

Rate-Distortion Optimized Bitstream Extractor for Motion Scalability in Wavelet-Based Scalable Video Coding

Meng-Ping Kao, *Student Member, IEEE*, and Truong Q. Nguyen, *Fellow, IEEE*

Abstract—Motion scalability is designed to improve the coding efficiency of a scalable video coding framework, especially in the medium to low range of decoding bit rates and spatial resolutions. In order to fully benefit from the superiority of motion scalability, a rate-distortion optimized bitstream extractor, which determines the optimal motion quality layer for any specific decoding scenario, is required. In this paper, the determination process first starts off with a brute force searching algorithm. Although guaranteed by the optimal performance within the search domain, it suffers from high computational complexities. Two properties, i.e., the monotonically nondecreasing property and the unimodal property, are then derived to accurately describe the rate-distortion behavior of motion scalability. Based on these two properties, modified searching algorithms are proposed to reduce the complexity (up to five times faster) and to achieve the global optimality, even for those decoding scenarios outside the search domain.

Index Terms—Bitstream extractor, motion scalability, rate distortion optimization, scalable video coding.

I. INTRODUCTION

A typical scalable video coding framework (SVCF), as shown in Fig. 1, is composed of three main building blocks, i.e., the encoder, the decoder, and the bitstream extractor. Compared to a conventional non-scalable video codec, the decoder in an SVCF is allowed to demand a variety of decoding specifications, including different combinations of spatial, temporal, and quality layers. It is the main task of the bitstream extractor to fulfill those requests by properly truncating the scalable bitstream.

In general, both the encoder and the bitstream extractor are not normative. Therefore, designers keep certain freedom in developing their own implementations, which also differentiates the performances of various codecs conforming to the same standard. There are, however, some requirements in designing a bitstream extractor. For example, the adapted bitstream must remain scalable itself and can be fed back to the extractor again for further truncations. On the other hand, the decoder should be

Manuscript received August 04, 2008; revised November 10, 2009. First published December 28, 2009; current version published April 16, 2010. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kiyoharu Aizawa.

The authors are with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093-0407 USA (e-mail: gyalpp@gmail.com; nguyent@ece.ucsd.edu).

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/TIP.2009.2039373

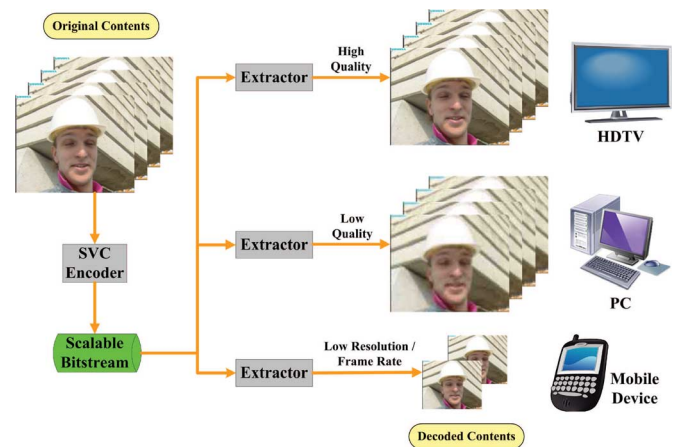


Fig. 1. Scalable video coding framework.

able to decode all adapted bitstreams that are legitimately generated by the extractor.

Besides these requirements, the designing criteria for a generic bitstream extractor can be rather trivial. We now take the SVC standard [1] for example, which is an extension, Annex G, to H.264/MPEG-4 AVC video compression standard [2]. Note that in order to avoid confusion, we refer SVC specifically to the aforementioned SVC standard, while SVCF and WSVC are referred respectively to the generic scalable video coding framework and the wavelet-based scalable video coding scheme we adopt for simulations throughout the paper. In SVC, the elementary unit of the bitstream is a network abstraction layer unit (NALU). Each NALU belongs to a certain temporal, spatial, and quality layer and is tagged accordingly through high level syntax, $temporal_id(T)$, $dependency_id(D)$, and $quality_id(Q)$. In the case where a specific spatio-temporal resolution is explicitly indicated by $T = T_t$ and $D = D_t$, the extraction can be easily done by dropping all NALUs with $T > T_t$ and $D > D_t$ [3]. A similar example based on a wavelet-based scalable video coding framework can be found in [4].

If there is an additional bit rate constraint imposed, which the remaining NALUs fail to meet, some NALUs with $Q > 0$ have to be further discarded. This is the case where different designing principles come into effect, among which the rate-distortion (RD) optimized extraction is the most popular one [5], [6]. The overall idea is to retain those NALUs with higher RD contribution and, therefore, to optimize the quality under the rate constraint.

There is yet another situation where only the rate constraint is present, not the spatio-temporal resolution. For example, in the case where the total bit rate is restricted by the applicable channel conditions, the extractor has the freedom to select the spatio-temporal resolution that best suits the display interest of the end device. Some researches have demonstrated the criteria of either optimizing the visual perception or minimizing the decoder complexity in this case [7]–[9].

When motion scalability [10]–[13] is taken into consideration, the bitstream extractor has an additional requirement, i.e., optimal bit allocation among motion and texture [12], [14]. Since motion scalability provides an additional option to flexibly distribute the budget bits among motion and texture, the optimal decision in terms of RD performance becomes another interesting topic in the extractor design. In this paper, we focus on the case where the decoding spatio-temporal resolution (T and D in SVC) is prespecified and fixed. Under this setup, one of the motion quality (MQ) layers, combining with the corresponding texture information, will provide the best reconstructed quality. As the target bit rate varies, however, the optimal motion quality layer also changes accordingly. The optimal motion quality layer as a function of decoding bit rate, if not provided by the encoder, will be determined by the extractor. Based on this function, the adapted bitstream is guaranteed with the best decoding quality throughout all possible rates, for this particular spatio-temporal resolution.

We propose three approaches for determining the optimal motion quality layer under a varying bit rate constraint, i.e., brute force, model-assisted, and model-based. The brute force method exercises an exhaustively searching algorithm among all possible motion quality layers. Note that the optimal motion quality layer can only change on a basis of group of pictures (GOP) throughout the entire sequence. Therefore, for each testing layer, a complete decoding process is performed on the target GOP and the peak signal-to-noise ratio (PSNR) of the reconstructed GOP is recorded. The optimal layer is chosen as the one that results in the highest PSNR. Note also that only a discrete finite set of the possible decoding bit rates can be tested in this approach.

The model-assisted method is based on two properties derived directly from the exponential RD model in the general source coding theory [15], [16] and from the additive distortion model for motion scalability [12]. The first one is the monotonically nondecreasing property of the optimal motion quality layer as a function of the decoding bit rate. The second one is the unimodal property of the decoded quality as a function of the motion quality layer. By applying these two properties, irrelevant testing scenarios can be omitted without sacrificing the extractor performance. Moreover, the monotonically nondecreasing property can further simplify the description of the optimal motion quality layers by recording only the critical rates at which the change in optimal motion quality layer occurs. One of the blind approaching methods for determination of these critical rates is the bisection method.

Given the critical rate representation, the model-based method aims at improving the approaching speed of a blind method, such as the bisection method. By explicitly applying the exponential RD model, a more accurate estimation of the

critical rate can be efficiently predicted. This estimation is in general better than the middle point estimate of the possible rate range, which is predicted using the bisection method. Through parameter estimation, the resulting model can adapt to the actual video contents. In general, if the adapted model fits well, a reduced number of trial and errors can be expected and the efficiency can be greatly improved.

The remainder of the paper is organized as follows. In Section II, we give a brief review of the scalable motion model (SMM) and the associated WSVC framework [10], on which the following experiments will be performed. A model-based theoretical justification of motion scalability is presented in Section III. Through the analysis, it will be clear how and when motion scalability can benefit the coding efficiency. In Section IV, a detailed description of the three proposed methods for optimal bitstream extraction is given. The properties that facilitate more efficient extractor designs are also derived here, based on the models introduced in the previous section. Finally, we provide the experimental results that verify the efficiency of our new extractor designs in Section V.

II. REVIEW OF THE SCALABLE MOTION MODEL AND THE WAVELET-BASED SCALABLE VIDEO CODING

The study of motion scalability began with the development of SVCF. Researchers had noticed that the lossless coding of motion information becomes inefficient when decoding the bitstream at lower resolutions [17]. Later on, the importance of scalable motion quality to low rate decoding was also well recognized [13]. In 2004, more than 20 publications addressed motion scalability in the context of different requirements and different SVCF's [18]–[20].

In our earlier work [10], a fully scalable motion model was proposed to solve the motion scalability problem from many aspects, along with tailored encoding techniques to minimize the coding overhead of scalability. Moreover, the associated rate distortion optimized estimation algorithm was also provided to achieve optimality throughout various decoding scenarios. In order to make this paper more self-contained, we highlight some relevant features of the proposed SMM in this section.

A. Scalable Motion Model

The basic cell of the proposed SMM is a macroblock (MB). Fig. 2 shows the structure of this basic cell. We explicitly implement both refining methods for motion scalability in our model, i.e., the motion vector (MV) accuracy dimension and the variable block size (VBS) dimension along horizontal and vertical axes, respectively.

Even though the proposed model provides two dimensions to fine tune the motion quality, the best motion for a target bit rate remains unique. Experimental results have shown that the MV accuracy dimension represents the gradual change of the underlying motion quality better than the other. Therefore, we assign a λ to each MV accuracy level according to its target operating bit rate. During the RD optimized motion estimation (ME) process, refinements from higher MV accuracy levels are not allowed in the current motion quality layer. Refinements from all VBS levels, on the other hand, are possible candidates and their values are purely determined by the estimation process.

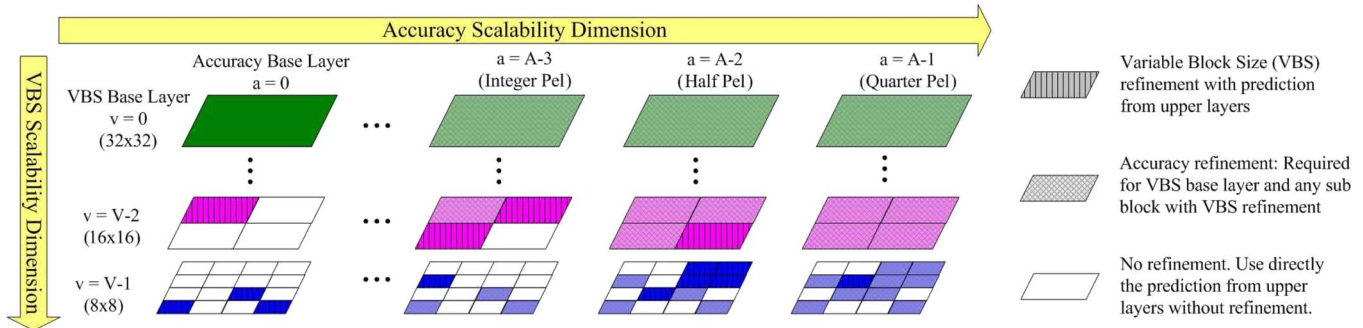


Fig. 2. Proposed fully scalable motion model.

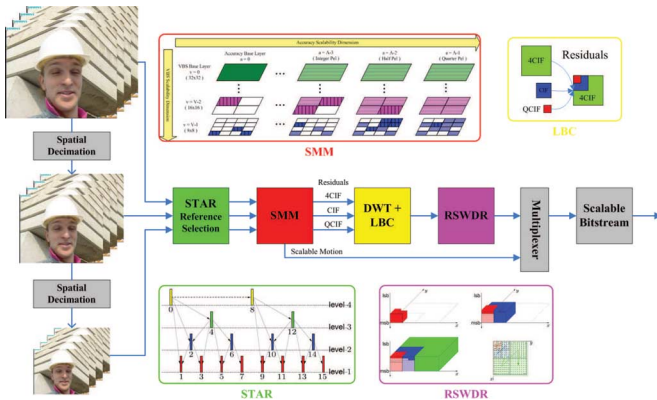


Fig. 3. WSVC system diagram with proposed SMM embedded.

The idea behind choosing the MV accuracy dimension as motion quality layers is that the VBS structure is more content dependent and it should be optimized to the underlying motion via RDO ME. It is, however, not to say that VBS scalability has no control on motion quality. Instead, it comes in great effect in a more implicit way. An increasing bit budget for the motion model could result in a more refined motion structure. This is the reason why the incomplete quadtree structure keeps growing in the example shown in Fig. 2 as the MV accuracy level increases.

B. WSVC Framework

The proposed SMM will be incorporated into a WSVC framework [21] to provide fully motion scalability as shown in Fig. 3. In this WSVC framework, successive temporal approximation and referencing (STAR), low band correction (LBC), and resolution scalable wavelet difference reduction (RSWDR) are adopted for realizing temporal, spatial and quality scalability, respectively.

The simplest way to reduce the frame rate is to drop irrelevant frames. STAR actually borrows this simple idea and shortens the end-to-end delay such that real time encoding/decoding becomes possible, which is similar to the hierarchical B-picture decomposition in SVC. LBC, on the other hand, is designed to merge all error frames from different resolutions into a single error frame. In this way, no overhead will be imposed while implementing spatial scalability. Perfect reconstruction is also guaranteed as long as some assumptions are strictly followed.

RSWDR is a modified subband bit plane coding algorithm from WDR [22]. It is originally designed to decouple the inter

band correlation that is inherited from WDR and to make independent decoding of different spatial resolution sequences possible. In RSWDR, a target bit rate can be chosen for the base resolution and different target bit rates can be chosen for higher resolutions. During the decoding process, decoupled sub-bitstreams representing different spatial resolutions are ordered from small to large and are consumed sequentially until the target bit rate is reached.

C. Modified Resolution Scalable Wavelet Difference Reduction

The original decoding algorithm in RSWDR follows a single decoding path from the sub-bitstream representing the smallest spatial resolution to the one representing the largest. The advantages include an easy truncation operation (specified by the target bit rate) and the minimization of incoherent motion compensation (MC) operations for smaller resolutions.

However, following this decoding rule might also result in skewed RD curves for higher resolutions, as shown in Fig. 4, which is not preferable to both the decoder and the extractor. The reason for the skewed RD curve is that the single decoding path does not properly reflect the ordered RD slopes of decoded coefficients. In the example shown in Fig. 4, coefficients with smaller RD slopes are decoded prior to others with larger slopes, just because they appear in the sub-bitstream that gets decoded first in the path. Later on when the coefficients with larger slopes are decoded, a discontinued jump in the RD curve comes up.

In order to avoid this undesired jump, we break the single decoding path and rearrange the priority of coefficients to be decoded by their RD slopes, or simply their bit planes. As observed in Fig. 4, the new RD curve becomes smoother and the possible MC incoherent problem from smaller resolutions is so subtle that only a slight decrease is observed around the target bit rate of QCIF sized sequence, which is 512 kbps in this example.

III. THEORETICAL JUSTIFICATION OF MOTION SCALABILITY

Motion information has traditionally been coded losslessly due to the complicated impacts that a corrupted motion may bring to the reconstructed quality. In a decoding scenario where a reduced target bit rate is requested, it is intuitively easier to discard the extra texture bits than to quantize the motion parameters. One reason is that the quantization effect of texture bits is easily predicted and quantified. On the contrary, quantization of motion bits might result in a nonpredictable behavior that is highly correlated to the individual video content. The second

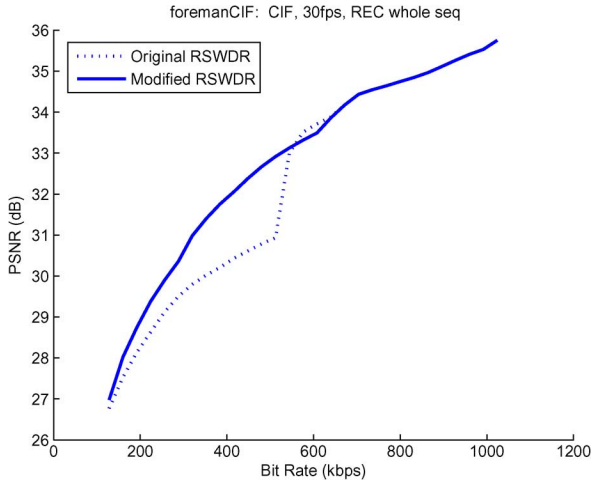


Fig. 4. Comparison of two RSWDR algorithms.

reason is that, even though motion information can be quantized, the corresponding texture that reflects the actual difference between the current picture and the new motion-predicted picture can not be re-encoded/transmitted. Only the unique version of texture that corresponds to the highest fidelity motion prediction is available for decoding. The mismatch between the quantized motion parameters and the original texture information prevents further investigation of the feasibility of motion scalability.

In this section, the aforementioned two doubts on motion scalability will be justified via simplified mathematical models, which are built based on some simplification assumptions. These models provide a clear insight to motion scalability and explain the question how an SVCF system can benefit from motion scalability. They are also the theoretical bases for designing a better bitstream extractor, as will be detailed in Section IV.

A. Linear Motion Distortion Model

The first work analyzing the distortion introduced by MV quantization is done by Secker [12]. We will briefly review his efforts here with some notation modifications in accordance with the remaining equations throughout this paper.

The motion distortion (or the distortion introduced by MV quantization) is defined as the mean squared error (MSE) between the motion compensated picture using the best motion information, which is used to generate the residual picture at the encoder, and the one using the quantized motion information. For simplicity, the following derivations are based on the assumptions that only a globally translational motion error is considered, i.e., a constant displacement MV error δ is applied for the entire picture, and that the picture boundary effect is ignored. Note also that the error propagation due to the MV quantization is not considered here as the perfect reference picture is assumed to be available.

Consider the perfect reference picture, $r[\mathbf{n}] \equiv r[n_1, n_2]$, and the motion warping operation, \mathcal{W}^* , which corresponds to the best, nonquantized motion. The motion compensated picture can be expressed as $m^*[\mathbf{n}] = \mathcal{W}^*(r)[\mathbf{n}]$. Suppose now \mathcal{W} is affected by MV quantization, resulting in the quantized oper-

ator \mathcal{W}' , which produces $m'[\mathbf{n}] = \mathcal{W}'(r)[\mathbf{n}]$. Under the assumption, $m'[\mathbf{n}] = m^*[\mathbf{n} - \delta]$, or in the frequency domain, $m'(\boldsymbol{\omega}) = m^*(\boldsymbol{\omega})e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}}$. By Parseval's theorem, the motion distortion $D_{m'}$ is given by

$$\begin{aligned} D_{m'} &\equiv \frac{1}{N_1 N_2} \sum_{n_1} \sum_{n_2} |m'[n_1, n_2] - m^*[n_1, n_2]|^2 \\ &= \frac{1}{N_1 N_2 (2\pi)^2} \\ &\quad \times \int_{\omega_1} \int_{\omega_2} S_{m^*}(\boldsymbol{\omega}) \left| (1 - e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}}) \right|^2 d\omega_1 d\omega_2 \end{aligned} \quad (1)$$

where $S_{m^*}(\boldsymbol{\omega}) = |m^*(\boldsymbol{\omega})|^2$ is the energy spectral density of $m^*[\mathbf{n}]$. For small δ , (1) can be approximated by applying the Taylor series expansion on $|1 - e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}}|^2$ and retaining the second order term $(\boldsymbol{\omega}^t\boldsymbol{\delta})^2$ only

$$D_{m'} \approx \frac{1}{N_1 N_2 (2\pi)^2} \int_{\omega_1} \int_{\omega_2} S_r(\boldsymbol{\omega}) (\boldsymbol{\omega}^t\boldsymbol{\delta})^2 d\omega_1 d\omega_2. \quad (2)$$

In (2), we have made a further assumption that $m^*[\mathbf{n}]$ and $r[\mathbf{n}]$ have similar energy spectral density, i.e., $S_{m^*}(\boldsymbol{\omega}) \approx S_r(\boldsymbol{\omega})$. Equation (2) can be expressed in the following form:

$$\begin{aligned} D_{m'} &\approx \Psi_{1,r} \delta_1^2 + \Psi_{2,r} \delta_2^2 + \Psi_{3,r} \delta_1 \delta_2 \\ &= (\Psi_{1,r} \cos^2(\theta_\delta) + \Psi_{2,r} \sin^2(\theta_\delta) \\ &\quad + \Psi_{3,r} \cos(\theta_\delta) \sin(\theta_\delta)) \|\boldsymbol{\delta}\|^2 \end{aligned} \quad (3)$$

where

$$\Psi_{1,r} = \frac{1}{N_1 N_2 (2\pi)^2} \int_{\omega_1} \int_{\omega_2} S_r(\boldsymbol{\omega}) \omega_1^2 d\omega_1 d\omega_2 \quad (4)$$

$$\Psi_{2,r} = \frac{1}{N_1 N_2 (2\pi)^2} \int_{\omega_1} \int_{\omega_2} S_r(\boldsymbol{\omega}) \omega_2^2 d\omega_1 d\omega_2 \quad (5)$$

$$\Psi_{3,r} = \frac{1}{N_1 N_2 (2\pi)^2} \int_{\omega_1} \int_{\omega_2} S_r(\boldsymbol{\omega}) \omega_1 \omega_2 d\omega_1 d\omega_2 \quad (6)$$

and θ_δ denotes the angle of $\boldsymbol{\delta}$ in polar form. By taking average over all θ_δ on both sides of (3), we get

$$D_{m'} \approx \frac{\Psi_{1,r} + \Psi_{2,r}}{2} \|\boldsymbol{\delta}\|^2 \equiv \Psi \|\boldsymbol{\delta}\|^2 \quad (7)$$

where Ψ is the isotropic motion sensitivity factor of the reference picture over all MV errors with magnitude $\|\boldsymbol{\delta}\|$. Strictly speaking, $D_{m'}$ is now also an averaged motion distortion over all MV errors with magnitude $\|\boldsymbol{\delta}\|$. Note that Ψ is a function of $S_r(\boldsymbol{\omega})$, which is highly content dependent.

To summarize, under a series of assumptions and approximations, we come up with a linear motion distortion model as shown in (7), which describes the linear relationship (with slope Ψ) between the MSE of MV errors and the corresponding MSE of MC errors, or simply known as the motion distortion.

B. Additive Distortion Model

In a generic video coding framework, the MC operation is followed by the texture/residual transform coding and quantization. The texture picture is defined as $t^*[\mathbf{n}] \equiv c[\mathbf{n}] - m^*[\mathbf{n}]$,

where $c[\mathbf{n}]$ is the current picture under encoding. The superscript, $*$, again represents the nonquantized texture picture, which differentiates from the texture picture after quantization, i.e., $t'[\mathbf{n}] \equiv \mathcal{Q}'(t^*)[\mathbf{n}]$. Note that the forward and the inverse transforms can be omitted here without confusion.

In the case where motion is non-scalable, the total distortion of the reconstructed picture, D , can be simply described by the texture distortion, $D_{t'}$, defined as the MSE between $t'[\mathbf{n}]$ and $t^*[\mathbf{n}]$

$$D_{t'} \equiv \frac{1}{N_1 N_2} \sum_{n_1} \sum_{n_2} |t'[n_1, n_2] - t^*[n_1, n_2]|^2. \quad (8)$$

However, if motion scalability is taken into consideration, the motion distortion may also contribute to the total distortion. The additive distortion model [12] states that the total distortion is the summation of the motion distortion and the texture distortion. The assumption behind this model is that the motion error, $m'[\mathbf{n}] - m^*[\mathbf{n}]$, and the texture error, $t'[\mathbf{n}] - t^*[\mathbf{n}]$, are orthogonal to each other

$$\begin{aligned} D_{m'} + D_{t'} &= \frac{1}{N_1 N_2} \left(\|m'[\mathbf{n}] - m^*[\mathbf{n}]\|^2 + \|t'[\mathbf{n}] - t^*[\mathbf{n}]\|^2 \right) \\ &= \frac{1}{N_1 N_2} \|m'[\mathbf{n}] + t'[\mathbf{n}] - c[\mathbf{n}]\|^2 = D. \end{aligned} \quad (9)$$

C. Beneficial Condition for Motion Scalability

Although the true distortion-rate model is data dependent and complicated, a simpler model has been derived and used for video texture coding [15], [16], [23]

$$D_t(R_t) = \sigma_t^2 \exp\left(-\frac{R_t}{a_t}\right) \quad (10)$$

where R_t is the texture bit rate and σ_t and a_t are content dependent parameters. Note that this exponential model is derived by applying the high-rate assumption, i.e., the quantization step size is small enough such that the quantization noise is almost uniformly distributed. This model provides an explicit way to quantify the texture distortion, D_t , according to the texture bit rate, R_t .

A similar exponential model can also be applied on MV coding, making the MV error, $\|\delta\|^2$, an exponential function of the motion bit rate, R_m . Therefore, (7) can be expressed as follows:

$$D_m(R_m) = \Psi \sigma_m^2 \exp\left(-\frac{R_m}{a_m}\right). \quad (11)$$

Applying the additive distortion model in (9), the total distortion-rate model becomes

$$D(R) = D_m(R_m) + D_t(R_t) \quad (12)$$

where $R = R_m + R_t$ is the total decoding bit rate.

Knowing the above distortion-rate relationships, we are able to derive a quantitative description of the beneficial condition of motion scalability. First of all, a video codec is said to be better than the other if its total distortion is smaller at the same

TABLE I
EXTRACTOR RD TABLE FOR THE BRUTE FORCE METHOD
(FOREMAN @ CIF 30 fps)

MQ Layer	Decoding Bit Rate (kbps)							
	128	256	384	512	640	768	896	1024
0	26.97	29.73	31.26	32.17	32.73	33.19	33.40	33.53
1	-	29.9	31.76	32.91	33.77	34.36	34.70	34.97
2	-	-	31.29	32.73	33.86	34.64	35.11	35.75

decoding bit rate. Consider the following two cases where the first one is coded with lossless motion, i.e., $D^*(R) = D_t(R - R_m^*)$, and the second one is coded with scalable motion (with MQ layer a), i.e., $D^a(R) = D_m(R_m^a) + D_t(R - R_m^a)$. The condition for scalable motion to outperform lossless motion is simply $D^a(R) < D^*(R)$, or

$$D_m(R_m^a) < D_t(R - R_m^*) - D_t(R - R_m^a) \quad (13)$$

where $R_m^a < R_m^*$. In other words, if the motion distortion is less than the texture distortion difference resulting from the applicable texture bit rate difference, i.e., $R_m^* - R_m^a$, the scalable motion case can achieve a better decoding quality. As can be derived from (10), the right hand side of (13) is a monotonically decreasing function of R . As a consequence, the satisfaction of (13) can be expected for a relatively smaller R , when R_m^a is fixed.

IV. OPTIMAL BITSTREAM EXTRACTOR DESIGN FOR MOTION SCALABILITY

We propose three approaches to realize the optimal bitstream extractor for motion scalability in this section. As mentioned earlier, the purpose is to determine the best motion quality layer for all possible target bit rates, as the decoding spatio-temporal resolution is fixed.

A. Brute Force Method

In the brute force method, the same video bitstream is decoded multiple times at the same bit rate, during each time a different motion quality layer is applied. The same process is repeated for all decoding bit rates of interest, resulting in an extractor RD table. An example for the first GOP (8 pictures) of the FOREMAN sequence is shown in Table I, with the fixed spatio-temporal resolution at CIF, 30 fps. Note that the entry marked with “-” indicates that the target bit rate is too low to be decodable. The entry marked with a bold face number reflects the best motion quality layer at its decoding bit rate.

A simplified table, as shown in Table II, records the effective bit rate range for each motion quality layer. A such table, one for each spatio-temporal resolution and GOP, contains all the required information for optimal bitstream adaptation. These tables are not large and can be efficiently compressed for transmission.

The accuracy of the recorded effective rate range is seriously affected by the number of testing bit rate in the brute force method. As observed in Table II, the range boundary is chosen as the average of two contiguous testing rates. The more testing

TABLE II
EXTRACTOR INFORMATION FOR THE BRUTE FORCE METHOD
(FOREMAN @ CIF 30 fps)

MQ Layer	Effective Bit Rate Range (kbps)
0	0 - 192
1	192 - 576
2	576 - 1024

bit rates, the better extractor performance, and, of course, the more computational burdens.

B. Model-Assisted Method

Two properties can be observed from the example shown in Table I. First, the optimal motion quality layer is a monotonically nondecreasing function of the decoding bit rate. Second, the decoding PSNR at a fixed bit rate is a unimodal function of the motion quality layer. With the help of those models built in Section III, we will now prove that these two properties follows directly as long as those models are assumed. The advantage of knowing these properties is to efficiently save some trials that are irrelevant to the final extractor table, as shown in Table II.

1) *Monotonically Nondecreasing Property*: Suppose at a certain decoding bit rate R_0 , the minimal distortion is achieved with motion quality layer i , which occupies a motion bit rate R_m^i . By plugging in (10), (11), and (12), we have

$$\begin{aligned} & \Psi \sigma_m^2 \exp\left(-\frac{R_m^i}{a_m}\right) + \sigma_t^2 \exp\left(-\frac{R_0 - R_m^i}{a_t}\right) \\ & \leq \Psi \sigma_m^2 \exp\left(-\frac{R_m^j}{a_m}\right) + \sigma_t^2 \exp\left(-\frac{R_0 - R_m^j}{a_t}\right), \quad \forall j \neq i \end{aligned} \quad (14)$$

and after some simplification yields

$$\begin{aligned} & \Psi \sigma_m^2 \left(\exp\left(-\frac{R_m^i}{a_m}\right) - \exp\left(-\frac{R_m^j}{a_m}\right) \right) \leq -\sigma_t^2 \exp\left(-\frac{R_0}{a_t}\right) \\ & \quad \times \left(\exp\left(\frac{R_m^i}{a_t}\right) - \exp\left(\frac{R_m^j}{a_t}\right) \right), \quad \forall j \neq i. \end{aligned} \quad (15)$$

Given an extra bit rate $\Delta R > 0$, the difference between the total distortion using motion quality layers i and j becomes

$$\begin{aligned} & D^i(R_0 + \Delta R) - D^j(R_0 + \Delta R) \\ & = \Psi \sigma_m^2 \left(\exp\left(-\frac{R_m^i}{a_m}\right) - \exp\left(-\frac{R_m^j}{a_m}\right) \right) \\ & \quad + \sigma_t^2 \exp\left(-\frac{R_0 + \Delta R}{a_t}\right) \\ & \quad \times \left(\exp\left(\frac{R_m^i}{a_t}\right) - \exp\left(\frac{R_m^j}{a_t}\right) \right) \\ & \leq \sigma_t^2 \exp\left(-\frac{R_0}{a_t}\right) \left(\exp\left(\frac{R_m^i}{a_t}\right) - \exp\left(\frac{R_m^j}{a_t}\right) \right) \\ & \quad \times \left(\exp\left(-\frac{\Delta R}{a_t}\right) - 1 \right). \end{aligned} \quad (16)$$

Note that the inequality in (16) comes from plugging in (15). Since both a_t and $\Delta R > 0$, we have $\exp(-\Delta R/a_t) - 1 < 0$. In addition, for those motion quality layers $j < i$, the corresponding motion bit rates are smaller, i.e., $R_m^j < R_m^i$. Therefore, we have $(\exp(R_m^i/a_t) - \exp(R_m^j/a_t)) > 0$. In summary,

TABLE III
EXTRACTOR INFORMATION FOR THE MODEL-ASSISTED METHOD
(FOREMAN @ CIF 30 fps)

MQ Layer	Critical Rate (kbps)
0 - 1	172
1 - 2	600

the right hand side of (16) is negative whenever $j < i$. In other words

$$D^i(R_0 + \Delta R) < D^j(R_0 + \Delta R), \quad \forall j < i. \quad (17)$$

if $D^i(R_0) \leq D^j(R_0), \forall j \neq i$.

Here, we have proven that when bit rate increases, the best motion quality layer never decreases, i.e., the monotonically nondecreasing property. By applying this property, many testing scenarios can be omitted without sacrificing the extractor performance. In Table I, for example, the MQ layer $a = 0$ need not be tested for decoding bit rates greater than 384 kbps, once we know the best MQ layer at 384 kbps is $a = 1$.

Moreover, the monotonically nondecreasing property also provides an even simpler way to describe the extractor information table than the one shown in Table II. A series of critical rates, $\{R^{a,*} | D^a(R^{a,*}) = D^{a+1}(R^{a,*}), a = 0, \dots, A - 2\}$, can be found and recorded instead. An example is shown in Table III. Note that the monotonically nondecreasing property limits the total number of critical rates to $A - 1$, where A denotes the total number of motion quality layers.

The critical rate, $R^{a,*}$, is defined as the total decoding bit rate at which both motion quality layers a and $a+1$ produce the same total distortion. Theoretically, $R^{a,*}$ is unique for each motion quality layer a , and can be gradually approached by closing the possible range defined by a lower bound $R^{a,l}$ and an upper bound $R^{a,u}$. The lower bound $R^{a,l}$ is the largest tested bit rate at which the optimal motion quality layer is a . Similarly, the upper bound $R^{a,u}$ is the smallest tested bit rate at which the optimal motion quality layer is $a + 1$.

Given $R^{a,l}$ and $R^{a,u}$, where $R^{a,l} < R^{a,*} < R^{a,u}$, the determination of $R^{a,*}$ can be approached using the bisection method. For each iteration, the middle bit rate, $R^{a,m} = (R^{a,l} + R^{a,u})/2$, is tested and the optimal motion quality layer is found among $\{a, a + 1\}$. The lower bound $R^{a,l}$ is replaced by $R^{a,m}$ if the optimal MQ layer is a , and the other way around. In theory, the process ends whenever $R^{a,*}$ is found. In practice, the iterative algorithm can be terminated whenever $|D^a(\hat{R}^{a,*}) - D^{a+1}(\hat{R}^{a,*})| < \epsilon$, where ϵ is a stopping threshold and $\hat{R}^{a,*}$ is an approximation to $R^{a,*}$. Finally, $\{\hat{R}^{a,*} | a = 0, \dots, A - 2\}$ is stored and transmitted as the optimal extractor information.

2) *Unimodal Property*: The property states that at a fixed decoding bit rate, the decoding PSNR as a function of the motion quality layer is unimodal, i.e., the decoding PSNR is monotonically decreasing on both sides of the optimal motion quality layer. This property is especially useful at finding the maximal decoding PSNR (or minimal decoding distortion). Once a decrease in decoding PSNR is identified, further decreasing with the following MQ layers can be expected, and, thus, the actual decoding processes can be saved.

The unimodal property can be proved as follows. First, from (10) and (11), we know that the first derivatives of both motion and texture distortion functions are monotonically increasing functions (of motion and texture bit rates)

$$D'_t(R_t) = -\frac{\sigma_t^2}{a_t} \exp\left(-\frac{R_t}{a_t}\right) \quad (18)$$

$$D'_m(R_m) = -\Psi \frac{\sigma_m^2}{a_m} \exp\left(-\frac{R_m}{a_m}\right). \quad (19)$$

We focus on one side of the total distortion function (of MQ layers) in the direction of increasing motion quality layers. The other side (decreasing MQ layers) can be proved in a similar manner. Again, suppose at a certain decoding bit rate R_0 , the minimal distortion is achieved with motion quality layer i

$$D_m(R_m^i) + D_t(R_t^i) \leq D_m(R_m^j) + D_t(R_t^j), \quad \forall j \neq i. \quad (20)$$

According to the mean value theorem, there exist $R_m^{ij}, R_m^i < R_m^{ij} < R_m^j$ and $R_t^{ij}, R_t^j < R_t^{ij} < R_t^i$ such that

$$\begin{aligned} D_m(R_m^i) - D_m(R_m^j) &= (R_m^i - R_m^j) D'_m(R_m^{ij}) \\ &= -\Delta R^{ij} D'_m(R_m^{ij}) \end{aligned} \quad (21)$$

and

$$\begin{aligned} D_t(R_t^j) - D_t(R_t^i) &= (R_t^j - R_t^i) D'_t(R_t^{ij}) \\ &= -\Delta R^{ij} D'_t(R_t^{ij}) \end{aligned} \quad (22)$$

where $\Delta R^{ij} \equiv R_m^j - R_m^i = R_t^i - R_t^j > 0$. By taking the difference of (21) and (22) and plugging back into (20), we have the following relationship:

$$D'_m(R_m^{ij}) \geq D'_t(R_t^{ij}). \quad (23)$$

Similarly, for another MQ layer $k > j$

$$\begin{aligned} D_m(R_m^j) - D_m(R_m^k) &= -\Delta R^{jk} D'_m(R_m^{jk}) \\ &< -\Delta R^{jk} D'_m(R_m^{ij}) \\ &\leq -\Delta R^{jk} D'_t(R_t^{ij}) \\ &< -\Delta R^{jk} D'_t(R_t^{kj}) \\ &= D_t(R_t^k) - D_t(R_t^j). \end{aligned} \quad (24)$$

Note that the first and the last inequalities in (24) result directly from (18) and (19), along with the fact that $R_m^{ij} < R_m^{jk}$ and $R_t^{kj} < R_t^{ij}$. The second inequality comes from (23).

The following relationship can now be concluded:

$$D^j(R_0) < D^k(R_0), \quad \forall \{j, k | i < j < k\}. \quad (25)$$

if $D^i(R_0) \leq D^j(R_0), \forall j \neq i$.

In other words, the decoding distortion is monotonically increasing (decreasing) on the increasing (decreasing) side of the optimal motion quality layer. This proves the unimodal property.

C. Model-Based Method

The bisection method for approaching the critical rates is based on the monotonically nondecreasing property. Despite of being more efficient and more accurate than the brute force method, it does not explicitly make full appreciation of the distortion-rate model. In fact, with the help of the distortion-rate model as shown in (10), during each iteration, a better prediction for the critical rate is possible (compared to the middle rate prediction).

Experimental results have shown that, for simplicity, the total distortion function in (12) can be well approximated by removing the motion contributions (both motion distortion and motion rate), i.e.,

$$D(R) \cong D_t(R) = \sigma_t^2 \exp\left(-\frac{R}{a_t}\right). \quad (26)$$

Since the distortion-rate plot is usually depicted in a logarithmic scale using PSNR representations, we have

$$\begin{aligned} PSNR(R) &= 10 \log_{10} \left(\frac{255^2}{D(R)} \right) \\ &\cong \left(\frac{10}{a_t \ln 10} \right) R + 10 \log_{10} \left(\frac{255^2}{\sigma_t^2} \right) \\ &\equiv \alpha R + \beta. \end{aligned} \quad (27)$$

Note that (α, β) can be estimated to reflect the individual characteristic of the video content from at least two tested points on the PSNR-rate curve. For example, the segment within $[R^{a,l}, R^{a,u}]$ of the PSNR-rate plot using MQ layer a , i.e., $PSNR^a(R)$, can be approximated by the line passing through $(R^{a,l}, PSNR^a(R^{a,l}))$ and $(R^{a,u}, PSNR^a(R^{a,u}))$. Similarly, $PSNR^{a+1}(R)$ can be approximated by the line passing through $(R^{a,l}, PSNR^{a+1}(R^{a,l}))$ and $(R^{a,u}, PSNR^{a+1}(R^{a,u}))$. In order to fast approach the critical rate $R^{a,*}$, a better estimate would be the bit rate at which two approximated lines intersect.

Because the actual PSNR-rate curve is approximated using a line with slope α and offset β , this approach is also called the linear model method. In the linear model method, each iteration for determining an estimate of $R^{a,*}$ requires at least four operating points. From one iteration to the next, two of these operating points should be updated with $(\hat{R}^{a,*}, PSNR^a(\hat{R}^{a,*}))$ and $(\hat{R}^{a,*}, PSNR^{a+1}(\hat{R}^{a,*}))$, where $\hat{R}^{a,*}$ is the linear model estimate of $R^{a,*}$.

V. EXPERIMENTAL RESULTS

The evaluation of the proposed bitstream extractors for motion scalability will be performed on the WSVC framework mentioned earlier in Section II. Test video sequences include BUS, FOOTBALL, FOREMAN, and MOBILE. The format of these input sequences is CIF at 30 fps. WSVC will generate a scalable bitstream with maximum bit rate supports, as shown in Table IV, for various spatio-temporal resolutions. The total number of MQ layers is limited to $A = 3$. The nice feature of scalability usually comes with a slight sacrifice of coding efficiency. This is also true for motion scalability. As a consequence, the number of MQ layers needs to be carefully designed in order to maintain a balance between scalability and efficiency. For more detailed information, please refer to [10].

TABLE IV
MAXIMUM BIT RATE ALLOCATION FOR THE WSVC ENCODER (kbps)

	30 fps	15 fps	7.5 fps
CIF	1536	768	-
QCIF	512	256	128

For each decoding spatio-temporal resolution, two experiments will be performed.

A. Discrete Testing Rates

In the first experiment, a discrete set of (equally spaced) bit rates is tested and the effective range of each MQ layer will be determined, as shown in Table II. The possible range of the decoding bit rate is uniformly divided into 2^N segments, which are indexed from 1 to 2^N . We compare the brute force (BF) method with the model assisted (MA) method that uses two searching methods, i.e., progressive search (MAPR) and bisection search (MABI). It is worth noting that MA tries to save irrelevant decoding operations based on the derived two properties. The omitted decoding scenarios depend highly on the actual testing order. As a consequence, different searching algorithms might result in different complexities, as well as different performances. The searching order of MAPR is from the lowest bit rate to the highest one. On the other hand, the order of MABI starts from the middle bit rate and recursively bisects the lower and upper halves.

The results are shown in Table V for $N = 3$. Note that this experiment is designed to evaluate how the BF method is compared with other methods, since the BF method is not applicable in the following experiment using the critical rate representation.

In Table V, the columns labeled with “ a_i ” denote the index (from 1 to 2^N) of the rate segment in which the optimal motion quality layer switches from $i - 1$ to i . Those entries marked with “-” indicate that the transition never happens. For example, decoding a QCIF sequence can never utilize the highest MQ layer, i.e., $a = 2$. Therefore, the a_2 columns are always “-” for QCIF decoding. The columns labeled with “#” denote the number of decoding times required to complete the extractor information table, as shown in Table II. This is a measure of the extractor complexity where a lower value indicates better performance. Note that the number of decoding times for the BF method is always $2^N A$.

As observed from Table V (columns a_i), both MAPR and MABI provide exactly the same results as BF, which is guaranteed the best one in the discrete testing rate experiment. At the same time, both MAPR and MABI save a tremendous amount of computations over BF (from columns #). This result verifies the effectiveness of the models built in Section III, from which the monotonically nonincreasing property and the unimodal property are derived. Moreover, the advantage of MABI over MAPR on reducing the complexity is also verified throughout various testing sequences and decoding scenarios.

B. Critical Rates

In the second experiment, a search is conducted for the critical rates, i.e., $\{R^{a,*} | a = 0, \dots, A - 2\}$. For practical reasons, the

TABLE V
EXTRACTOR COMPARISON WITH DISCRETE TESTING RATES

		BUS			FOOTBALL			FOREMAN			MOBILE		
		a_1	a_2	#	a_1	a_2	#	a_1	a_2	#	a_1	a_2	#
CIF	BF	3	6	24	3	7	24	2	5	24	2	7	24
30 fps	MAPR	3	6	12	3	7	13	2	5	11	2	7	15
	MABI	3	6	10	3	7	10	2	5	11	2	7	11
CIF	BF	3	7	24	4	-	24	2	6	24	2	-	24
15 fps	MAPR	3	7	14	4	-	15	2	6	12	2	-	16
	MABI	3	7	11	4	-	10	2	6	10	2	-	10
QCIF	BF	4	-	16	7	-	16	3	-	16	3	-	16
30 fps	MAPR	4	-	6	7	-	10	3	-	5	3	-	5
	MABI	4	-	5	7	-	5	3	-	6	3	-	6
QCIF	BF	6	-	16	-	-	16	4	-	16	4	-	16
15 fps	MAPR	6	-	9	-	-	11	4	-	6	4	-	6
	MABI	6	-	6	-	-	5	4	-	5	4	-	5
QCIF	BF	-	-	16	-	-	16	8	-	16	6	-	16
7.5 fps	MAPR	-	-	12	-	-	9	8	-	13	6	-	10
	MABI	-	-	5	-	-	3	8	-	6	6	-	6

TABLE VI
EXTRACTOR COMPARISON WITH CRITICAL RATES

		BUS			FOOTBALL			FOREMAN			MOBILE		
		a_1	a_2	#	a_1	a_2	#	a_1	a_2	#	a_1	a_2	#
CIF	MABI	216	816	26	336	880	22	172	600	29	148	832	27
30 fps	MBLM	215	816	15	337	882	20	172	599	23	148	831	23
CIF	MABI	138	480	27	224	-	24	112	384	12	96	-	22
15 fps	MBLM	139	481	25	224	-	24	118	384	9	93	-	15
QCIF	MABI	240	-	9	400	-	9	180	-	14	176	-	10
30 fps	MBLM	244	-	13	397	-	7	180	-	14	166	-	12
QCIF	MABI	162	-	14	-	-	15	124	-	11	120	-	9
15 fps	MBLM	163	-	8	-	-	15	124	-	7	117	-	9
QCIF	MABI	-	-	9	-	-	11	118	-	12	84	-	10
7.5 fps	MBLM	-	-	9	-	-	11	119	-	10	84	-	14

search stops whenever $|PSNR^a(\hat{R}^{a,*}) - PSNR^{a+1}(\hat{R}^{a,*})| \leq 0.01$ dB. The approximate critical rates $\{\hat{R}^{a,*}\}$ are recorded in the extractor information table. We compare the model-assisted method using bisection search (MABI) with the model-based method using the linear model (MBLM). The results are shown in Table VI. Note that the columns marked with a_i now denote the approximated critical rates (in kbps) at which MQ layers $i - 1$ and i produce the same PSNR.

It is worth noting that the critical rates representation provides a more accurate decision boundary than the discrete testing rates representation does. Therefore, a decoder that is based on the critical rates bitstream extractor always performs equally or better than that based on the discrete testing rates, especially for those decoding bit rates that fall within the transition segments, e.g., a_1 and a_2 columns in Table VI. That being said, the discrete testing rates representation should only be adopted when the critical rates representation is not applicable to the derivation algorithms, e.g., BF and MAPR. For MABI, both representations are applicable and produce virtually the same results as N approaches infinity. As for MBLM, since

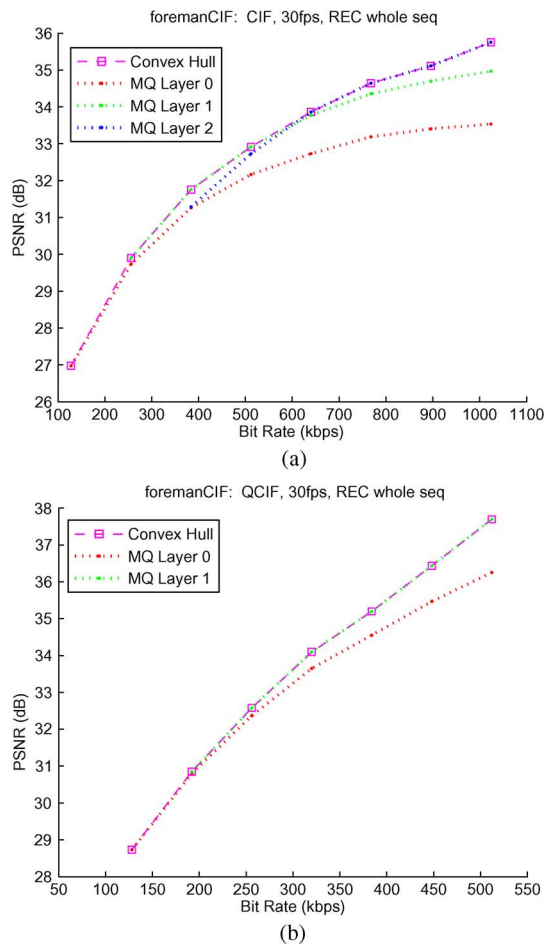


Fig. 5. FOREMAN reconstructed PSNR-rate plots. (a) CIF 30 fps. (b) QCIF 30 fps.

the testing rates are generated on the fly from each iteration to the next, the discrete testing rates representation (with a predetermined set of equally spaced testing rates) is in general not applicable. The nature of adaptively generated testing rates in MBLM provides the potential to outperform MABI, as long as the underlying linearity model is applicable.

Some observations can be drawn from Table VI. MBLM demonstrates better or equal performances than MABI in about 85% of the cases. Note that here we refer the performance to the computational complexity required for deriving the complete set of critical rates, which leads to the optimal decoding for all possible scenarios. Therefore, smaller complexity implies better performance. On the other hand, the linear model assumption seems to work better for the QCIF than the CIF sequences. This observation can be justified by the reconstructed PSNR-rate plots for the FOREMAN sequence, as shown in Fig. 5. In the QCIF plot, the overall curve (convex hull curve) is more linear than that from the CIF plot. Since MBLM is derived based on the linearity assumption of the PSNR-rate curve, it is reasonable to expect better performances for the QCIF than the CIF sequences.

Although MBLM provides better or equal performance than MABI in 85% of the testing scenarios, it is not always better. The reason is that the performance of MBLM depends highly

on the linearity assumption of the PSNR-rate curve. As observed from Fig. 5, the actual curve is not perfectly linear and the deviation from the perfect linearity also varies from sequences to sequences, which results in the variation of the MBLM performances. As a final note, while the overall curve might not be perfectly linear, it can still be accurately approximated as linear locally within a smaller rate range. In the experiments, we do observe a better performance for MBLM in latter iterations, which operates on smaller rate ranges.

VI. CONCLUSION

With the rapid development of SVC and motion scalability, a bitstream extractor aiming at determining the optimal motion quality layer in the rate-distortion sense is essential. In this paper, several algorithms have been proposed to solve this problem, with the designing principle to reduce the complexity. In particular, the linear model based approach using the critical rate representation achieves the lowest complexity, without sacrificing the optimality. Experimental results have verified the effectiveness of the proposed methods, which are mainly based on some mathematical models on the rate-distortion characteristics of a compressed video bitstream.

REFERENCES

- [1] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
- [2] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [3] M. Wien, H. Schwarz, and T. Oelbaum, "Performance analysis of SVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1194–1203, Sep. 2007.
- [4] T. Zgaljic, N. Sprljan, and E. Izquierdo, "Bitstream syntax description based adaptation of scalable video," in *Proc. European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology*, Nov. 2005, pp. 173–178.
- [5] I. Amonou, N. Cammas, S. Kervadec, and S. Pateux, "Optimized rate-distortion extraction with quality layers in the scalable extension of H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1186–1193, Sep. 2007.
- [6] W.-H. Peng, L.-S. Huang, J. K. Zao, J.-S. Lu, T.-W. Wang, H.-T. Huang, and L.-C. Kuo, "Rate-distortion optimized SVC bitstream extraction for heterogeneous devices: A preliminary investigation," in *Proc. IEEE Int. Symp. Multimedia Workshops*, Dec. 2007, pp. 407–412.
- [7] T. Kim and M. Ammar, "Optimal quality adaptation for scalable encoded video," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 344–356, 2005.
- [8] Y. Kim, Y. Jung, T. Thang, and Y. Ro, "Bit-stream extraction to maximize perceptual quality using quality information table in SVC," *Proc. SPIE*, vol. 6077, no. 1, p. 607723, 2006 [Online]. Available: <http://link.aip.org/link/?PSI/6077/607723/1>
- [9] Y. Wang, S. Chang, and A. Loui, "Subjective preference of spatio-temporal rate in video adaptation using multi-dimensional scalable coding," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2004, vol. 3, pp. 1719–1722.
- [10] M.-P. Kao and T. Nguyen, "A fully scalable motion model for scalable video coding," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 908–923, Jun. 2008.
- [11] V. Bottreau, M. Benetiere, B. Felts, and B. Pesquet-Popescu, "A fully scalable 3D subband video codec," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001, vol. 2, pp. 1017–1020.
- [12] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1029–1041, Aug. 2004.
- [13] M. Mrak, N. Sprljan, and E. Izquierdo, "Evaluation of techniques for modeling of layered motion structure," in *Proc. IEEE Int. Conf. Image Process.*, 2006, pp. 1905–1908.

- [14] J. Barbarien, A. Munteanu, F. Verdicchio, Y. Andreopoulos, J. Cornelis, and P. Schelkens, "Motion and texture rate-allocation for prediction-based scalable motion-vector coding," *EURASIP Signal Process.: Image Commun.*, vol. 20, pp. 315–342, Apr. 2005.
- [15] H. Hang and J. Chen, "Source model for transform video coder and its application. I. Fundamental theory," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 287–298, Feb. 1997.
- [16] Y. Yu and C. Tsai, "A model-based rate allocation mechanism for wavelet-based embedded image and video coding," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2005, vol. 6, pp. 6066–6069.
- [17] M. Mrak, G. Abhayaratne, and E. Izquierdo, "On the influence of motion vector precision limiting in scalable video coding," in *Proc. IEEE Int. Conf. Image Processing*, 2004, vol. 2, pp. 1143–1146.
- [18] D. Maestroni, A. Sarti, M. Tagliasacchi, and S. Tubaro, "Scalable coding of variable size blocks motion vectors," in *Proc. IEEE Int. Conf. Image Processing*, 2004, vol. 2, pp. 1333–1336.
- [19] R. Xiong, J. Xu, F. Wu, S. Li, and Y. Zhang, "Layered motion estimation and coding for fully scalable 3D wavelet video coding," in *Proc. IEEE Int. Conf. Image Processing*, 2004, vol. 4, pp. 2271–2274.
- [20] J. Xu, R. Xiong, B. Feng, G. Sullican, M. C. Lee, F. Wu, and S. Li, "3D subband video coding using barbell lifting," in *Proc. ISO/IEC JTCl/SC29/WG11, 68th MPEG Meet.*, Munich, Germany, Mar. 2004, M10569/S05.
- [21] M.-P. Kao and T. Nguyen, "Motion vector field manipulation for complexity reduction in scalable video coding," in *Proc. IEEE Asilomar Conf. Signals Syst. Comput.*, 2006, pp. 1095–1098.
- [22] K. R. Rao and P. C. Yip, *The Transform and Data Compression Handbook*. Boca Raton, FL: CRC, 2001.
- [23] H. Gish and J. Pierce, "Asymptotically efficient quantizing," *IEEE Trans. Inf. Theory*, vol. IT-14, no. 5, pp. 676–683, 1968.



Meng-Ping Kao (S'06) was born in Taipei, Taiwan, R.O.C., in 1978. He received the B.S. and M.S. degrees in electrical engineering from National Taiwan University, Taipei, in 2000 and 2004, respectively, and the Ph.D. degree in electrical and computer engineering from the University of California at San Diego, La Jolla, in 2008.

He is currently a senior video system engineer with Qualcomm Inc., San Diego. His research interests are in the field of scalable video compression and processing.



Truong Q. Nguyen (F'06) is currently a Professor at the ECE Department, University of California at San Diego, La Jolla. He is the coauthor (with Prof. G. Strang) of the popular textbook *Wavelets and Filter Banks* (Wellesley-Cambridge Press, 1997) and the author of several Matlab-based toolboxes on image compression, electrocardiogram compression, and filter bank design. He has over 200 publications. His research interests are video processing algorithms and their efficient implementation.

Prof. Nguyen received the IEEE TRANSACTIONS ON SIGNAL PROCESSING Paper Award (Image and Multidimensional Processing area) for the paper he co-wrote with Prof. P. P. Vaidyanathan on linear-phase perfect-reconstruction filter banks (1992). He received the NSF Career Award in 1995 and is currently the Series Editor (Digital Signal Processing) for Academic Press. He served as Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1994–1996), the IEEE SIGNAL PROCESSING LETTERS (2001–2003), the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS (1996–1997, 2001–2004), and the IEEE TRANSACTIONS ON IMAGE PROCESSING (2004–2005).