Evaluating the Streaming of FGS–Encoded Video with Rate–Distortion Traces

Institut Eurécom Technical Report RR–03–078

June 2003

Philippe de Cuetos
Institut EURECOM
2229, route des Crêtes
06904 Sophia Antipolis, France
Email: philippe.de-cuetos@eurecom.fr

Martin Reisslein
Arizona State University
Goldwater Center MC 7206
Tempe AZ 85287–7206, USA
Email: reisslein@asu.edu

Keith W. Ross
Polytechnic University
Six MetroTech Center
Brooklyn, NY 11201, USA
Email: ross@poly.edu

Abstract

The MPEG–4 video coding standard has recently been enriched with Fine Granularity Scalability (FGS) a new coding technique expressly designed for video streaming. With FGS coding the video stream can be flexibly truncated at very fine granularity to adapt to the available network resources. In this report we provide a framework for the evaluation of the streaming of FGS-encoded video. The framework consists of evaluation metrics and a library of rate-distortion traces. This framework enables networking researchers without access to video codecs and video sequences to develop and evaluate rate-distortion optimized streaming mechanisms for FGS-encoded video. We apply our evaluation framework to investigate the rate-distortion optimized streaming at different video frame aggregation levels. We find that compared to the optimization for each individual video frame, optimization at the level of video scenes reduces the computational effort dramatically, while reducing the video quality only very slightly.

Index Terms

Fine granularity scalability, multimedia communications, performance evaluation, rate–distortion optimized streaming, scalable video, video coding, video streaming.

Rate–Distortion traces available at `http://trace.eas.asu.edu/indexfgs.html`. To avoid any conflict with copyright laws, we emphasize that all image processing, encoding, and analysis was done for scientific purposes. The encoded video sequences have no audio stream and are not publicly available. We make only the rate-distortion traces available to researchers.

The work of M. Reisslein is supported in part by the National Science Foundation under Grant No. Career ANI-0133252 and Grant No. ANI-0136774 as well as the state of Arizona through the IT301 initiative, and two matching grants from Sun Microsystems.

I. Introduction

Fine Granularity Scalability (FGS) has recently been added to the MPEG-4 video coding standard [1] in order to increase the flexibility of video streaming. With FGS coding the video is encoded into a base layer (BL) and one enhancement layer (EL). Similar to conventional scalable video coding, the base layer must be received completely in order to decode and display a basic quality video. In contrast to conventional scalable video coding, which requires the reception of complete enhancement layers to improve upon the basic video quality, with FGS coding the enhancement layer stream can be cut anywhere before transmission. The received part of the FGS enhancement layer stream can be successfully decoded and improves upon the basic video quality [2], [3]. Similar to conventional scalable encoding, the FGS enhancement layer is hierarchical in that "higher" bits require the "lower" bits for successful decoding. This means that when cutting the enhancement layer bit stream before transmission, the lower part of the bit stream (below the cut) needs to be transmitted and the higher part (above the cut) can be dropped. The FGS enhancement layer can be cut at the granularity of bits. This fine granular flexibility was the key design objective of FGS coding, along with good rate–distortion coding performance. With the fine granularity property, FGS–encoded videos can flexibly adapt to changes in the available bandwidth in wired and wireless networks. This flexibility can be exploited by video servers to adapt the streamed video to the available bandwidth in real–time (without requiring any computationally demanding re–encoding). In addition, the fine granularity property can be exploited by intermediate network nodes (including base stations in wireless networks) to adapt the video stream to the currently available downstream bandwidth.

FGS video coding has the potential to fundamentally change the video streaming in networks. With conventionally encoded video the goal of the streaming mechanism is to deliver the complete video stream (or complete layers) in a timely fashion so as to avoid the loss (starvation) of video data at the decoder. Network streaming mechanisms for conventional video typically focus on minimizing the loss of video data subject to the available resources (such as available bandwidth, buffers, start-up latency, etc.). This is very challenging due to the variabilities in the video traffic (bit rate) and the typically varying bandwidth available for video streaming. The key performance metric for conventional video streaming is typically the probability (or long run rate) of lost video data, i.e., data that misses its decoding and playout deadline at the client. This loss probability is a convenient metric for characterizing the performance of a video streaming mechanism as it can be obtained from video traffic models or frame size traces and does not require experiments with actual video codecs and video sequences. However, the loss probability is essentially a "network" metric and does not provide much quantitative insight into the video quality perceived by the user. Clearly, on a qualitative basis, a smaller starvation probability results generally in better video quality, but quantifying this relationship is very difficult without conducting experiments with actual video codecs and video sequences. This difficulty is due to the fact that conventional video coding is not explicitly designed to tolerate losses. Thus, the encoder rate–distortion curve, which relates the bit rate at the encoder output to the video quality obtained by decoding the entire video stream, can not directly be employed to assess the video quality after lossy network transport. (We note that for conventional scalable encoded video the decoded video quality can be obtained from the encoder rate-distortion curve at the granularity of complete layers.) Assessing the video quality is further complicated by the motion compensation and the resulting dependencies among the different frame types in MPEG–encoded video. Also, a number of techniques have been developed to attempt to repair (conceal) losses [4] or to make the encoded video more resilient to losses [5], [6]. All these issues need to be taken into consideration when assessing the decoded video quality after lossy network transport. We note that an approximate heuristic that relates the loss of video data to the decoded video quality has been examined [7], but in general determining the video quality after network transport requires experiments with actual video, see for instance [8].

In contrast to conventionally coded video, the FGS enhancement layer is designed to be cut (truncated) anywhere. The received part (below the cut) can be decoded and contributes to the video quality according the rate–distortion curve of the enhancement layer. More precisely, the received enhancement layer part of a given video frame contributes to the decoded quality of that frame according to its rate-distortion curve. In contrast to conventionally encoded video (i) it is not crucial to deliver the entire enhancement layer stream, and (ii) the decoded quality corresponding to the received and decoded part of the enhancement layer can be determined directly from the enhancement layer rate-distortion curve at the granularity of bits. Providing and analyzing these rate–distortion curves for different videos and explaining their use in networking studies is the main focus of this report. The provided FGS enhancement layer rate–distortion curves make it possible to assess the quality of the decoded video after lossy network transport with good accuracy. (We note here that the subjectively perceived video quality is very complex to assess and the topic of ongoing research; our evaluation framework allows for complex metrics, but uses the Peak Signal to Noise Ratio (PSNR) for numerical studies.) With the evaluation framework for FGS video streaming provided in this report it becomes fairly straightforward to use the video quality as performance metric for video streaming and to develop rate–distortion optimized streaming mechanisms even if video codecs and video sequences are not available.

Generally, the goal of rate–distortion optimized streaming [9], [10], [11] is to exploit the rate–distortion characteristics of the encoded video to maximize the overall video quality at the receiver while meeting the constraints imposed by the underlying network. The maximization of the overall quality is generally achieved by maximizing the quality of the individual video frames and by minimizing the variations in quality between consecutive video frames [11]. (Similar goals, albeit on the much coarser basis of layers, are pursued by the streaming mechanisms for conventionally layered video that maximize the number of delivered layers and minimize the changes in the number of completely delivered layers, see for instance [12], [13], [14].) The optimization of the overall video quality is in general approached by algorithms that take the rate–distortion functions of all individual video frames into account. With FGS–encoded video the optimization procedure at the server is to find the optimal number of enhancement layer bits to send for each image, subject to the bandwidth constraints. We apply our evaluation framework to examine this optimization on the basis of individual video frames. We find that due to the generally convex shape of the rate–distortion curve of the FGS enhancement layer the optimization per video frame can be computationally demanding, which may reduce the number of simultaneous streams that a high–performing server can simultaneously support. We explore an alternative optimization approach where the server groups several consecutive frames of the video into sequences and performs rate–distortion optimization over the sequences. In this approach, each frame within a given sequence is allocated the same number of bits. We demonstrate that by exploiting the strong correlations in quality between consecutive images, this aggregation approach has the potential to decrease the computational requirement of the optimization procedure, and thereby the computational load on video servers.

This report is organized as follows. In the following subsection we discuss the related work. In Section II we give a brief overview of FGS video coding. In Section III we present our framework for the evaluation of FGS video streaming. Our framework consists mainly of (i) metrics for assessing the traffic and quality characteristics of FGS video, and (ii) traces of rate–distortion characteristics of FGS–encoded video. We define metrics based on individual video frames as well as metrics based on aggregations of video frames (such as Groups of Pictures (GoPs) or visual scenes). We detail our method for generation of rate–distortion traces. We provide and analyze the traces for a short "Clip" as well as a representative library of long videos from different genres. Long traces are essential to obtain statistically meaningfully performance results for video streaming mechanisms. All traces and statistics are made publicly available on our web site at `http://trace.eas.asu.edu/indexfgs.html`. Since the rate–distortion characteristics depend strongly on the semantic video content it is important to consider videos from a representative set of genres. In Section IV we apply our evaluation framework to compare the rate–distortion optimized streaming for different video frame aggregation levels. We summarize our findings in Section V.

A. Related Work

Over the past few years, streaming video over the Internet has been the focus of many research efforts (see [15], [16] for comprehensive surveys). Because of the best–effort nature of the Internet, streaming video should adapt to the changing network conditions. One of the most popular techniques for network–adaptive streaming of stored video is using scalable video (see for instance [17], [18]). Video streaming applications should also adapt to the properties of the particular encoded video [10]. Recently, rate–distortion optimized streaming algorithms have been proposed (e.g., [9], [19]) to minimize the end–to–end distortion of media, for transmission over the Internet. Our work complements these studies by providing a framework for the evaluation of video streaming mechanisms for FGS encoded video.

Significant efforts have gone into the development of the FGS amendment to the MPEG–4 standard, see for instance [2], [3] for an overview of these efforts. Following standardization, the refinement and evaluation of the FGS video coding has received considerable interest [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30]. Recently, the streaming of FGS video has been examined in a number of studies, all of which are complementary to our work. General frameworks for FGS video streaming are discussed in [31], [32], [33]. The error resilience of FGS video streaming is studied in [11], [34], [35]. In [36] the FGS enhancement layer bits are assigned to different priority levels, which represent the importance of the carried content. In [37] a real–time algorithm for the network adaptive streaming of FGS–encoded video is proposed. The proposed algorithm does not take the rate distortion characteristics of the encoded video into consideration. The concept of scene–based streaming is briefly introduced in [38], but not evaluated with rate–distortion data. Streaming mechanisms which allocate the FGS enhancement layer bits over fixed length segments are studied in [39], [40] and evaluated using the well–known short MPEG test sequences; in contrast, in Section IV of this report, we study allocation of the FGS enhancement layer bits on individual frame, fixed–length segment, and video scene basis using our evaluation framework. A packetization and packet drop

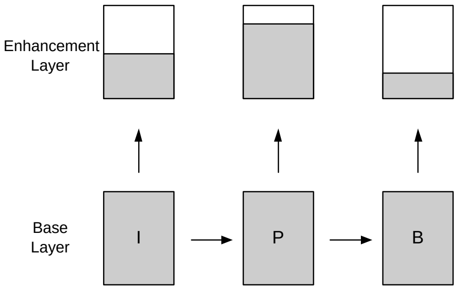


Fig. 1. Example of truncating the FGS enhancement layer before transmission

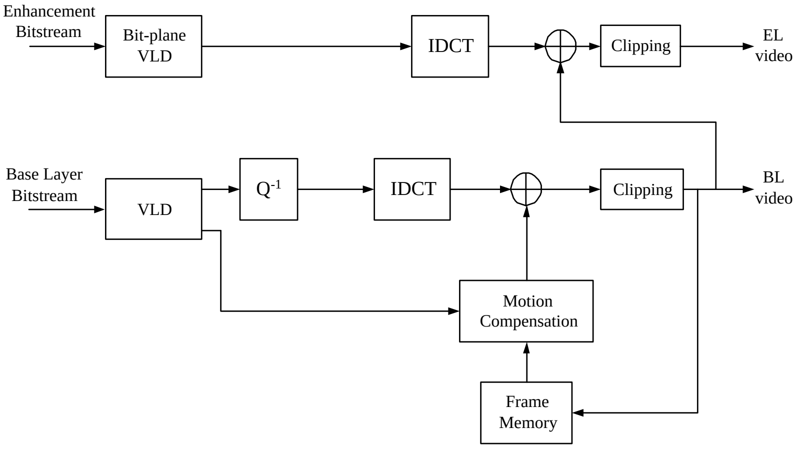


Fig. 2. MPEG–4 FGS decoder structure: Both layer bit streams are variable length decoded (VLD). In addition, the enhancement layer is bit–plane decoded and the base layer is inverse quantized (Q^{-1}). Both layers are then passed through the inverse discrete cosine transform (IDCT).

policy for FGS video streaming is proposed in [41]. An efficient approach for the decoding of streamed FGS video is proposed in [42]. Finally, streaming of FGS video over multicast [43] and to wireless clients [44], [45], [46], [47] has also been considered, while issues of FGS complexity scaling and universal media access are addressed in [48], [49].

This report is in many respects a follow–up of our earlier study on MPEG-4 encoded video [50]. In [50] we studied the traffic characteristics of single–layer (non–scalable) MPEG–4 and H.263 encoded video for different video quality levels. The quality level was controlled by the quantization scale of the encoder. However, neither the video quality nor the relationship between video traffic (rate) and video quality (distortion) were quantitatively studied in [50]. In the technical report series [51] the video traffic, quality, and rate–distortion characteristics of video encoded into a single layer and video encoded with the conventional temporal and spatial scalability modes have been quantitatively studied. In contrast to [50] and [51], in this report we consider the new *fine granularity scalability* mode of MPEG–4 and study quantitatively the video traffic (rate), video quality (distortion), as well as their relationship (rate–distortion) for FGS encoded video.

II. Overview of Fine Granularity Scalability (FGS)

Fine Granularity Scalability (FGS) has been introduced in the MPEG–4 standard, specifically for the transmission of video over the Internet [1]. The unique characteristic of FGS encoding compared with conventional scalable encoding is that the enhancement layer bit stream can be truncated anywhere and the remaining part can still be decoded. Figure 1 shows an example of truncating the FGS enhancement layer before transmission. For each frame, the shaded area in the enhancement layer represents the part of the FGS enhancement layer which is actually sent by the server to the client. Truncating the FGS enhancement layer for each frame before transmission allows the server (or intermediate network nodes or gateways) to adapt the transmission rate finely to changing network conditions. At the client side, the decoder can use the truncated enhancement layer to enhance the quality of the base layer stream.

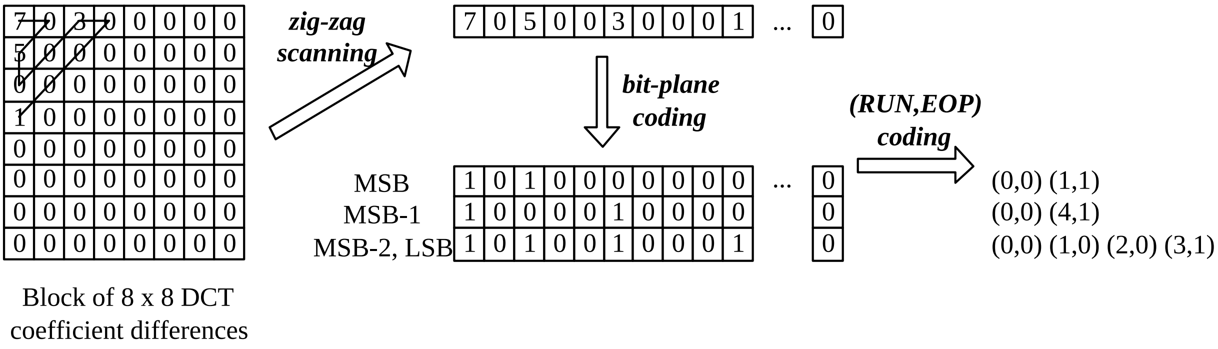


Fig. 3. Example of bit–plane coding

In this report, we focus on the MPEG–4 Signal to Noise Ratio (SNR) Fine Granularity Scalability [2], [52], where the FGS enhancement layer contains an encoding of the quantization error between the original image and the corresponding base layer decoded image. Figure 2 illustrates the architecture of the MPEG–4 SNR FGS decoder. According to the MPEG–4 standard, and as illustrated in the figure, only the base layer frames are stored in frame memory and used for motion compensation (and predictive encoding using the Intra-coded, Predicted, and Bi-directionally predicted (I, P, and B) frame types). There is no motion compensation within the FGS enhancement layer. This makes the enhancement layer highly resilient to transmission errors, and subsequently well suited to the transmission over error–prone networks such as the best–effort Internet. A typical scenario for transmitting MPEG–4 FGS encoded videos over the Internet has been proposed by the MPEG–4 committee in [53]. In this scenario the base layer is transmitted with high reliability (achieved through appropriate resource allocation and/or channel error correction) and the FGS enhancement layer is transmitted with low reliability (i.e., in a best effort manner and without error control).

The fine granularity property of the FGS enhancement layer comes from the bit–plane encoding of the enhancement layer discrete cosine transform coefficients. In digital video each pixel is represented by one luminance value and two chrominance values. The pixels are grouped into blocks of typically 8x8 pixels. The 64 luminance values in the block are then quantized and subsequently transformed using the discrete cosine transform (DCT) to produce a block of 8x8 DCT transform coefficients. (The chrominance values are processed in similar fashion, but are typically sub–sampled prior to quantization and transformation.) With conventional single–layer encoding the DCT coefficients are zig–zag scanned and then compressed using run–level coding. The run–level symbols are then variable–length coded. The base layer of SNR scalable encoding is obtained by quantizing the original luminance (and chrominance) values and then carrying out the DCT transform, zig–zag scan, and coding as just outlined. To obtain the enhancement layer of a conventional SNR scalable encoding, the base layer is decoded and the difference (mostly due to the quantization) between the decoded base layer and the original image is obtained. The difference is then quantized with a smaller quantization step size and subsequently DCT transformed, zig–zag scanned, and run–level as well as variable–length coded. (Multiple enhancement layers are obtained by carrying out this encoding of the differences with successively smaller quantization step sizes.)

The main difference between conventional encoding and FGS encoding is that the DCT coefficients of the enhancement layer are not run–level encoded in the FGS encoding. Instead the DCT coefficients are bit–plane coded, which we now illustrate with an example. (Another difference is that there is only one enhancement layer with FGS encoding. To obtain this one enhancement layer the difference between the decoded base layer and the original image is DCT coded without prior quantization.) Consider the 8x8 block of enhancement layer DCT coefficients in the left part of Fig. 3. The coefficients are scanned in zig–zag order to give the sequence of 64 integers starting with 7, 0, 5, $\ldots$. Each integer is then represented in the binary format (e.g., 7 is represented by 111, 3 is represented by 011). The representation for each integer is written in a vertical column as illustrated in the middle of Fig. 3 to form an array that is 64 columns wide and 3 rows deep (as the largest integer in this example has a 3 bit binary representation, in practice 8 bit representations are typically used). The bitplanes are obtained by scanning the rows of the array horizontally. Scanning the row containing the most significant bit (the top row in the illustration) gives the most significant bit–plane (MSB). Scanning the row containing the least significant bit (the bottom row in the illustration) gives the least significant bit–plane (referred to as MSB-2 in this example, or more generally, LSB). Next, each bit–plane is encoded into $(RUN,\ EOP)$ symbols. RUN gives the number of consecutive "0"s before a "1". EOP is set to 0 if there are some "1"s left in the bit–plane, if there is no "1" left in the bit plane then EOP is set to 1, as illustrated in Fig. 3. The $(RUN,\ EOP)$ symbols are finally variable–length coded.

We close this brief overview of MPEG–4 FGS encoding by noting that the MPEG–4 standard includes several refinements to the basic SNR FGS approach outlined above and also a temporal scalable FGS mode, which are beyond the scope of our study. (A streaming mechanism adapting the video by adding and dropping the SNR FGS and temporal FGS enhancement layers is studied in [54].) We also note that a Progressive FGS (PFGS) refinement has recently been proposed [11], [30], but not yet standardized. In contrast to MPEG–4 FGS, PFGS allows for partial motion compensation among the FGS bit–planes, while still achieving the fine granularity property. This motion compensation typically improves the coding efficiency, but lowers the error resilience of the enhancement layer [16].

III. Framework for Evaluating Streaming Mechanisms

In this section, we present a framework for evaluating streaming mechanisms for FGS–encoded video. This framework consists of (i) definitions of metrics that characterize the traffic and quality on the basis of individual video frames as well as on the basis of scenes (or more generally any arbitrary aggregation of video frames), and (ii) rate–distortion traces of a short clip as well as several long videos.

A. Notation

Throughout this report, we use the terms "images" and "video frames" interchangeably. We assume that the frame period (display time of one video frame) is constant and denote it by T seconds. Let N denote the number of frames in a given video and let n, $n = 1, \ldots, N$, index the individual video frames. Frame n is supposed to be decoded and displayed at the discrete instant $t = n \cdot T$. Suppose the base layer was encoded with fixed quantization scale, resulting in variable base layer frame sizes (as well as variable enhancement layer frame sizes). For our evaluation framework we assume that the transmission of each individual variable size frame is spread out equally over the frame period preceding the actual display frame period, i.e., the frame is transmitted at a constant bit rate over one frame period such that it arrives just in time for its display. (This evaluation framework can be adapted to streaming mechanisms that transmit frames ahead of time in a straightforward fashion.) More formally, let X_n^b denote the size of the base layer of frame n (in bit or byte).

Let $X^e_{n,comp}$ denote the size of the complete FGS enhancement layer of frame n, i.e., the enhancement layer without any cuts. The base layer is transmitted with constant bit rate $r^b_n = X^b_n/T$ during the period from $t = (n-1) \cdot T$ to $t = n \cdot T,\ n = 1, \dots, N$. Similarly, the complete enhancement layer would be streamed at the constant bit rate $C_{n,comp} = X^e_{n,comp}/T$ from $t = (n-1) \cdot T$ to $t = n \cdot T$. Now, note that, according to the FGS property, the FGS enhancement layer can be truncated anywhere before (or during) the transmission through the network. The remaining — actually received — part of the FGS enhancement layer is added to the reliably transmitted base layer and decoded. We refer to the part of the enhancement layer of a frame that is actually received and decoded as *enhancement layer subframe*. More formally, we introduce the following terminology. We say that the enhancement layer subframe is encoded at rate $C_n,\ 0 \leq C_n \leq C_{n,comp}$, when the first $C_n \cdot T$ bits of frame n are received and decoded together with the base layer. In other words, the enhancement layer subframe is said to be encoded with rate C_n when the last $(C_{n,comp} - C_n) \cdot T$ bits have been cut from the FGS enhancement layer and are not decoded.

For the scene based metrics the video is partitioned into consecutive scenes. Let S denote the total number of scenes in a given video. Let $s,\ s = 1, \dots, S$, denote the scene index and N_s the length (in number of images) of scene number s. (Note that $\sum_{s=1}^{S} N_s = N$.) All notations that relate to video scenes can be applied to any arbitrary sequence of successive frames (e.g., GoP). In the remainder of the report, we explicitly indicate when the notation relates to GoPs rather than to visual scenes.

B. Image-based Metrics

Let $Q_n(C),\ n = 1, \dots, N$, denote the quality of the nth decoded image, when the enhancement layer subframe is encoded with rate C; for ease of notation we write here and for all image related metrics C instead of C_n. (In our framework we consider a generic abstract quality metric, which could be the frame PSNR or some other metric. In Sec. III-D we explain how to use the PSNR (and MSE) as an instantiation of the abstract $Q_n(C)$ metric.) Let $Q^b_n = Q_n(0)$, denote the quality of the same image, when only the base layer is decoded. We define $Q^e_n(C) = Q_n(C) - Q^b_n$ as the improvement (increase) in quality which is achieved when decoding the enhancement layer subframe encoded with rate C together with the base layer of frame n.

The mean and sample variance of the image quality, are estimated as:

$$\begin{aligned} \bar{Q}(C) &= \frac{1}{N} \sum_{n=1}^{N} Q_n(C), && (1) \\ \sigma^2_Q(C) &= \frac{1}{N-1} \sum_{n=1}^{N} [Q_n(C) - \bar{Q}(C)]^2 = \frac{1}{N-1} \left\{ \sum_{n=1}^{N} [Q_n(C)]^2 - [\bar{Q}(C)]^2 \right\}. && (2) \end{aligned}$$

The coefficient of quality variation is given by:

$$CoV_Q(C) = \frac{\sigma_Q(C)}{\bar{Q}(C)}. \qquad (3)$$

The autocorrelation coefficient of the image qualities $\rho_Q(C, k)$ for lag $k,\ k = 1, \dots, N$, is estimated as:

$$\rho_Q(C, k) = \frac{1}{N-k} \sum_{n=1}^{N-k} \frac{[Q_n(C) - \bar{Q}(C)] \cdot [Q_{n+k}(C) - \bar{Q}(C)]}{\sigma^2_Q(C)}. \qquad (4)$$

Let $Q_{s,n}(C)$, $s = 1, \dots, S$, $n = 1, \dots, N_s$, denote the quality of the nth decoded image of scene s, when the enhancement layer subframe is encoded with rate C. Similar to $Q_n(C)$, we denote $Q_{s,n}(C) = Q^b_{s,n} + Q^e_{s,n}(C)$. The mean and sample variance of the qualities of the images within scene s are denoted by $\bar{Q}_s(C)$ and $\sigma^2_{Q_s}(C)$. They are estimated in the same way as the mean and sample variance of individual image quality over the entire video.

We denote the total size of image n by $X_n(C) = X^b_n + X^e_n(C)$, when the enhancement layer subframe is encoded with rate C, whereby $X^e_n(C) = C \cdot T$.

The key characterization of each FGS encoded frame is the rate–distortion curve of the FGS enhancement layer. This rate–distortion curve of a given frame n is a plot of the improvement in image quality Q^e_n as a function of the enhancement layer subframe bitrate C. This rate–distortion curve is very important for evaluating network streaming mechanisms for FGS encoded video. Suppose that for frame n the streaming mechanism was able to deliver the enhancement layer subframe at rate C. Then we can read off the corresponding improvement in quality as $Q^e_n(C)$ from the rate–distortion curve for video frame n. Together with the base layer quality Q^b_n we obtain the decoded image quality as $Q_n(C) = Q^b_n + Q^e_n(C)$.

In order to be able to compare streaming mechanisms at different aggregation levels, we monitor the maximum variation in quality between consecutive images within a given scene s, $s = 1, \dots, S$, when the enhancement layer subframes of all images in the considered scene are coded with rate C. We denote this *maximum variation in image quality* by $Var_s(C)$:

$$Var_s(C) = \max_{n=2,\dots,N_s} \{|Q_{s,n}(C) - Q_{s,n-1}(C)|\}. \tag{5}$$

We define the *average maximum variation in image quality* of a video with S scenes as

$$\overline{Var}(C) = \frac{1}{S} \sum_{s=1}^{S} Var_s(C). \tag{6}$$

We also define the *minimum value of the maximum quality variation* of a video with S scenes as

$$minVar(C) = \min_{1 \leq s \leq S} Var_s(C). \tag{7}$$

C. Scene–based Metrics

Typically, long videos feature many different scenes composed of successive images with similar visual characteristics. Following Saw [55], we define a *video scene* as a sequence of images between two scene changes, where a scene change is defined as any distinctive difference between two adjacent images. (This includes changes in motion as well as changes in the visual content.)

In this section we define metrics for studying the quality of long videos scene by scene. We first note that the mean image quality of a scene, $\bar{Q}_s(C)$ defined in (1), may not necessarily give an indication of the overall quality of the scene. This is because the quality of individual images does not measure temporal artifacts, such as mosquito noise (moving artifacts around edges) or drifts (moving propagation of prediction errors after transmission). In addition, high variations in quality between successive images within the same scene may decrease the overall perceptual quality of the scene. For example, a scene with alternating high and low quality images may have the same mean image quality as when the scene is rendered with medium but constant image quality, but the quality perceived by the user is likely to be much lower.

For these reasons we let $\Theta_s(C)$ denote the *overall quality* of video scene number s, $s = 1, \ldots, S$, when the enhancement layer subframes have been coded at rate C for all images of the scene. (This overall quality is again an abstract quality metric. In Section III-D we explain how to use the average MSE as an instantiation of this metric.) Similar to the measure of quality of the individual images, we define $\Theta_s(C) = \Theta_s^b + \Theta_s^e(C)$, where $\Theta_s^b = \Theta_s(0)$ denotes the overall quality of scene s when only the base layer is decoded, and $\Theta_s^e(C)$ the improvement in quality achieved by the enhancement layer subframes coded at rate C. We analyze the mean $\bar{\Theta}(C)$, sample variance $\sigma_\Theta^2(C)$, coefficient of variation $CoV_\Theta(C)$, the minimum to average ratio $\Theta_{\min}/\bar{\Theta}$, and the autocorrelation coefficients $\rho_\Theta(C, k)$ of the scene qualities. These metrics are estimated in analogous fashion to the corresponding image–based metrics.

Note that our measure for overall scene quality, $\Theta_s(C)$, does not account for differences in the length of the successive scenes. Our analysis with a measure that weighed the scene qualities proportionally to the scene length gave very similar results as the scene length independent metric $\Theta_s(C)$. We consider therefore the metric $\Theta_s(C)$ throughout this study. Moreover, it should be noted that the perception of the overall quality of a scene may not be linearly proportional to the length of the scene, but may also depend on other factors, such as the scene content (e.g., the quality of a high action scene may have higher importance than the quality of a very low action scene).

The rate–distortion characteristic of a given scene s is obtained by plotting the curve $\Theta_s(C)$, analogous to the rate–distortion curve of an individual image.

The mean and variance of the scenes' qualities give an overall indication of the perceived quality of the entire video. However, the variance of the scene quality does not capture the differences in quality between successive video scenes, which tend to cause a significant degradation of the perceived overall video quality. To capture these quality transitions between scenes, we introduce a new metric, called *average scene quality variation*, which we define as:

$$V(C) = \frac{1}{S-1} \sum_{s=2}^{S} |\Theta_s(C) - \Theta_{s-1}(C)|. \tag{8}$$

Also, we define the *maximum scene quality variation* between two consecutive scenes as:

$$V_{\max}(C) = \max_{2 \le s \le S} |\Theta_s(C) - \Theta_{s-1}(C)|. \tag{9}$$

Let $\bar{X}_s(C)$ denotes the mean size of the frames in scene s ($\bar{X}_s(C) = \frac{1}{N_s} \sum_{n=1}^{N_s} X_{s,n}$). The correlation coefficient between the mean frame size $\bar{X}_s(C)$ of a scene and the overall quality $\Theta_s(C)$ of a scene is estimated as:

$$\rho_{X,\Theta}(C) = \frac{1}{S-1} \sum_{s=1}^{S} \frac{(\bar{X}_s(C) - \bar{X}(C))(\Theta_s(C) - \bar{\Theta}(C))}{\sigma_X(C) \cdot \sigma_\Theta(C)}, \tag{10}$$

where $\bar{X}(C)$ denotes the mean of the successive mean frame sizes of all scenes in the video ($\bar{X}(C) = \sum_{s=1}^{S} \bar{X}_s(C)/S$). We denote the correlation coefficient between the base layer quality of a scene and the aggregate base and enhancement layers quality of a scene by $\rho_{\Theta^b,\Theta}(C)$. It is estimated the same way as $\rho_{X,\Theta}(C)$.

Finally, we monitor the length (in video frames) of the successive scenes N_s, $s = 1, \ldots, S$. We denote the mean and sample variance of N_s as $\bar{N} = N/S$ and σ_N^2.

D. MSE and PSNR Measures

The evaluation metrics defined in Sections III-B and III-C are general in that any specific quality metric can be used for the image quality $Q_n(C)$ and the overall scene quality $\Theta_s(C)$. In this section we explain how to use the Peak Signal to Noise Ratio (PSNR) (derived from the Mean Square Error (MSE)) as an instantiation of these general metrics. The choice of PSNR (MSE) is motivated by the recent Video Quality Expert Group (VQEG) report [56]. This report describes extensive experiments that compared several different objective quality measures with subjective quality evaluations (viewing and scoring by humans). It was found that none of the objective measures (some of them quite sophisticated and computationally demanding) performed better than the computationally very simple PSNR (MSE) in predicting (matching) the scores assigned by humans.

For video images of size $X \times Y$ pixels, the PSNR of the video sequence between images n_1 to n_2 is defined by [57]:

$$\text{PSNR}(n_1, n_2) = 10 \cdot \log \frac{M^2}{\text{MSE}(n_1, n_2)}, \tag{11}$$

where M is the maximum value of a pixel (255 for 8–bit grayscale images), and MSE(n_1, n_2) is defined as:

$$\text{MSE}(n_1, n_2) = \frac{1}{X \cdot Y \cdot (n_2 - n_1 + 1)} \sum_{n=n_1}^{n_2} \sum_{y=1}^{Y} \sum_{x=1}^{X} [I(x, y, n) - \tilde{I}(x, y, n)]^2, \tag{12}$$

where $I(x, y, n)$ and $\tilde{I}(x, y, n)$ are the gray–level pixel values of the original and decoded frame number n, respectively. The PSNR and MSE are well–defined only for luminance values, not for color [58]. Moreover, as noted in [56], the Human Visual System (HVS) is much more sensitive to the sharpness of the luminance component than that of the chrominance component. Therefore, we consider only the luminance PSNR.

To use the PSNR as an instantiation of the generic image quality $Q_n(C)$ and scene quality $\Theta_s(C)$, we set:

$$Q_n(C) = \text{PSNR}(n, n), \tag{13}$$

$$Q_{s,n}(C) = \text{PSNR}(T_s + n - 1, T_s + n - 1), \tag{14}$$

$$\Theta_s(C) = \text{PSNR}(T_s, T_{s+1} - 1), \tag{15}$$

where $T_s \in \{1, \ldots, N\}$ is the absolute frame number of the first frame of scene s, i.e., $T_s = 1 + \sum_{j=1}^{s-1} N_j$. Equation (15) assumes that all enhancement layer subframes within scene s are encoded with constant bitrate C. We note again that we use the MSE and PSNR as an instantiation of our general metrics and to fix ideas for our numerical experiments. Our general evaluation metrics defined in Sections III-B and III-C accommodate any quality metric [57], e.g., the ANSI metrics motion energy difference and edge energy difference [59] in a similar manner.

We close this section by noting that the above setting for scene quality $\Theta_s(C)$ uses the average of the MSEs of the individual images and then transforms this average MSE mathematically to give the PSNR (in dB). An alternative would be to set the scene quality to the arithmetic average of the PSNRs of the individual images (i.e., to $\bar{Q}_s(C)$). There is the following subtle difference between these two approaches to calculate the scene quality. The MSE of a given image is the arithmetic average of the distortion between the pixels of the decoded image and the original image (Eqn. (12) with $n_1 = n_2$). When we consider a sequence of video frames $(n_2 > n_1)$, the MSE of the video sequence is the arithmetic average of the MSEs for the individual images of the video (or equivalently, the average of the distortions for all pixels of all frames of the video sequence). The PSNR of the video sequence is then just a mathematical transformation of the MSE (see Eqn. (11)), which gives the overall quality of the video sequence in dB. On the other hand, the arithmetic average of the PSNRs of the individual images of a video sequence (each PSNR value obtained from the corresponding image MSE) gives the average of the quality for each image, rather than the overall quality of the video sequence. In practice both approaches give typically very close results (the difference is usually on the order of 0.1 dB). However, we think that the average MSE approach is more sound and more intuitive and use it throughout this study.

E. Generation of Traces and Limitations

In our experiments, we used the Microsoft MPEG–4 software encoder/decoder [60] with FGS functionality. We generated our traces according to the following methodology:

1) First, we encode the video using 2 different sets of quantization parameters for the base layer. This gives compressed base layer bitstreams of high quality (with quantization parameters $(4, 4, 4)$ for (I, P, B) frames) and low quality (with quantization parameters $(10, 14, 16)$), as well as the associated enhancement layer bitstreams. The Group of Pictures (GoP) structure of the base layer is set to IBBPBBPBBPBB. The frame period is $T = 1/30$ sec. throughout.
2) We segment the video into S successive scenes. This can be done based on the compressed base layer bitstream or the source video, according to the segmentation tool which is used. We obtain a file containing the image numbers delimiting the scenes (`scene-nb(`s`), last-image-nb(`$T_s + N_s - 1$`))`.
3) For each base layer quality, we cut the corresponding FGS enhancement layer at the increasing and equally spaced bitrates $C = 200,\ 400,\ 600,\ \ldots$ kbps.
4) For each tuple of compressed bitstreams (base layer quality, enhancement layer substream encoded at rate C), we compute the PSNR for each image after decoding, and then the PSNR for each scene.

Finally, for each base layer quality, we obtain the following traces:

- a file containing the base layer statistics for each image number (`image-nb(`n`), decoding-timestamp(`$n\cdot T$`), image-type, frame-size(`X_n^b`), PSNR-Y(`Q_n^b`), PSNR-U, PSNR-V`),
- a file containing the size of each enhancement layer bit–plane (up to 8 bit–planes) for each image number (`image-nb(`n`), size-of-BP1, ..., size-of-BP8`),
- a file, for each enhancement layer encoding rate C, containing the image quality (in PSNR) obtained after decoding the base layer and the truncated enhancement layer for all frames (`image-nb(`n`),`

`PSNR-Y(`$Q_n(C)$`), PSNR-U, PSNR-V`).

Note that videos are processed in the YUV format (Y is the luminance component, U and V are color components of an image).

1) Limitations: Due to a software limitation in the encoder/decoder, some PSNR results (particularly at some low enhancement layer bitrates) are incoherent (outliers). This has a minor impact for the short videos, because the trend of the rate–distortion curves for all individual images and video scenes is clear enough to estimate the quality that will be reached without considering the outliers. However, for the long videos, only the high quality base layer encoding gave valid results for most enhancement layer bitrates; thus, we only consider the high base layer quality for long videos.

Also, due to an encoder limitation, we had to encode separately two 30 minute sequences of our 1 hour videos and then concatenate the traces. For the video *News*, a few bidirectionally predicted frames at the end of the sequence are skipped at the encoder, so we repeated the last encoded frame until the original end of the sequence (this is visible on the base layer traces when the frame–type stays constant for some frames at the end of a 54000 image sequence). Since this only concerns 4 frames of the videos, we do not expect it to change the statistical results.

Because the automatic extraction of scene boundaries is still a subject of ongoing research (e.g., [61], [62], [63]), we restricted the segmentation of the video to the coarser segmentation into *shots* (also commonly referred to as *scene shots*). A shot is the sequence of video frames between two director's cuts. Since shot segmentation does not consider other significant changes in the motion or visual content, a shot may contain several distinct scenes (each in turn delimited by any distinctive difference between two adjacent frames). Nevertheless, distinct scene shots are still likely to have distinct visual characteristics, so we believe that performing shot–segmentation instead of a finer scene segmentation does not have a strong effect on the conclusions of our analysis in Section IV. A finer segmentation would only increase the total number of distinct video scenes, and increase the correlation between the qualities of the frames in a scene. Many commercial applications can now detect shot cuts with good efficiency. We used the MyFlix software [64], a MPEG–1 editing software which can find cuts directly in MPEG–1 compressed videos. (For the shot segmentation we encoded each video into MPEG–1, in addition to the FGS MPEG–4 encoding.)

2) Organization of the Web Site: All our traces, together with some statistics, can be found on our public web site. The site is organized as follows. For each long video encoded at high base layer quality, we have the following directories:

- **stats/**, which contains the traces of the bit–plane sizes, the boundaries of the scenes and the total (base layer and enhancement layer) coding rate by scene and GoP. It also features some overall statistics, such as statistics for scene length (S, $\bar{N}$ and σ_N), and the graphs of scene and GoP quality statistics as a function of the FGS rate ($\bar{\Theta}(C)$, $\sigma_\Theta(C)$, $V(C)$, $\rho_{\bar{X},\Theta}(C)$, $\rho_{\Theta^b,\Theta}(C)$) for $C = 0, 800, 1000, \cdots, 2000$ kbps. Note that for the graphs in this directory, we did not plot the statistics corresponding to the FGS cutting rates $C = 200, 400, 600$ kbps because of the phenomenon explained in section III-E.1.
- **srd/**, which contains the rate–distortion trace files for each scene ($\Theta_s(C)$).
- **q0/** . . . **q2000/**, which contain, for each FGS cutting rate $C = 0, \cdots, 2000$ kbps, the trace of individual image quality (n, $Q_n(C)$), the graphs of the autocorrelation in scene or GoP quality ($\rho_\Theta(C, k)$), the

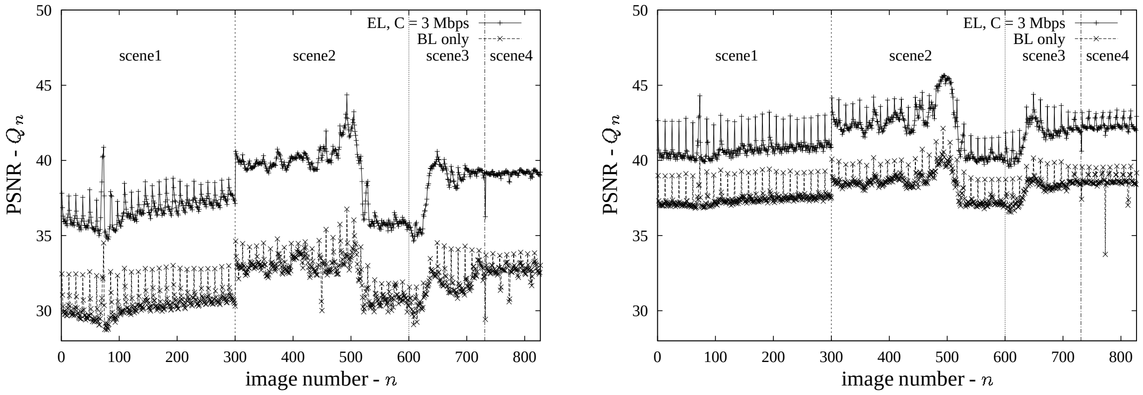
graph of the scene quality as a function of the scene number and the graph of the GoP quality as a function of the GoP number ($\Theta_s(C)$).

For the short video clip, which is described and analyzed in the next section, we have the following directories for both high quality base layer and low quality base layer versions of the video:

- **stats/**, which contains the trace of the bit–plane sizes, the boundaries of the scenes, and the graphs of image quality mean and variance as a function of the FGS rate for each scene ($\bar{Q}_s(C)$,$\sigma_{Q_s}(C)$).
- **srd/**, which contains the rate–distortion trace files for each image ($Q_n(C)$).
- **q0/**, which contains the trace of the base layer (n, X_n^b, $Q_n^b(C)$), and the graphs of quality and frame size as a function of the image number.

F. Analysis of Traces from a Short Clip

In this section, we present the analysis of a short video clip of 828 frames encoded in the CIF format. This clip was obtained by concatenating the well–known sequences *coastguard*, *foreman*, and *table* in this order. We segmented (by hand) the resulting clip into 4 scenes ($T_1 = 1$, $T_2 = 301$, $T_3 = 601$, $T_4 = 732$) corresponding to the 4 shots of the video (the *table* sequence is composed of 2 shots).

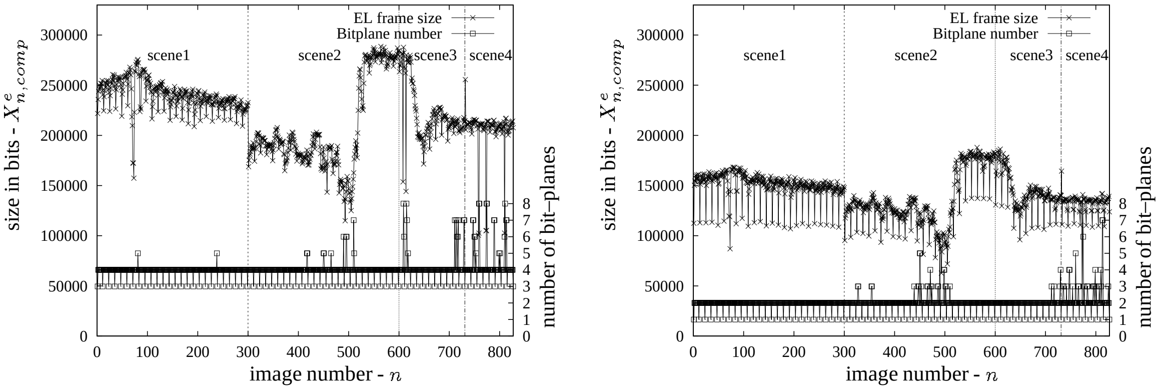


(a) low quality base layer

(b) high quality base layer

Fig. 4. Image PSNR Q_n (in dB) as a function of image number n for “Clip”

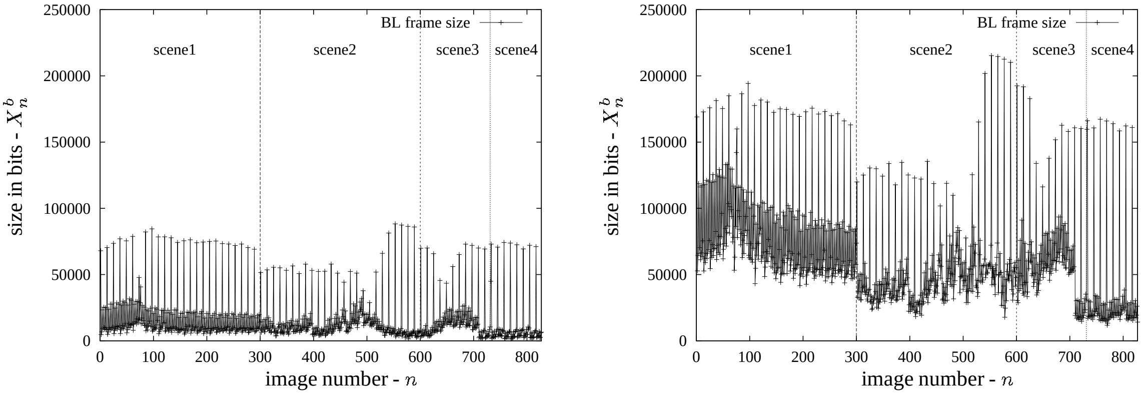
Figure 4 shows the quality of the successive images Q_n when only the base layer is decoded and when FGS enhancement layer subframes of rate $C = 3$ Mbps are added to the base layer. We make the following observations for both low and high base layer qualities. (i) First, the average image quality changes from one scene to the other for both base layer–only and EL–enhanced streaming. (ii) For a given scene, we see that for the base layer there are significant differences in the quality for successive images. Most of these differences are caused by the different types of base layer images (I, P, B) — the frames with the highest quality correspond to I–frames. When adding a part of the enhancement layer (at rate $C = 3$ Mbps in the figure), we see that these differences are typically still present, but may have changed in magnitude. This suggests to distinguish between the different types of images in order to study the rate–distortion characteristics of the FGS enhancement layer. (iii) We notice that scenes 2 and 3 feature high variations of image quality even for a given frame type within a given scene. Scene 2 corresponds to the *foreman* sequence in which the camera pans from the foreman's face to the building. A finer scene segmentation than shot–based segmentation would have segmented scene 2 into two different scenes, since the foreman's face and the building have different visual complexities.



(a) low quality base layer

(b) high quality base layer

Fig. 5. Size of complete enhancement layer frames $X^e_{n,comp}$ and number of bit–planes as a function of image number n for "Clip"



(a) low quality base layer

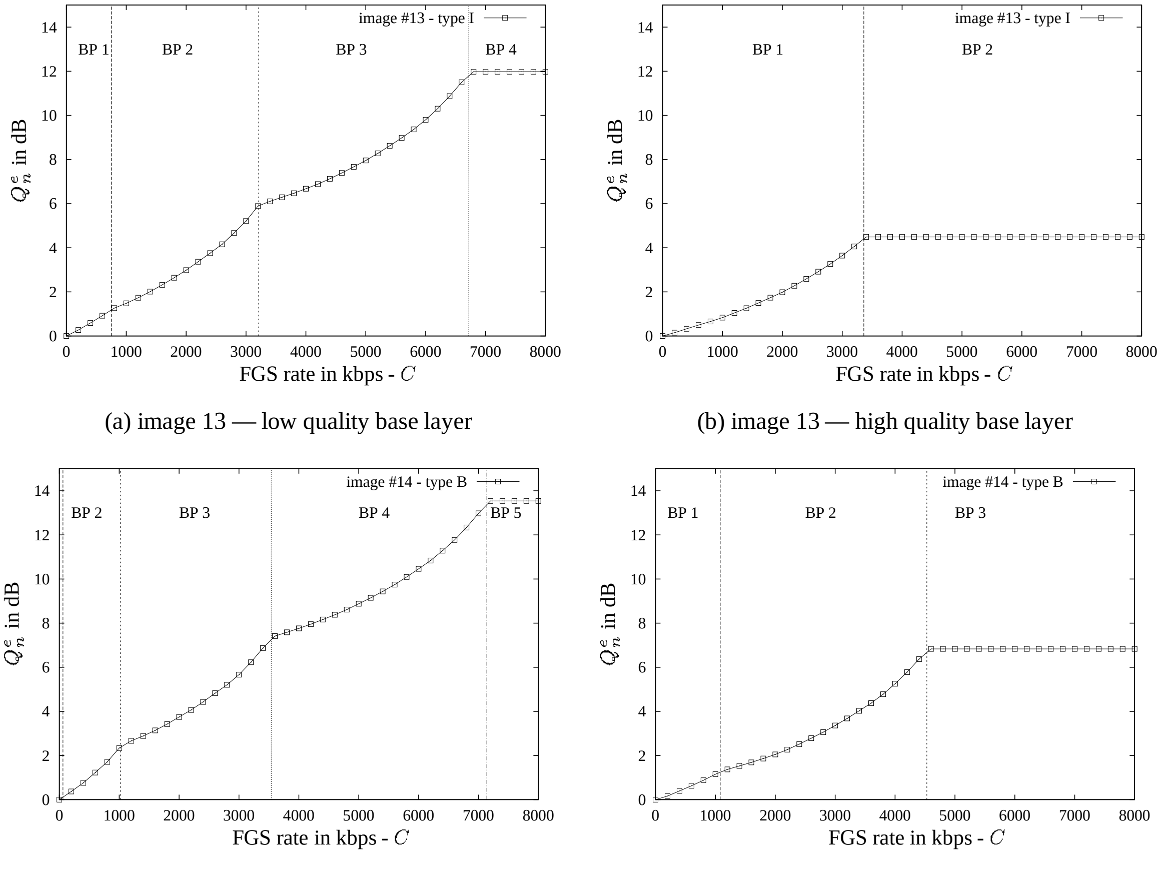
(b) high quality base layer

Fig. 6. Size of base layer images X^b_n as a function of image number n for "Clip"

Figure 5 shows the size of the complete enhancement layer, $X^e_{n,comp}$, and the number of bit–planes needed to code the enhancement layer of each image. First, we focus on a given scene. We observe that, in general, I images have fewer bit–planes than P or B images and that the total number of bits for the enhancement layer images is larger for P and B images than for I images. This is because I images have higher base layer quality. Therefore, fewer bit–planes and fewer bits are required to code the enhancement layer of I images. For the same reason, when comparing different high and low base layer qualities, we see that the enhancement layer corresponding to the high base layer quality needs, for most images, fewer bit–planes than the enhancement layer corresponding to the low base layer quality. For low base layer quality, the enhancement layer contains, for most images, 4 bit–planes, whereas, for the high base layer quality, it usually contains 2 bit–planes.

Next, we conduct comparisons across different scenes. Figure 6 shows the size of the base layer frames,

X_n^b. When comparing the average size of the enhancement layer frames for the individual scenes (Fig. 5) with the average size of the corresponding base layer frames (Fig. 6), we see that the larger the average base layer frame size of a scene the larger the average enhancement layer frame size of the scene. This can be explained by the different complexities of the scenes. For example, for a given base layer quality, we see that it requires more bits to code I images in scene 1 than in the first part of scene 2. This means that the complexity of scene 1 images is higher than the complexity of scene 2. Therefore, the average number of bits required to code the enhancement layer of scene 1 images is larger than for the first part of scene 2.



(a) image 13 — low quality base layer

(b) image 13 — high quality base layer

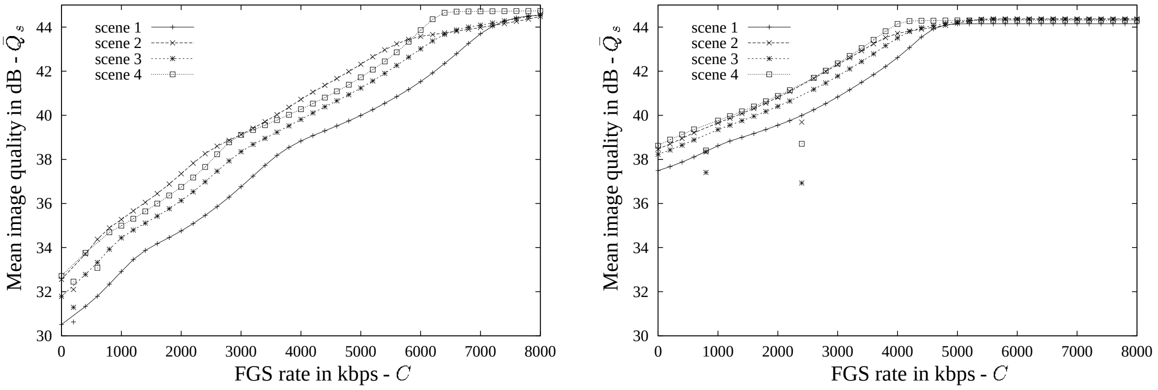
(c) image 14 — low quality base layer

(d) image 14 — high quality base layer

Fig. 7. Improvement in PSNR $Q_{1,13}^e$ and $Q_{1,14}^e$ as function of the FGS bitrate C for successive I and B images in scene 1 of “Clip”

In Figure 7 we plot the RD functions $Q_{1,13}^e(C)$ and $Q_{1,14}^e(C)$ (improvement in quality brought by the enhancement layer as a function of the encoding rate of the FGS enhancement layer) for different types of images within the same GoP. These plots give rise to a number of interesting observations, which in turn have important implications for FGS video streaming and its evaluation. First, we observe that the rate–distortion curves are different for each bit–plane. The rate–distortion curves of the lower (more significant) bit–planes tend to be almost linear, while the higher (less significant) bit planes are clearly non–linear. (Note that the most significant bit–plane (BP1) for image 14 with low quality base layer has a very small size.) More specifically, the rate–distortion curves of the higher bit–planes tend to be convex. In other words, the closer we get to the end of a given bit–plane, the larger the improvement in quality for a fixed amount of additional bandwidth. This appears to be due to the bit–plane headers. Indeed, the more bits are kept in a given bit–plane after truncation, the smaller the share of the bit–plane header in the total data for this bit–plane. An implication of this phenomenon for the design of streaming mechanisms is that it may be worthwhile to prioritize the enhancement layer cutting toward the end of the bitplanes.

Recall that the plots in Fig. 7 are obtained by cutting the FGS enhancement layer every 200 kbps. We observe from the plots here that a piecewise linear approximation of the curve using the 200 kbps spaced sample points gives an accurate characterization of the rate–distortion curve. We also observe that approximating the rate distortion–curves of individual bit–planes or the entire rate–distortion curve up to the point of saturation in quality (reached for instance with bitplane 4 in frame 1,13 of low quality base layer) by straight lines (one for each bitplane) or one straight line for the entire curve gives rise to significant errors which are typically in the range from 0.5 – 0.75 dB. It is therefore recommended to employ a piecewise linear approximation based on the 200 kbps spaced sample points. An interesting avenue for future work is to fit analytical functions to our empirically measured rate–distortion curves.

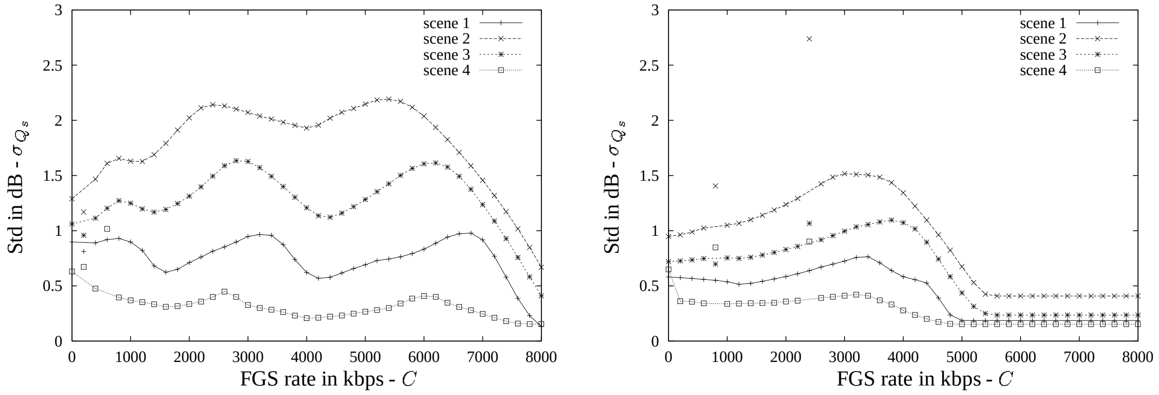


(a) low quality base layer

(b) high quality base layer

Fig. 8. Average image quality by scene $\bar{Q}_s$ as a function of the FGS enhancement layer bitrate C for “Clip”

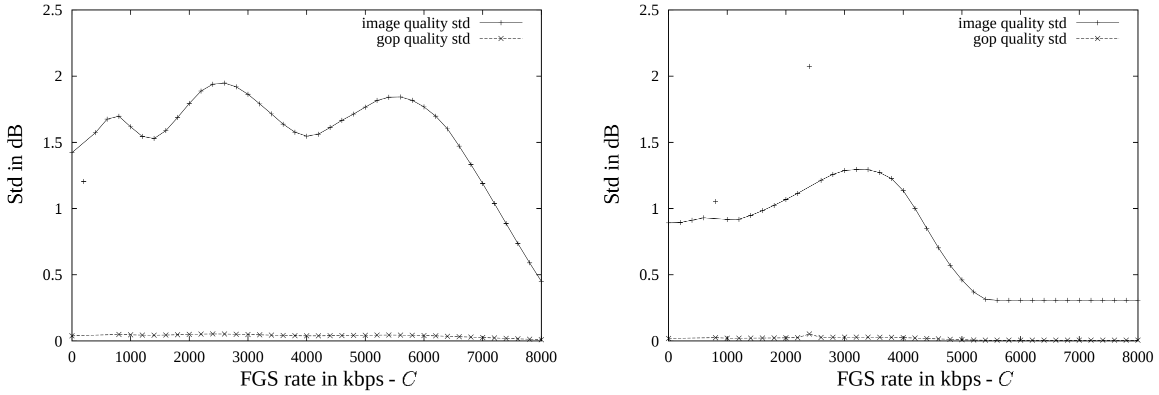
So far we have considered the rate–distortion curves of individual frames. We now aggregate the frames into scenes and study the rate–distortion characteristics of the individual scenes. Figure 8 shows the average image quality (from base plus enhancement layer) of the individual scenes in the “Clip” as a function of the FGS enhancement layer rate. (The outliers at low FGS bitrates are due to the software limitation discussed in Section III-E.1.) We observe that the scenes differ in their rate–distortion characteristics. For the low quality base layer version, the PSNR quality of scene 1 (*coastguard*) is about 2 dB lower than the PSNR quality of scene 2 (*foreman*) for almost the entire range of enhancement layer rates. This quality difference falls to around 1 dB for the high quality base layer video. This appears to be due to the higher level of motion in *coastguard*. Encoding this motion requires more bits with MPEG–4 FGS, which has no motion compensation in the enhancement layer. Overall, the results indicate that it is prudent to (i) analyze FGS encoded video on a scene by scene basis (which we do in the next section for long video with many scenes), and (ii) to take the characteristics of the individual scenes into consideration when streaming FGS video (which we examine in some more detail in Section IV).



(a) low quality base layer

(b) high quality base layer

Fig. 9. Standard deviation of image quality σ_{Q_s} for individual scenes as a function of the FGS bitrate C for "Clip"



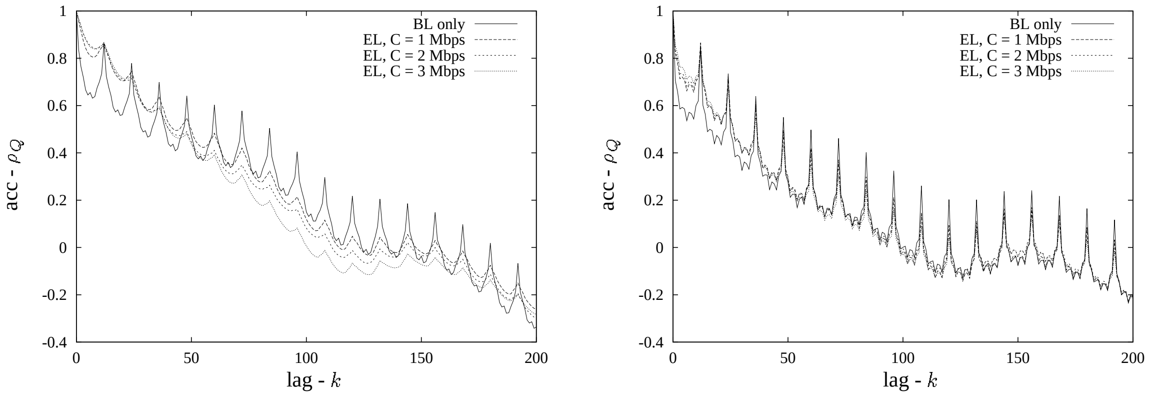
(a) low quality base layer

(b) high quality base layer

Fig. 10. Standard deviation of image quality σ_Q and GoP quality σ_Θ for "Clip"

As noted in the introduction, the perceived video quality depends on the qualities of the individual frames as well as the variations in quality between successive frames. To examine the quality variations, we plot in Figure 9 the standard deviation of the image quality σ_{Q_s} for the different scenes. For both base layer qualities, we observe that overall scene 2 (*foreman*) is the scene with the largest variance. This is due to the change of the visual complexity within the scene as the camera pans from the foreman's face to the building behind him. We also observe that for a given scene, the variance in quality can change considerably with the FGS enhancement layer rate. To examine the cause for these relatively large and varying standard deviations, we plot in Figure 10 the standard deviation of both image quality σ_Q and GoP quality σ_Θ for the entire video clip. We see that the standard deviation of the GoP quality is negligible compared to the standard deviation of the image quality. This indicates that most of the variations in quality are due to variations in image quality between the different types of images (I, P, and B) within a given GoP. Thus, it is, as already noted above, reasonable to take the frame type into consideration in the streaming.

To take a yet closer look at the quality variations, we plot in Figure 11 the autocorrelation function ρ_Q of the image quality for the base layer and the FGS enhancement layer coded at rates $C = 1$, 2 and 3 Mbps.



(a) low quality base layer

(b) high quality base layer

Fig. 11. Autocorrelation coefficient of image quality ρ_Q for "Clip"

We observe periodic spikes which correspond to the GoP pattern. We verify that, at small lags there are high correlations (i.e., relatively smooth transitions) in quality for the different types of images, especially for high FGS enhancement layer rates. This means that the higher FGS enhancement layer rate smoothes the difference in quality between near images. Indeed, for the same number of FGS enhancement layer bits added to the base layer, the gain in quality is different for consecutive I, P, and B frames. In general, the gain in quality for I frames is smaller than the gain in quality for P or B frames: as indicated earlier, the base layer has higher quality for I frames; so the enhancement layer bits provide higher (less visible) spatial frequencies for the I frames than for the P and B frames.

G. Analysis of Traces from Long Videos

In this section we analyze the traces of long videos. All videos have been captured and encoded in QCIF format (176×144 pixels), except for the movie *Silence of the Lambs* which has been captured and encoded in CIF format (352×288 pixels). All videos have been encoded with high base layer quality. The image based metrics defined in Section III-B and studied in Section III-F apply in analogous fashion to the long videos and lead to similar insights as found in Section III-F. In contrast to the short "Clip", the long videos contain many different scenes and thus allow for a statistically meaningful analysis at the scene level, which we give an overview of in this section.

Table I gives the scene shot length characteristics of the long videos along with the elementary base layer traffic statistics. We observe that the scene lengths differ significantly among the different videos. *Toy Story* has shortest scenes, with an average scene length of just about 2.9 seconds (= 88 frames/30 frames per second). Comparing Oprah with commercials (*Oprah* $+$ *comm*) with *Oprah* (same video with commercials removed), we observe that the commercials significantly reduce the average scene length and increase the variability of the scene length in the videos. The *lecture* video, a recording of a class by Prof. M. Reisslein at ASU has by far the longest average scene length, with the camera pointing to the writing pad or blackboard for extended periods of time. The scene length can have a significant impact on the required resources (e.g., client buffer) and the complexity of streaming mechanisms that adapt on a scene by scene basis. (In Section IV we compare scene by scene based streaming with other streaming mechanisms from a video

TABLE I

Scene shot length and base layer traffic characteristics for the long videos

	run time	S	N	CoV_N	$N_{\max}/N$	$\bar{r}_b$ (Mbps)	X^b (bits)	CoV_{X^b}	$X^b_{\max}/X^b$
The Firm	1h	890	121	0.94	9.36	0.65	21765	0.65	6.52
Oprah+com	1h	621	173	2.46	39.70	2.73	91129	0.14	1.94
Oprah	38mn	320	215	1.83	23.86	1.69	56200	0.19	2.33
News	1h	399	270	1.67	9.72	0.74	24645	0.54	5.30
Star Wars	1h	984	109	1.53	19.28	0.49	16363	0.65	6.97
Silence CIF	30mn	184	292	0.96	6.89	1.74	57989	0.72	7.85
Toy Story	1h	1225	88	0.95	10.74	1.08	36141	0.49	5.72
Football	1h	876	123	2.34	31.47	0.97	32374	0.53	3.90
Lecture	49mn	16	5457	1.62	6.18	1.54	51504	0.29	2.72

TABLE II

Scene quality statistics of long videos for the base layer and FGS enhancement layer substream bitrates $C = 1$ and 2 Mbps

	BL only			$C = 1$ Mbps			$C = 2$ Mbps		
	Θ (dB)	CoV_Θ	$\Theta_{\min}/\Theta$	Θ (dB)	CoV_Θ	$\Theta_{\min}/\Theta$	Θ (dB)	CoV_Θ	$\Theta_{\min}/\Theta$
The Firm	36.76	0.013	0.97	40.10	0.017	0.88	43.70	0.003	0.99
Oprah+com	35.71	0.015	0.99	38.24	0.013	0.99	42.30	0.010	0.99
Oprah	35.38	0.003	1.00	38.18	0.003	1.00	42.84	0.007	0.99
News	36.66	0.018	0.97	39.65	0.027	0.96	43.76	0.021	0.98
Star Wars	37.48	0.025	0.95	41.14	0.031	0.94	43.83	0.013	0.99
Silence CIF	37.88	0.015	0.96	NA	NA	NA	39.70	0.020	0.96
Toy Story	36.54	0.021	0.97	39.57	0.029	0.97	43.95	0.013	0.97
Football	37.42	0.034	0.95	40.69	0.041	0.94	43.97	0.018	0.99
Lecture	35.54	0.001	1.00	38.48	0.002	1.00	43.64	0.007	0.99

quality perspective.)

The base layer traffic statistics in Table I are quite typical for encodings with a fixed quantization scales (4,4,4). We include these statistics here for completeness and refer the interested reader to [50] for a detailed study of these types of traffic traces.

More relevant for the study of FGS–encoded video are the average scene quality statistics in Table II. Table II indicates that the average scene PSNR is very different from one video to the other. In particular, while *Oprah with commercials* and *Oprah* have the highest base layer encoding rates (see Table I), the average overall PSNR quality achieved for the base layer for both videos is low compared to the average PSNR quality achieved by the other videos. This appears to be due to the high motion movie trailers featured in the show as well as noise from the TV recording, both of which require many bits for encoding. (We note that these objective PSNR qualities do not necessarily reflect the subjective video quality, but the PSNR is as good an objective indication of video quality as any other more sophisticated objective metric [56].) We observe that for a given video, each additional 1 Mbps of enhancement layer increases the average PSNR by roughly 3–4 dB. (The relatively large bitrate and low PSNR for the *Lecture* video are due to the relatively noisy copy (of a copy of a copy) of the master tape.) We also observe from Table II that the standard deviation of the scene qualities is quite small and the normalized minimum scene quality $\Theta_{\min}/\bar{\Theta}$ is very close to 1. This is one reason why we defined the average scene quality variation (8) and maximum scene quality variation (9),

TABLE III

Average scene quality variation V (in dB) and maximum scene quality variation $V_{\max}$ (in dB) of long videos for the base layer and FGS bitrates $C = 1$ and 2 Mbps

	base layer only		$C = 1$ Mbps		$C = 2$ Mbps	
	V	$V_{\max}$	V	$V_{\max}$	V	$V_{\max}$
The Firm	0.06	1.83	0.16	2.37	0.00	1.26
Oprah+com	0.04	12.15	0.05	11.31	0.03	7.32
Oprah	0.00	0.36	0.00	0.42	0.00	1.13
News	0.11	3.15	0.29	3.17	0.12	2.36
Star Wars	0.29	8.25	0.57	8.83	0.13	6.28
Silence CIF	0.05	1.42	NA	NA	0.25	4.45
Toy Story	0.19	9.77	0.40	11.25	0.12	6.23
Football	0.51	9.79	0.72	10.12	0.19	6.36
Lecture	0.00	0.14	0.00	0.20	0.00	0.64

TABLE IV

Scene-based correlation statistics of long videos for the base layer and FGS bitrates $C = 1$ and 2 Mbps

	base layer only		$C = 1$ Mbps		$C = 2$ Mbps	
	$\rho_{X,\Theta}$	$\rho_{\Theta^b,\Theta}$	$\rho_{X,\Theta}$	$\rho_{\Theta^b,\Theta}$	$\rho_{X,\Theta}$	$\rho_{\Theta^b,\Theta}$
The Firm	-0.71	1.00	0.00	0.92	0.52	-0.18
Oprah+com	-0.20	1.00	0.01	0.99	0.07	0.83
Oprah	0.42	1.00	0.00	084	-0.04	0.48
News	-0.66	1.00	0.00	0.83	0.25	0.25
Star Wars	-0.47	1.00	-0.02	0.97	0.51	0.67
Silence CIF	-0.80	1.00	NA	NA	0.00	0.53
Toy Story	-0.39	1.00	-0.01	0.98	0.16	0.90
Football	-0.54	1.00	-0.02	0.97	0.33	0.81
Lecture	-0.14	1.00	0.00	0.52	-0.11	-0.17

which focus more on the quality change from one scene to the next (and which we will examine shortly). The other point to keep in mind is that these results are obtained for fixed settings of the FGS enhancement layer rate C. When streaming over a real network, the available bandwidth is typically variable and the streaming mechanism can exploit the fine granularity property of the FGS enhancement layer to adapt to the available bandwidth, i.e., the enhancement layer rate C will become a function of time. The challenge for the streaming mechanism is to adapt the enhancement layer rate C so as to minimize CoV_Θ and maximize $\Theta_{\min}/\bar{\Theta}$ while staying within the available resources (bandwidth, buffer, start–up latency, etc.). The CoV_Θ and $\Theta_{\min}/\bar{\Theta}$ reported in Table II for constant C are thus useful reference values for evaluating streaming mechanisms.

In Table III, we first observe that *Oprah* has the smallest average scene quality variation V at all FGS rates, whereas the *Football* video has the largest average scene quality variation for the base layer and $C = 1$ Mbps. For most videos, V and $V_{\max}$ are both minimum at $C = 2$ Mbps. We see from $V_{\max}$ that the difference in quality between successive scenes can be as high as 12 dB and is typically larger than 2 dB at all FGS rates for most videos. This indicates that there are quite significant variations in quality between some of the successive video scenes, which in turn may very visibly affect the video quality.

Figures 12 and 13 give, for *The Firm* and *News* respectively, the scene quality as a function of the scene number ($\Theta_s(C)$) for the base layer and FGS cutting rates $C = 1$ and 2 Mbps, as well as the average encoding bitrates for the base layer and the base layer plus complete enhancement layer. The plots illustrate the quite significant variations in scene quality for an enhancement layer rate of C = 1 Mbps. For the base layer the quality variations are less pronounced and for $C = 2$ Mbps, the quality is almost constant for most scenes. With C = 2 Mbps we may have come close to the maximum encoding rate for most scenes (see plot on right), i.e., most scenes are encoded at close to maximum achievable quality.

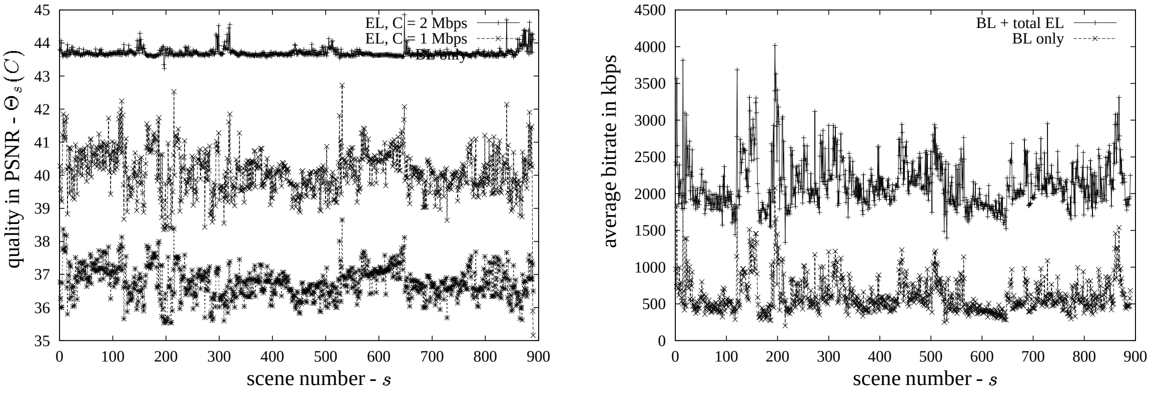


Fig. 12.   Scene PSNR $\Theta_s(C)$ (left) and average encoding bitrate $\bar{X}_s/T$ (right) as a function of scene number s for *The Firm*

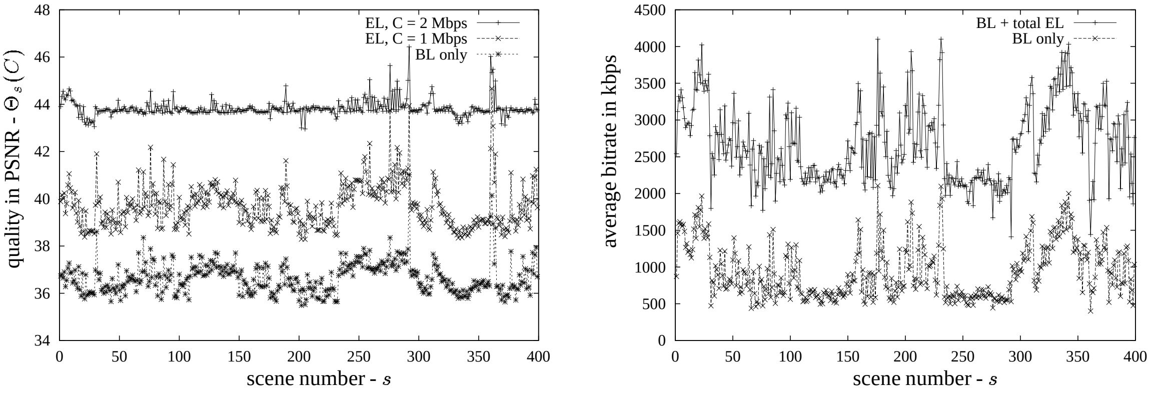
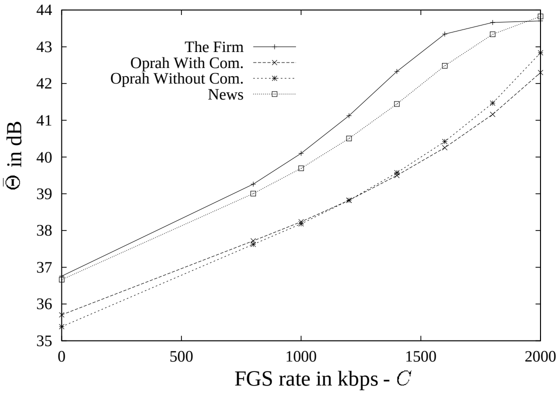


Fig. 13.   Scene PSNR $\Theta_s(C)$ (left) and average encoding bitrate $\bar{X}_s/T$ (right) as a function of scene number s for *News*

Figure 14 shows the average scene quality for *The Firm*, *Oprah*, *Oprah with commercials* and *News*, as a function of the FGS rate ($\bar{\Theta}(C)$). We observe that the slope of the quality increase with increasing FGS enhancement layer rate is about the same for all considered videos. We also observe that there is a difference of around 1 dB between the average base layer quality for *The Firm* or *News* and the average base layer quality for *Oprah* or *Oprah with commercials*; this difference roughly remains constant at all FGS rates. This indicates that the average quality achieved by a video at all FGS rates strongly depends on the visual content of the videos and on the average quality of the base layer. This is confirmed in Figure 16 which shows the coefficient of scene correlation between the base layer and the aggregate base and enhancement layer quality



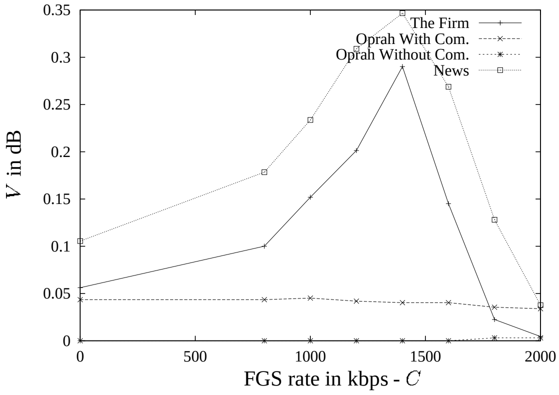
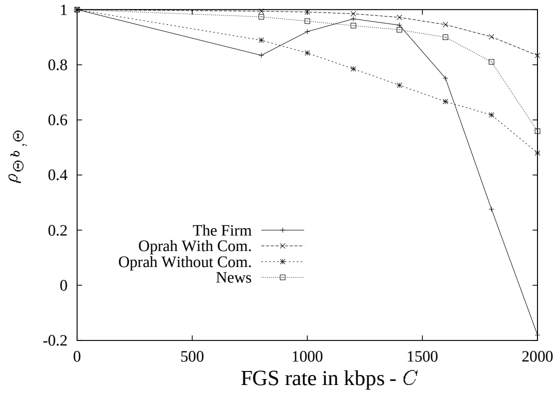


Fig. 14. Average scene quality $\bar{\Theta}$ as a function of the FGS enhancement layer rate C

Fig. 15. Average scene quality variability V as a function of the FGS enhancement layer rate C

as a function of the FGS rate ($\rho_{\Theta^b,\Theta}(C)$). The correlation decreases slightly with the FGS rate but stays high at all rates (see Table IV for complete statistics for all videos).

Figure 15 shows the average scene quality variation as a function of the FGS rate ($V(C)$). As we see, the difference in quality between the successive scenes first increases with the FGS rate for *The Firm* and *News*. This is probably because, according to the performance of the VBR base layer encoding, some scenes can achieve maximum quality with a small number of enhancement layer bits (low complexity scenes), while other scenes require a higher number of bits to achieve maximum quality (high complexity scenes). At high FGS rates the variability starts to decrease because all scenes tend to reach the maximum quality (as confirmed in Table III for most videos). For *Oprah* and *Oprah with commercials*, the variability stays very low at all FGS rates, which is mainly due to the fact that the VBR–base layer encoder has been able to smooth the differences in scene quality quite well.



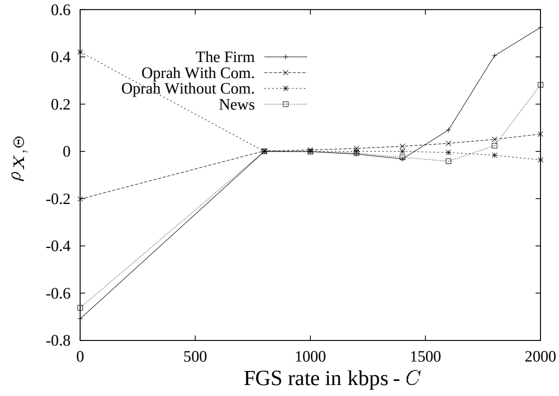


Fig. 16. Coefficient of correlation between scene base layer quality and scene overall quality $\rho_{\Theta^b,\Theta}$, as a function of the FGS bitrate C

Fig. 17. Coefficient of correlation between average size and quality of scenes $\rho_{X,\Theta}$, as a function of the FGS bitrate C

Figure 17 shows the coefficient of correlation between the average size and the quality of the scenes ($\rho_{X,\Theta}(C)$). Except for *Oprah*, the coefficient of correlation is negative for the base layer; this appears to be due to the base layer encoder allocating more bits to the more complex scenes. Then, as shown in Table IV, the coefficient of correlation globally increases with the FGS rate and becomes positive for most videos.

Indeed, for high base layer quality encodings, most of the complexity of the diverse scenes appears to have been absorbed by the VBR base layer. The *Oprah* video is a special case: the coefficient of correlation is already positive for the base layer and then decreases with the FGS rate. In this case, the diversity of scene complexity seems to have been almost totally absorbed by the base layer, as we have mentioned for Figure 15. For all videos, a detailed video content based analysis may shed more light on this correlation.

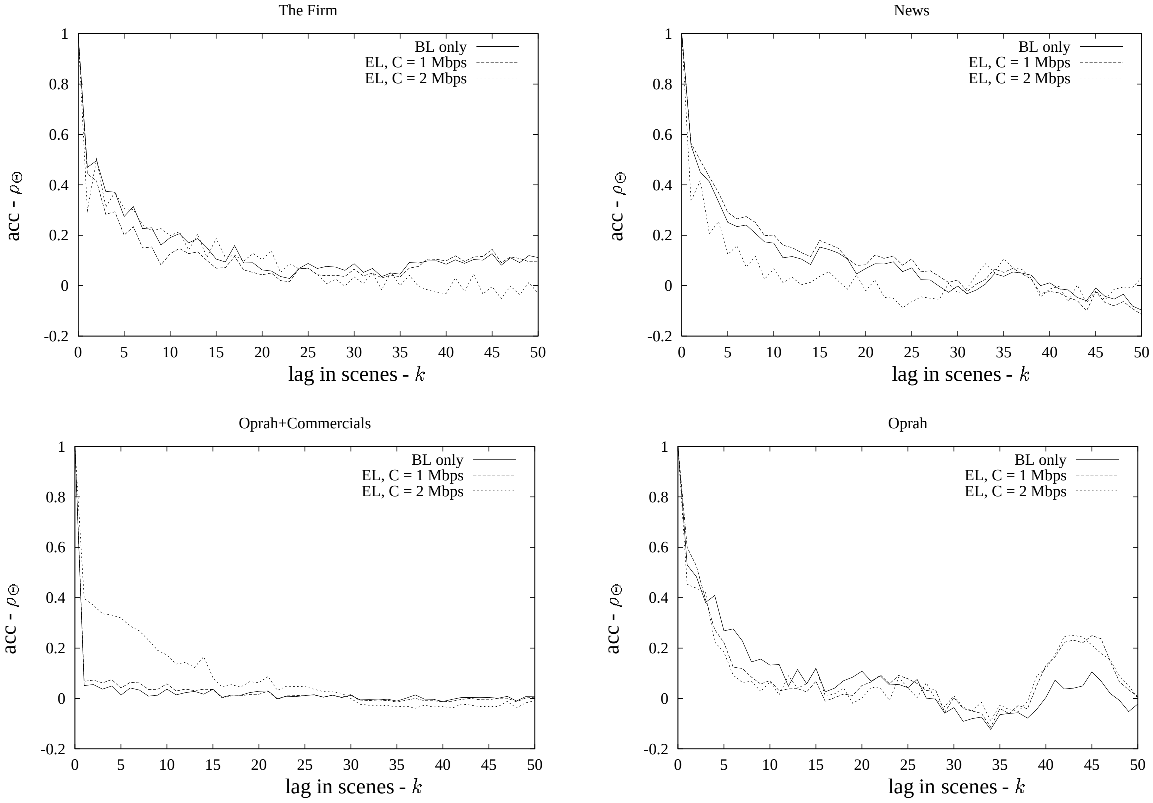


Fig. 18. Autocorrelation in scene quality ρ_Θ for videos encoded with high quality base layer

Finally, Figure 18 shows, for each video, the autocorrelation in scene quality ρ_Θ for the base layer and FGS rates $C = 0$, 1, and 2 Mbps. For the four videos, we observe that the autocorrelation functions drop off quite rapidly for a lag of a few scene shots, indicating that there is a tendency of abrupt changes in quality from one scene to the next. Also, for a given video, the autocorrelation function for the aggregate base and enhancement layers follows closely the autocorrelation function for the base layer only, except for *Oprah with commercials* at $C = 2$ Mbps. The difference in autocorrelation at low lags between *Oprah* and *Oprah with commercials* can be explained by the higher diversity of successive scene types when adding commercials.

IV. Comparison of Streaming Optimization at Different Aggregation Levels

In this section we apply our evaluation framework to compare the rate–distortion optimized streaming of FGS–encoded video at different levels of image aggregation.

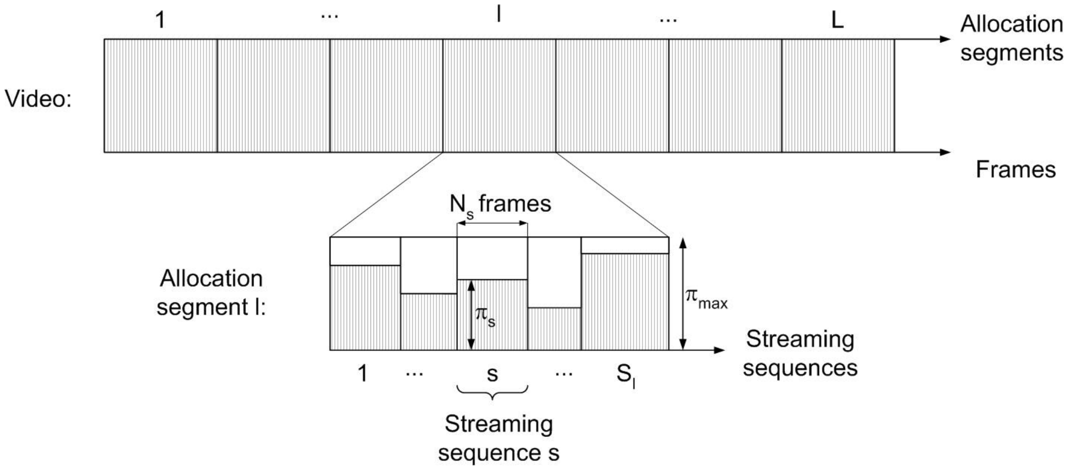


Fig. 19. The video is partitioned into L allocation segments, each consisting of S_l streaming sequences.

A. Problem Formulation

We suppose that the transmission of the base layer is made reliable, and we focus on the streaming of the enhancement layer. When streaming video over the best–effort Internet, the available bandwidth typically fluctuates over many time–scales. However, for non real–time applications such as streaming stored video, the user can usually tolerate an initial build–up delay, during which some initial part of the video is prefetched into the client before the start of the playback. Maintaining a sufficient playback delay throughout the rendering allows the application to accommodate future bandwidth variations (see for instance [17], [18]).

To account for bandwidth variability, we model bandwidth constraints and client buffering resources as follows. As shown on Figure 19, the video is partitioned into L *allocation segments*, with each allocation segment l containing the same number of frames $\lfloor N/L \rfloor$. While the server is streaming the video, for each allocation segment l, the server assigns a maximum bandwidth budget $B_{\max} = C_{\max} \cdot \lfloor N/L \rfloor \cdot T$ bits to be allocated across all the frames in the segment, where the maximum average bit rate $C_{\max}$ varies from one segment to the next. The values for $C_{\max}$ are determined by a coarse–grain streaming strategy, such as those given in [17], [37]. In this section, we focus on the fine–grain streaming strategy, namely, the allocation of the bandwidth budget to the individual frames within a segment. In our experiments, we use allocation segments consisting of 1000 frames, which correspond to about 30 seconds of a 30 frame/sec video.

Due to the client buffering, the server has great flexibility in allocating the given bandwidth budget to the frames within a segment. As discussed earlier, given the rate–distortion functions of all images, the server can optimize the streaming within the segment by allocating bits from the bandwidth budget to the individual frames so as to maximize the video quality. Alternatively, the server can group several consecutive images of an allocation segment into sub–segments, referred to as *streaming sequences*, and perform rate–distortion optimization on the granularity of streaming sequences. In this case, each frame in a streaming sequence (that is sub–segment) is allocated the same number of bits. We denote by S_l the number of streaming sequences in a given allocation segment l, $l = 1, \ldots, L$.

We consider four aggregation cases for streaming sequences:

- images — each image from the current allocation segment forms a distinct streaming sequence ($S_l = \lfloor N/L \rfloor$).
- GoPs — we group all images from the same GoP into one streaming sequence. In this case, the number of streaming sequences in allocation segment l is equal to the number of distinct GoPs in the allocation segment ($S_l = \lfloor N/(12 \cdot L) \rfloor$ with the 12 image GoP used in this study.
- scenes — we group all images from the same video scene into one streaming sequence. In this case $S_l = S_l^{scene}$, where S_l^{scene} denotes the number of distinct scenes in allocation segment l, according to the initial segmentation of the video (shot–based segmentation in this study).
- constant — allocation segment l is divided into $S_l^{const} = S_l^{scene}$ streaming sequences, each containing the same number of frames. Consequently, each streaming sequence contains a number of frames equal to the average scene length of the allocation segment.
- total — all the images from allocation segment l form one streaming sequence ($S_l = 1$).

In the following, we focus on the streaming of a particular allocation segment l. In order to simplify the notation, we remove the index l from all notations whenever there is no ambiguity. Let S be the number of streaming sequences in the current allocation segment. Let N_s be the number of frames in streaming sequence s, $s = 1, \ldots, S$ (see Figure 19). For a given allowed average rate $C_{\max}$, let π_s denote the number of bits allocated to each of the N_s images in streaming sequence s. Define $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_S)$ as the streaming policy for the allocation segment. We denote by $\pi_{\max} = C_{\max}/T$ the maximum number of enhancement layer bits that can be allocated to any image of the video.

Extending the scene quality metric defined in Section III-C to allocation segments, we define $\Theta(\boldsymbol{\pi})$ as the overall quality of the current allocation segment under the streaming policy $\boldsymbol{\pi}$. We denote $D(\boldsymbol{\pi})$ for the corresponding total distortion and $B(\boldsymbol{\pi})$ for the total number of bits to stream under this policy. As explained in Section III-D, the total distortion of a sequence of successive frames is measured in terms of the average MSE, obtained by averaging the individual frames' MSEs. The overall quality of a sequence is measured in terms of the PSNR and computed directly from the average MSE of the sequence. We denote by $d_{s,n}(\pi)$ the distortion (in terms of MSE) of image n, $n = 1, \ldots, N_s$, of streaming sequence s, when its enhancement layer subframes are encoded with π bits. We denote by $D_s(\pi) = \frac{1}{N_s}\sum_{i=1}^{N_s} d_{s,i}(\pi)$, the total distortion of streaming sequence s when all enhancement layer subframes contain π bits. With these definitions we can formulate the streaming optimization problem as follows:

For the current allocation segment, a given bandwidth constraint $C_{\max}$, and a given aggregation case, the optimization procedure at the server consists of finding the policy $\boldsymbol{\pi}^* = (\pi_1^*, \ldots, \pi_S^*)$ that optimizes:

$$\text{minimize } D(\boldsymbol{\pi}) = \sum_{s=1}^{S}\left(\frac{1}{N_s}\sum_{i=1}^{N_s} d_{s,i}(\pi_s)\right)$$
$$\text{subject to: } B(\boldsymbol{\pi}) = N_1 \cdot \pi_1 + \cdots + N_S \cdot \pi_S \leq B_{\max},$$
$$\pi_s \leq \pi_{\max}, \quad s = 1, \ldots, S.$$

We denote $D^* = D(\boldsymbol{\pi}^*)$ (respectively Θ^*) for the minimum total distortion (maximum overall quality) achieved for the current allocation segment. Our problem is a resource allocation problem, which can be solved by dynamic programming [65]. Dynamic programming is a set of techniques that are used to solve various decision problems. In a typical decision problem, the system transitions from state to state, according to the decision taken for each state. Each transition is associated with a profit. The problem is to find the optimal decisions from the starting state to the ending state of the system, i.e., the decisions that maximize the total profit, or in our context minimize the average distortion.

The most popular technique to solve such an optimization problem is called *recursive fixing*. Recursive fixing recursively evaluates the optimal decisions from the ending state to the starting state of the system. This is similar to the well–known Dijkstra algorithm which is used to solve shortest–path problems. Another popular technique is marginal analysis. Basically, marginal analysis starts with a feasible policy and increments the policy that gives the best profit. In our case, the profit corresponds to the distortion associated with each policy. Marginal analysis is computationally less demanding than recursive fixing. However it requires the profit function to be concave. As we observed in Section III-F, the rate-distortion functions of the enhancement layer consist of convex segments. Therefore, we need to use the computationally more expensive recursive fixing.

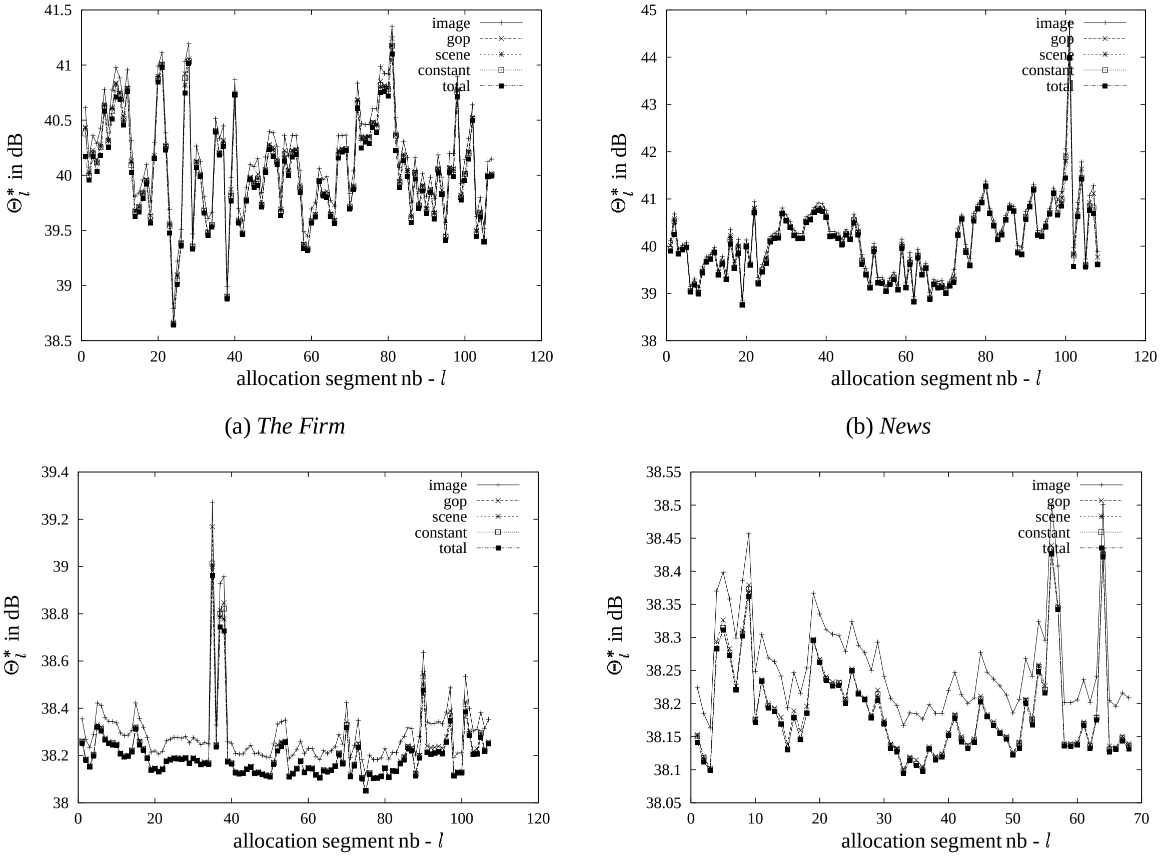
As we have observed in Figure 7 the rate–distortion curves can not easily be modeled by a simple function. Therefore, we implemented recursive fixing to resolve our dynamic programming problem by sampling the possible values of enhancement layer bits per image in steps of 833 bytes ($= 200$ kbit/sec $\times$ 0.033 sec$/8$), with a maximum of $\pi_{\max} = 8333$ bytes per image. (Recall that our long traces have been obtained by cutting the enhancement layer bit stream at 200 kbps, 400 kbps, 600 kbps, $\ldots$, 2 Mbps. A finer granularity solution to the optimization problem could be obtained by interpolating the rate-distortion curve between the 200 kbps spaced points and using a smaller sampling step size in the recursive fixing, which in turn would increase the required computational effort.)

The computational effort required for resolving our problem depends on the aggregation case which is considered (image, scene, constant, or total), on the length of an allocation segment, as well as on the number of scenes within a given allocation segment. Since scene shots are usually composed of tens to thousands of frames, the reduction in computational complexity when aggregating frames within a shot (scene case) or aggregating an arbitrary number of frames (constant case) is typically quite significant. For instance, in *The Firm*, with an encoding at 30 frames/sec, there are on average only around 8 scene shots in one allocation segment of 1000 frames.

B. Results

Figure 20 gives plots of the maximum overall quality Θ_l^* as a function of the allocation segment number l, for an average target rate of $C_{\max} = 1000$ kbps for *The Firm*, *Oprah*, *Oprah with commercials* and *News*. Not surprisingly, for all allocation segments, the overall quality with image–by–image streaming optimization is higher than that for the other aggregation cases. This is because the image–by–image optimization is finer. However, the overall quality achieved by the other aggregation cases is very close to that of image–by–image streaming: the difference is usually less than 1 dB in PSNR. This is due to the high correlation between the enhancement layer rate distortion functions of successive frames.

We show in Figure 21 the optimal quality averaged over all allocation segments for the entire videos as



(a) *The Firm*

(b) *News*

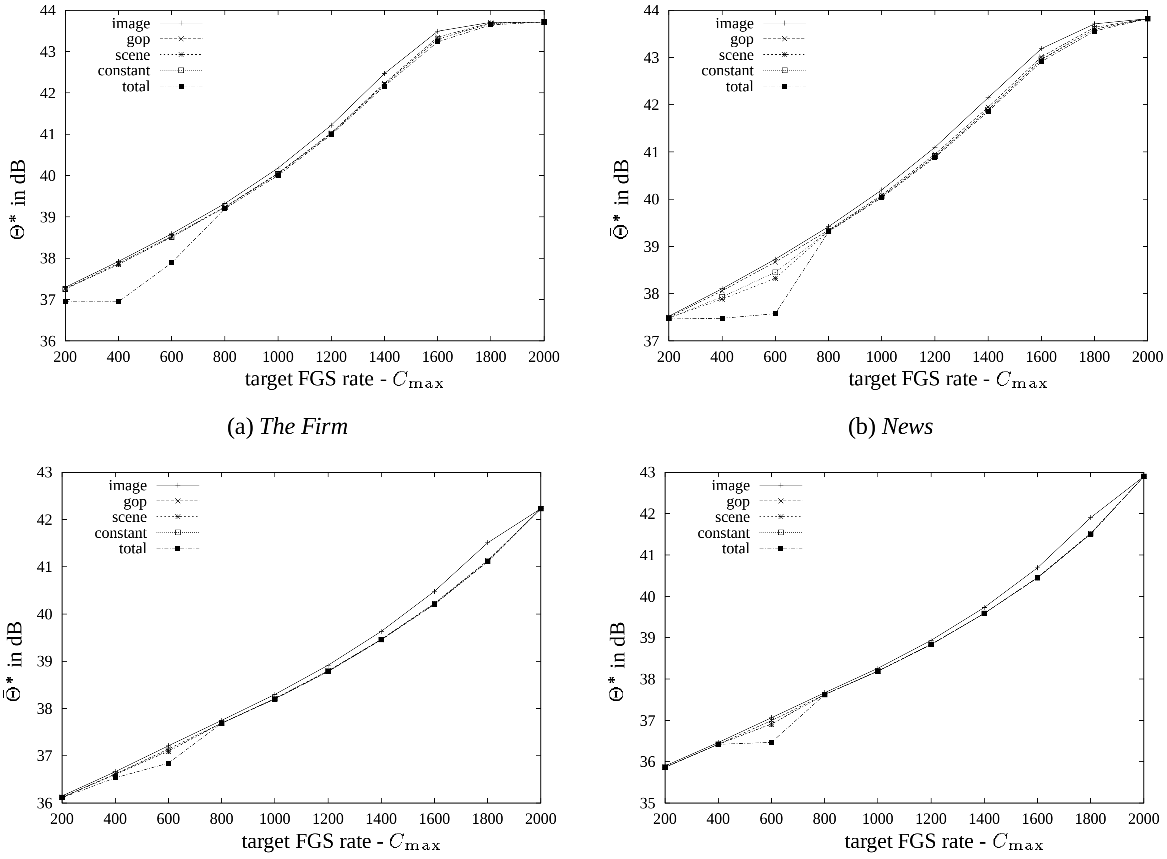
(c) *Oprah with commercials*

(d) *Oprah*

Fig. 20. Maximum overall quality (PSNR) Θ_l^* as a function of allocation sequence number l with $C_{\max} = 1000$ kbps ($\lfloor N/L \rfloor = 1,000$ frames, fixed)

a function of the target rate constraint $C_{\max}$. We observe that each of the aggregation cases gives about the same optimal quality for all target rates. We observed in more extensive experiments, which are not shown here, that the qualities are very similar when the allocation segments contain more than 1000 frames.

As we have seen from the average MSE based metrics plotted in the previous figures, there seems to be very little difference between the maximum quality achieved for all allocation segments when aggregating over scenes or arbitrary sequences. However, the actual perceived quality may be somewhat different. The reason is that the MSE does not account for temporal effects, such as the variations in quality between consecutive images: two sequences with a same average MSE may have different variations in image MSE, and thus different perceived quality. To illustrate this phenomenon, we monitor the maximum quality variation between consecutive images Var_s of a given streaming sequence (defined in (5)). For the streaming sequence s, $s = 1, \ldots, S_l$, with each enhancement layer subframe encoded with π bits, the maximum variation is $Var_s(\pi/T) = \max_{i=2,\ldots,N_s}\{|Q_{s,i}(\pi/T) - Q_{s,i-1}(\pi/T)|\}$. For allocations segments of 1000 frames, Table V shows the average maximum variation in quality $\overline{Var}$ (defined in (6)) for different FGS rates. We observe that the average maximum variation in quality for a given FGS rate is always smaller with scene aggregation than with constant or total aggregation. This means that selecting a constant number of bits for



(a) *The Firm*

(b) *News*

(c) *Oprah with commercials*

(d) *Oprah*

Fig. 21. Average maximum quality (PSNR) $\bar{\Theta}^*$ as a function of the enhancement layer (cut off) bit rate $C_{\max}$ ($\lfloor N/L \rfloor = 1,000$ frames, fixed)

the enhancement layer of all images within a given video shot yields on average a smaller maximum variation in image quality than selecting a constant number of bits for an arbitrary number of successive images. Therefore, it is preferable to choose streaming sequences that correspond to visual shots rather than segmenting the video arbitrarily. This result is intuitive since frames within a given shot are more likely to have similar visual complexity, and thus similar rate–distortion characteristics, than frames from different shots. This is confirmed in Figure 22, which shows the minimum value of the maximum variations in quality over all scenes of a given allocation segment $minVar$ (defined in (7)). We observe that the min–max variation in image quality is typically larger for arbitrary segmentation. This indicates that the minimum jump in quality in the streaming sequences of a given allocation segment is larger for arbitrary segmentation. In the case of shot segmentation the minimum jumps in quality are smaller; when the shot consists of one homogeneous video scene, $minVar_l$ is close to 0 dB. As shown in Figure 22, for some allocation segments, the difference with arbitrary segmentation can be more than 1 dB.

More generally, we expect the difference in rendered quality between shot–based segmentation and arbitrary segmentation to be more pronounced with a scene segmentation that is finer than shot–based segmentation. A finer segmentation would further segment sequences with varying rate–distortion characteristics,

TABLE V

AVERAGE MAXIMUM VARIATION IN QUALITY $\overline{Var}$ (IN dB) FOR LONG VIDEOS

	$C_{\max} = 800$ kbps			$C_{\max} = 1600$ kbps		
	scene	const.	total	scene	const.	total
The Firm	1.84	1.99	2.51	0.81	0.92	1.44
OprahWith	2.64	2.77	2.99	2.68	2.76	2.93
Oprah	2.43	2.47	2.60	2.60	2.64	2.75
News	2.22	2.55	3.71	1.43	1.64	2.89
StarWars	1.90	2.11	3.44	0.85	0.97	1.84
Silence	1.33	1.37	1.82	1.37	1.40	1.88
Toy Story	2.34	2.54	3.54	1.46	1.74	2.96
Football	2.21	2.56	4.92	1.09	1.38	3.49
Lecture	2.64	2.69	2.73	2.06	2.08	2.12

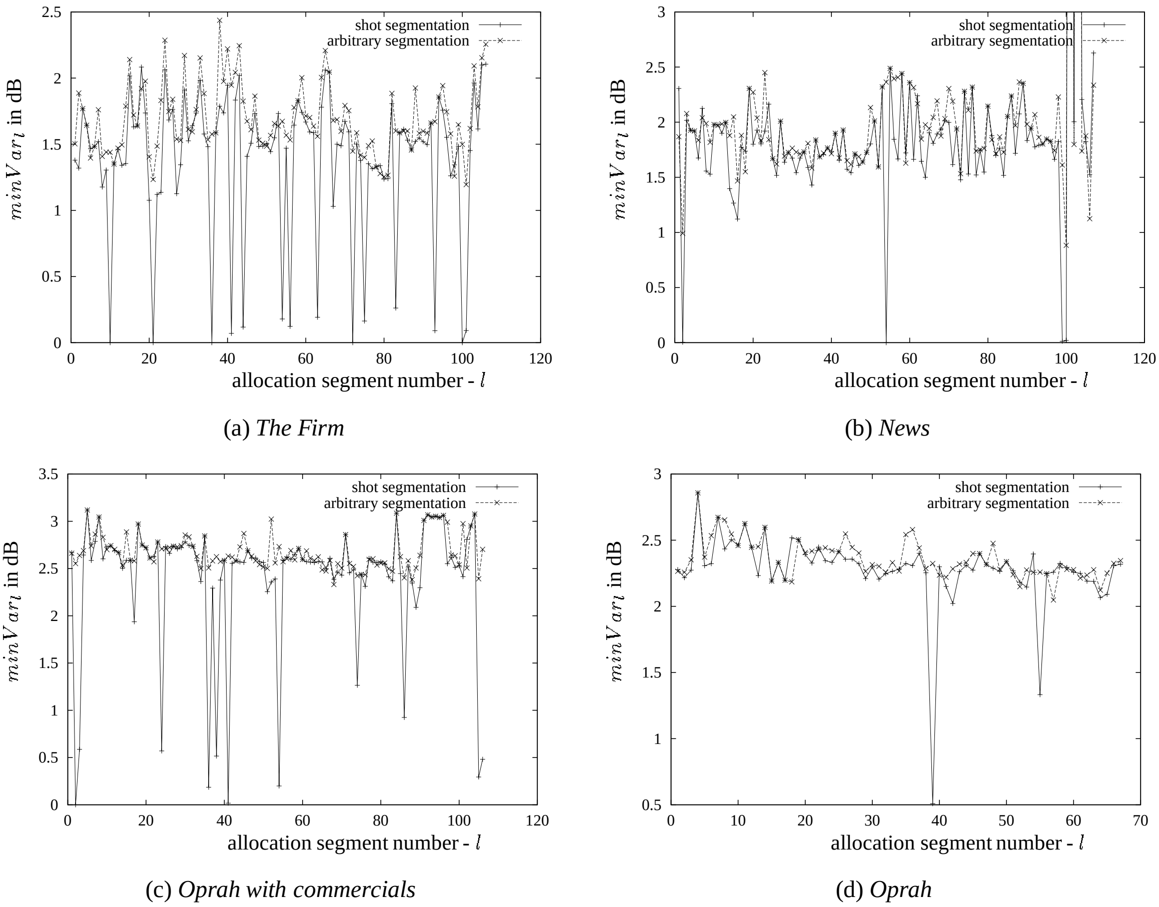
e.g., sequences with changes in motion or visual content other than director's cuts. This would increase the correlation between the qualities of the frames in a same scene, which would further reduce the quality degradation due to scene–based streaming over image–based streaming.

V. CONCLUSIONS

We have developed a framework, consisting of evaluation metric definitions and rate–distortion traces, for the evaluation of the streaming of FGS–encoded video. The defined evaluation metrics capture the quality of the received and decoded video both at the level of individual video frames (images) as well as aggregations of images (GoP, scene, etc). The rate–distortion traces provide the rate–distortion characteristics of the FGS enhancement layer for a set of long videos from different genres. Together, the defined evaluation metrics and the rate–distortion traces allow for the accurate evaluation of the quality of streamed FGS–encoded video without requiring experimentation with actual video.

Our analysis of the rate–distortion traces provides a number of insights that are useful for the design of streaming mechanisms for FGS–encoded video. First, the convex form of the rate–distortion curves of the individual bitplanes suggests to prioritize the cutting of the bit stream close to the end of the bit planes. (Note however that cutting the enhancement layer bit stream only at bit–plane boundaries would provide coarser grained scalability in adapting video quality to varying network conditions.) Secondly, the base layer frame types (I, P, and B) and in general the base layer coding tend to have a significant impact on the total quality obtained from the base layer plus FGS enhancement layer stream. We observed that, for fixed FGS enhancement layer cut–off rates, significant variations in the base layer quality correspond to significant variations in the total (base + enhancement layer) quality. This suggests to take the different base layer frame types into consideration in the streaming of the FGS enhancement layer frames. We also observed that, for fixed FGS enhancement layer cut–off rates, the total video quality tends to vary according to the different semantic content of the different video scenes. This suggests to take the scene structure into consideration in the enhancement layer streaming.

We have illustrated the use of our evaluation framework with an investigation of the rate–distortion optimized streaming at different image aggregation levels. We have found that the optimal scene–by–scene



(a) *The Firm*

(b) *News*

(c) *Oprah with commercials*

(d) *Oprah*

Fig. 22. Min–max variations in quality $minVar$ as a function of the allocation segment number ($\lfloor N/L \rfloor = 1,000$ frames, fixed)

adjustment of the FGS enhancement layer rate reduces the computational complexity of the optimization significantly compared to image–by–image optimization, while having only a very minor impact on the video quality. We also found that reducing the computational optimization effort by aggregating the images arbitrarily (without paying attention to the scene structure) tends to result in significant quality deteriorations.

Acknowledgements

We are grateful to Osama Lotfallah and Sethuraman Panchanathan of Arizona State University for explaining the intricacies of the MPEG–4 reference software to us, and to Frank Fitzek of Acticom GmbH, Berlin, Germany, and Patrick Seeling of Arizona State University who both helped in setting up the website. We are grateful to the Telecommunications Research Center at Arizona State University for the support of Philippe de Cuetos' visit at ASU in the spring of 2002.

References

[1] *ISO/IEC JTC1/SC29/WG11 Information Technology - Generic Coding of Audio-Visual Objects : Visual ISO/IEC 14496-2 / Amd X*, December 1999.

[2] W. Li, "Overview of Fine Granularity Scalability in MPEG–4 Video Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 301–317, March 2001.

[3] H. Radha, M. van der Schaar, and Y. Chen, “The MPEG-4 Fine-Grained Scalable Video Coding Method for Multimedia Streaming over IP,” *IEEE Transactions on Multimedia*, vol. 3, no. 1, pp. 53–68, March 2001.

[4] Y. Wang and Q. Zhu, “Error control and concealment for video communication: A review,” *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[5] Y. Wang, S. Wenger, J. Wen, and A. Katsaggelos, “Error resilient video coding techniques,” *IEEE Signal Processing Magazine*, vol. 17, no. 4, pp. 61–82, July 2000.

[6] J. Kim, R. M. Mersereau, and Y. Altunbasak, “Error-resilient image and video transmission over the internet using unequal error protection,” *IEEE Transaction on Image Processing*, vol. 12, no. 2, pp. 121–131, Feb. 2003.

[7] N. Duffield, K. Ramakrishnan, and A. Reibman, “Issues of quality and multiplexing when smoothing rate adaptive video,” *IEEE Transactions on Multimedia*, vol. 1, no. 4, pp. 53–68, Dec. 1999.

[8] X. Lu, R. O. Morando, and M. ElZarki, “Understanding video quality and its use in feedback control,” in *Proceedings of Packet Video Workshop*, Pittsburgh, PA, 2002.

[9] P. A. Chou and Z. Miao, “Rate-Distortion Optimized Streaming of Packetized Media,” *submitted to IEEE Transactions on Multimedia*, February 2001.

[10] R. Rejaie and A. Reibman, “Design Issues for Layered Quality—Adaptive Internet Video Playback,” in *Proc. of the Workshop on Digital Communications*, Taormina, Italy, September 2001, pp. 433–451.

[11] Q. Zhang, W. Zhu, and Y.-Q. Zhang, “Resource Allocation for Multimedia Streaming over the Internet,” *IEEE Transactions on Multimedia*, vol. 3, no. 3, pp. 339–355, September 2001.

[12] S. Bajaj, L. Breslau, and S. Shenker, “Uniform versus priority dropping for layered video,” in *Proceedings of ACM SIGCOMM*, Vancouver, Canada, September 1998.

[13] S. Nelakuditi, R. R. Harinath, E. Kusmierek, and Z.-L. Zhang, “Providing smoother quality layered video stream,” in *Proceedings of The 10th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Chapel Hill, NC, June 2000.

[14] R. Rejaie, M. Handley, and D. Estrin, “Layered quality adaptation for internet video streaming,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2530–2543, Dec. 2000.

[15] M.-T. Sun and A. R. Reibman, *Compressed Video over Networks*. Marcel Dekker, 2001.

[16] D. Wu, Y. T. Hou, W. Zhu, Y.-Q. Zhang, and J. M. Peha, “Streaming Video over the Internet: Approaches and Directions,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 1–20, March 2001.

[17] R. Rejaie, D. Estrin, and M. Handley, “Quality Adaptation for Congestion Controlled Video Playback over the Internet,” in *Proc. of ACM SIGCOMM*, Cambridge, September 1999, pp. 189–200.

[18] D. Saparilla and K. W. Ross, “Optimal Streaming of Layered Video,” in *Proc. of IEEE INFOCOM*, Tel Aviv, Israel, March 2000, pp. 737–746.

[19] Z. Miao and A. Ortega, “Expected Run–time Distortion Based Scheduling for Delivery of Scalable Media,” in *Proc. of International Conference of Packet Video*, Pittsburg, PA, April 2002.

[20] C. Buchner, T. Stockhammer, D. Marpe, G. Blattermann, and G. Heising, “Progressive texture video coding,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001, pp. 1813–1816.

[21] H.-C. Huang, C.-N. Wang, and T. Chiang, “A Robust Fine Granularity Scalability Using Trellis-Based Predictive Leak,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 372–385, June 2002.

[22] R. Kalluri and M. van der Schaar, “Fine Granular Scalability for H.26L-Based Video Streaming,” in *Proc. of IEEE International Conference on Consumer Electronics*, 2002, pp. 346–347.

[23] E. Lin, C. Podilchuk, A. Jacquin, and E. Delp, “A Hybrid Embedded Video Codec Using Base Layer Information for Enhancement Layer Coding,” in *Proc. of IEEE International Conference on Image Processing*, 2001, pp. 1005–1008.

[24] A. Luthra, R. Gandhi, K. Panusopone, K. Mckoen, D. Baylon, and L. Wang, “Performance of MPEG-4 Profiles used for streaming video,” in *Proceedings of Workshop and Exhibition on MPEG-4*, 2001, pp. 103–106.

[25] R. Rajendran, M. van der Schaar, and S.-F. Chang, “FGS+: Optimizing the Joint SNR–Temporal Video Quality in MPEG-4 Fine Grained Scalable Coding,” in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, 2002, pp. 445–448.

[26] A. R. Reibman, U. Bottou, and A. Basso, “DCT-Based Scalable Video Coding with Drift,” in *Proceedings of IEEE International Conference on Image Processing*, 2001, pp. 989–992.

[27] M. van der Schaar and Y.-T. Lin, “Content-Based Selective Enhancement for Streaming Video,” in *Proc. of IEEE International Conference on Image Processing*, 2001, pp. 977–980.

[28] M. van der Schaar and H. Radha, “A Hybrid Temporal–SNR Fine–Granular Scalability for Internet Video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 318–331, March 2001.

[29] Q. Wang, Z. Xiong, F. Wu, and S. Li, “Optimal rate allocation for progressive fine granularity scalable video coding,” *IEEE Signal Processing Letters*, vol. 9, no. 2, pp. 33–39, February 2002.

[30] F. Wu, S. Li, and Y.-Q. Zhang, “A Framework for Efficient Progressive Fine Granularity Scalable Video Coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 332–344, March 2001.

[31] H. Radha and Y. Chen, “Fine–Granular–Scalable video for Packet Networks,” in *Proc. of Packet Video Workshop*, 1999.

[32] H. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, “Scalable Internet Video Using MPEG-4,” *Signal Processing: Image Communication*, vol. 15, no. 1-2, pp. 95–126, September 1999.

[33] K. W. Stuhlmuller, M. Link, B. Girod, and U. Horn, “Scalable Internet Video Streaming with Unequal Error Protection,” in *Proc. of Packet Video Workshop*, 1999.

[34] M. van der Schaar and H. Radha, “Unequal Packet Loss Resilience for Fine-Granular-Scalability Video,” *IEEE Transactions on Multimedia*, vol. 3, no. 4, pp. 381–393, December 2001.

[35] X. K. Yang, C. Zhu, Z. Li, G. N. Feng, S. Wu, and N. Ling, “A Degressive Error Protection Algorithm for MPEG-4 FGS Video Streaming,” in *Proc. of IEEE International Conference on Image Processing*, 2002, pp. 737–740.

[36] T. M. Liu, W. Qi, H. Zhang, and F. Qi, “Systematic Rate Controller for MPEG-4 FGS Video Streaming,” in *Proc. of IEEE International Conference on Image Processing*, 2001, pp. 985–988.

[37] P. de Cuetos and K. W. Ross, “Adaptive Rate Control for Streaming Stored Fine-Grained Scalable Video,” in *Proc. of NOSSDAV*, Miami, Florida, May 2002, pp. 3–12.

[38] P. de Cuetos, P. Guillotel, K. W. Ross, and D. Thoreau, “Implementation of Adaptive Streaming of Stored MPEG–4 FGS Video,” in *Proc. of IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, August 2002.

[39] R. Cohen and H. Radha, “Streaming Fine-Grained Scalable Video over Packet-Based Networks,” in *Proc. of IEEE Globecom*, 2000, pp. 288–292.

[40] L. Zhao, J.-W. Kim, and C.-C. J. Kuo, "Constant Quality Rate Control for Streaming MPEG-4 FGS Video," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, 2002, pp. 544–547.

[41] H.-F. Hsiao, Q. Liu, and J.-N. Hwang, "Layered Video over IP Networks by Using Selective Drop Routers," in *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS)*, 2002, pp. I–411 – I–444.

[42] Y.-S. Tung, J.-L. Wu, P.-K. Hsiao, and K.-L. Huang, "An Efficient Streaming and Decoding Architecture for Stored FGS Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 8, pp. 730–735, August 2002.

[43] J. Vieron, T. Turletti, X. Hjnocq, C. Guillemot, and K. Salamatian, "TCP-Compatible Rate Control for FGS Layered Multicast Video Transmission Based on a Clustering Algorithm," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, 2002, pp. 453–456.

[44] J. Liu, B. Li, B. Li, and X. Cao, "Fine-Grained Scalable Video Broadcasting over Cellular Networks," in *Proc. of IEEE Conference on Multimedia and Expo*, 2002, pp. 417–420.

[45] T. Stockhammer, H. Jenkac, and C. Weiss, "Feedback and error protection strategies for wireless progressive video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 465–482, June 2002.

[46] M. van der Schaar and H. Radha, "Motion–compensation Fine–Granular–Scalability (MC-FGS) for Wireless Multimedia," in *Proc. of Fourth IEEE Workshop on Multimedia Signal Processing*, 2001, pp. 459–458.

[47] ——, "Adaptive Motion-Compensation Fine–Granular–Scalability (AMC-FGS) for Wireless Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 360–371, June 2002.

[48] R. Y. Chen and M. van der Schaar, "Complexity-Scalable MPEG-4 FGS Streaming for UMA," in *Proc. of IEEE International Conference on Consumer Electronics*, 2002, pp. 270–271.

[49] R. Chen and M. van der Schaar, "Resource-Driven MPEG-4 FGS for Universal Multimedia Access," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, Lausanne, Switzerland, 2002, pp. 421–424.

[50] F. Fitzek and M. Reisslein, "MPEG–4 and H.263 video traces for network performance evaluation," *IEEE Network*, vol. 15, no. 6, pp. 40–54, November/December 2001.

[51] M. Reisslein, J. Lassetter, S. Ratnam, O. Lotfallah, F. Fitzek, and S. Panchanathan, "Traffic and quality characterization of scalable encoded video: A large-scale trace-based study, part 1: Overview and definitions, part 2: Statistical analysis of single-layer encoded video, part 3: Statistical analysis of temporal scalable encoded video, part 4: Statistical analysis of spatial scalable encoded video," Dept. of Electrical Eng. Arizona State University, Tech. Rep., 2002, available at `http://trace.eas.asu.edu`.

[52] W. Li, F. Ling, and X. Chen, "Fine Granularity Scalability in MPEG-4 for Streaming Video," in *Proc. of ISCAS*, Geneva, Switzerland, May 2000.

[53] *ISO/IEC JTC1/SC29/WG11 N4791 — Report on MPEG–4 Visual Fine Granularity Scalability Tools Verification Tests*, May 2002.

[54] T. Kim and M. Ammar, "Optimal quality adaptation for MPEG-4 fine-grained scalable video," in *Proceedings of IEEE Infocom*, San Francisco, CA, Apr. 2003.

[55] Y.-S. Saw, *Rate Quality Optimized Video Coding*. Kluwer Academic Publishers, 1999.

[56] A. M. Rohaly and al., "Video Quality Experts Group: Current Results and Future Directions," in *Proc. SPIE Visual Communications and Image Processing*, vol. 4067, Perth, Australia, June 2000, pp. 742–753.

[57] S. Olsson, M. Stroppiana, and J. Baina, "Objective Methods for Assessment of Video Quality: State of the Art," *IEEE Trans. on Broadcasting*, vol. 43, no. 4, pp. 487–495, December 1997.

[58] S. Winkler, "Vision Models and Quality Metrics for Image Processing Applications," Ph.D. dissertation, EPFL, Switzerland, 2000.

[59] "ANSI T1.801.03 — Digital Transport of One–Way Video Signals — Parameters for Objective Performance Assessment," 1996.

[60] Microsoft, "ISO/IEC 14496 Video Reference Software," Microsoft–FDAM1–2.3–001213.

[61] A. M. Dawood and M. Ghanbari, "Scene Content Classification From MPEG Coded Bit Streams," in *Proc. of the 3rd Workshop on Multimedia Signal Processing*, 1999, pp. 253–258.

[62] C.-L. Huang and B.-Y. Liao, "A Robust Scene-Change Detection Method for Video Segmentation," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1281–1288, December 2001.

[63] G. Lupatini, C. Saraceno, and R. Leonardi, "Scene break detection: a comparison," in *Proc. of the Int. Workshop on Continuous–Media Databases and Applications*, 1998, pp. 34–41.

[64] MediaWare Solutions, "MyFlix 3.0," `http://www.mediaware.com.au/MyFlix.html`.

[65] E. V. Denardo, *Dynamic Programming: Models and Applications*. Prentice–Hall, 1982.