A NOVEL PARADIGM FOR OPTIMIZED SCALABLE VIDEO TRANSMISSION BASED ON JPEG2000 WITH MOTION

Aous Thabit Naman and David Taubman

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

ABSTRACT

A novel paradigm for optimized scalable video streaming is presented. The paradigm proposes transmission of motion vectors and selected code-blocks of the JPEG 2000 representation of each new frame, instead of frame differences as in existing methods. These code-blocks are selected to achieve the highest MSE. This paradigm overcomes the flexibility and accessibility limitations imposed by predictive motion compensated video by relying on the JPEG 2000 stream features for spatial scalability and on motion compensation and server-optimized conditional replenishment for temporal redundancy reduction. It is expected that real-time and interactive applications, such as teleconferencing and surveillance, would benefit most from this paradigm. This paper introduces the paradigm, formulates an optimization procedure for one simple case where it can be applied and compares its performance with alternate strategies.

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

1. INTRODUCTION

Current video compression techniques can be classed as scalable or non-scalable. Existing video compression standards such as MPEG-1 through MPEG-4 and H.261 through H.264 offer at most limited scalability. Scalable video coders aim to produce layered (or embedded) bit-streams, corresponding to progressively higher qualities and/or resolutions. Indeed, promising results have been obtained for scalable video coding within the last few years [1] [2] [3]. Even for scalable video coding, the structure of the coder imposes accessibility limitations for streaming applications. As an example, if a remote client is particularly interested in just one frame, the server must send sufficient bits to reconstruct a larger number of frames in order to invert the motion compensated transform used during compression. We shall consider other examples below.

Here, we propose a novel paradigm for serving video, in which motion information is used by the decoder, but does not affect the compression of source content. Specifically, we rely on JPEG 2000 to independently compress the original video frames, providing quality scalability and spatial resolution scalability. To exploit inter-frame redundancy, we rely on a client policy for combining received (distorted) frames in accordance with a motion model, as well as a server policy for determining the number of quality layers which should be delivered for each code-block in order to optimize the result at the client side. A closely related paradigm has shown promising results for the interactive browsing of 3D scenes in [4].

In the extreme case, the server could attempt to fully model the implications of all possible transmission scenarios at the client, selecting the best one. However, in this paper we show how it is possible for the server to use a small set of statistics for each code-block, combined with a simplified version of the motion model to derive a good transmission policy. In the extreme case, a client might attempt to deduce the motion itself, estimate distortions and explore a variety of rendering policies based on the information it has received from the server. Again, however, we prefer to work with much simpler client policies, assisted by motion and some additional side information which is supplied by the server. In any case, the policy naturally tends to decouple the roles of the client and server policies so that the communication paradigm does not fail if the server policy departs from that expected by the client or vice-versa.

We now briefly consider some benefits of the proposed paradigm. Clearly, since motion was not used during the original source compression, the server is free to send whatever motion information it likes, depending on available bandwidth, optimizing its delivery of the source material accordingly. This also allows delay sensitive clients to be served very differently to others. Moreover, for off-line applications the video could be browsed backwards as efficiently as forwards, possibly with different motion parameters. The server is also free to skip frames or frame regions which are of no interest to the client.

The proposed paradigm has other implications for lossy transmission environments. For example, if the server discovers that a packet has been lost from a reference frame, it can rectify the problem by sending a code-block from the subsequent frame instead of attempting to re-send the lost packet, as in existing video streaming technologies.

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

2. CLIENT POLICY

For simplicity, we consider here only the special case in which each pair of frames has only one motion model, which can be used to produce a motion compensated estimate of the second (predicted) frame f_p from the first (reference) frame f_r. For each code-block C_k^β of each frame f_k the client receives some number of quality layers q_k^β, possibly 0. We write $\tilde{C}_k^\beta$ for the dequantized samples which can be recovered from these layers and $D_k^\beta = \left\| \tilde{C}_k^\beta - C_k^\beta \right\|^2$ for the corresponding distortion. As an alternative to $\tilde{C}_p^\beta$ in the second frame, the client could use the corresponding subband samples, $\tilde{C}_{r\to p}^\beta$, obtained from a motion compensated version of the first

frame. The tilde-notation here is used to remind the reader that this version also involves quantization distortion. The distortion associated with this second choice is given by

$$\begin{aligned} D^{\beta}_{r\to p} &= \left\| \tilde{C}^{\beta}_{r\to p} - C^{\beta}_{p} \right\|^2 \\ &= 2\left\langle C^{\beta}_{r\to p} - C^{\beta}_{p}, \tilde{C}^{\beta}_{r\to p} - C^{\beta}_{r\to p} \right\rangle \\ &\quad + \underbrace{\left\| C^{\beta}_{r\to p} - C^{\beta}_{p} \right\|^2}_{D^{\mathrm{M},\beta}_{r\to p}} + \underbrace{\left\| \tilde{C}^{\beta}_{r\to p} - C^{\beta}_{r\to p} \right\|^2}_{D^{\mathrm{Q},\beta}_{r\to p}} \approx D^{\mathrm{M},\beta}_{r\to p} + D^{\mathrm{Q},\beta}_{r\to p} \end{aligned} \tag{1}$$

Here, $D^{\mathrm{M},\beta}_{r\to p}$ and $D^{\mathrm{Q},\beta}_{r\to p}$ represent the distortion contributions due to motion compensation and quantization errors in the first frame, respectively. We assume that motion and quantization errors are likely to be uncorrelated in practice, allowing us to ignore the cross term.

Ideally, the client policy reconstructs f_p using the motion compensated version $\tilde{C}^{\beta}_{r\to p}$, if $D^{\beta}_{p} > D^{\beta}_{r\to p}$, and $\tilde{C}^{\beta}_{p}$ otherwise. In practice, we simplify this to $D^{\beta}_{p} > D^{\mathrm{M},\beta}_{r\to p}$. In this case, the number of layers $q^{\beta}_{p,\min}$ beyond which $D^{\beta}_{p} \le D^{\mathrm{M},\beta}_{r\to p}$ can be precomputed during content creation and delivered as side information by the server, where appropriate. If the value of $q^{\beta}_{p,\min}$ is not known, it defaults to ∞, so that the server needs only communicate $q^{\beta}_{p,\min}$ for those blocks which cannot reliably be predicted by motion compensation.

3. SERVER POLICY

To simplify matters, we consider the joint allocation of a fixed byte budget $L_{\max}$ to the code-blocks of each pair of frames, so as to minimize the expected distortion experienced by the client. In the present formulation, we assume that the client has no pre-existing content for the two frames, although the derivation is readily extended to the case of pre-existing cached content. In this case, the client's distortion can be expressed as

$$D_r + D_p = \sum_{\beta\in f_r} D^{\beta}_{r} + \sum_{\beta\ni q^{\beta}_{p}<q^{\beta}_{p,\min}} D^{\beta}_{r\to p} + \sum_{\beta\ni q^{\beta}_{p}\ge q^{\beta}_{p,\min}} D^{\beta}_{p}$$

Such a problem can be formulated as the minimization of

$$\begin{aligned} J_\lambda = &\sum_{\beta\in f_r} \left(D^{\beta}_{r} + \lambda \left| q^{\beta}_{r} \right| \right) + \sum_{\beta\ni q^{\beta}_{p}<q^{\beta}_{p,\min}} D^{\beta}_{r\to p} \\ &+ \sum_{\beta\ni q^{\beta}_{p}\ge q^{\beta}_{p,\min}} \left(D^{\beta}_{p} + \lambda \left| q^{\beta}_{p} \right| \right) \end{aligned} \tag{2}$$

where $\left| q^{\beta}_{k} \right|$ denotes the number of bytes in q^{β}_{k} layers of C^{β}_{k} and the Lagrangian parameter λ is adjusted until the solution which minimizes J_λ satisfies the length constraint $L_{\max}$.

Direct optimization of equation (2) appears difficult, since the central term depends on both $\left\{q^{\beta}_{p}\right\}_{\beta}$ and $\left\{q^{\beta}_{r}\right\}_{\beta}$ in a complicated way. It turns out, however, that near optimal values for the q^{β}_{p} terms can be found without considering the influence of $\left\{q^{\beta}_{r}\right\}_{\beta}$, based solely on the value of λ. To see this consider Figure 1, which shows a typical rate-distortion curve for a code-block C^{β}_{p}. Each circle on the figure represents one quality layer. Also shown on the figure is the distortion $D^{\beta}_{r\to p}$ which can be achieved by using motion compensated prediction instead of $\tilde{C}^{\beta}_{p}$. Note that the D^{β}_{p} vs. $\left| q^{\beta}_{p} \right|$ characteristic is guaranteed to be convex by virtue of the rate-distortion optimization procedure performed during JPEG 2000 compression of the original images.

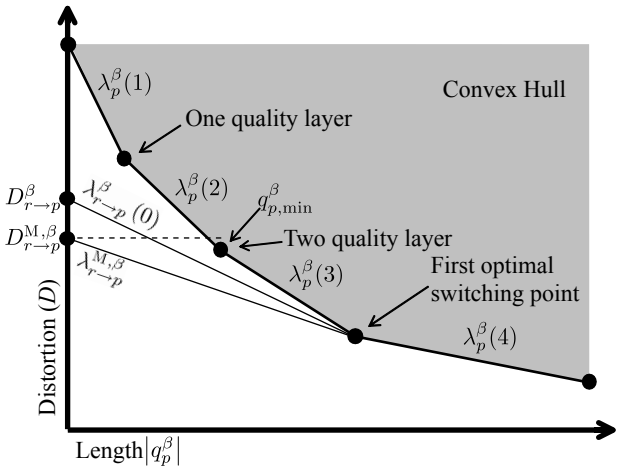

Fig. 1. A typical rate-distortion curve for a code-block C^{β}_{p} showing the quality layers.

In the absence of a previous frame, the optimal choice for q^{β}_{p} is given by

$$q^{\beta}_{p} = \max\left\{ q \mid \lambda^{\beta}_{p}(q) > \lambda \right\} \tag{3}$$

where $\lambda^{\beta}_{p}(q) = (D^{\beta}_{p}(q-1) - D^{\beta}_{p}(q))/(|q| - |q-1|)$ is the distortion-length slope associated with layer q. The availability of the previous frame, however, reduces the effective distortion associated with the choice $q^{\beta}_{p} = 0$ to $D^{\beta}_{r\to p}$. Figure 1 shows the impact of this option on the effective distortion-length convex hull. The new maximum distortion-length slope is written as $\lambda^{\beta}_{r\to p}(0)$ and the optimal choice for q^{β}_{p} becomes

$$q^{\beta}_{p} = \begin{cases} 0 & \text{if } \lambda > \lambda^{\beta}_{r\to p}(0) \\ \max\left\{ q \mid \lambda^{\beta}_{p}(q) > \lambda \right\} & \text{if } \lambda \le \lambda^{\beta}_{r\to p}(0) \end{cases}$$

This means that no information for C^{β}_{p} should be sent by the server unless $\lambda \le \lambda^{\beta}_{r\to p}(0)$. Beyond this point, the selection of q^{β}_{p} is independent of $D^{\beta}_{r\to p}$ and hence $\left\{q^{\beta}_{r}\right\}_{\beta}$.

We still have the problem that the threshold $\lambda^{\beta}_{r\to p}(0)$ depends in a complicated way upon $\left\{q^{\beta}_{r}\right\}_{\beta}$. Equation (1), however, shows that $D^{\beta}_{r\to p}$ is composed of two parts; $D^{\mathrm{Q},\beta}_{r\to p}$ tends to be small in comparison to $D^{\mathrm{M},\beta}_{r\to p}$ at most bit-rates of practical interest, partly because the service policy itself encourages the assignment of more bits to the reference frame wherever motion compensated prediction is effective. Therefore, we propose using $D^{\mathrm{M},\beta}_{r\to p}$ instead of $D^{\beta}_{r\to p}$ for the purpose of estimating $\lambda^{\beta}_{r\to p}(0)$. However, a more sophisticated server could iteratively update this estimate, once values for the q^{β}_{r} terms have been assigned. Writing $\lambda^{\mathrm{M},\beta}_{r\to p}$ for our estimate of $\lambda^{\beta}_{r\to p}(0)$, the server's policy becomes

$$q^{\beta}_{p} = \begin{cases} 0 & \text{if } \lambda > \lambda^{\mathrm{M},\beta}_{r\to p} \\ \max\left\{ q \mid \lambda^{\beta}_{p}(q) > \lambda \right\} & \text{if } \lambda \le \lambda^{\mathrm{M},\beta}_{r\to p} \end{cases} \tag{4}$$

Figure 1 also shows $q^{\beta}_{p,\min}$, introduced earlier. It should be clear that the value of q^{β}_{p}, selected in accordance with equation (4), will always be at least as large as $q^{\beta}_{p,\min}$, if non-zero. It is also worth

noting that the value of $\lambda_{r\to p}^{\mathrm{M},\beta}$ can be computed ahead of time, since it does not depend on the dynamic state of any client.

We turn attention now to the problem of finding the q_r^β terms. According to equation (2), these terms influence both D_r^β and those $D_{r\to p}^{\beta'}$ for which $\lambda > \lambda_{r\to p}^{\mathrm{M},\beta'}$. To evaluate the impact of q_r^β on $D_{r\to p}^{\beta'}$, consider the following. The error in frame f_r, can be expressed in term of the errors at each location $\mathbf{k}$ in each of its subbands b, as

$$\delta f_r = \sum_b \sum_{\mathbf{k}} \delta B_r^b[\mathbf{k}] \cdot S_{\mathbf{k}}^b$$

where $S_{\mathbf{k}}^b$ denotes the relevant synthesis vectors (images). Let $\overrightarrow{\mathcal{W}}$ denote the forward motion compensated mapping from frame f_r to f_p. The error at the $\mathbf{p}^{\text{th}}$ location in the predicted subband b' of f_p, due to quantization in the $\mathbf{k}^{\text{th}}$ location of subband b in f_r is then given by

$$\delta B_{r\to p}^{\mathrm{Q},b'}[\mathbf{p}] = \sum_b \sum_{\mathbf{k}} \delta B_r^b[\mathbf{k}] \cdot \left\langle \overrightarrow{\mathcal{W}}(S_{\mathbf{k}}^b), A_{\mathbf{p}}^{b'} \right\rangle$$

where $A_{\mathbf{p}}^{b'}$ is the analysis function for subband b' at location $\mathbf{p}$. Assuming that quantization errors in the subbands are approximately uncorrelated, the distortion power for some region $\mathcal{R}'$ in subband b' can then be approximated by

$$\sum_{\mathbf{p}\in\mathcal{R}'} \left|\delta B_{r\to p}^{\mathrm{Q},b'}[\mathbf{p}]\right|^2 \approx \sum_b \underbrace{\sum_{\mathbf{p}\in\mathcal{R}'} \sum_{\mathbf{k}} \left|\delta B_r^b[\mathbf{k}]\right|^2 \cdot \left\langle \overrightarrow{\mathcal{W}}(S_{\mathbf{k}}^b), A_{\mathbf{p}}^{b'} \right\rangle^2}_{D_{\mathcal{R}'}^{b\to b'}}$$

A result of the decay in the finite support operators $\overrightarrow{\mathcal{W}}(S_{\mathbf{k}}^b)$ and $A_{\mathbf{p}}^{b'}$ is that $D_{\mathcal{R}'}^{b\to b'}$ depends mainly on the distortion contributions $\delta B_r^b[\mathbf{k}]$ inside and around the region $\mathcal{R}$; the projection of $\mathcal{R}'$ in subband b. If $\mathcal{R}'$ is small enough such that the distortion around it can be approximated by a single average quantization noise power, we have

$$D_{\mathcal{R}'}^{b\to b'} \approx \frac{D_{\mathcal{R}}^b}{|\mathcal{R}|} \cdot \sum_{\mathbf{p}\in\mathcal{R}'} \underbrace{\sum_{\mathbf{k}} \left\langle \overrightarrow{\mathcal{W}}(S_{\mathbf{k}}^b), A_{\mathbf{p}}^{b'} \right\rangle^2}_{W_{\mathbf{p}}^{b\to b'}}$$

Here, $W_{\mathbf{p}}^{b\to b'}$ represents a weight for the contribution of the average quantization noise power around $\mathcal{R}$ in subband b to the distortion at location $\mathbf{p}$ in subband b'. Denoting the average quantization noise power $D_{\mathcal{R}}^b/|\mathcal{R}|$ around $\mathcal{R}$ by $\bar{D}_r^b[\mathbf{k}]$ gives

$$\left|\delta B_{r\to p}^{\mathrm{Q},b'}[\mathbf{p}]\right|^2 \approx \sum_b \bar{D}_r^b\left[\overleftarrow{\mathcal{W}}_{b'\to b}(\mathbf{p})\right] \cdot W_{\mathbf{p}}^{b\to b'}$$

where $\overleftarrow{\mathcal{W}}_{b'\to b}$ is the reverse motion mapping, which takes a location from subband b' in f_p to a location in subband b of f_r. Equivalently, the predicted code-block distortions are given by

$$D_{r\to p}^{\mathrm{Q},\beta'} \approx \sum_{\mathbf{p}\in C_p^{\beta'}} \sum_{\beta\ni\overleftarrow{\mathcal{W}}_{b'\to b}(\mathbf{p})\in C_r^\beta} D_r^\beta \frac{W_{\mathbf{p}}^{b\to b'}}{\left|C_r^\beta\right|}$$

So the impact of D_r^β on the central term in equation (2) can be expressed as

$$D_r^\beta \cdot \underbrace{\sum_{\beta'\ni\lambda>\lambda_{r\to p}^{\mathrm{M},\beta'}} \sum_{\mathbf{p}\in C_p^{\beta'}} \sum_{\beta\ni\overleftarrow{\mathcal{W}}_{b'\to b}(\mathbf{p})\in C_r^\beta} \frac{W_{\mathbf{p}}^{b\to b'}}{\left|C_r^\beta\right|}}_{\theta_r^\beta}$$

Although the last equation looks complicated, the interpretation is simple. θ_r^β is a state variable associated with each code-block in f_r. For a given value of λ, we identify all code-blocks β' in the predicted frame such that $\lambda > \lambda_{r\to p}^{\mathrm{M},\beta'}$, meaning that they will be assigned no bits. For each location in each such code-block, we find the corresponding location in each subband of f_r and add $W_{\mathbf{p}}^{b'\to b}/\left|C_r^\beta\right|$ to the corresponding state variable, θ_r^β. In practice, we actually perform this state mapping process on a coarse grid of 4×4 samples, using pre-computed values for $W_{\mathbf{p}}^{b'\to b}$ which are stored at the server on this grid.

Hence, finding q_r^β simplifies to the minimization of

$$J_\lambda = \sum_{\beta\in f_r} \left[\left(1+\theta_r^\beta\right)\cdot D_r^\beta + \lambda\left|q_r^\beta\right|\right] \tag{5}$$

since $D_{r\to p}^{\mathrm{M},\beta}$ is independent of the quantization distortion and q_p^β have already been determined. Equation (5) can thus be solved for each code-block independently by solving

$$q_r^\beta = \arg\min\left[\left(1+\theta_r^\beta\right)\cdot D_r^\beta + \lambda\left|q_r^\beta\right|\right] \tag{6}$$

The calculations required for each iteration of the optimization are acceptable and we believe real-time implementation is possible for a reasonable number of concurrent clients.

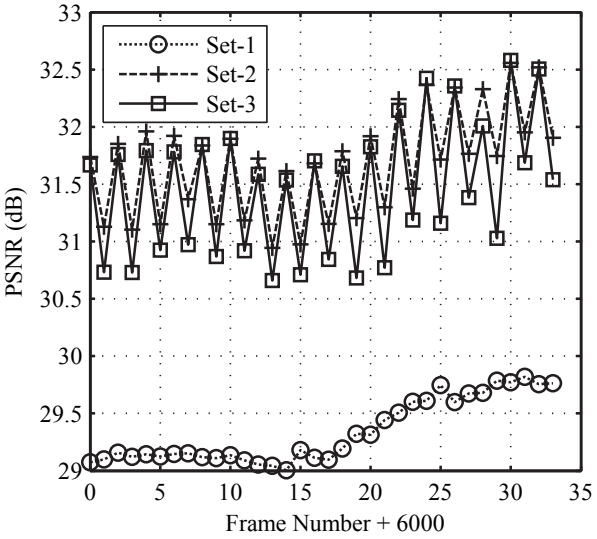

Fig. 2. Frames PSNR for the three test sets at 5000 bytes/frame.

4. EXPERIMENTAL RESULTS

The result presented here are for a sequence of frames from the DCI[1] StEM content, which is referred to as "Clip 2" during the DCI compression test, starting from frame 6000 up to 6033, with filenames MM_4K_XYZ_06000.tif up to MM_4K_XYZ_06033.tif. The images has a 4096×1714 resolution with a bit-depth of 16-bit per component in the XYZ domain, however, only the Y component is used after being gamma-corrected and truncated to 8 bits. Also,

[1]Digital Cinema Initiatives (DCI) and The American Society of Cinematographers (ASC), StEM mini-movie access procedure available at http://www.dcimovies.com/.

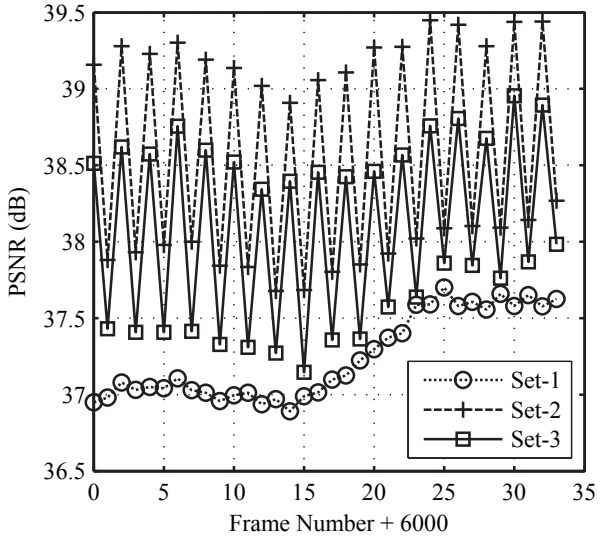

Fig. 3. Frames PSNR for the three test sets at 30000 bytes/frame.

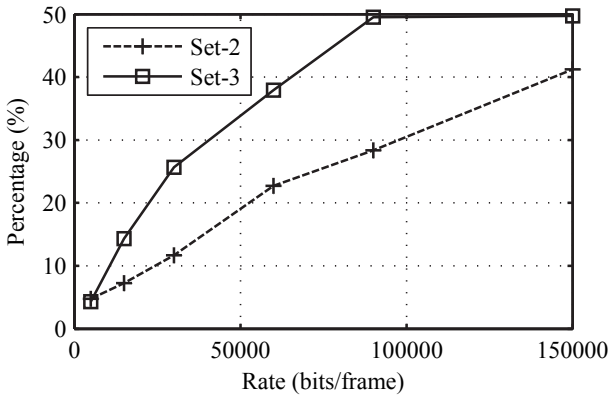

Fig. 4. The percentage of the bytes transmitted from f_p relative to the combined two frame bytes budget $L_{\max}$ at different frame rates for test sets 2 and 3.

the images are cropped to 4096×1712 by removing the bottom two rows due to limitations in the available motion compensation subroutines. Those images are compressed into the JPEG2000 format using the Kakadu[2] implementation employing five levels of wavelet decomposition, 20 quality layers, and a code-block size of 32×32. Motion compensation employs a deformable triangular mesh motion model that uses a node spacing of 16 pixels and is applied only to the highest resolution. Motion is estimated to $1/8$ of a pixel accuracy employing a 7×7 interpolation kernel obtained by windowing cubic spline functions [5].

The paradigm is compared against two sets of results. The first set, referred to as "Set-1," comes from jointly optimizing two full frames at a time such that their rate-distortion slope is the same. That is minimizing

$$J_\lambda = \sum_{\beta \in f_r} \left(D_r^\beta + \lambda \left| q_r^\beta \right| \right) + \sum_{\beta \in f_p} \left(D_p^\beta + \lambda \left| q_p^\beta \right| \right)$$

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

The second set, referred to as "Set-2," comes from jointly optimizing the first frame; f_r, and the second frame's residues from motion-compensating the first frame; $f_{res} = \overrightarrow{\mathcal{W}}(f_r) - f_p$. One main difference for Set-2 is that the distortion for f_r is multiplied by 2 since it contributes to the distortion of two frame. That is minimizing

$$J_\lambda = \sum_{\beta \in f_r} \left(2 \cdot D_r^\beta + \lambda \left| q_r^\beta \right| \right) + \sum_{\beta \in f_{res}} \left(D_{res}^\beta + \lambda \left| q_{res}^\beta \right| \right)$$

The third set of results, referred to as "Set-3," comes from the proposed paradigm.

Figure 2 and Figure 3 show the PSNR for the three test sets for bit-rates of 5000 bytes/frame and 30000 bytes/frame, respectively. The figures indicate that using the proposed paradigm performs almost as good as Set-2 and much better than Set-1. As the bit-rate is increased to 90000 bytes/frame (not shown here), the proposed paradigm has slightly better performance than Set-1, which performs much better than Set-2, since f_p frames in Set-2 suffers from errors in both f_r and f_{res}.

Figure 4 shows the allocation of the byte budget between the two frames at different frame rates. It can be seen that at low bit-rates the optimization algorithm prefers spending more bits enhancing f_r as at these low rates the improvements in f_r results in improvements to f_p as well. At high bit-rates, however, it is better to divide the rate between the two frames equally as the motion distortion becomes the dominant factor.

5. CONCLUSIONS

We presented a novel paradigm, which is very flexible and has a reasonable performance. The approximations introduced are fair and performs reasonably well. For the particular implementation scenario addressed here, results indicate that at low-bit rates and high-bit rates the proposed paradigm has a favorable performance; however, at moderate rates it performs modestly. Moreover, the calculations required are moderate and real-time performance is possible for a reasonable number of clients. Further benefits are expected if more than two frames are optimized at a time.

6. REFERENCES

[1] 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.

[2] H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand, "Combined scalability support for the scalable extension of H.264/AVC," *Proc. Int. Conf. on Multimedia and Expo, 2005*, July 2005.

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

[4] P. Zanuttigh, N. Brusco, D. Taubman, and G.M. Cortelazzo, "Server policies for interactive transmission of 3D scenes," *IEEE Multimedia Signal Processing Workshop*, pp. 59–64, October 2006.

[5] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Proc.*, vol. 13, no. 8, pp. 1029–1041, August 2004.