RATE-DISTORTION OPTIMIZED JPEG2000-BASED SCALABLE INTERACTIVE VIDEO (JSIV) WITH MOTION AND QUANTIZATION BIN SIDE-INFORMATION

Aous Thabit Naman and David Taubman

University of New South Wales,
School of Electrical Engineering and Telecommunications,
Sydney, Australia.

ABSTRACT

The authors have recently proposed a paradigm that can potentially provide for considerably better interactivity compared to existing practices and can adapt immediately to interactive changes in client interests, such as forward or backward playback and zooming into individual frames. The proposed paradigm relies on JPEG2000 format for providing scalability, flexibility, and accessibility; and on transmitting a server-optimized selection of code-blocks and motion side-information. Motion compensation and conditional replenishment are employed to reduce needed bandwidth. This work extends the previous work by providing server and client policies that allow for a realistic implementation and by introducing the use of coarsely quantized code-blocks in improving prediction. This work introduces the concepts, formulates the policies and optimization problems, proposes solutions, and compares the performance to alternate strategies.

Index Terms— Teleconferencing, video signal processing, image coding, image communication

1. INTRODUCTION

Video is one of the main applications of the Internet today; the interactivity provided, however, is limited most of the time to play, stop, and at best random access to a set of predetermined access points. The limitations in accessibility are the result of an encoder's attempt to exploit most of the redundancy in the source. The streamed video is encoded in one of the existing video coding standards such as MPEG-1 through MPEG-4 and H.261 through H.264; which at best offer limited scalability and accessibility. The recently approved scalable video coding (SVC) extension to H.264/AVC [1] provides improved scalability options; however its wide spread application is yet to be seen. Scalable video coding [1][2][3], in general, can solve some of the accessibility problems by accommodating the varying needs of different clients from one base source file; and by dynamically adapting to available network bandwidth gracefully degrading the streamed video quality. For this reason, it has been an active area of research in the last twenty years.

Recently, the authors have presented a new approach for serving scalable video [4] [5] [6] [7]. To provide for quality and spatial resolution scalability, this approach relies on JPEG2000 to independently compress the individual frames. For inter-frame redundancy reduction, the approach relies on motion compensation and optimized selection of JPEG2000 code-blocks. These goals are achieved through the cooperation of loosely coupled server and client policies. The server policy dynamically determines which code-blocks to send, while the client policy determines how best to make use of the data which it received from the server, possibly in conjunction with some motion model.

In [4], we presented a realistic implementation of the approach albeit its redundancy exploitation is limited to disjoint pairs of frames. In [5], the server optimization policy was extended to a sliding window of sequential frames, with the potential to exploit the redundancy between any two frames within the window. In [6], the server optimization policy was extended to a hierarchical group of frames, similar to the hierarchical B-frames dyadic prediction structure used by the SVC extension of H.264/AVC. In [7], we proposed the use of approximate distortion estimation rather than reconstructing the video to calculate real distortions. In this paper, we turn our attention to improving prediction samples by exploiting the knowledge about quantization bins of the received samples; we also propose realistic server and client policies.

A recent work by Mavlankar et. al. [8] proposes a way of dynamically providing pan/tilt/zoom features to different clients with varying regions of interest. The proposed method breaks a high resolution video into tiles and streams them simultaneously using H.264 compression and employing some peer-to-peer delivery techniques. Another recent work is that of Devaux et al. [9] which investigates a problem similar to the one we investigated in [5]; however, that work deals only with sequential prediction and does not employ motion-compensation. Cheung and Ortega [10] proposed recently the use of motion information and residual distortion as side information to enable flexible video delivery; which is conceptually similar to our work, except that it is based on distributed video coding.

The paradigm proposed here provides considerable interactivity and we expect it to benefit a diverse range of applications, such as surveillance video browsing and teleconferencing. As one example, the server can easily serve its content at reduced frame rate or reduced resolution without the need to re-encode its content[1]; we examine such a scenario in this paper. Additional examples, include region of interest access, selective delivery of novel content in the event of communication errors and isolated frame enhancement; the interested reader is referred to [4] and [5] for a discussion of various application scenarios.

The remainder of the paper is structured as follows. Sections 2 and 3 elaborate on our client and server policies, explaining their theoretical aspects. Section 4 provides experimental results. Finally, Section 5 states our conclusions.

2. CLIENT POLICY

Frames are arranged in a dyadic hierarchical structure with temporal decimation levels $T_0, T_1, \ldots T_K$. Figure 1 shows the structure for the case of $K = 3$. Each frame belongs to one or more temporal decimation level T_k depending on its position. Frames at level T_K are not predicted from any other frame (intra frames). At temporal

[1] under the assumption that the JPEG2000 content have more than one level of discrete wavelet transform, which is the norm

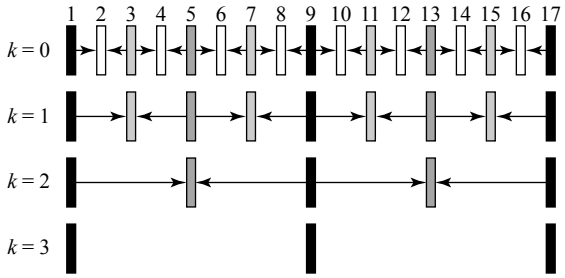

Fig. 1. Two groups of pictures in the dyadic hierarchical structure. The arrows show prediction directions and the numbers at the top are frame indices.

levels with $k < K$, the code-blocks can either come from the frame itself (intra code-blocks) or be predicted from the two nearby frames as shown in Figure 1. We write $f^k_{\to n}$ for the predicted frame given by

$$f^k_{\to n} = \frac{1}{2}\left(\mathcal{W}_{n-p\to n}\left(f^k_{n-p}\right) + \mathcal{W}_{n+p\to n}\left(f^k_{n+p}\right)\right) = \frac{1}{2}\left(f^k_{n-p\to n} + f^k_{n+p\to n}\right) \quad (1)$$

where $\mathcal{W}_{a\to b}$ is the motion compensation operator mapping f_a to f_b and $p = 2^k$ is the distance between a frame and the two adjacent frames at the k^{th} temporal level.

For each code-block C^β_n of each frame f_n, the client receives zero or more quality layers q^β_n. Consequently, the de-quantized samples $\tilde{C}^\beta_n(q^\beta_n)$ of that code-block have an associated distortion given by $\tilde{D}^\beta_n = \|\tilde{C}^\beta_n - C^\beta_n\|^2$. For frame reconstruction, it is also possible for the client to use $C^\beta_{\to n}$ obtained from the two-dimensional discrete wavelet transform (2D-DWT) of frame $f_{\to n}$ with an associated distortion given by $D^\beta_{\to n} = \|C^\beta_{\to n} - C^\beta_n\|^2$. In this work, these samples are improved by using the quantization bins of $\tilde{C}^\beta_n$ when one or more quality layers are available for that code-block as explained in the following paragraphs. We write $C^\beta_{\to n}(q^\beta_n)$ for the improved samples whenever we need to stress this fact.

JPEG2000 uses a fractional bit-plane encoder [11] for each code-block; as such, it is possible that different samples in the same code-block are quantized with different step widths. Let us denote a particular sample value by v_i and the lower and upper boundaries of its quantization bin by v_i^- and v_i^+, respectively. Knowledge of the bin can be utilized to improve prediction by limiting the predicted samples μ_i to the interval $\left[v_i^-, v_i^+\right]$. In the simplest case, the modified samples v_i' are given by

$$v_i' = \begin{cases} v_i^-, & \text{if } \mu_i < v_i^- \\ \mu_i, & \text{if } v_i^- \le \mu_i \le v_i^+ \\ v_i^+, & \text{if } \mu_i > v_i^+ \end{cases}$$

This strategy guarantees that v_i' is always closer to v_i than μ_i and that the prediction error $v_i - v_i'$ decreases monotonically with the availability of more quality layers. To see this, remember that the width of quantization bins decreases with the availability of more quality layers. This reduces prediction error for predicted samples that are outside or are becoming outside the quantization bin interval, while having no effect on samples inside the interval.

To further improve prediction, let us model the actual sample value v by a random variable V with a Laplace distribution given by $f_V(v) = \frac{1}{2}\alpha e^{-\alpha|v-\mu|}$, shown in Figure 2; where α is related to the variance by $\sigma_V^2 = \frac{2}{\alpha^2}$. This model is widely used in the literature to

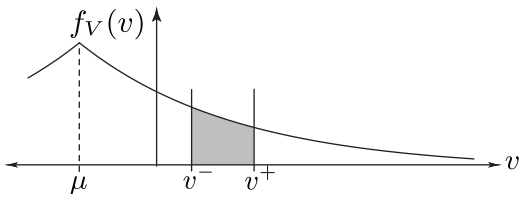

Fig. 2. Probability distribution function for residual distortion for the case of $\mu < v^-$. The shaded are is $P\left\{v^- \le v \le v^+\right\}$.

model residual distortion. Then, the improved prediction samples $\bar{v}_i$ are given by

$$\bar{v}_i = \begin{cases} \mu_i, & \text{if } v_i^- \le \mu_i \le v_i^+ \\ E\{v_i | v_i^- \le v_i \le v_i^+\}, & \text{otherwise} \end{cases} \quad (2)$$

where α is estimated from the residual distortion variance σ_V^2, calculated over code-block samples v_i' obtained from the simple prediction procedure outlined above. In practice, α changes considerably with the increase in the number of quality layers; however, α has only a small impact on the observed performance of the method.

To help the client decide when to use improved prediction, a simple client policy is used. For each code-block, the client receives a prediction flag $\mathcal{F}^\beta_n$ such that $\mathcal{F}^\beta_n = 1$ when the use of prediction can be beneficial for this code-block. For code-blocks with $q^\beta_n > 0$, the client also receives a quality layer threshold $\bar{q}^\beta_n$ such that $\tilde{D}^\beta_n(q^\beta_n) < D^\beta_{\to n}(q^\beta_n)$ when $q^\beta_n \ge \bar{q}^\beta_n$, as shown in Figure 3. We do not send $\bar{q}^\beta_n$ for code-blocks with $q^\beta_n = 0$ since these thresholds are of no use to the client for these code-blocks. More details about $\bar{q}^\beta_n$ are given in Section 3. Thus, the client policy can be summarized by:

$$C^\beta_n = \begin{cases} C^\beta_{\to n}(q^\beta_n), & \text{if } \mathcal{F}^\beta_n = 1 \text{ and } q^\beta_n < \bar{q}^\beta_n \\ \tilde{C}^\beta_n, & \text{otherwise} \end{cases} \quad (3)$$

In Section 3, we show that this policy guarantees that

$$D^\beta_n = \min\left\{\tilde{D}^\beta_n, D^\beta_{\to n}\right\} \quad (4)$$

3. SERVER POLICY

Frames are divided into groups of pictures $\mathcal{G}$, each with $2^K + 1$ frames. Frames at the $T_K{}^{\text{th}}$ decimation level are part of two consecutive groups; frame 9 in Figure 1 for example is part of $\mathcal{G}_0$ and $\mathcal{G}_1$. Each $\mathcal{G}$ is jointly optimized subject to a transmission budget of $L_{\max}$; as such, frames in T_K have two chances of receiving data. Using an additive model, the distortion in one $\mathcal{G}$ is given by

$$D = \sum_{n\in\mathcal{G}_s}\sum_{\beta\in f_n} D^\beta_n \quad (5)$$

The minimization of D subject to length constraint $L_{\max}$ can be (approximately) recast as the minimization of a family of Lagrangian functionals,

$$J_\lambda = \sum_{n\in\mathcal{G}_s}\sum_{\beta\in f_n}\left(D^\beta_n + \lambda\cdot|q^\beta_n|\right) \quad (6)$$

where $|q^\beta_n|$ denotes the number of bytes in q^β_n quality layers of C^β_n. The Lagrangian parameter λ is adjusted until the solution which minimizes J_λ satisfies the length constraint.

Direct minimization of (6) has two difficulties. The first is in selecting q^β_n since D^β_n for $f_n \notin T_K$ depend on other frames whose contribution is also being optimized. The second difficulty is that each C^β_n might contribute to code-blocks in other frames, in which

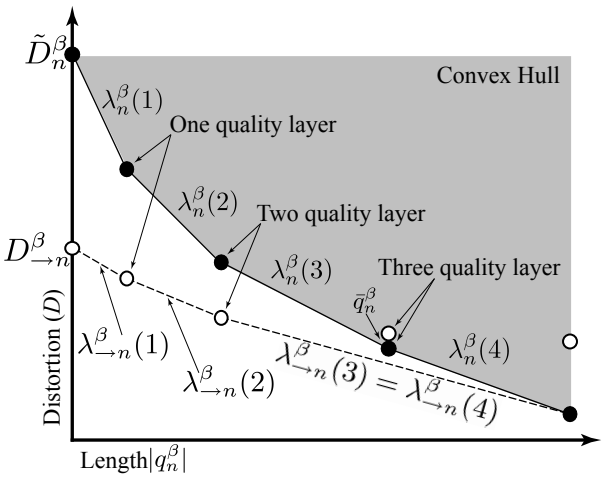

Fig. 3. A typical convex distortion-length curve for a code-block C_n^β for the case of $D_{\to n}^\beta(0) < \tilde{D}_n^\beta(0)$. Filled dots are for $\tilde{D}_n^\beta(q_n^\beta)$ and empty dots are for $D_{\to n}^\beta(q_n^\beta)$. Each dot represent one quality layer. The figure also shows the quality layer threshold $\bar{q}_n^\beta$.

case its distortion contribution should be weighted accordingly; however, this contribution is not known until the decisions for those other frames are made.

To deal with these two difficulties, we propose a two pass approach. In the first pass (PASS-1), q_n^β are determined for each C_n^β. In the second pass (PASS-2), the contributions of each C_n^β are evaluated and contribution weights w_n^β are determined as explained in Section 3.1. These weights are used in the next iteration of PASS-1. Multiple iterations of PASS-1 followed by PASS-2 are possible, where last iteration can skip PASS-2. However, we found that very little improvement is obtained beyond the second iteration. For PASS-1, we notice that f_n^K are independent of any other frames and q_n^β can be easily determined for a given λ as explained next. Once these are determined, the distortions D_n^β for all the f_n^{K-1} are known and their q_n^β can similarly be determined. This process is repeated until all q_n^β are decided.

We turn our attention to determining q_n^β. Figure 3 shows a typical distortion-length curve for a code-block C_n^β, which is guaranteed to be convex by construction [11]. Each circle in the figure represents one quality layer. Also shown in the figure is the distortion $D_{\to n}^\beta$. We define the distortion-length slope associated with each q_n^β by $\lambda_n^\beta(q) = (\tilde{D}_n^\beta(q-1) - \tilde{D}_n^\beta(q))/(|q| - |q-1|)$.

In the absence of any prediction data or when $D_{\to n}^\beta(0) \geq \tilde{D}_n^\beta(0)$, q_n^β is optimally determined from

$$q_n^\beta = \max\left\{q \mid \lambda_n^\beta(q) > \lambda\right\} \tag{7}$$

For f_n^k where $k < K$, the existence of prediction sources reduces the effective distortion to $D_{\to n}^\beta(q_n^\beta)$ when $q_n^\beta < \bar{q}_n^\beta$. This creates a new distortion-length convex hull; which can be computed by utilizing an algorithm for convex hull and slope computation similar to those proposed in [11]. Once these new slopes $\lambda_{\to n}^\beta$ are established; q_n^β can be determined by using (7). As can be seen from (3), the client uses $C_{\to n}^\beta(q_n^\beta)$ when $q_n^\beta < \bar{q}_n^\beta$, and $\tilde{C}_n^\beta(q_n^\beta)$ otherwise; which explains (4). Finally, $\mathcal{F}_n^\beta$ is set whenever $D_{\to n}^\beta(0) < \tilde{D}_n^\beta(0)$.

To obtain a policy that can be used for a variety of serving conditions, we calculate $D_{\to n}^\beta(q_n^\beta)$, $\bar{q}_n^\beta$, and $\mathcal{F}_n^\beta$ using full quality reference frames. Obviously, these parameters depend to some extent on the accuracy of the motion model used for motion compensation; therefore, we use two different accuracies for the motion models to cover the wide range of bit-rates used for this work. A more proper way is to use only one scalable motion model while modeling the effects of motion field quantization. This is left for future work.

3.1. Distortion Estimation and Weights

In [6] and [7], we derived the formulas that govern the propagation of quantization distortion from a reference frame to a predicted frame taking into account the effects of motion compensation. To avoid repetition while maintaining the completeness of this work, we summarize the results.

Distortion is estimated rather than calculated over *grid-blocks* finer than the code-block grid. It is assumed that distortion is constant over these grid-blocks. These distortions are mapped from one sub-band in the reference frame to many sub-bands in predicted frames using some mapping factors that depend on the sub-bands in question and on motion compensation operators. These mapping factors are pre-calculated and stored in the server. When predicted frames are themselves the source of further prediction, the residual and quantization distortions at those source frames are added and then mapped into the motion compensated target sub-bands. In doing this, we are effectively assuming not only that quantization and motion compensation errors are uncorrelated, so that their squared error distortions add, but also that the motion compensation errors produced at one level of the temporal hierarchy are uncorrelated with those produced at the next level. The validity of this assumption may be questionable, but seems necessary for the development of a workable distortion estimation strategy.

One difficulty encountered when mapping distortions from a source frame to a predicted target frame is that each sub-band in the source frame generally affects all sub-bands in the target frame. In this work, we limit our distortion estimation procedure to consider the distortion which propagates only into sub-bands at the same spatial resolution, the next higher resolution and the next lower resolution in the target frame's 2D-DWT than the originating sub-band in the source frame. This approximation improves upon that used previously in [7].

Weights are also estimated over grid-blocks and they are mapped from one sub-band in the predicted frame to many sub-bands in the source frames. For code-blocks with weight $w_n^\beta > 0$, (7) becomes

$$q_n^\beta = \max\left\{q \mid (1 + w_n^\beta) \cdot \lambda_n^\beta(q) > \lambda\right\} \tag{8}$$

4. EXPERIMENTAL RESULTS

The results presented here are for two test sequences. The first is the MPEG test sequence "Crew," while the second is "OldTownCross"[2]. Only the first 49 frames of the Y-component in "Crew" are used. These frames have a resolution of 704×576 with a bit depth of 8 bits and a frame rate of 30 fps. Similarly, only the first 65 frames of the Y-component in "OldTownCross" are used; these frames are cropped to a resolution of 3840×2048 and the bit depth is reduced to 8 bits at a frame rate of 50 fps.

Both sequences are compressed to JPEG2000 format using Kakadu[3], employing five levels of wavelet decomposition, 20 quality layers, and a code-block size of 32×32. Motion compensation employs an advanced hierarchical block-based motion model with

[2]European Broadcasting Union HD test sequences; `http://www.ebu.ch/en/technical/hdtv/test_sequences.php`

[3]`http://www.kakadusoftware.com/`, Kakadu software, version 5.2.4.

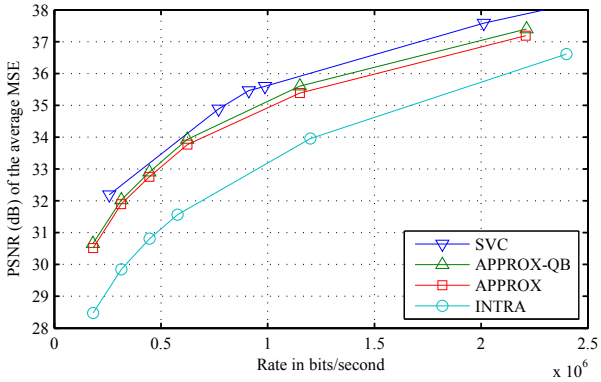

Fig. 4. Performance comparison between the various algorithms for the "Crew" sequence at various bit-rates in PSNR of the average MSE.

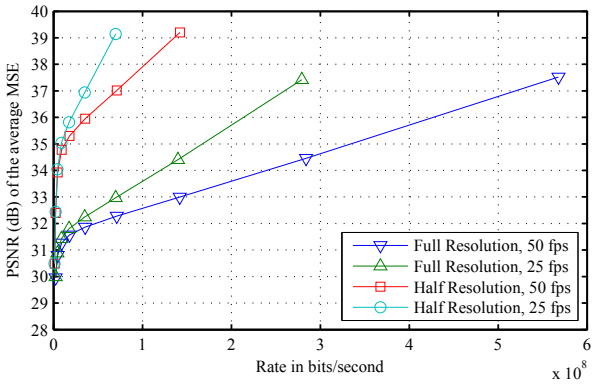

Fig. 5. Performance comparison for different clients accessing the "OldTownCross" media but with different requirements measured in PSNR.

a variable grid size and $1/4$ pixel motion vector accuracy. For both sequences, the rates given here are for the encoded sub-band samples and encoded motion vectors; they exclude headers and client signaling that might be required in a complete system.

For the "Crew" sequence, four methods are compared in Figure 4: the method that uses quantization bins and approximate distortion estimation, identified as "APPROX-QB"; the method that uses approximate distortion estimation without the use of quantization bins, identified as "APPROX"; the method that individually optimizes each frame subject to the rate constraint, identified as "INTRA"; and the SVC[4] extension of H.264/AVC, identified as "SVC". All results are expressed in terms of PSNR calculated from the average MSE.

For "APPROX", the server policy is slightly different to the policy presented here, as the server is aware that the client is incapable of improving prediction and therefore does not send any layers for predicted frames unless they are more than $\bar{q}_n^{\beta}$. For "SVC", three enhancement layers are used with three levels of MGS giving a total of 10 quality layers. These layers cover the quality range from 32.2dB to 42.2dB (not shown in full) and each point is one layer.

The flexibility of the paradigm is demonstrated in Figure 5, which shows four clients accessing the same "OldCrossTown" sequence contents but with different requirements, some at 25 fps and some at 50fps, some at full resolution and some at half resolution.

5. CONCLUSION

The proposed paradigm provides better support for interactive video delivery compared to existing video coding technologies; it allows for forward/backward, pan/tilt/zoom, region of interest, and individual frame access. This extended flexibility can be provided with a rather small negative impact on quality when compared to existing video coding technologies, under test conditions favorable to the latter. However, we expect the proposed approach to perform considerably better in areas where existing technologies fail, such as region of interest video delivery. Hierarchical arrangement of frames provides good exploitation of temporal redundancies for the proposed method. The use of quantization bins from a given code-block, in conjunction with motion compensated prediction from surrounding frames in the hierarchy, slightly improves delivered video quality. More work remains to be done to define comprehensive client-server signaling, possibly as an amendment to the JPIP standard (IS15444-9).

[4]JSVM version 9.12.2 obtained using CVS from its repository at `garcon.ient.rwth-aachen.de`

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. on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[2] N. Mehrseresht and D. Taubman, "An efficient content-adaptive motion compensated 3D-DWT with enhanced spatial and temporal scalability," *IEEE Trans. Image Proc.*, pp. 1397–1412, June 2006.

[3] J.-R. Ohm, "Advances in scalable video coding," *Proc. of the IEEE*, vol. 93, January 2005.

[4] A.T. Naman and D. Taubman, "A novel paradigm for optimized scalable video transmission based on JPEG2000 with motion," *Proc. IEEE Int. Conf. Image Proc. 2007*, Septmeber 2007.

[5] A.T. Naman and D. Taubman, "Optimized scalable video transmission based on conditional replenishment of JPEG2000 code-blocks with motion compensation," *MV '07: Proceedings of the International Workshop on Workshop on Mobile Video*, pp. 43–48, Septmeber 2007.

[6] A.T. Naman and D. Taubman, "Rate-distortion optimized delivery of JPEG2000 compressed video with hierarchical motion side information," *Proc. IEEE Int. Conf. Image Proc. 2008*, pp. 2312–2315, October 2008.

[7] A.T. Naman and D. Taubman, "Distortion estimation for optimized delivery of JPEG2000 compressed video with motion," *IEEE 10th Workshop on Multimedia Signal Processing, 2008, MMSP 2008*, pp. 433–438, October 2008.

[8] A. Mavlankar, J. Noh, P. Baccichet, and B. Girod, "Peer-to-peer multicast live video streaming with interactive virtual pan/tilt/zoom functionality," *Proc. IEEE Int. Conf. Image Proc. 2008*, pp. 2296–2299, October 2008.

[9] F.-O. Devaux, J. Meessen, C. Parisot, J.-F. Delaigle, B. Macq, and C. De Vleeschouwer, "A flexible video transmission system based on JPEG2000 conditional replenishment with multiple references," *Proc. IEEE Int. Conf. Acoust. Speech and Sig. Proc.*, April 2007.

[10] N.-M. Cheung and A. Ortega, "Flexible video decoding: A distributed source coding approach," *IEEE 9th Workshop on Multimedia Signal Processing, 2007, MMSP 2007*, pp. 103–106, 1-3 Oct. 2007.

[11] D.S. Taubman and M.W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Boston, 2002.