RATE-DISTORTION OPTIMIZED DATA PARTITIONING FOR VIDEO USING BACKWARD ADAPTATION

Jong Chul Ye and Yingwei Chen

Philips Research, 345 Scarborough Road, Briarcliff Manor, NY 10510
Email: {jong.ye, yingwei.chen}@philips.com

ABSTRACT

While data partitioning in conjunction with unequal error protection provides superb error resiliency, insufficient video quality when only the base partition is available prevents its wide deployment in high-quality video applications. In this work, we develop a new scheme for data partitioning of motion-compensated DCT coded video in an operational rate-distortion context. Unlike the conventional data partitioning scheme which adapts the DCT break points at slice or video packet level, our new partitioning algorithm adapts the partitioning points at as low as the DCT block level with virtually no overhead using backward adaptation, hence produces superior video quality over the conventional data partitioning scheme. Simulation results show that significant PSNR gain can be achieved using the new algorithm.

I. INTRODUCTION

Data partitioning (DP) is a form of layered coding where a single layer coded bitstream is segmented into two or more bitstreams in a hierarchical fashion by splitting DCT coefficients into base and enhancement layer [1]. When both partitions are received and decoded, transparent quality (compared with single-layer coded video) is achieved at the same total bit rate. When part or all of the enhancement partition is not decoded, partial quality is achieved with graceful degradation. Because of this graceful quality degradation, data partitioning is an effective error resilience tool especially when combined with unequal error protection [2].

Advantages of data partitioning include very low rate overhead at high bitrates and low encoding and decoding complexity overhead. However, one major drawback of DP is the prediction drift at low bitrates as a result of the single-loop prediction structure used. Therefore, a major problem for data partitioning is how to choose the DCT break point for each block such that the base partition quality at a given base partition rate (or ratio) is optimal. Eleftheriadis and Anastassiou [1] presented an algorithm to determine optimal partitioning points for each slice in the operational rate distortion framework. Since the optimal algorithm has high complexity and delay, a causally optimal algorithm based on Lagrangian optimization was proposed to optimally solve the problem for intra (I) pictures, while providing an optimal solution for predicted/interpolated (P/B) pictures when the additional constraints of causal operation and/or low-delay are imposed. A memoryless version of the algorithm, theoretically optimal for intra pictures only, was shown to perform almost identically but with significantly less computational complexity.

While the work in [1] is effective in optimizing the conventional data partitioning scheme, there exists a fundamental performance limit. In order to achieve the minimum distortion for the base layer, the partitioning point should be allowed to vary at the DCT block level. However, such a fine control of the breakpoint introduces significant rate overhead due to the explicit encoding/transmission of breakpoint values. Therefore, in practice, break point signaling is possible only at the slice level or above, in which case all DCT blocks in a slice are partitioned at the same DCT run-level pair location.

This paper introduces a new data partitioning scheme that overcomes fundamental limitations of the conventional DP scheme. In the new scheme, each DCT block can be partitioned differently *without* explicit coding of the break point. The exact location of the breakpoint can be easily deduced at the encoder and decoder synchronously from the preceding DCT run-level pairs using *backward adaptation*. Furthermore, if the rate-distortion curve for each DCT block is convex, the new DP algorithm can be shown to be rate-distortion *near optimal* in the sense that only one excessive run-level pair is included into the base partition compared to the optimal partitioning. Hence, we refer to this data partitioning scheme as rate-distortion optimized data partitioning using backward adaptation (RDDP) . Simulation results confirm the superior performance of RDDP in PSNR over the conventional DP. In addition, visible subjective quality improvement is achieved. Furthermore, unlike the conventional DP, the rate-distortion performance of RDDP is independent of slice size, and the same rate-distortion performance can be achieved even in extreme scenarios where a video frame contains a single slice. Hence, from a joint source channel coding point of view, the optimization problem of RDDP is much simpler than that of DP since packetization scheme does not impact the base layer video quality.

Such a large improvement of RDDP stems from a close look at the conventional DP and the classical Lagrangian based rate-distortion optimization technique. Hence, this paper is organized as follows. Sec. II-A presents the problem statement with a brief overview of the classical Lagrangian approach for the rate-distortion optimization. The conventional DP algorithm is also presented. In Sec. III, we introduce the new RDDP algorithm. Sec. IV provides simulation results.

II. OVERVIEW OF DATA PARTITIONING SCHEME

II-A. Problem Statement

The system diagram of the data partitioning scheme is shown in Figure 1. Between encoding and decoding, the bitstream (coded at rate B Mbps) is split into two partitions, each being transmitted on a different "channel". Throughout the paper, we will assume that channel 0 is perfect without errors with a given fixed available bandwidth $\hat{B} < B$, while channel 1 is assumed to have random packet losses.

Partitioning is performed at well-defined points (break points) in the bitstream syntax. Usually, partitioning will only affect the number of DCT run-level coefficients that will be carried in partition 0 (base layer or partition), while the rest will be assigned

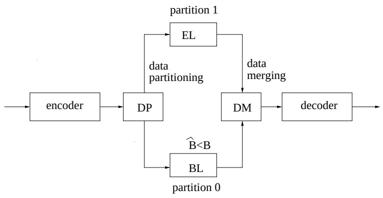


Fig. 1. Overview of Data partitioning.

to partition 1 (enhancement layer or partition). Since channel 1 is assumed to have random losses, we consider the deterministic problem of minimizing the maximum possible error.

The optimization window is not specified, and it can span from just a part of a picture up to a group of pictures (GOP). Due to the error accumulation process, partitioning decisions made for a given picture will have an effect on the quality and partitioning decisions of subsequent pictures. As a result, an optimal window size should cover a complete group of pictures (Intra-to-Intra). Not only the computational overhead would be extremely high, but the delay would be unacceptable as well. In [1], Eleftheriadis and Anastassiou empirically showed that a memoryless version of the optimization problem, theoretically optimal for intra picture only, is shown to perform almost identically but with significantly less computational complexity than a simplified version of the optimal algorithm. Therefore, in this paper, we only focus on the memoryless version of the optimization problem where the selection of the window is a single frame. In this case, the target bit budget R_{budget} of each picture can be set to: $R_{budget} = (\hat{B}/B)R - R_0$, where R is the size (in bits) of the currently processed frame, and R_0 are the number of bits spend for coding components of the bitstream that are not subjective to data partitioning. Allocated bits that are left over from one picture are carried over to the subsequent picture.

Therefore, if X is the input signal for a frame, $\hat{X}$ the quantized output corresponding to a coded video signal, and $\tilde{X}$ the output frame of the decoder assuming that the entire partition 1 is lost, we seek to minimize the MSE distortion between X and $\tilde{X}$ given the coded video signal $\hat{X}$, subject to a total coding-bit budget of R_{budget} in partition 0 for $\tilde{X}$:

$$\min \left[D(X,\tilde{X}) | \hat{X} \right] \quad \text{subject to} \quad R(\tilde{X}) \leq R_{budget} \ , \quad (1)$$

where $R(\tilde{X})$ denotes the rate for the partition 0. The optimization problem (1) is solved for each frame independently.

In the memoryless version of the DP optimization algorithm, the temporal dependency between pictures are not taken into consideration. Consequently, the partitioning error will simply consists of the DCT coefficients that were assigned to partition 1. Using the orthonormality of the DCT and assuming that distortion due to quantization error is negligible, this can be expressed as follows:

$$D(X,\tilde{X}) \;=\; \sum_{i=1}^{N} \sum_{j \in S_i} D_i^j(B_i^j) = \sum_{i=1}^{N} \sum_{j \in S_i} \sum_{k > B_i^j} \left[\hat{C}_i^j(k) \right]^2 \quad (2)$$

where $B_i^j \in \{0, \cdots, 64\}$ is the breakpoint value (not the DCT coefficient location!) for the j-th block of the i-th slice i (run-level codes greater than B_i^j will go to partition 1), and $D_i^j(B_i^j)$ denotes the resultant distortion for the j-th block of the i-th slice i, N is the number of slice considered, S_i are the blocks in slice i (or video packet for MPEG4 cases), $\hat{C}_i^j(k)$ are quantized DCT coefficients of the k-th run in the j-th block of the i-th slice. The rate can be represented as follows:

$$R(\tilde{X}) = \sum_{i=1}^{N} \sum_{j \in S_i} R_i^j(B_i^j) = \sum_{i=1}^{N} \sum_{j \in S_i} \sum_{k \leq B_i^j} L_i^j(k) \ , \quad (3)$$

where $R_i^j(B_i^j)$ denotes the rate of using breakpoint value B_i^j in the j-th block of the i-th slice in partition 0, and $L_i^j(k)$ denotes the code length of the k-th run in the j-th block of the i-th slice.

II-B. Optimal Data Partitioning

The "hard" constrained optimization problem (1) can be solved by being converted to an "easy" equivalent unconstrained problem by "merging" rate and distortion through the Lagrangian multiplier λ [3]. The constrained optimization problem becomes the determination of that set of coefficients, which results in the minimum total Lagrangian cost defined as [3]:

$$J(\lambda) \equiv D(X,\tilde{X}) + \lambda R(\tilde{X}) \ . \quad (4)$$

The desired optimal slope value λ^* is not known a priori and depends on the particular target budget or quality constraint but is obtained using a fast convex search using the bisection algorithm [3].

The main advantage of the Lagrangian approach is its independent optimization property for each signal elements such as slice, macroblock or block depending on its minimal coding unit. More specifically, the theoretical performance limit of the data partitioning can be achieved by minimizing the cost function:

$$\min_{B_i^j} \left\{ D_i^j(B_i^j) + \lambda R_i^j(B_i^j) \right\} \quad , \quad \text{for all } i,j \ . \quad (5)$$

II-C. Conventional Data Partitioning Scheme

In the conventional DP, the optimal break points $\{B_i^j\}$ for the optimization problem (5) are *explicitly* encoded and transmitted. This introduces a significant increase of rate overhead. Therefore, in MPEG2 standard the breakpoint values are set to equal for all the blocks in the slice and transmitted only once for each slice. This is equivalent to impose a constraints:

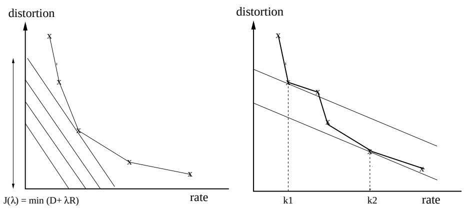
$$B_i \equiv B_i^1 = B_i^2 = \cdots \quad , \quad i = 1, \cdots, N \ . \quad (6)$$

The performance loss is caused by the constraints (6). Indeed, the optimization problem can be no more decoupled up to the block level as in (5), but only up to the slice. Therefore, the optimal rate-distortion performance from (5) is no more feasible.

III. RATE-DISTORTION OPTIMIZED DATA PARTITIONING

Consider a typical convex R-D curve in Fig. 2(a). The minimum Lagrangian function is obviously achieved for that point which is "hit" first by the plane wave of absolute slope λ impinging on the rate-distortion curve. If every admissible operating point lies on the convex hull, then the absolute slope before the optimal operating point is greater than λ, while the absolute slope after the optimal point is less than or equal to λ. This implies that

$$\frac{D_i^j(k) - D_i^j(k-1)}{R_i^j(k-1) - R_i^j(k)} > \lambda \quad , \quad \text{if } k \leq B_i^{j*}$$
$$\frac{D_i^j(k) - D_i^j(k-1)}{R_i^j(k-1) - R_i^j(k)} \leq \lambda \quad , \quad \text{if } k > B_i^{j*} \ , \quad (7)$$



(a) (b)

Fig. 2. (a) Geometric interpretation of the Lagrangian optimization problem. (b)Loss of optimality of RDDP for a non-convex R-D curve.

where $D_i^j(k)$ denotes the distortion of j-th block of the i-th slice if up to the k-th (run,level) pairs are included, and B_i^{j*} denotes the optimal partitioning points. Using Eqs. (2) and (3), Eq. (7) can be written as:

$$\frac{\left[\hat{C}_i^j(k)\right]^2}{L_i^j(k)} > \lambda, \text{ if } k \le B_i^{j*} \quad , \quad \frac{\left[\hat{C}_i^j(k)\right]^2}{L_i^j(k)} \le \lambda, \text{ if } k > B_i^{j*} \tag{8}$$

Eq. (8) shows that all DCT run-level pairs are split into two partitions based on the magnitude of a test variable $\left[\hat{C}_i^j(k)\right]^2 / L_i^j(k)$ with respect to the quality factor λ. Since the values of $\hat{C}_i^j(k)$ and $L_i^j(k)$ are known for both encoder and decoder, the basic idea of the RDDP is that instead of encoding and transmitting B_i^{j*}, *only the quality factor λ is encoded and transmitted to the decoder and then the break points B_i^{j*} are deduced from $\hat{C}_i^j(k)$ and $L_i^j(k)$ during run-time (backward adaptation)*. This significantly reduces the overhead of the data partitioning.

However, due to the look-ahead nature of the partition rule in (8) which requires the decoding of run-level pairs before the decision, it is not always possible to split the bitstream at the optimal partitioning point B_i^{j*} using backward adaptation. However, the following algorithm allows the encoder and decoder to split the bitstream *near optimally* without sending explicit information of the breakpoint value.

Encoding rule:

```
Encode λ into partition 0.
for i = 1, ··· , N {   // for each slice (or video packets)
    for j ∈ S_i {   // for each blocks
        for k = 1, ··· , K(j) {   // for each (run,level) pair
            Compute Ĉ_i^j(k) and L_i^j(k).
            Put the k-th (run,level) VLC into partition 0.
            If [Ĉ_i^j(k)]^2 /L_i^j(k) < λ break.
        }
    Put the remaining (run,level) pairs of i-th block into partition 1.
    }
}
```

Decoding rule:

```
Decode λ from partition 0.
for i = 1, ··· , N {   // for each slice (or video packets)
    for j ∈ S_i {   // for each blocks
        for k = 1, ··· , K(j) {   // for each (run,level) pair
            Read k-th (run,level) VLC from partition 0.
            Compute Ĉ_i^j(k) and L_i^j(k).
            If [Ĉ_i^j(k)]^2 /L_i^j(k) < λ break.
        }
    Read the remaining (run,level) pairs of i-th block from partition 1.
    }
}
```

Note that partition 0 is over-partitioned in the sense that all (run,level) VLC pairs at $B_i^{j*}+1$ are included at the partition 0. However, we observe that this over-partitioning does not incur much performance loss. What affects the performance of RDDP is, however, the non-convexity of R-D planes in some DCT blocks. This is especially true in some motion residual blocks, where R-D planes for the DCT blocks could be non-convex due to strong high frequency components. In these cases, the test variable $\left[\hat{C}_i^j(k)\right]^2 / L_i^j(k)$ is no longer monotonic with respect to k and the partitioning rule given by (8) is not valid, so the near-optimality of the RDDP can be broken. For example, in Fig. 2(b), the optimal breakpoint value should be k_2 while the RDDP algorithm provides k_1, which makes the base layer under-partitioned. However, even in this case, we can easily see that the encoding and decoding algorithms can be perfectly synchronized since they both switch to partition 1 after the first (run,level) pair which satisfies $\left[\hat{C}_i^j(k)\right]^2 / L_i^j(k) < \lambda$.

IV. NUMERICAL RESULTS

All the codecs described in this paper are based on ISO-MPEG4 codec and have been tested on 300 frames of standard sequences including “foreman”, “mobile”, “carphone”, “stefan”, “coastguard”, and “table tennis” in CIF resolution (352×288) at 25 fps. Since MPEG4 does not have DCT data-partitioning tool for scalability, we modified the MPEG4 codec accordingly. Intra-frame encoding is performed once every 15 frames to enable a full refresh. We use two B-frames in between I and/or P frames. The total bit-rate is 2 Mbps, while the ratio between the base and enhancement layer is variable. For each simulation run, we compute the peak signal-to-noise ratio (PSNR) of the base layer only assuming that all the enhancement layer is lost, using formula $PSNR = 10\log\left(\frac{255^2}{MSE}\right)$, where MSE is the mean squared error between the original uncoded image and the decoded image. The resulting PSNR of each simulation run is averaged over the 300 frames to obtain values presented in this paper. Due to the drift effect in motion-compensated video coder, the overall PSNR performance is quite dependent on the base layer partitioning ratio of I,P, and B frames. Therefore, for a fair comparison, the following choices of frame-type dependent base layer partitioning ratios are used for both DP and the RDDP to compute the rate-distortion plots. RDDP and the conventional DP incur small amount of over-

	I frame	P frame	B frame
case 1	0.9	0.7	0.3
case 2	0.9	0.7	0.5
case 3	0.9	0.8	0.7
case 4	0.9	0.9	0.9

head for duplicating start codes and for including the quality factor λ or the breakpoint values in each video packet header. In addition, it may require additional bits to ensure byte alignment in the two layers. In our experiments, data partitioning incurred less than 1% overhead for all splits, and this overhead did not detract from the overall quality of the video. In our simulation, the quality factor λ is uniformly quantized with 10 bits and is duplicated for each video packet header for error resiliency.

Recall that the RDDP algorithm is not sensitive to the video packet structure since breakpoints are determined block-by-block rather than fixed in slice level. Hence, we compute the R-D plots for different video packet sizes, one with 14,500 bytes and the other with 1,450 bytes. The former choice of video packet size includes all the P and B frames within one packets and I frames within two packets, whereas the latter choice of packet size allows multiple packets even within P and B frames. The breakpoint values of DP and RDDP are optimally computed using the same bisection algorithm in [3] for a fair comparison. Figure 3 show R-D plots for several test sequences. For "foreman", "carphone" and "mobile" sequences, we can find that RDDP consistently outperforms than the conventional DP algorithm with different slice sizes. Furthermore, we can observe that the RDDP is not sensitive to the slice size, unlike DP. However, for "coastguard" and "stefan" sequence, where high frequency component are prominent in motion compensated residual images, the convexity assumption of the RD curve does not hold, so small performance loss of the RDDP are observed as shown at the forth plot of Figs. 3. The performance for this case could be improved by separate processing of nonconvex RD blocks and/or frames, and we are now currently investigating methods to extend the idea even for the non-convex R-D plain cases.

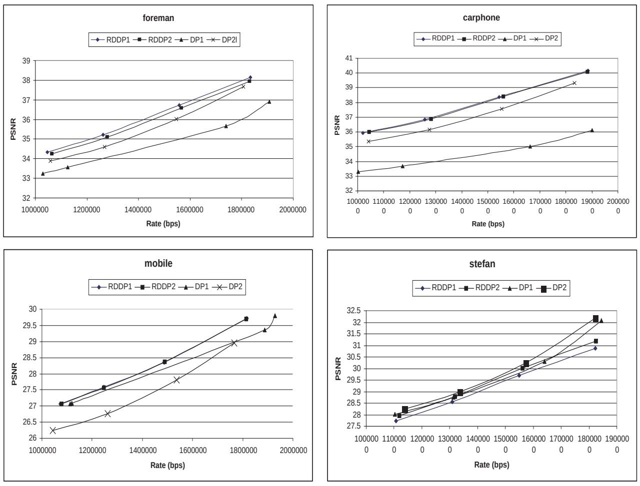


Fig. 3. Rate-distortion plots for "foreman", "carphone", "mobile", and "stefan" sequences for the base layer of of RDDP with packet size of 14,500 bytes (RDDP1) and 1,450 bytes (RDDP2), and DP with packet size of 14,6500 bytes (DP1) and 1,450 bytes (DP2), respectively.

To demonstrate the superior subjective quality of RDDP, Figure 4 illustrate reconstruction of the 133-th frame of the "mobile" sequences by RDDP and the DP using packet size of 1,450 bytes, respectively. Here, the video is encoded with base layer partitioning ratio of I:P:B=0.9:0.7:0.5, which results in approximate 1.25Mbps base layer. Note the areas within black boxes, where the DP reconstruction is much blurred compared to RDDP. Especially, the image within the left box in DP reconstruction is so blurred that the head of a sheep and grasses are not visible, while the reconstruction by RDDP does not have these artifacts. This is because RDDP adapts the partition by block level while the conventional DP does not. This kind of subjective quality improvement can be observed consistently for all sequences. In general, the base layer bit budget for RDDP is distributed uniformly to contribute overall sharpness inhancement contrary to the conventional DP case.

Fig. 4. The 133-th frames of "mobile" sequence decoded from base layer only by RDDP (top) and DP (bottom), respectively.

V. CONCLUSIONS AND FUTURE WORK

In this work, we developed a new scheme for data partitioning of motion-compensated DCT coded video in an operational rate-distortion context. Unlike the conventional data partitioning scheme which adapts the DCT break points at slice or video packet level, our new partitioning algorithm adapts the breakpoints at DCT block level with minimal overhead, hence produces superior video quality over the conventional data partitioning scheme. Such an improvement comes from that at the rate-distortion optimality the slope of the convex hull of R-D curve of each DCT block are equal, hence rather than sending the breakpoints explicitly only Lagrangian parameter is transmitted as a side information. Simulation results show that if the convexity assumption for the RD plain is satisfied, a significant quality improvement in both PSNR and subjective sense can be obtained. We are now currently investigating a method to extend the idea even for the non-convex R-D plain cases.

VI. REFERENCES

[1] A. Eleftheriadis and D. Anastassiou, "Optimal data partitioning of MPEG-2 coded video," in *Proc. IEEE Int. Conf. Image Proc., ICIP*, pp. 273 –277, 1994.

[2] R. Aravind, M. Civanlar, and A. Reibman, "Packet loss resilience of MPEG-2 scalable video coding algorithms," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 426–435, October 1996.

[3] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. on Image Processing*, vol. 2, pp. 160–175, April 1993.