RATE-DISTORTION OPTIMIZED BITSTREAM EXTRACTOR FOR MOTION SCALABILITY IN SCALABLE VIDEO CODING

Meng-Ping Kao and Truong Nguyen

University of California, San Diego, Department of ECE
http://videoprocessing.ucsd.edu, mkao@ucsd.edu, nguyent@ece.ucsd.edu

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 or 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 each 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 has high computational complexity. Two properties, i.e. the monotonically non-decreasing 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 by a factor up to 5.

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

1. INTRODUCTION

A typical scalable video coding (SVC) infrastructure, 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 SVC is allowed to demand a variety of decoding specifications, including different combinations of spatial, temporal, and quality layers. It is the main task of an SVC bitstream extractor to fulfill those requests by properly truncating the scalable bitstream.

The designing criteria for a generic SVC bitstream extractor can be rather trivial. In the SVC standard [1], for example, each network abstraction layer unit (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 $D > D_t$. If, however, 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 [2]. The overall idea is to retain those NALUs with higher RD contribution and therefore to optimize the quality under the rate constraint.

When motion scalability [3] is taken into consideration, the bitstream extractor has an additional requirement, i.e. optimal bit allocation among motion and texture [4]. In this paper, we focus on the case where the decoding spatio-temporal resolution (T and D in the SVC standard) is pre-specified 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 MQ layer also changes accordingly. The optimal MQ 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.

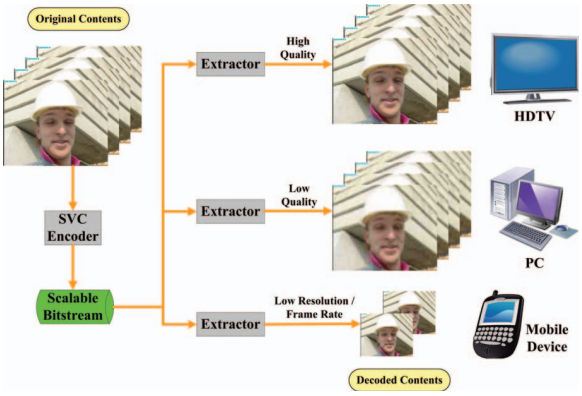

Fig. 1. Scalable video coding.

The remainder of the paper is organized as follows. In Section 2, a model-based theoretical justification of motion scalability is presented. It will be clear how and when motion scalability can benefit the coding efficiency. In Section 3, we propose three approaches for optimal bitstream extraction, i.e. the brute force, model-assisted, and model-based methods. The properties that facilitate more efficient extractor designs are also derived here. Finally, experimental results are provided in Section 4 to verify the effectiveness of our new designs .

2. 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 video quality. In this section, we briefly review some models that have been developed to describe the behavior of motion scalability. Combined with the exponential distortion-rate model from the source coding theory, we are able to derive a beneficial condition for motion scalability.

2.1. Linear Motion Distortion Model

The first work analyzing the distortion introduced by MV quantization is done by Secker [5]. Under a series of assumptions and approximations, he proposed a linear motion distortion model, as shown in (1), which describes the linear relationship (with slope Ψ) between the squared MV error, $||\boldsymbol{\delta}||^2$, and the corresponding mean squared MC error, or simply known as the motion distortion, D_m.

$$D_m \approx \Psi ||\boldsymbol{\delta}||^2 \tag{1}$$

Note that Ψ is the isotropic motion sensitivity factor, averaged over all MV errors with magnitude $||\boldsymbol{\delta}||$, of the reference picture. It is a function of the energy spectral density of the corresponding reference picture, which is highly content dependent.

2.2. Additive Distortion Model

In a generic video coding framework, the MC operation is followed by the texture/residual transform coding and quantization. In the case where motion coding is non-scalable, the total distortion of the reconstructed picture, D, can be simply described by the texture distortion, D_t, which is introduced by the texture quantization operation.

However, if motion scalability is taken into consideration, the motion distortion may also contribute to the total distortion. The additive distortion model [5] states that the total distortion is the summation of the motion distortion and the texture distortion. The assumption behind 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.

$$\begin{aligned} D_m + D_t &= \frac{1}{N_1 N_2} \left(||m[\mathbf{n}] - m^*[\mathbf{n}]||^2 + ||t[\mathbf{n}] - t^*[\mathbf{n}]||^2 \right) \\ &= \frac{1}{N_1 N_2} ||m[\mathbf{n}] + t[\mathbf{n}] - c[\mathbf{n}]||^2 = D, \end{aligned} \tag{2}$$

where $c[\mathbf{n}] = m^*[\mathbf{n}] + t^*[\mathbf{n}]$ is the original picture without distortion.

2.3. 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 [6].

$$D_t(R_t) = \sigma_t^2 \exp\left(-\frac{R_t}{a_t}\right), \tag{3}$$

where R_t is the texture bit rate and σ_t and a_t are content dependent parameters. 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 to MV coding, making the squared MV error, $||\boldsymbol{\delta}||^2$, an exponential function of the motion bit rate, R_m. Therefore, (1) can be expressed as follows.

$$D_m(R_m) = \Psi \sigma_m^2 \exp\left(-\frac{R_m}{a_m}\right) \tag{4}$$

Applying the additive distortion model in (2), the total distortion-rate model becomes

$$D(R) = D_m(R_m) + D_t(R_t) = D_m(R_m) + D_t(R - R_m), \tag{5}$$

where $R = R_m + R_t$ is the total 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. $D^a(R) = D_m(R_m^a) + D_t(R - R_m^a)$, where $R_m^a < R_m^*$. The condition for scalable motion to outperform lossless motion is simply $D^a(R) < D^*(R)$, or

$$D_m(R_m^a) < D_t(R - R_m^*) - D_t(R - R_m^a). \tag{6}$$

As can be derived from (3), the right hand side of (6) is a monotonically decreasing function of R. As a consequence, the satisfaction of (6) can be expected for a relatively smaller R, given a fixed R_m^a.

Table 1. Extractor RD table (FOREMAN @ CIF 30 *fps*)

	Decoding Bit Rate (*kbps*)							
MQ Layer	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**

Table 2. Effective rate information (FOREMAN @ CIF 30 *fps*)

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

3. OPTIMAL BITSTREAM EXTRACTOR DESIGN FOR MOTION SCALABILITY

3.1. 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 MQ layer is applied. The same process is repeated for all decoding bit rates of interest, resulting in an extractor RD table, as shown in Table 1. 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 MQ layer at its decoding bit rate. A simplified table, as shown in Table 2, records the effective bit rate range for each MQ 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 2, the range boundary is chosen as the average of two contiguous testing rates. The more testing bit rates, the better extractor performance, and, of course, the more computational burdens.

3.2. Model-Assisted Method

Two properties can be observed in Table 1. First, the optimal MQ layer is a monotonically non-decreasing function of the decoding bit rate. Second, the decoding PSNR at a fixed bit rate is an unimodal function of the MQ layer. With the help of those models built in Section 2, we will now prove that these two properties follows directly. The advantage of knowing these properties is to efficiently save some trials that are irrelevant to the final extractor table, as shown in Table 2.

Table 3. Critical rate information (FOREMAN @ CIF 30 *fps*)

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

3.2.1. Monotonically Non-Decreasing Property

Suppose at a certain decoding bit rate R_0, the minimal distortion is achieved with MQ layer i, which occupies a motion bit rate R_m^i.

$$D^i(R_0) \leq D^j(R_0), \forall j \neq i \tag{7}$$

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

$$\begin{aligned} & D^i(R_0+\triangle R) - D^j(R_0+\triangle R) \\ =& \Psi\sigma_m^2\left(\exp\left(-\frac{R_m^i}{a_m}\right) - \exp\left(-\frac{R_m^j}{a_m}\right)\right) \\ & + \sigma_t^2 \exp\left(-\frac{R_0+\triangle R}{a_t}\right)\left(\exp\left(\frac{R_m^i}{a_t}\right) - \exp\left(\frac{R_m^j}{a_t}\right)\right) \\ \leq& \sigma_t^2 \exp\left(-\frac{R_0}{a_t}\right)\left(\exp\left(\frac{R_m^i}{a_t}\right) - \exp\left(\frac{R_m^j}{a_t}\right)\right) \\ & \left(\exp\left(-\frac{\triangle R}{a_t}\right) - 1\right) \end{aligned} \tag{8}$$

The inequality in (8) comes directly from (7),(3), (4) and (5). Since both $a_t > 0$ and $\triangle R > 0$, we have $\exp\left(-\triangle R/a_t\right) - 1 < 0$. In addition, for those MQ layers $j < i$, the corresponding motion bit rates are smaller, i.e. $R_m^j < R_m^i$. Therefore, we have $\left(\exp\left(R_m^i/a_t\right) - \exp\left(R_m^j/a_t\right)\right) > 0$. In summary, the right hand side of (8) is negative whenever $j < i$. In other words, if $D^i(R_0) \leq D^j(R_0), \forall j \neq i$,

$$D^i(R_0+\triangle R) < D^j(R_0+\triangle R), \forall j < i. \tag{9}$$

Here we have proven that when the bit rate increases, the best MQ layer can never decrease, i.e. the monotonically non-decreasing property. By applying this property, many testing scenarios can be omitted without sacrificing the extractor performance. In Table 1, for example, the MQ layer $a = 0$ does not need to 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 non-decreasing property enables a simpler alternative to describe Table 2. A series of critical rates, $\{R^{a,*} | D^a(R^{a,*}) = D^{a+1}(R^{a,*}), a = 0, \cdots, A-2\}$, can now be found and recorded. An example is shown in Table 3. Note that the monotonically non-decreasing property limits the total number of critical rates to $A-1$, where A denotes the total number of MQ layers.

3.2.2. Unimodal Property

This property states that at a fixed bit rate, the decoding PSNR as a function of the MQ layer is unimodal, i.e. the decoding PSNR is monotonically decreasing on both sides of the optimal MQ 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 skipped.

The unimodal property can be proved as follows. We focus only on one side of the total distortion function (of MQ layers) in the direction of increasing MQ layers. The other side (decreasing MQ layers) can be proved in a similar manner. First, from (3) and (4), we know that the first derivatives of both motion and texture distortion functions, i.e. $D_t'(R_t)$ and $D_m'(R_m)$, are monotonically increasing. Again, suppose at a certain decoding bit rate R_0, the minimal distortion is achieved with MQ layer i.

$$D_m(R_m^i) + D_t(R_t^i) \leq D_m(R_m^j) + D_t(R_t^j), \forall j \neq i \tag{10}$$

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

$$D_m(R_m^i) - D_m(R_m^j) = -\triangle R^{ij} D_m'(R_m^{ij}) \tag{11}$$

$$D_t(R_t^j) - D_t(R_t^i) = -\triangle R^{ij} D_t'(R_t^{ji}) \tag{12}$$

where $\triangle R^{ij} = R_m^j - R_m^i = R_t^i - R_t^j > 0$. By taking the difference of (11) and (12) and plugging back into (10), we have the following relationship:

$$D_m'(R_m^{ij}) \geq D_t'(R_t^{ji}) \tag{13}$$

Similarly, for another MQ layer $k > j$,

$$\begin{aligned} & D_m(R_m^j) - D_m(R_m^k) = -\triangle R^{jk} D_m'(R_m^{jk}) < -\triangle R^{jk} D_m'(R_m^{ij}) \\ & \leq -\triangle R^{jk} D_t'(R_t^{ji}) < -\triangle R^{jk} D_t'(R_t^{kj}) = D_t(R_t^k) - D_t(R_t^j) \end{aligned} \tag{14}$$

Note that the first and the last inequalities in (14) result directly from the monotonically increasing characteristic of $D_t'(R_t)$ and $D_m'(R_m)$, along with the fact that $R_m^{ij} < R_m^{jk}$ and $R_t^{kj} < R_t^{ji}$. The second inequality comes from (13). The following relationship can now be concluded. If $D^i(R_0) \leq D^j(R_0), \forall j \neq i$,

$$D^j(R_0) < D^k(R_0), \forall \{j,k | i < j < k\}. \tag{15}$$

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

3.3. Model-Based Method

The bisection method for approaching the critical rates is based on the monotonically non-decreasing 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 models. Experimental results have shown that, for simplicity, the total distortion function in (5) can be well approximated by removing the motion contributions (both motion distortion and motion rate), i.e.

$$D(R) \cong D_t(R) = \sigma_t^2 \exp\left(-\frac{R}{a_t}\right). \tag{16}$$

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

$$PSNR(R) = 10\log_{10}\left(\frac{255^2}{D(R)}\right) \cong \alpha R + \beta. \tag{17}$$

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. Because the actual PSNR-rate curve is approximated using a line with slope α and offset β, this approach is called

Table 4. 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

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 $R^{a,*}$.

4. EXPERIMENTAL RESULTS

The evaluation of the proposed bitstream extractors for motion scalability will be performed on the wavelet-based SVC framework [3]. Test video sequences include BUS, FOOTBALL, FOREMAN, and MOBILE. The format of these input sequences is CIF at 30 *fps*. The total number of MQ layers is limited to $A = 3$. For each decoding spatio-temporal resolution, two experiments will be performed.

In the first experiment, a discrete set of (equally spaced and indexed from 1 to 2^N) bit rates is tested and the effective range of each MQ layer will be determined. 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). 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 4 for $N = 3$. The columns labeled with "a_i" denote the index (from 1 to 2^N) of the rate segment in which the optimal MQ layer switches from $i-1$ to i. The columns labeled with "#" denote the number of decoding times required to complete the extractor information table, as shown in Table 2. Note that the number of decoding times for the BF method is always $2^N A$.

As observed from Table 4 (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 2, from which the monotonically non-increasing 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.

In the second experiment, a search is conducted for the critical rates, i.e. $\{R^{a,*}|a = 0, \cdots, A-2\}$. For practical reasons, the 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 5. 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. As observed from Table 5, MBLM demonstrates better or equal performances than MABI in about 85% of the cases.

Table 5. 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

5. 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.

6. 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, Sept. 2007.

[2] 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, Sept. 2007.

[3] 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.

[4] 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 Processing: Image Communication*, vol. 20, pp. 315–342, Apr. 2005.

[5] 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.

[6] H.M. Hang and J.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, Apr. 1997.