computation. Moreover, (4) shows that the more accurate the R-D models are, the better λ can be achieved. Thus, many algorithms of this kind were proposed in recent years, such as the recommended method in the H.264/AVC reference software [10]–[12] and ρ-domain based algorithm [13].

B. Dynamic Programing

Normally, Lagrangian Optimization works quite well in hybrid video coding. However, it does have a shortcoming, i.e., it is not able to reach points which are not on the convex hull of the R-D curve [3]. In such a case, dynamic programming method may be used.

In fact, all kind of constrained optimization over finite discrete sets can be solved by dynamic programming [14]. To achieve the optimal solution to RDO, a tree representing all possible solutions is created. Each stage of the tree denotes one coding unit and each node of the stage indicates a possible coding mode with corresponding rate and distortion. Then the optimization process is converted to find a best path with the requested total rate and minimal distortion.

Definitely, dynamic programing is a more general method than Lagrangian optimization. However, its computational complexity grows exponentially when the number of stages in the tree increases, which is not affordable for practical applications. Therefore, most algorithms by this approach are focused on dependent RDO which is hard to solve directly by Lagrangian optimization [5], [15], [16].

C. Organization of the Paper

In order to improve the overall coding efficiency for practical applications, a Laplace distribution based Lagrangian RDO algorithm for one-pass coding is presented in this paper. Basically, accurate rate and distortion models are first developed based on Laplace distribution of transformed residuals. Then the Lagrange multiplier λ is derived from the models. When the statistical properties of input sequences cannot be well captured by the models, λ will be refreshed by specially designed escape methods. Finally, the RDO is performed with λ. Compared with the recommend algorithm in the H.264/AVC reference software, the proposed algorithm shows a better or similar performance in all scenarios. Particularly, significant gains of 1.79 and 1.60 dB in PSNR are achieved for slow sequences and B-frames, respectively.

The rest of the paper is organized as follows. First, the rate and distortion models are developed in Section II. Second, the proposed RDO algorithm is detailedly discussed in Section III. Subsequently, the performance of the proposed method is verified by comprehensive simulations in Section IV. Finally, conclusions and future work are presented in Section V.

II. Rate and Distortion Models

As mentioned in Section I-A, the accuracy of R-D models greatly affects the performance of Lagrangian RDO. In this section, rate and distortion models and related λ in the literature are firstly reviewed. Then new R-D models are proposed based on the Laplace distribution of transform residuals.

A. R-D Models and Related λ in the Literature

In the past decades, a lot of R-D models were proposed. However, most of them aimed at rate control and are not applicable for RDO since they were not focused on the accuracy but the connection between the rate and quantization step size, such as the famous quadratic model [17] in MPEG-4 [18] reference software [19].

Currently, the most popular R-D model in RDO was proposed in [10], [11]. Assuming a sufficiently high rate environment, the R model can be derived as shown in (5) according to the typical high-rate approximation curve for entropy-constrained scalar quantization [20]

$$R(D) = a \log_2 \left(\frac{b}{D}\right) \tag{5}$$

where a and b are two constants. For the D model, if the same "high rate" assumption is preserved, the source probability distribution can be approximated as uniform within each quantization interval [20], leading to

$$D = \frac{(2Q)^2}{12} = \frac{Q^2}{3}. \tag{6}$$

Note that here the quantizer value Q is half the distance of the quantization interval. Putting (5) and (6) into (4), the final λ can be determined by

$$\lambda_{HR} = -\frac{\mathrm{d}D}{\mathrm{d}R} = c \cdot Q^2 \tag{7}$$

where c is a constant which is experimentally suggested to be 0.136[1] for H.264/AVC environment [12]. In summary, this method, or in short High Rate λ selection (HR-λ), is practical and efficient. Therefore, it was adopted into the reference softwares not only by H.263 [21], [22], but also by H.264/AVC [1], [12], the state-of-art video coding standard. However, there are some drawbacks with this algorithm. First, λ_{HR} is only related to Q and no property of the input signal is considered, which means that it cannot adapt to different videos dynamically. More important, the "high rate" assumption is not realistic all the time, which will result in a poor performance for both rate and distortion models, especially in the case of low bit-rate coding.

In order to achieve adaptivity, Chen and Garbacea proposed an adaptive λ estimation method based on the ρ-domain method [13]. Defining ρ as the percentage of zero coefficients among quantized transformed residuals, the R and D can be rewritten as (8) and (9), respectively [23], [24]

$$R(\rho) = \theta \cdot (1 - \rho) \tag{8}$$
$$D(\rho) = \sigma^2 \cdot e^{-\alpha(1-\rho)} \tag{9}$$

where θ and α are coding constants, σ^2 is the variance of transformed residuals. Considering (4) at the same time, Chen finally derived λ as shown in (10) [13]

$$\lambda_\rho = \beta \cdot \left(\ln\left(\frac{\sigma^2}{D}\right) + \delta\right) \cdot \frac{D}{R}. \tag{10}$$

[1]Reference [11]claimed c was 0.85. However for H.264/AVC environment, c is actually $0.85/2.5^2$ since the real Q is 2.5 times of what they claimed [12].

where β and δ are both coding constants. In summary, such a method represents a first shot at adaptivity. Thanks to the introduction of the variance of transformed residuals, it is able to adapt input videos dynamically. However, it decouples quantization information Q from λ_ρ, which is sometimes inconvenient, such as in the case of rate control. More important, R and D are directly included into λ_ρ calculation, which may result in improper λ_ρ due to error propagation.

B. Proposed R-D Models

To be more general than high-rate assumption, the property of the input sequence should be considered. Therefore, the proposed R-D models are based on the distribution of transformed residuals.

In the literature, there are three widely accepted distributions for transform residuals , i.e., Laplace distribution [25], generalized Gaussian distribution (GGD) [26] and Cauchy distribution [27]. As Laplace distribution is a special case of GGD, the latter shows a higher accuracy in offline analysis [28]. However, compared with Laplace and Cauchy distribution, GGD has one more control parameter to be determined during the fitting process. It is much more difficult to accurately update the model to capture the property of input sequences on the fly. Actually, recent research on dynamically updating models for rate control shows that the predictability of the control parameters is even more important than the accuracy of the models [29]. Thus, GGD is not satisfiable for one-pass coding. For Cauchy distribution, it indeed shows higher accuracy in the case of heavy tails of transformed residuals [27]. Nevertheless, it is hard to be applied in RDO since the mean and variance of Cauchy distribution are not defined/converged. On the contrary, Laplace distribution does not have these problems. Moreover, it shows a good tradeoff between the computational complexity and accuracy. Therefore, it is finally chosen as the base distribution of the proposed algorithm.

Suppose that the transformed residuals obey a zero-mean Laplace distribution, i.e.,

$$\begin{aligned} f_{Lap}(x) &= \frac{\Lambda}{2} e^{-\Lambda |x|} \\ \Lambda &= \frac{\sqrt{2}}{\sigma} \end{aligned} \tag{11}$$

where x represents the transformed residual, σ is their standard deviation which indicates the property of the input sequence, and Λ here is called Laplace parameter.[2] Because of the one-to-one mapping between σ and Λ, the latter will be used in the following expressions and discussions for simplicity.

Considering the uniform reconstruction quantizer used in recent coding standards, such as H.263 [21] and H.264/AVC [1], the probability P_n of transformed residuals inside nth quantization interval can be calculated as

$$\begin{aligned} \mathrm{P}_0 &= \int_{-(Q-\gamma Q)}^{Q-\gamma Q} f_{Lap}(x)dx \\ P_n &= \int_{nQ-\gamma Q}^{(n+1)Q-\gamma Q} f_{Lap}(x)dx. \end{aligned} \tag{12}$$

[2]To be distinguishable with Lagrange multiplier, Λ is used instead of λ though the latter is normally used in Laplace distribution.

where Q is the quantization interval, γQ represents the rounding offset and γ is between $(0, 1)$, such as $1/6$ for H.264/AVC inter frame coding [12]. Here, it is worth mentioning that P_0 actually represents the probability of quantized zeros.

With the probability function, the entropy H of the quantized transformed residuals can be derived according to the entropy definition.

$$H = -P_0 \cdot \log_2 P_0 - 2\sum_{n=1}^{\infty} \mathrm{P_n} \cdot \log_2 \mathrm{P_n}. \tag{13}$$

Intuitively, (13) may be regarded as the rate model. Unfortunately, it is far from the truth since such entropy is for the case of independent coding. In hybrid video coding, quantized transformed residuals are always dependently entropy coded at block level, such as the case of run-length coding. After the skip mode is introduced, the situation is even worse since a single macroblock may be skipped and nothing but only a flag is coded [1]. Therefore, the entropy H is refined by two steps to approximate the real rate.

First, the affect by marcoblock/block skipping should be counteracted. In our previous work [30], the percentage of skipped macroblcok in a single frame is introduced to compensate such a kind of affect. It is efficient but sometimes not stable. In this paper, a more theoretical approach is used instead. Since skipped blocks represent a group of quantized zeros which will not be entropy coded at all, the real P_0 and P_n in (13) should be calculated as $P_0^* = (P_0 - P_s)/(1 - P_s)$ and $P_n^* = P_n/(1 - P_s)$, where P_s represents the probability of skipped blocks. Then the refined entropy H^* is

$$\begin{aligned} H^* = (1 - P_s) \cdot \Bigg[&- \frac{P_0 - P_s}{1 - P_s} \cdot \log_2 \frac{P_0 - P_s}{1 - P_s} \\ &- 2\sum_{n=1}^{\infty} \frac{P_n}{1 - P_s} \cdot \log_2 \frac{P_n}{1 - P_s} \Bigg] \end{aligned} \tag{14}$$

where the multiplier $(1 - P_s)$ outside the bracket is to keep H^* as the average entropy for all quantized transformed residuals since only those of non-skipped blocks are counted in the entropy calculation inside the bracket. Defining

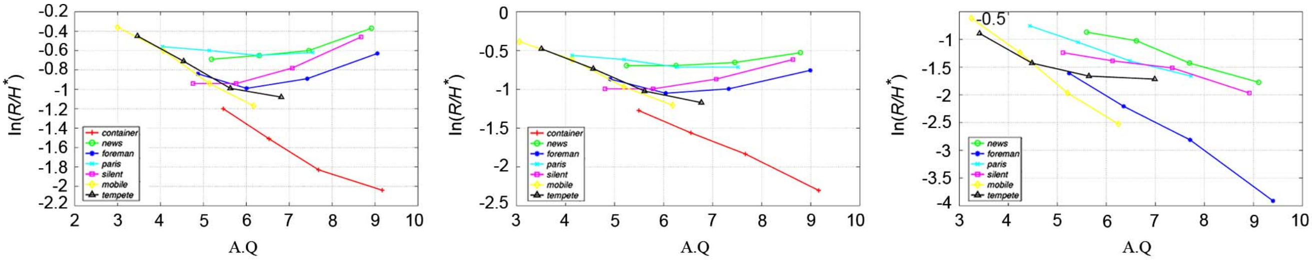
$$r = \frac{P_s}{P_0} \tag{15}$$

(14) turns to

$$\begin{aligned} H^* = H + P_0 \cdot [r \cdot \log_2 P_0 - (1 - r) \cdot \log_2(1 - r)] \\ + (1 - r \cdot P_0) \cdot \log_2(1 - r \cdot P_0) \end{aligned} \tag{16}$$

where r should be between $(0, 1)$ since the probability of skipped blocks should be no more than that of the quantized zeros according to its physical meaning. Furthermore, it should be noted that r is a very stable parameter which actually indicates an inherent property of the input sequence.

Second, the inaccuracy caused by dependent entropy coding should be compensated. Unfortunately, it is a very difficult problem which is highly related to the entropy coding method in use. In our previous work [32], only the entropy of quantized nonzeros is counted for the rate model since normally the bits spent on quantized zeros are relatively negligible in run-length



(a) P frames in Baseline Profile (CAVLC): first 100 frames (I+99P) in IPPP mode, each point indicates an average of 99 P frames.

(b) P frames in Main Profile (CABAC): first 100 frames (I+99P) in IPPP mode, each point indicates an average of 99 P frames.

(c) B frames in Main Profile (CABAC): first 148 frames (I+49P+98B) in IBBP mode, each point indicates an average of 98 B frames.

Fig. 1. Relationship between (R/H^*) (the ratio of the real luma rate to the estimated entropy) and $(\Lambda \cdot Q)$. Test conditions are derived from [31]: *container* (QCIF), *news* (QCIF), *foreman* (QCIF), *paris* (CIF), *silent* (QCIF), *mobile* (CIF) and *tempete* (CIF) are coded in H.264/AVC (JM13.1), with five reference frames and fixed quantization step size (QP =28, 32, 36 and 40).

coding. Nevertheless, a more stable relationship between the real coded rate R and refined entropy H^* is noticed. In fact, the relationship between $\ln(R/H^*)$ and (ΛQ) can be investigated in Fig. 1. [3] From the figure, an approximately linear relationship can be observed for each sequence at practical rates ($\mathrm{QP} = 28$, 32, 36, 40 [31]):

$$\ln\left(\frac{R}{H^*}\right) = a \cdot (\Lambda Q) + b \tag{17}$$

where a and b are both constants. Consequently, the rate model R can be approximated by

$$R = S \cdot H^* \cdot e^{-\xi \Lambda Q} \tag{18}$$

where S is a constant at sequence level, ξ is dependent on the input sequence and frame type. Notice that the last term of (18) is similar to the closed form of the probability of quantized nonzeros $(1 - P_0 = e^{-(1-\gamma)\Lambda Q})$ which is highly related to the entropy coding. Therefore, this term, to some extent, can be regarded as a correction factor for the entropy coding method in use. Intuitively, ξ should be adaptive to different entropy coding methods as well.

Considering (11)–(13), (16) and (18), the closed form of R model can be derived as in (19), shown at the bottom of the page.

Note that (18) and (19) only describe the rate of quantized transformed residuals and no side information, such as macroblock type and motion vectors, is considered. In early video coding standards, such as MPEG-1/2 and H.261 [33]–[35], the performance of rate models only based on residuals is quite good since the rate on side information is small when compared with that on residuals. However, for modern video coding standards, like H.263 and H.264/AVC [1], [21], they are not negligible. Moreover, motion vectors will influence the performance of motion compensation and finally affect total rate and distortion, especially for low rate video coding. Though there are some rate models for motion vectors proposed, such as [36], they cannot be applied in RDO since they only focus on rate and no distortion effect is considered. Consequently, the target of RDO, namely the tradeoff between rate and distortion, cannot be analytically achieved by such models. Currently in the literature, the only available model to describe the relationship between the rate on motion vectors and the corresponding distortion is the multi-hypothesis model [37], [38]. Based on the power spectral model, it is good for general analysis. While for practical applications, it is not mature since it is very hard to accurately calculate the power spectrum and count the real noise. Thus, in this section, no effect by side information is assumed. While in practice, an empirical strategy is used to compensate such a problem, which will be discussed in Section III-C.

Compared with the rate model, the distortion model D is easier to obtain. With the distribution, distortion in each quantization interval can be calculated. Then the total distortion is achieved by summing them up as

$$D = \int_{-(Q-\gamma Q)}^{Q-\gamma Q} x^2 f_{Lap}(x)dx + 2\sum_{n=1}^{\infty}\int_{nQ-\gamma Q}^{(n+1)Q-\gamma Q} (x-nQ)^2 f_{Lap}(x)dx. \tag{20}$$

[3] No *container* data is presented in Fig. 1(c) because of undefined $\ln(R/H^*)$. In fact for B-frames in *container*, when $\mathrm{QP} \geq 36$, $R = 0$ for luma signals.

$$R = \frac{Se^{-\xi\Lambda Q}}{\ln 2}\Bigg\{(1-e^{-(1-\gamma)\Lambda Q})[r\ln(1-e^{-(1-\gamma)\Lambda Q}) - (1-r)\ln(1-r)] - (1-e^{-(1-\gamma)\Lambda Q})\ln(1-e^{-(1-\gamma)\Lambda Q}) + [1-r(1-e^{-(1-\gamma)\Lambda Q})]\cdot\ln[1-r(1-e^{-(1-\gamma)\Lambda Q})] + e^{-(1-\gamma)\Lambda Q}\left(\ln 2 - \ln(1-e^{-\Lambda Q}) - \gamma\Lambda Q + \frac{\Lambda Q}{1-e^{-\Lambda Q}}\right)\Bigg\}. \tag{19}$$

Putting (11) into (20), the closed form of D model can be derived as

$$D = \frac{\Lambda Q \cdot e^{\gamma \Lambda Q}(2 + \Lambda Q - 2\gamma \Lambda Q) + 2 - 2e^{\Lambda Q}}{\Lambda^2(1 - e^{\Lambda Q})}. \qquad (21)$$

So far, both the R-D models have been achieved. In the next section, the proposed RDO algorithm will be discussed in detail.

III. Laplace Distribution Based Lagrangian RDO

In this section, the proposed Lagrange multiplier λ_{Lap} is first derived from the R-D models developed in Section II. Then the implementation details on λ_{Lap} calculation are depicted. Subsequently, several escaped methods are proposed for the cases which cannot be well captured by the R-D models. Finally, the whole RDO algorithm, Lap-λ in short, is summarized and the computational complexity is discussed.

A. Proposed Lagrange Multiplier λ_{Lap}

In Section II-B, the closed forms of R-D models (19) and (21) were derived. They show that R and D are functions of Q, Λ and r, among which the last two are the inherent properties of sequences. Therefore, such two properties can be regarded as independent with Q when Q varies in a small range for successive frames, which is the case of practical rate control and fixed Q coding. Then λ_{Lap} can be derived according to (4), namely

$$\lambda_{Lap} = -\frac{\mathrm{d}D}{\mathrm{d}R} = -\frac{\frac{\partial D}{\partial Q}}{\frac{\partial R}{\partial Q}}. \qquad (22)$$

Finally, λ_{Lap} is determined by plugging (19) and (21) into (22). Unfortunately, the closed form of (22) is too long to be presented here. Nevertheless, it can be easily computed by mathematical softwares or numerical methods. Of course, in practice, a lookup table can also be used to save computations.

To further study the relationship between λ_{HR} and λ_{Lap}, a special case is investigated. In fact, it is well known that uniform distribution is a special case of Laplace distribution when Λ approaches zero. In such a case, the limit of λ_{Lap} is

$$\lim_{\Lambda \to 0} \lambda_{Lap} = \frac{2\ln(2)(3\gamma^2 - 3\gamma + 1)}{3S} \cdot Q^2. \qquad (23)$$

As S and γ are both constants, (23) matches (7) mathematically, which indicates that **the proposed $\boldsymbol{\lambda_{\mathrm{Lap}}}$** is actually a general form of $\boldsymbol{\lambda_{\mathrm{HR}}}$.

B. Implementation Details for λ_{Lap} Calculation

With (22), λ_{Lap} can be calculated. However, for real applications, there are several practical aspects related to the implementation worth mentioning. For convenience, the H.264/AVC environment is taken as an example.

First, it has to be decided whether the luminance and chrominance should be both considered. In Baseline Profile and Main Profile of H.264/AVC, input videos are requested in 4:2:0 format [1]. As chrominance is much less important than luminance in such a case, only Λ of the luminance is considered in the current implementation. Of course, λ_{Lap} can be easily extended to the cases of 4:2:2 and 4:4:4 by counting Λ for chrominance as well.

Second, the two constants S and ξ in (18) have to be determined. To guarantee that λ_{Lap} will reduce to λ_{HR}, (23) is forced to (7)

$$\lim_{\Lambda \to 0} \lambda_{Lap} = \lambda_{HR}. \qquad (24)$$

Then S can be derived as

$$S = \frac{2\ln(2)(3\gamma^2 - 3\gamma + 1)}{3c} = \begin{cases} 1.133 & \text{(Intra frames)} \\ 1.982 & \text{(Inter frames)} \end{cases}. \qquad (25)$$

where c is the constant in (7), and γ is in line with the reference software [12], i.e., $1/3$ for intra frames and $1/6$ for inter frames, respectively. As mentioned in Section II-B, ξ should be adaptive to input sequence, frame type and entropy coding method. However in practice, it is some hard to determine a good value for ξ on the fly. Considering the efficiency of entropy coding is comparatively stable, while the properties of input sequence and frame type are some versatile, a tradeoff is made to ease the implementation where a certain adaptivity is kept for entropy coding methods while those for sequences and frame types are sacrificed. In the current design, two fixed values are employed for CAVLC and CABAC, respectively. To determine these values, comprehensive coding efficiency tests are conducted. Finally, $\xi = 0.35$ for CAVLC and $\xi = 0.3$ for CABAC are selected from a group of predefined values.

To perform the RDO, λ_{Lap} should be determined before coding the ith frame. However, Λ and r for the frame are both unavailable at that time. Thus, they have to be predicted. Fortunately, as the inherent properties of input sequences, Λ and r can be reasonably regarded as constants during a short time when there is no shot transition. Therefore, in the current implementation, both of them are estimated by the averages of those in five previously coded same-type frames, i.e.,

$$\hat{\Lambda}^i = \frac{1}{5}\sum_{n=1}^{5} \Lambda^{i-n}$$
$$\hat{r}^i = \frac{1}{5}\sum_{n=1}^{5} r^{i-n}. \qquad (26)$$

where the superscript ^{i} indicates the value for the ith same-type frame, $\hat{\Lambda}^i$ and $\hat{r}^i$ represent the estimations on Λ^i and r^i, respectively. Taking $\hat{\Lambda}^i$ and $\hat{r}^i$ into (22), λ^i_{Lap} can be determined.

Considering λ^i_{Lap} actually represents a tradeoff between the distortion and rate, its dynamic range should be limited to ensure neither distortion nor rate will receive too much weighting in practical applications. Due to the good performance of λ^i_{HR}, it is employed as a reference in the range determination of λ^i_{Lap}. Intuitively, λ^i_{Lap} should be within $(b_l \cdot \lambda^i_{HR}, b_u \cdot \lambda^i_{HR})$, where $b_l < 1$ and $b_u > 1$. To determine b_l and b_u, different values are heuristically tried by comprehensive coding tests. Finally, $b_l = 0.9$ and $b_u = 5$ are chosen in the current implementation. Moreover, to avoid undesired performance oscillation among successive frames, λ^i_{Lap} is further limited between $0.8\lambda^{i-1}_{Lap}$ and $1.2\lambda^{i-1}_{Lap}$, where the bounds are empirically determined in a similar way to the forementioned b_l and b_u.

After coding the ith frame, Λ^i and r^i are updated according to the statistics on the frame. First, Λ^i is obtained from the standard deviation of transformed residuals which are those after motion estimation for inter frames or those after intra prediction for intra frames. Then, r^i is calculated according to (15), where P_s in the current implementation is defined as the percentage of skipped 8×8 luma blocks which are actually all-zero 8×8 blocks after quantization.

C. Escape Methods

Definitely, video sequences are quite complex to model. During the development of R-D models and λ_{Lap}, three assumptions, i.e., "Laplace distribution of transformed residuals" (Section II-B), "Negligible side information" (Section II-B) and "No scene changes" (Section III-B), were made to ease the modeling. Although in most cases these assumptions are valid, there are still some rare cases in which they may not hold. Unfortunately, in such situations, the overall performance of λ_{Lap} may degrade seriously due to the improper weighting between rate and distortion. For example, in sequence *soccer* (4CIF) which contains large movement and gradually scene changes, a loss up to 0.5 dB may be observed by straightforward λ_{Lap}. To keep a reasonable performance in these kind of conditions, escape methods are proposed in this sub-section. Note that the target of the escape methods is not to improve coding efficiency but to avoid performance dropping.

Basically, there are two steps in escape methods. First, detection on whether the assumption is still valid is performed. Then a refresh process on λ_{Lap}, actually a compulsive range limitation backward to λ_{HR}, is applied according to the detection result.

1) Detection on Scene Change: When scene change occurs, the predictions on Λ and r are not reliable so that the calculated λ_{Lap} may not be good. Normally a lower distortion is better for the first frame of a new shot because of a lower error propagation effect. Thus, a smaller λ_{Lap} is preferred for potential scene changes.

In the literature, there are a lot of methods to detect scene changes before the real coding. [39] proposed a simple while efficient algorithm based on $\mathrm{PSNR}_{\mathrm{drop}}$. Following a similar idea, a detection method based on the difference between two successive same-type frames is used. Define σ_0 as the standard deviation of the frame difference between the reconstruction of the last coded same-type frame and the current frame to be coded, namely

$$\sigma_0^i = \sqrt{\frac{1}{W \cdot H} \sum_{x=1}^{W} \sum_{y=1}^{H} \left(I_{x,y}^i - \hat{I}_{x,y}^{i-1} \right)^2} \tag{27}$$

where W and H are the width and height of the frame, subscript (x, y) denotes the pixel position in a frame, I and $\hat{I}$ represent the source frame and reconstructed frame, respectively. Since σ_0 denotes the residuals with zero motion, it should vary moderately within a shot while increase sharply when a scene change occurs. Therefore, when the ratio I_S in (28) is below a certain threshold, a potential scene change should be considered.

$$I_S^i = \frac{\sigma_0^{i-1}}{\sigma_0^i}. \tag{28}$$

In the current implementation, the refresh strength RS for scene change is derived as

$$RS^i = \begin{cases} 4\ (RS_{HR}) & I_S^i < T_{\mathrm{SC}} \\ 3\ (RS_{Strong}) & I_S^i < T_{\mathrm{PSC}} \end{cases} \tag{29}$$

where $T_{\mathrm{SC}} = 0.3$ and $T_{\mathrm{PSC}} = 0.8$ are empirical thresholds for scene change and possible scene change, respectively.

2) Detection on Non-Laplace Distribution: Laplace distribution of transformed residuals is the basis of λ_{Lap}. In the case of non-Laplace distribution, the analytical tools developed in Section II-B are in a less accuracy. To keep the performance in such a situation, freshing backward to HR-λ is a good way.

Generally, there are many ways to check whether a set of points is in Laplace distribution, such as directly calculating the deviation from the observed points to the ideal distribution. However, to save computation, a simple indicator is employed in the current implementation.

Define an RD_{Lap} as

$$RD_{Lap} = \frac{\frac{R_r}{R_m}}{\frac{D_r}{D_m}} \tag{30}$$

where R_r and D_r are the real frame rate and distortion collected after a frame coding, R_m and D_m are those estimated by the models developed in II-B . Ideally, RD_{Lap} equaling one indicates a perfect match between the practice and R-D models. On the contrary, a big RD_{Lap} is regarded as a non-Laplace case. In the current implementation, RS for non-Laplace case is derived as

$$RS^i = \begin{cases} 2\ (RS_{Weak}) & \frac{1}{5}\sum_{n=1}^{5} RD_{Lap}^{i-n} > T_{ARD} \\ 1\ (RS_{KeepMax}) & RD_{Lap}^{i-1} > T_{\mathrm{PRD}} \end{cases} \tag{31}$$

where $T_{ARD} = 50$ and $T_{\mathrm{PRD}} = 10$ correspond to a continuous non-Laplace case and a single non-Laplace frame, respectively. It should be noted that, although (30) works well, other indicators are also possible.

3) Detection on Effect by Side Information: As mentioned in Section II-B, on one hand, side information may greatly affect the overall coding efficiency. On the other hand, there are no analytical models to describe the effect. Generally, fast/complex sequences need more bits for side information to obtain a good prediction so that even more bits can be saved for residuals. While for slow/simple sequences, less side information is preferred since the gain by accurate prediction may be largely counteracted by the cost of side information. Therefore, a refresh should be applied when the side information is quite positive to the coding efficiency.

To evaluate the importance of side information, R_{Gap} and D_{Gap} are defined as

$$\begin{aligned} R_{\mathrm{Gap}}^i &= \frac{R(\Lambda_0^i, r^i, Q^i) - R(\Lambda^i, r^i, Q^i)}{R_{rs}^i} \\ D_{\mathrm{Gap}}^i &= 10 \cdot \log_{10} \frac{D(\Lambda_0^i, Q^i)}{D(\Lambda^i, Q^i)} \end{aligned} \tag{32}$$

where $\Lambda_0^i = \sqrt{2}/\sigma_0^i$, $R(\cdot)$ and $D(\cdot)$ are the R-D models developed in Section II-B, and R_{rs}^i is the real rate for side information in ith frame. Since σ_0^i is the standard deviation of residuals of zero motion, the two terms in the numerator of $R^i_{\rm Gap}$ represent the calculated frame luma rate without and with motion compensation, respectively. Consequently, $R^i_{\rm Gap}$ indicates the relative gain over the case of no side information and so is $D^i_{\rm Gap}$. With $R^i_{\rm Gap}$ and $D^i_{\rm Gap}$, the effect by side information can be roughly assessed. When $R^i_{\rm Gap}$ and $D^i_{\rm Gap}$ are above some certain thresholds, the positive effect by side information should be considered and maximal λ_{Lap} should be limited by refresh process. In the current implementation, RS for the case of positive side information is derived as

$$RS^i = \begin{cases} 2 & R^{i-1}_{\rm Gap} > T_{R{\rm Gap}H} \text{ and } D^{i-1}_{\rm Gap} > 0 \\ 2 & R^{i-1}_{\rm Gap} > 1 \text{ and } D^{i-1}_{\rm Gap} > T_{D{\rm Gap}H} \\ 1 & R^{i-1}_{\rm Gap} > T_{R{\rm Gap}} \text{ and } D^{i-1}_{\rm Gap} > 0 \\ 1 & R^{i-1}_{\rm Gap} > 1 \text{ and } D^{i-1}_{\rm Gap} > T_{D{\rm Gap}} \end{cases} \tag{33}$$

where $T_{R{\rm Gap}H} = 15$, $T_{D{\rm Gap}H} = 5$, $T_{R{\rm Gap}} = 7.5$ and $T_{D{\rm Gap}} = 3$ are thresholds suggested by simulations.

4) Refresh Process: The refresh process is actually an empirical range limitation on λ_{Lap}. Since the refresh process will be used only in the cases which cannot be well captured by the proposed models, it is designed to limit λ_{Lap} backward to λ_{HR} in order to keep a similar performance with HR-λ under the worst situations.

As mentioned in previous subsections, there are totally four refresh strengths, from the highest $RS_{HR} = 4$ to the lowest $RS_{KeepMax} = 1$, assigned by the detection processes. The detailed refresh process is described

$$\lambda^i_{Lap} = \begin{cases} \lambda^i_{HR} & RS^i = RS_{HR} \\ \text{clip}\,(\lambda^i_{HR}, \lambda^i_{Lap}, 0.8\lambda^{i-1}_{Lap}) & RS^i = RS_{Strong} \\ \text{clip}\,(\lambda^i_{HR}, \lambda^i_{Lap}, 0.9\lambda^{i-1}_{Lap}) & RS^i = RS_{Weak} \\ \text{clip}\,(\lambda^i_{HR}, \lambda^i_{Lap}, \lambda^{i-1}_{Lap}) & RS^i = RS_{KeepMax} \\ \lambda^i_{Lap} & RS^i = 0 \end{cases} \tag{34}$$

where $RS = 0$ indicates no refresh is necessary, the function $\text{clip}(a, b, c)$ represents a clipping process on b with a and c as the minimal and maximal allowed values.

In fact, the idea behind the refresh design is that the higher refresh strengths are designed for the case of scene change. This is because the detection on scene change is based on the statistics of the current frame, which is much more reliable. On the contrary, lower strengths are always assigned by the detection processed based on the data of previously coded frames. Consequently, the refresh process will not degrade the coding efficiency much if the statistical property of the current frame is quite different with the expectation based on the previously coded frames.

Algorithm 1: Encode one Frame with the proposed RDO

```
Initialize i^th frame
```

begin λ^i_{Lap} calculation in Section III-B
  $\lambda^i_{Lap} \leftarrow \lambda_{Lap}(Q^i, \hat{\Lambda}^i, \hat{r}^i)$
end

begin Refresh Process in Section III-C
  if $I^i_S < T_{SC}$ **then**
    $RS^i \leftarrow RS_{HR}$
  else if $I^i_S < T_{PSC}$ **then**
    $RS^i \leftarrow RS_{Strong}$
  else if $\frac{1}{5}\sum_{n=1}^{5} RD^{i-n}_{Lap} > T_{ARD}$ **then**
    $RS^i \leftarrow RS_{Weak}$
  else if $RD^{i-1}_{Lap} > T_{PRD}$ **then**
    $RS^i \leftarrow RS_{KeepMax}$
  else if $(R^{i-1}_{Gap} > T_{RGapH}$ *and* $D^{i-1}_{Gap} > 0)$ **then**
    $RS^i \leftarrow RS_{Weak}$
  else if $(R^{i-1}_{Gap} > 1$ *and* $D^{i-1}_{Gap} > T_{DGapH})$ **then**
    $RS^i \leftarrow RS_{Weak}$
  else if $(R^{i-1}_{Gap} > T_{RGap}$ *and* $D^{i-1}_{Gap} > 0)$ **then**
    $RS^i \leftarrow RS_{KeepMax}$
  else if $(R^{i-1}_{Gap} > 1$ *and* $D^{i-1}_{Gap} > T_{DGap})$ **then**
    $RS^i \leftarrow RS_{KeepMax}$
  switch RS^i **do**
    case RS_{HR}
      $\lambda^i_{Lap} \leftarrow \lambda^i_{HR}$
    case RS_{Strong}
      $\lambda^i_{Lap} \leftarrow \text{clip}(\lambda^i_{HR}, \lambda^i_{Lap}, 0.8\lambda^{i-1}_{Lap})$
    case RS_{Weak}
      $\lambda^i_{Lap} \leftarrow \text{clip}(\lambda^i_{HR}, \lambda^i_{Lap}, 0.9\lambda^{i-1}_{Lap})$
    case $RS_{KeepMax}$
      $\lambda^i_{Lap} \leftarrow \text{clip}(\lambda^i_{HR}, \lambda^i_{Lap}, \lambda^{i-1}_{Lap})$
  end
end

begin Range Limitation in Section III-B
  $\lambda^i_{Lap} \leftarrow \text{clip}(0.8\lambda^{i-1}_{Lap}, \lambda^i_{Lap}, 1.2\lambda^{i-1}_{Lap})$
  $\lambda^i_{Lap} \leftarrow \text{clip}(0.9\lambda^i_{HR}, \lambda^i_{Lap}, 5\lambda^i_{HR})$
end

```
Encode i^th frame with λ^i_Lap
```

begin Update Process in Section III-B
  Update Λ^i and r^i
  Calculate RD^i_{Lap}, R^i_{Gap} and D^i_{Gap}
end

```
Finish i^th frame
```

D. Summary of the Whole Algorithm

For convenience, the whole RDO method is summarized as pseudo code in Algorithm 1, where the frame initialization, encoding and finishing indicate the corresponding process in H.264/AVC reference software [12]. It should be noted that, for the encoding, λ_{Lap} is used in the same way as the original λ_{HR}, in both motion estimation and mode decision.

From Algorithm 1, it can also be observed that no much computation is introduced by the proposed algorithm. The most complex part in the algorithm is the computation on σ^i_0 and Λ^i which totally cost $2 \cdot W \cdot H$ multiplications and additions plus $W \cdot H$ subtractions for one single frame where W and H are the frame width and height, respectively. Note that the SAD operation for 16 × 16 block matching in motion compensation module will cost $n \cdot W \cdot H$ subtractions, absolute operations and additions for only one reference frame, where n is the average matching times. Considering the multiple-reference-frame and

TABLE I
SIMULATION RESULTS OF JVT-I AND JVT-II

sequences	QP	JVT-I					JVT-II							
		Lap-λ over HR-λ					Lap-λ over HR-λ							
		$\overline{RS}$	$\frac{\bar{\lambda}_{Lap}}{\lambda_{HR}}$	ΔP (dB)	ΔR	ΔP_{peak} (dB)	$\overline{RS}$	$\frac{\bar{\lambda}_{Lap}}{\lambda_{HR}}$	ΔP (dB)	ΔR	ΔP_{peak} (dB)	ΔP^B (dB)	ΔR^B	ΔP^B_{peak} (dB)
container (QCIF, 30 Hz)	28	0.04	288.12%	1.26	-28.80%	1.79	0.05	430.97%	0.59	-14.50%	0.90	N/A	-8.33%	N/A
	32	0.04	426.17%				0.05	443.64%						
	36	0.04	475.56%				0.05	459.41%						
	40	0.04	475.55%				0.06	470.68%						
news (QCIF, 30 Hz)	28	0.11	212.26%	0.38	-8.89%	0.57	0.56	227.83%	0.25	-5.55%	0.40	0.54	-13.85%	1.47
	32	0.12	269.84%				0.35	313.16%						
	36	0.11	356.91%				0.20	382.05%						
	40	0.19	429.63%				0.41	431.68%						
foreman (QCIF, 30 Hz)	28	0.44	153.46%	0.14	-3.11%	0.61	0.81	176.74%	0.04	-0.93%	0.29	0.76	-25.23%	1.25
	32	0.37	207.59%				0.74	269.79%						
	36	0.23	308.67%				0.72	388.82%						
	40	0.09	418.50%				0.35	447.19%						
paris (CIF, 30 Hz)	28	0.33	150.48%	0.15	-3.41%	0.52	0.54	156.79%	0.14	-3.63%	0.34	0.32	-9.42%	0.76
	32	0.05	197.20%				0.25	209.92%						
	36	0.04	260.73%				0.24	291.42%						
	40	0.07	354.37%				0.18	391.58%						
silent (QCIF, 30 Hz)	28	0.19	185.00%	0.42	-11.84%	0.54	0.39	184.92%	0.40	-10.92%	0.52	0.66	-21.30%	0.94
	32	0.13	237.87%				0.33	246.44%						
	36	0.04	327.00%				0.22	358.01%						
	40	0.07	453.86%				0.20	441.98%						
mobile (CIF, 30 Hz)	28	1.01	94.60%	0.00	-0.07%	0.02	1.45	98.14%	-0.04	1.00%	0.00	0.19	-6.30%	0.93
	32	0.18	129.04%				1.18	99.82%						
	36	0.51	142.94%				0.82	178.53%						
	40	0.21	224.09%				0.47	382.24%						
tempete (CIF, 30 Hz)	28	0.08	113.45%	0.05	-1.47%	0.30	0.51	115.29%	0.06	-1.95%	0.19	0.40	-15.17%	0.98
	32	0.04	155.90%				0.19	184.43%						
	36	0.04	218.52%				0.35	298.35%						
	40	0.04	325.49%				0.14	409.82%						
average	-	-	-	0.34	-8.23%	-	-	-	0.21	-5.21%	-	0.48	-14.23%	-

variable-block-size motion compensation in H.264/AVC, those cost on σ_0^i and Λ^i are negligible. Simulations on Pentium IV 3.2G PC show that, on average, the proposed RDO takes less than 0.05% coding time when fast full search (search range of ± 32), all MB coding modes and five reference frames are employed in JM 13.1 [12].

IV. SIMULATIONS AND DISCUSSIONS

The proposed RDO algorithm Lap-λ is verified by the most recent H.264/AVC reference software JM13.1 [12], [40]. To investigate the performance in different scenarios, four sets of simulations are conducted: JVT-I (Tests on H.264/AVC Baseline Profile), JVT-II (Tests on H.264/AVC Main Profile), HB-3 (Tests on Hierarchical B-Frames) and High-Res (Tests on High Resolution Sequences).

The general simulation environment is derived from the common conditions for coding efficiency tests defined by JVT [31], [41], i.e., five reference frames, fixed quantization parameters ($\mathrm{QP} = 28, 32, 36, 40$), high complexity RDO, one I frame followed by all inter frames.

In practice, RDO affects not only the performance of one luminance channel, but also those of two chrominance channels. Thus, for a fair comparison, the coding efficiency of the all these three components should be jointly evaluated. As the test sequences are all in 4:2:0 format where the weighting among the three components is 4:1:1, a combined PSNR P as in (35) is used instead of three independent PSNRs in the following assessments and discussions for convenience.

$$P = \frac{1}{6}(4P_Y + P_U + P_V). \tag{35}$$

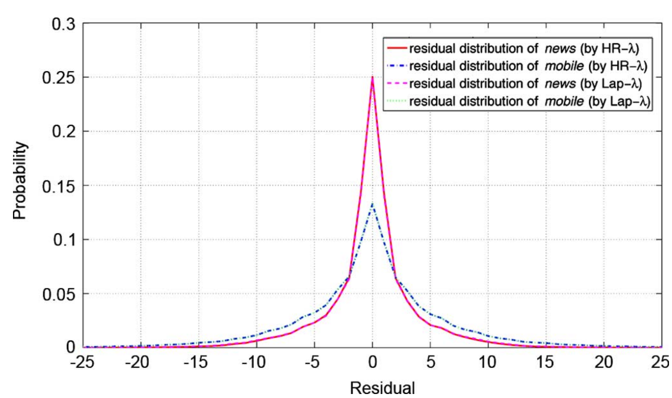


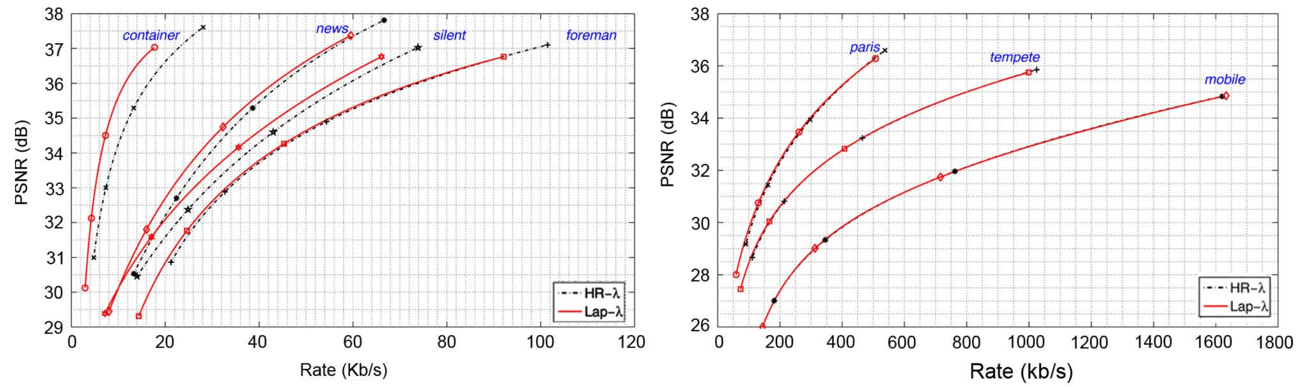
Fig. 2. Distribution of transformed residuals in JVT Test I ($\mathrm{QP} = 28$).

where P_Y, P_U, and P_V are PSNRs for one luminance and two chrominance channels, respectively.

For quantitative comparisons, the difference between two R-D curves is measured by the average gain ΔP in PSNR or the average rate reduction ΔR which are both calculated by the PSNR calculator [42] recommended by JVT. Furthermore, the peak gain ΔP_{peak} in PSNR from the related R-D curve is also derived as a byproduct during the calculation on ΔP. It should also be noted that the first I frame is not included in the results since they are always coded identically.

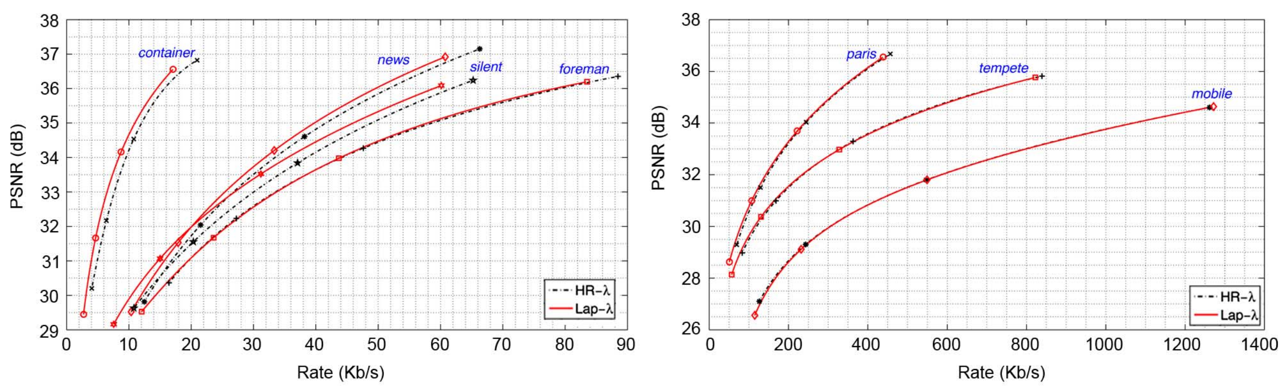
A. JVT Test I

In this test, the performance of the proposed algorithm in the environment of H.264/AVC Baseline Profile is evaluated. Seven sequences requested in [31] are tested. The entropy coding is



(a) R-D curves for *container, news,foreman and silent* (b) R-D curves for *paris, mobile and tempete*

Fig. 3. R-D curves of JVT-I.



(a) R-D curves for *container, news,foreman and silent* (b) R-D curves for *paris, mobile and tempete*

Fig. 4. R-D curves of JVT-II.

CAVLC, frame mode is IPPP, and the first 100 frames $(\mathrm{I}+99\mathrm{P})$ of each sequence are coded without frame skipping.

Table I (left part) summaries the simulation results. On average, 0.34 dB gain in PSNR or 8.23% rate reduction is achieved by the proposed Lap-λ. From the RD-curves shown in Fig. 3, it can be observed that the peak gains are even much bigger: for the first five sequences, the peak gains in PSNR are all above 0.5 dB, for *container*, it is even up to 1.79 dB or 32% rate reduction.

Generally, the sequences in this test are in simple scenarios and thus the escape methods are seldom used, as indicated by the average refresh strength $\overline{RS}$ in Table I. Therefore, those gains are from the proposed λ_{Lap}. Nevertheless, the gains are not even-distributed. On one hand, significant gains are achieved for slow sequences, like *container*. On the other hand, very limited gains are observed for videos with big movements, rotations or zooming, such as *mobile* and *tempete*. Such results are actually in line with the mathematical analysis in Section III-A. Normally, the residuals for the fast videos are much bigger than those for slow ones because of less accurate predictions. Accordingly, the distribution for fast sequences is comparatively closer to the assumption of uniform distribution, as shown in Fig. 2. In such a case, λ_{HR} and λ_{Lap} are close to each other, as indicated in the column "$\bar{\lambda}_{Lap}/\bar{\lambda}_{HR}$" of Table I, so that HR-λ and Lap-λ share a similar performance. On the contrary, for slow videos, the assumption of HR-λ fails, and λ_{Lap} is much bigger than λ_{HR}, which results in a preference on the MB mode with less bits during the encoding. For example, the percentage of skipped MB for *container* may be increased by 10% when the proposed Lap-λ is applied. Consequently, the bit-rate is greatly reduced at the cost of a bit higher distortion so that the overall performance is improved for the whole sequence. In fact, bigger Λ leading to better performance has already been partly observed in [43], though no analytical explanation was provided.

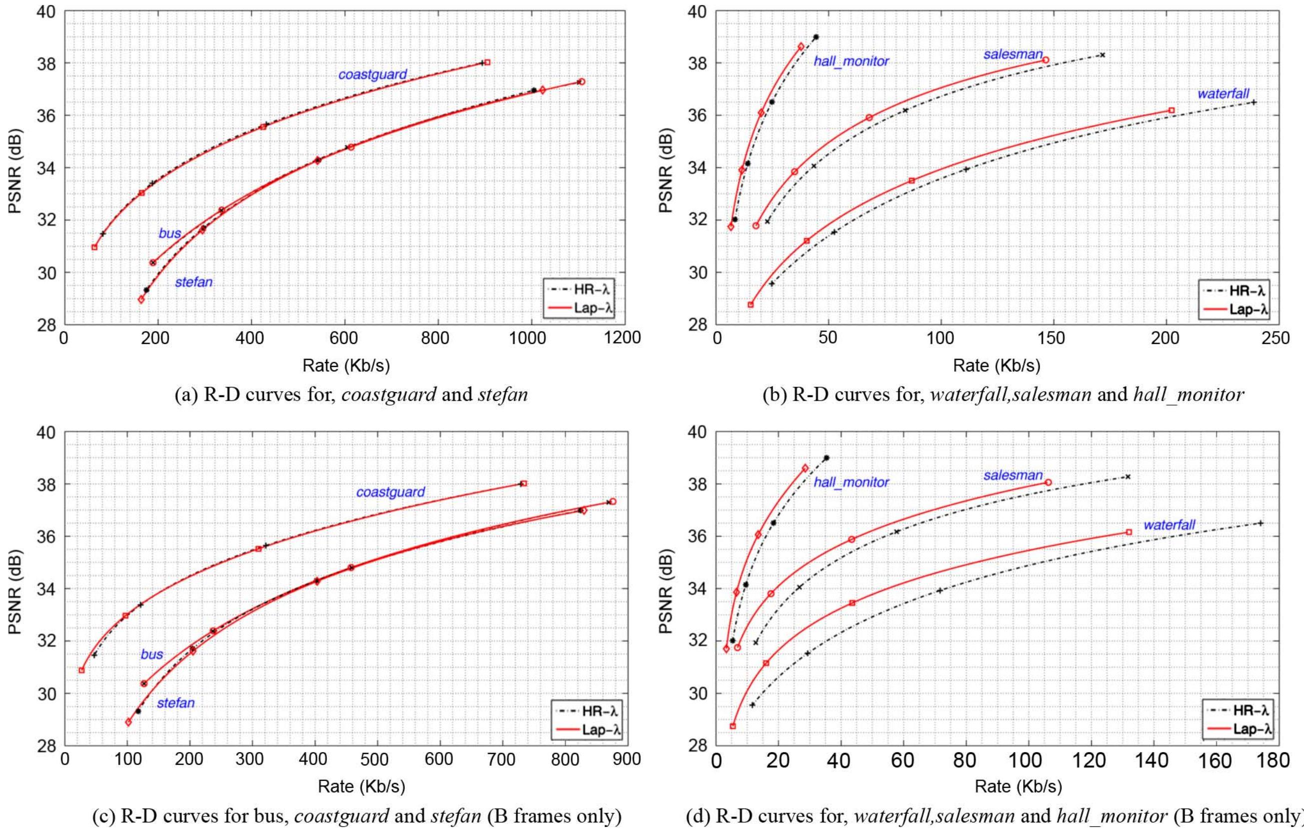
B. JVT Test II

In this test, the performance of the proposed algorithm in the environment of H.264/AVC Main Profile is evaluated. The same seven sequences as those in JVT-I are tested. The entropy coding is CABAC, frame mode is IBBP, and the first 148 frames $(\mathrm{I}+49\mathrm{P}+98\mathrm{B})$ of each sequence are coded.

Table I (right part) presents the simulation results. On average, 0.21 dB gain in PSNR or 5.21% rate reduction is achieved. For the peek gain, up to 0.90 dB gain for *container* can be observed in Fig. 4(a).

In addition, the performances of B-frames is also listed as ΔP^B and ΔR^B in Table I. On average, 0.48 dB gain in PSNR or 14.23% rate reduction are obtained.[4] And for *foreman*, the average gain is even 0.76 dB. In general, such significant gains are

[4]For *container*, ΔP^B and $\Delta P^B_{\mathrm{peak}}$ are both listed as N/A. This is because the calculator [42] failed with this calculation due to the fact that a part of the corresponding RD curve is vertical.



(a) R-D curves for, *coastguard* and *stefan*

(b) R-D curves for, *waterfall,salesman* and *hall_monitor*

(c) R-D curves for bus, *coastguard* and *stefan* (B frames only)

(d) R-D curves for, *waterfall,salesman* and *hall_monitor* (B frames only)

Fig. 5. R-D curves of HB-3 Test.

due to big λ_{Lap} applied on B-frames. As residuals in B-frames are normally quite small, λ_{Lap} becomes quite big so that the importance of lower rate is greatly emphasized in the RDO process. Then B-frames are coded with a little higher distortion but a much lower rate. Consequently, the overall performance for B-frames is highly improved. In fact, a very recent research [44] even observed that a big improvement can be obtained when residual encoding is deactived for B-frames at low to medium rates, which can be actually regarded as an extreme case where the residual encoding is totally suppressed by a very large λ.

When comparing ΔP and ΔP^B in Table I (right part), an interesting fact is noticeable. On one hand, a significant gain ΔP^B is achieved for B-frames. On the other hand, the overall gain ΔP for all inter frames is much less than the related ΔP^B even tough $2/3$ of inter frames were coded as B-frames. Why B-frames contribute so little to ΔP? As mentioned previously, the big ΔP^B is mainly from the lower rate R^B of B-frames. However, when examining the performance for all inter frames, the average rate R is largely depend on the rate R^{P} of P frames since R^{P} is much higher than R^B. In such a case, the average rate R is not that low so that the overall gain ΔP is much less than ΔP^B for B-frames.

C. HB-3 Test

In this test, the performance of the proposed algorithm for the hierarchical B coding structure in the environment of H.264/AVC Main Profile is investigated. To be more representative, another six sequences with different properties are tested, i.e fast movements with details: *bus*, *stefan*, median motion with scene change: *coastguard*, zooming with details: *waterfall*, and small movements: *salesman*, *hall_monitor*. The entropy coding is CABAC, three hierarchical B layers are used, and first 233 frames $(\mathrm{I} + 29\mathrm{P} + 203\mathrm{B})$ [5] of each sequence are coded.

Table II shows the simulation results. On average, 0.20 dB gain in PSNR or 5.50% rate reduction is achieved. And for the peak gain, up to 0.74 dB gain for $hall\_monitor$ can be observed in Fig. 5(b).

As reference frame, a smaller λ is normally preferred to reduce the error propagation. Therefore, hierarchical B scheme is usually a hard case for RDO since many B-frames will be used as reference frames as well. Even in such a case, a big gain is still obtained by the proposed algorithm, i.e., on average, 0.41 dB gain in PSNR or 12.53% rate reduction. And for *salesman*, up to 1.60 dB gain in PSNR or 42% rate reduction can be observed. In fact, according to the hierarchical structure, the referenced B-frames are separated in some distance, so that their residuals are not as small as those in normal B-frames. Consequently, proper λ_{Lap} can be adaptively derived and serious error propagation is automatically avoided. Obviously, this adaptivity by the proposed algorithm is better than the empirical λ adjusting

[5]Limited by sequence length, first 113 frames $(\mathrm{I} + 14\mathrm{P} + 98\mathrm{B})$ and 153 frames $(\mathrm{I} + 19\mathrm{P} + 133\mathrm{B})$ are coded for *bus* and *salesman*, respectively.

TABLE II
SIMULATION RESULTS OF HB-3

sequences	QP	Lap-λ over HR-λ								
		$\overline{RS}$	$\frac{\bar{\lambda}_{Lap}}{\bar{\lambda}_{HR}}$	ΔP (dB)	ΔR	ΔP_{peak} (dB)	ΔP^B (dB)	ΔR^B	ΔP^B_{peak} (dB)
bus (CIF, 30 Hz)	28 32 36 40	1.90 1.72 1.11 1.11	98.56% 99.56% 99.94% 100.00%	-0.01	0.30%	0.02	0.01	-0.26%	0.02
waterfall (CIF, 30 Hz)	28 32 36 40	0.24 0.21 0.19 0.07	391.61% 431.42% 453.50% 463.58%	0.34	-11.31%	0.44	0.76	-28.43%	1.03
coastguard (CIF, 30 Hz)	28 32 36 40	1.30 0.71 0.41 0.38	106.63% 233.08% 383.84% 432.95%	-0.04	1.60%	0.02	0.09	-3.24%	0.55
salesman (CIF, 30 Hz)	28 32 36 40	0.36 0.32 0.29 0.30	261.31% 397.96% 433.16% 444.40%	0.43	-12.76%	0.65	0.75	-23.46%	1.60
stefan (CIF, 30 Hz)	28 32 36 40	1.76 1.60 1.31 1.09	102.59% 110.24% 127.67% 262.91%	-0.03	0.76%	0.00	-0.02	0.74%	0.27
hall_monitor (QCIF, 30 Hz)	28 32 36 40	0.26 0.24 0.24 0.32	358.62% 409.26% 442.11% 453.49%	0.51	-11.60%	0.74	0.85	-20.53%	1.54
average	-	-	-	0.20	-5.50%	-	0.41	-12.53%	-

method for hierarchical B-frames in H.264/AVC reference software [12].

D. High Resolution Test

In this test, the performance of the proposed algorithm for high resolution sequences is assessed in H.264/AVC Main Profile environment. Three 576i (720×576) sequences (*mobcal*, *stockholm*, *shields*[6]) and three 4CIF sequences (*city*, *harbour*, *soccer*) are tested. The entropy coding is CABAC, frame mode is IBBP, and first 148 frames $(\mathrm{I}+49\mathrm{P}+98\mathrm{B})$ of each sequence are coded.

Table III gives the simulation results. This time, a similar performance with HR-λ is observed since the tested sequences are all quite complex with a lot of details involved. More important, these sequences also represent the cases which cannot be well captured by the models, such as gradually scene changes, non-Laplace distribution and very positive side information. Thus, the refresh process is applied more frequently, which can be noticed from comparing the column of $\overline{RS}$ in Tables I and III. In such cases, λ_{Lap} is refreshed backward to λ_{HR}, as shown in the column of $\bar{\lambda}_{Lap}/\bar{\lambda}_{HR}$ of the table. Consequently, the overall performance is similar to HR-λ. On the other hand, it is can also be regarded as a proof on the effectiveness of escape methods proposed in Section III-C. In fact, without these escape methods, the coding efficiency may degrade much for some complex sequences.

E. Discussions on Dependency and Local Adaptivity

So far the proposed RDO is on a frame level. In fact, it is implicitly supposed that the optimizations are independent so that the global optimality can be achieved when each frame is optimally coded. However, there are dependencies among frames because of the motion estimation. Very recent researches [16], [30] believed that by considering these dependencies, a joint optimization on a group of frames could further increase the coding efficiency. Generally, in such a case, the desired distortion for a frame will be determined by the potential error propagation, i.e., lower distortion should be preferred to keep a lower error propagation if the frame will be referenced a lot in the future. Otherwise, a higher distortion with a lower rate may be more efficient. Thus, if the error propagation is considered in the R-D models, an improvement can be expected. Unfortunately, it is quite difficult to apply such an idea in one-pass coding scheme

TABLE III
SIMULATION RESULTS OF HIGH-RESOLUTION SEQUENCES

sequences	QP	Lap-λ over HR-λ			
		$\overline{RS}$	$\frac{\bar{\lambda}_{Lap}}{\bar{\lambda}_{HR}}$	ΔP (dB)	ΔR
mobcal (576i, 30 Hz)	28 32 36 40	1.82 1.75 1.22 0.69	146.38% 157.05% 195.63% 440.18%	-0.02	0.98%
shields (576i, 30 Hz)	28 32 36 40	2.03 2.00 1.84 1.21	99.68% 100.25% 101.44% 116.69%	0.00	-0.05%
stockholm (576i, 30 Hz)	28 32 36 40	2.03 2.03 1.98 1.49	100.00% 100.00% 100.00% 100.00%	0.00	0.00%
city (4CIF, 30 Hz)	28 32 36 40	2.03 1.73 1.54 0.75	100.00% 100.00% 100.00% 316.23%	-0.04	1.46%
harbour (4CIF, 30 Hz)	28 32 36 40	1.24 0.94 0.90 0.88	98.96% 130.42% 182.79% 284.77%	-0.02	0.78%
soccer (4CIF, 30 Hz)	28 32 36 40	2.03 1.59 1.25 1.26	99.75% 100.00% 100.00% 221.41%	-0.03	0.92%
average	-	-	-	-0.02	0.68%

[6][Online]. Available: ftp.ldv.e-technik.tu-muenchen.de/dist/test_sequences/601/

since how to predict the error propagation in video sequence is still unknown.

Another interesting problem is about local adaptivity within a frame. Intuitively, a good local adaptivity is able to further improve the coding efficiency. Recent results [7]–[9] show that in H.264/AVC JM distortion, when RDO is off, a method of empirically adaptive λ at macroblock level is able to achieve a similar performance with the case of RDO on. However, it is hard to introduce such an idea into the proposed analytical scheme. First, the proposed models are based on distributions. If the considered area is small, such as one macroblock, the related statistical properties, like standard deviation, are not representative. Moreover, the proposed λ_{Lap} is a statistical conclusion, which indicates that an optimal performance can be achieved, at least in theory, if the whole area is coded with it. But for a single macroblock in the area, it may be not the best. Therefore, it is almost impossible to find a best λ for a single macroblock by a statistical model. Of course, adaptivity for bigger areas, such as foreground and background, may be a good way. But how to make the separation and how to handle macroblocks along edges of the areas are still open problems.

V. Conclusion and Future Work

In this paper, a Laplace distribution based Lagrangian RDO algorithm Lap-λ for one-pass hybrid video coding is presented. Based on Laplace distribution of transformed residuals, accurate rate and distortion models are first developed. Then an adaptive Lagrange multiplier λ_{Lap} is derived from the models. In fact, it has been proved that the proposed λ_{Lap} is mathematically a general form of λ_{HR} which is widely used, such as in the reference software of H.264/AVC. Furthermore, to keep a reasonable performance in the cases which cannot be well captured by the models, escape methods are also designed. So far, a complete solution for RDO is achieved. Comprehensive simulations verify that, compared with the most recent H.264/AVC reference software JM13.1, the proposed algorithm shows a much better, or at least a similar performance in all scenarios. Particularly, significant gains are achieved for slow sequences and B-frames, such as up to 1.79 dB gain in PSNR or 32% rate reduction for *container* in JVT-I test, and up to 1.60 dB gain in PSNR or 42% rate reduction for *salesman* B-frames in HB-3 test.

Actually, the contribution of this paper is not only for RDO. The proposed R-D models may be used in other aspects as well, such as rate control. Intuitively, the more accurate the R-D models are, the more gain can be expected for the rate control.

Nevertheless, the proposed Lap-λ currently works in a fixed-QP environment. How to embed the algorithm into the floating-QP scheme, like rate control, is under research. Moreover, replacing the current escape methods with analytical algorithms is worth consideration. Definitely, extending this work to the environment of scalable video coding and even multi-view video coding is also a very interesting topic.

References

[1] *JVT, Advanced Video Coding (AVC)*, ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 Part 10), 3rd ed., 2004.

[2] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, 1998.

[3] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Oper. Res.*, vol. 11, no. 3, pp. 399–417, 1963.

[4] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a ratedistortion sense," *IEEE Trans. Image Process.*, vol. 2, no. 2, pp. 160–175, feb. 1993.

[5] T. Wiegand, M. Lightstone, D. Mukherjee, T. Campbell, and S. Mitra, "Rate-distortion optimized mode selection for very low bit-rate video coding and the emerging H.263 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 182–190, Apr. 1996.

[6] J. Choi and D. Park, "A stable feedback control of the buffer state using the controlled Lagrange multiplier method," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 546–558, 1994.

[7] J. Zhang, X. Yi, N. Ling, and W. Shang, "Context adaptive Lagrange multiplier (CALM) for motion estimation in JM (JVT-S028)," in *Joint Video Team (JVT) of ISO/IEC MPEG ITU-T VCEG*, Geneva, Switzerland, Mar./Apr. 2006.

[8] J. Zhang, X. Yi, N. Ling, and W. Shang, "Context adaptive Lagrange multiplier (CALM) for motion estimation in JM—improvement (JVT-T046)," in *Joint Video Team (JVT) of ISO/IEC MPEG ITU-T VCEG*, Klagenfurt, Austria, Jul. 2006.

[9] J. Zhang, X. Yi, N. Ling, and W. Shang, "Chroma coding efficiency improvement with context adaptive Lagrange multiplier (CALM)," in *Proc. IEEE Int. Symp. Circuits Systems (ISCAS)*, New Oreleans, LA, May 2007, pp. 293–296.

[10] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, 1998.

[11] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Thessaloniki, Greece, 2001, vol. 3, pp. 542–545.

[12] JVT, H.264/AVC Reference Software (JM13.1), Dec. 2007 [Online]. Available: http://iphome.hhi.de/suehring/tml/

[13] L. Chen and I. Garbacea, "Adaptive lambda estimation in Lagrangian rate-distortion optimization for video coding," presented at the Visual Commun. Image Process. (VCIP), San Jose, CA, Jan. 2006.

[14] D. Bertsekas, *Dynamic Programming : Deterministic and Stochastic Models*. Upper Saddle River, NJ: Prentice-Hall, 1987.

[15] Y. Yang and S. Hemami, "Generalized rate-distortion optimization for motion-compensated video coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 6, pp. 942–955, Sep. 2000.

[16] E. Yang and X. Yu, "Rate distortion optimization for H.264 interframe coding: A general framework and algorithms," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1774–1784, Jul. 2007.

[17] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 246–250, Feb. 1997.

[18] *ISO/IEC, Coding of Audio-Visual Objects—Part 2: Visual*, ISO/IEC 14496-2 (MPEG-4 Part 2), 1999.

[19] V. Group, "MPEG-4 video verification model version 5.1," *ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Asssociated Audio* 1996.

[20] H. Gish and J. Pierce, "Asymptotically efficient quantizing," *IEEE Trans. Inf. Theory*, vol. 14, no. 5, pp. 676–683, Oct. 1968.

[21] *ITU-T, Video Coding for Low Bitrate Communication*, ITU-T Rec. H.263 Version3 (H.263++), 2000.

[22] *Video Codec Test Model, Near Term, Version 10 (TMN-10), Draft 1*, ITU-T/SG16/Q15-D-65, 1998.

[23] Z. He and S. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 970–982, Nov. 2002.

[24] Z. He and S. Mitra, "Optimum bit allocation and accurate rate control for video coding via ρ-domain source modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 840–849, Oct. 2002.

[25] E. Lam and J. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1661–1666, Oct. 2000.

[26] G. Yovanof and S. Liu, "Statistical analysis of the DCT coefficients and their quantization error," in *Proc. 30th Asilomar Conf. Signals, Syst. Comput.*, 1996, vol. 1, pp. 601–605.

[27] Y. Altunbasak and N. Kamaci, "An analysis of the DCT coefficient distribution with the H.264 video coder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2004, vol. 3, pp. III-177–80.

[28] R. Joshi and T. Fischer, "Comparison of generalized Gaussian and Laplacian modeling in DCT image coding," *IEEE Signal Process. Lett.*, vol. 2, no. 5, pp. 81–82, May 1995.
[29] J. Dong and N. Ling, "On model parameter estimation for H.264/AVC rate control," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2007, pp. 289–292.
[30] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Extended Lagrange multiplier selection for hybrid video coding using interframe correlation," presented at the Picture Coding Symposium (PCS), Lisbon, Portugal, Nov. 2007.
[31] G. Sullivan, "Recommended simulation common conditions for H.26L coding efficiency experiments on low-resolution progressive-scan source material (VCEG-N81)," in *ITU-T SG16 Q.6 (VCEG)*, Santa Barbara, USA, Sep. 2001.
[32] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Advanced Lagrange multiplier selection for hybrid video coding," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Beijing, China, Jul. 2007, pp. 364–367.
[33] *Coding of Moving Pictures and Associated Audio for Digital Storage Media at Up to About 1.5 Mbits/s—Part2: Video*, ISO/IEC 11172-2 (MPEG-1 Part 2), 1993.
[34] *Generic Coding of Moving Pictures and Associated Audio Information—Part 2: Video*, ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2 Part 2), 1994.
[35] *Video Codec for Audiovisual Services at px64 kbits/s*, ITU-T Rec. H.261 Version 2, 1993.
[36] D.-K. Kwon, M.-Y. Shen, and C. Kuo, "Rate control for H.264 video with enhanced rate and distortion models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 5, pp. 517–529, May 2007.
[37] 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.
[38] M. Flierl, T. Wiegand, and B. Girod, "Rate-constrained multihypothesis prediction for motion-compensated video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 957–969, Nov. 2002.
[39] M. Jiang and N. Ling, "An improved frame and macroblock layer bit allocation scheme for H.264 rate control," in *IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2005, vol. 2, pp. 1501–1504.
[40] A. Tourapis, A. Leontaris, K. Suehring, and G. Sullivan, "H.264/MPEG-4 AVC reference software manual (JVT-X072)," in *Joint Video Team (JVT) of ISO/IEC MPEG ITU-T VCEG*, Geneva, Switzerland, Jun.-Jul. 2007.
[41] A. Hallapuro, M. Karczewicz, and H. Malvar, "Low complexity transform and quantization—Part II: Extensions (JVT-B039)," in *Joint Video Team (JVT) of ISO/IEC MPEG ITU-T VCEG*, Geneva, Switzerland, Jan. 2002.
[42] G. Bjontegaard, "Calculation of average PSNR differences between RD curves (VCEG-M33)," in *ITU-T SG16 Q.6 (VCEG)*, Austin, TX, Apr. 2001.
[43] M. Jiang and N. Ling, "On Lagrange multiplier and quantizer adjustment for H.264 frame-layer video rate control," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 5, pp. 663–669, May 2006.
[44] N. Rodrigues, E. Silva, M. Carvalho, S. Faria, and V. Silva, "Overriding H.264/AVC B-slice predicted residue coding," presented at the Picture Coding Symposium (PCS), Lisbon, Portugal, Nov. 2007.

Xiang Li (S'02) received the B.Sc. and M.Sc. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2001 and 2004, respectively. Since 2005, he has been working toward the Ph.D. degree at the Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, Germany. During this time, he is sponsored by Siemens doctoral scholarship and works in Networks and Multimedia Communications department at Siemens Corporate Technology in Munich, Germany

From 2004 to 2005 he was with Siemens Corporate Technology, Beijing, China, where he served as a Senior Video Engineer and independently developed a real time codec for H.264/AVC Main Profile on PC platform. .

Norbert Oertel received the Dipl. Inform. Univ. degree in computer science from Technical University of Munich, Germany, in 1999.

Since 1999 he has been with Siemens Corporate Technology, Munich, Germany, where he contributed to the standardization of H.264 at ITU and MPEG and now leads innovation projects related to video communications with the Siemens business units. His current areas of interest are video compression technologies, error resilient video transmission over IP networks, efficient implementation of video codecs on general purpose hardware platforms and the design of video systems.

Andreas Hutter (M'06) received the diploma and Dr.-Ing. degrees in communications engineering from the Munich University of Technology, Munich, Germany, in 1993 and 1999, respectively.

From 1993 to 1999 he was as a Research Assistant with the Institute for Integrated Circuits of the Munich University of Technology, where he mainly worked on algorithms for video coding and on the implementation of multimedia systems for mobile terminals. He joined Siemens Corporate Technology, Munich, Germany, in 1999, where he is currently leading the competence centre for video and multimedia communications. He has been an active member of MPEG from 1995 where he contributed to the MPEG-4, the MPEG-7 and the MPEG-21 standards. He was co-editor of the MPEG-7 Systems standard and he was acting as HoD (Head of Delegation) of the German National Body at MPEG. He has also been actively involved in several European research projects, where he has been work package leader of the EU-IST projects ISIS and DANAE.

André Kaup (M'96–SM'99) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from RWTH Aachen University, Germany, in 1989 and 1995, respectively.

From 1989 to 1995 he was with the Institute for Communication Engineering, Aachen University of Technology, where he was responsible for industrial as well as academic research projects in the area of high resolution printed image compression, object-based image analysis and coding, and models for human perception. In 1995 he joined the Networks and Multimedia Communications Department, Siemens Corporate Technology, Munich, Germany, where he chaired work packages in several European research projects in the area of very low bit-rate video coding, image quality enhancement and mobile multimedia communications. In 1999 he was appointed Head of the mobile applications and services group in the same department, with research focusing on multimedia adaptation for heterogeneous communication networks. Since 2001 he is Full Professor and Head of the Chair of multimedia communications and signal processing at Friedrich-Alexander-University of Erlangen-Nuremberg, Germany. From 1998 to 2001 he served as Adjunct Professor at Technical University of Munich and the University of Erlangen-Nuremberg, teaching courses on image and video communication. He has published over 120 scientific journal and conference papers and holds 26 patents.

Prof. Kaup was elected Siemens inventor of the year 1998 and is recipient of the 1999 ITG Award. From 2005 to 2007, he was vice-speaker of the DFG Collaborative Research Center 603 "Modeling and Analysis of Complex Scenes and Sensor Data." He is member of the German ITG.