Rate-Distortion Models for FGS-encoded Video Sequences

Cheng-Hsin Hsu
School of Computing Science
Simon Fraser University
Surrey, BC, Canada

Mohamed Hefeeda
School of Computing Science
Simon Fraser University
Surrey, BC, Canada

***Abstract*— Fine granularity scalability (FGS) coding enables finer bitrate scalability and better error resiliency than traditional multi-layer coding, because it allows truncating a bitstream at arbitrary bits. This flexibility enables streaming applications a more space to optimize quality (i.e., minimize distortion) at a given channel bandwidth. This optimization relies on accurate estimation of the rate-distortion (R-D) characteristics of video sequences. In this paper, we analyze and compare the performance of the R-D models proposed in the literature for FGS coding systems. We analyze the models by following their mathematical derivations and scrutinizing their assumptions. We perform the comparison by implementing the models and conducting an extensive experimental study using a large set of video sequences with diverse image and motion complexities. The results of our experiments provide guidelines for choosing the appropriate R-D model for rate-distortion optimized streaming applications.**

I. Introduction

Video streaming on the Internet is increasingly getting very popular. The best-effort service offered by the Internet, however, poses unique challenges for high-quality video streaming. These challenges include heterogeneity and bandwidth variability in network channels between streaming servers and clients. These challenges require streaming systems to support bitrate scalability and error resiliency. Traditional streaming systems partially cope with these challenges using either multi-layer or multi-description encoding of streams. These solutions, however, provide limited (coarse-grain) rate scalability: clients receiving incomplete layers or descriptions can not use them to enhance display quality. These solutions also suffer from poor error resiliency because the loss or corruption of a few bits render the entire layer useless.

In contrast to traditional multi-layer video coding, fine granularity scalability (FGS) coding has been proposed to provide finer bitrate scalability and better error resiliency [1]. An FGS encoder compresses video data into two layers: a base layer which provides basic quality, and a single enhancement layer that adds incremental quality refinements proportional to the number of bits received. Arbitrary truncation (at the bit level) of the enhancement layer to achieve a target bitrate is possible, and, more importantly, it does not require any complex or resource-intensive operations from the streaming servers or their proxy caches. This in turn enables streaming servers to scale to larger and more heterogeneous set of clients.

Given the flexibility of controlling the bitrate provided by FGS encoders and the constraints on and the variability of the channel bandwidth, researchers seek to optimize the received video stream quality. A common method in the literature to achieve such quality-optimized systems is through the use of rate-distortion (R-D) functions. R-D functions describe the relationship between the bitrate and the expected level of distortion in the reconstructed video stream. Knowing the R-D functions enables us, for example, to determine the required bitrate to achieve a target quality, to optimally allocate a given bandwidth among frames, and to prioritize bits within the same frame.

There are two approaches for determining R-D functions of a given video sequence: empirical and analytic. Empirical models rely on statistical observations rather than theoretical derivations. Each empirical model proposes a *parameterized* function that is thought to approximate the actual R-D function. The parameters of the model are computed by fitting actual R-D data to the proposed function. This requires coding a video sequence several times at different rates to get enough samples for estimating the parameters. Analytic models, on the other hand, break the system into components and describe each component with a model using theoretical bases. The components are then put together for a complete R-D model. Several analytic models have been proposed in the literature. Unfortunately, we are not aware of any previous work that rigorously analyzes and compares the accuracy and applicability of the models.

In this paper, we analyze and compare the performance of the R-D models proposed in the literature for FGS coding systems. We analyze the models by following their mathematical derivations and scrutinizing their assumptions. We perform the comparison by implementing the models and conducting an extensive experimental study using a large set of video sequences with diverse image and motion complexities. The results of our experiments provide guidelines for choosing the appropriate R-D model for rate-distortion optimized streaming applications.

The rest of this paper is organized as follows. In Section II, we present the mathematical foundations of several FGS R-D models. In Section III, we describe our experimental setup, the selection of test sequences, and our results. We conclude the paper in Section IV.

Due to space limitations, in this paper we present only a synopsis of our analysis and a small sample of our experi-

mental results. Interested readers are referred to [2] for the details.

II. FGS Rate-Distortion Models

The FGS enhancement layer employs a different quantization mechanism from the base layer. Instead of quantizing transform coefficients with different quantization parameter Q, the FGS enhancement layer drops bits from transform coefficients, which is equivalent to gauging the bitrate R. This suggests that the traditional R-D models may not capture the characteristics of the enhancement layer. We carefully studied several traditional models, such as the quadratic model and ρ-domain model, and found most of them are not applicable to the enhancement layer. More details are given in [2]. In the following subsections, we present three R-D models that are explicitly designed for the FGS enhancement layer.

A. Square Root Model

To develop the square root model, Dai et al. [3] first statistically analyzed FGS-encoded sequences. They found that the enhancement layer coefficients can be modeled by a linear combination of two Laplacian distributions:

$$f(x) = p\frac{\lambda_0}{2}e^{-\lambda_0|x|} + (1-p)\frac{\lambda_1}{2}e^{-\lambda_1|x|}, \tag{1}$$

where p is a weight parameter between the two Laplacian distributions, which have λ_0 and λ_1 as their distribution parameters.

Suppose there are n bitplanes, and let z denote the number of transmitted bitplanes. This means the receiver is not aware of the last $(n-z)$ bitplanes, and has to reconstruct the coefficients without these bitplanes. This is essentially a quantizer with a uniform quantization step $\Delta(z) = 2^{(n-z)}$, and the reconstruction levels at:

$$L(x) = \begin{cases} \lfloor x/\Delta \rfloor \times \Delta & x \geq 0; \\ \lceil x/\Delta \rceil \times \Delta & x < 0. \end{cases} \tag{2}$$

Next, we write the distortion D (in terms of the mean square error) as a function of the quantization step Δ:

$$D(\Delta) = 2\sum_{m=0}^{N}\sum_{n=m\Delta}^{(m+1)\Delta-1}(n-m\Delta)^2 f(n), \tag{3}$$

where $f(n)$ is the source probability distribution, and $N = 2^z$ is the total number of positive quantization bins. The outer summation iterates through all quantization bins, while the inner summation covers integers within each quantization bin. The $m\Delta$ is the reconstruction level of the quantization index m. Substituting the the source model in Eq. (1) for $f(x)$, we get the D-Q function.

Meanwhile, the size of each bitplane is extracted by scanning the compressed bitstream. Comparing the bitplane size against the target bitrate, we can find out the corresponding z (the last transmitted bitplane). This defines the quantization step Δ, but only at the bitplane boundaries.

To get the R-Q function, the following heuristic function is used [3]:

$$R(z) = e_1 z^2 + e_2 z + e_3, \tag{4}$$

where e_1, e_2, and e_3 are polynomial coefficients, which are derived by fitting the function against the R-z mappings at bitplane boundaries.

B. Logarithm Model

The logarithm R-D model also assumes a mixture Laplacian source and is derived as follows [4]. Let C denote the collection of all enhancement layer coefficients, and let $M = |C|$ denote the total number of coefficients. For any given quantization step Δ, the coefficients falling in the interval $(-\Delta, \Delta)$ will have zero reconstructed level. C is divided into two subsets: C_z and C_{nz} for zero and non-zero quantized coefficients. It was found that the C_z contributes the majority of distortion at low and medium bitrates [4]. Define D_z and D_{nz} to be the distortion contributed by C_z and C_{nz}, respectively. The authors of [4] propose to carefully compute D_z and roughly approximate D_{nz}, since the D_z represents the majority of distortion.

Because coefficients in C_z have zero reconstruction level, D_z (in mean square error) can be written as:

$$D_z(\Delta) = \sum_{c_i \in C_z} |c_i|^2 . \tag{5}$$

Since bitplane quantization is very close to uniform quantization, the logarithm model uses the following D-Q function to approximate D_{nz}:

$$D(\Delta) = \frac{\Delta^2}{12}. \tag{6}$$

Adding and normalizing D_{nz} and D_Z result in the D-Q function:

$$D(\Delta) = \frac{D_z + D_{nz}}{M} = \frac{1}{M}\sum_{c_i \in C_Z} |c_i|^2 + \rho\frac{\Delta^2}{12}, \tag{7}$$

where ρ is the percentage of non-zero quantized coefficients.

Dai et al. found that the linear bitrate model proposed for traditional transform coders [5] is also valid in the FGS enhancement layer and proposed to employ it as the R-Q model [4]. The linear bitrate model defines the bitrate R as a linear function of ρ with slope γ. To estimate the slope γ, we first compute all the mapping between ρ and bitrate R at bitplane boundaries, and then do a polynomial fitting.

C. Generalized Gaussian Model

To accurately model the enhancement layer coefficients, the generalized Gaussian model introduces higher flexibility by applying a zero-mean generalized Gaussian function to each frequency [6]. The R-D function is separately derived for each of the 64 frequencies. Then, the complete R-D function is calculated by aggregating all 64 functions together.

The R-D function is approximated by:

$$\begin{aligned} R(\Delta) &= h(f) - \log_2 \Delta, \\ D(\Delta) &= \Delta^2/3, \end{aligned} \tag{8}$$

where $h(f)$ is the differential entropy of $f(x)$. Note that, the derivation of this R-D function holds under the assumption that $f(x)$ is a zero-mean symmetric distribution. Furthermore, the $D(\Delta)$ function is deviated from the classic uniform quantizer approximation (Eq. (6)) due to the unique reconstruction levels in bitplane coding (Eq. (2)).

High computational complexity is a main concern for this R-D model. We have to estimate and store 64 pair of distribution parameters for each frame, instead of 1 set of parameters in other models. The differential entropy $h(f)$ computation involves intensive integrations over the whole real line. The intensity is even higher for a complex density function $f(x)$, like the generalized Gaussian.

III. Experimental Study

A. Set up

In our experiments, we use the MPEG-4 Reference Software Version 2.5 [7] developed by Microsoft as an experimental package for the MPEG-4 standard. We instrument the reference software to extract various statistics of a video sequence. For instance, we collect the transform coefficients, number of bitplanes, and size of each bitplane in the enhancement layer. This information is then used to estimate the parameters of different R-D models. We have implemented the following R-D models for the FGS enhancement layer: the square root, the logarithm, and the generalized Gaussian. To thoroughly evaluate the above R-D models for FGS-encoded video sequences, we perform the following steps:

1) Choose a test video sequence (see Section III-B).
2) Fix the bitrate R_b for encoding the base layer, and FGS-encode the sequence. The encoder creates two files: one for the base layer and the other for the enhancement layer.
3) For each frame, equally divide every bitplane into 5 segments. This defines 6 sampling bitrates R_e for each bitplane, including the bitrates at bitplane boundaries. The minimum rate is 0, while the maximum rate is the rate at which all bitplanes of a frame are included in the stream, i.e., full quality. For each value of R_e, do the following:
    a) For each R-D model considered in the study, extract the needed information from the enhancement layer file. Then compute the parameters of the R-D model and estimate the distortion at R_e.
    b) Truncate the enhancement bitstream at R_e and save it as a new file. Compute the empirical distortion by decoding this truncated file and comparing the reconstructed and original video frames (both are uncompressed).
4) Randomly choose several frames from the video sequence and plot the R-D curves computed in step 3 for these frames.
5) For each R-D model, compute the absolute average error per frame by doing the following:
    a) Compute the absolute difference (in PSNR) between the empirical and the estimated distortion at all considered R_e.
    b) Take the mean value of the difference to get the average absolute error.
6) Repeat steps 1-5 for the next test video sequence.

B. Test Sequences

To form a set of test sequences, we select twenty video sequences from various sources [8], [9]. We adopt the neighborhood difference as the complexity metric [10], and categorize these sequences into three complexity classes: low, medium, and high. Our extensive experimental results [2] indicate that video sequences belonging to the same complexity class exhibit similar R-D characteristics.

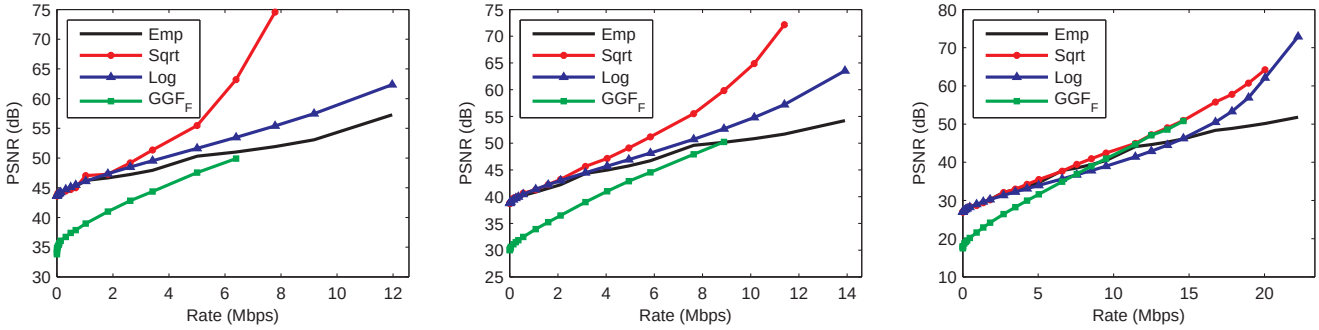
To illustrate our classification, we describe three sample sequences: Akiyo (low complexity), Foreman (medium complexity), and Mobile (high complexity). In Akiyo, a female reporter reads news with very limited head movements in front of a fixed camera. The Foreman sequence also features a talking person, but it was taken with a hand-held device that introduces camera movements. The Mobile sequence contains saturated colors and several moving objects, thus falls in the highest complexity class. More details on sequence complexity and classification are given in [2].

C. Sample Results

The accuracy of the R-D models across different complexity classes is depicted in Figure 1. The accuracy of the models are compared against the actual R-D function (denoted as the empirical (Emp) model in the Figures) of each video sequence. First, we find that the square root model deviates dramatically at high bitrate. This is because its D-Q function (Eq. (3)) assumes that there is no rounding error, and returns $D(\Delta) = 0$ when all bitplanes have been transmitted.

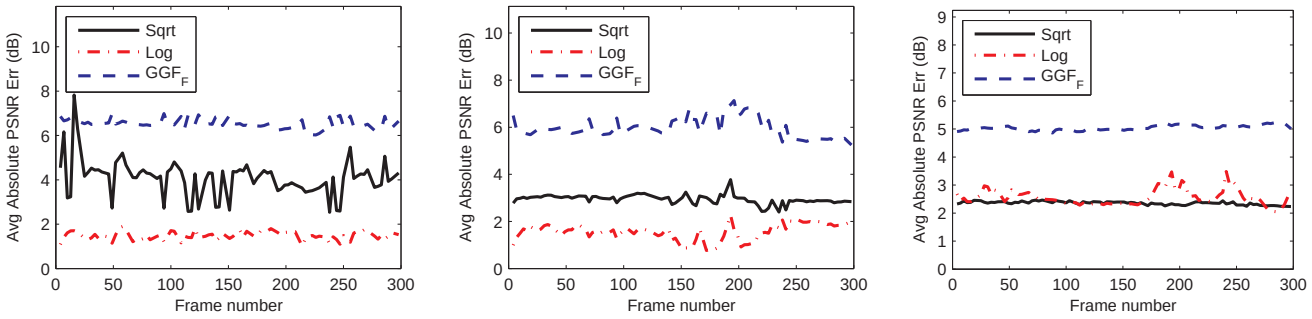
Second, the generalized Gaussian model works poorly on following the empirical R-D curves. It has larger deviation at low bitrates, and it does not produce any results at high bitrates. The large errors at low bitrates can be explained by the derivation of its R-Q and D-Q functions, where the quantization steps are assumed to be very small and all samples in the same quantization bin share equal probability. This introduces tremendous errors when Δ is large, e.g., at low bitrates. On the other hand, the D-Q function (Eq. (8)) stops working when $\Delta = 1$ (high bitrate). Hence, the generalized Gaussian model can not handle high bitrates.

Last, the logarithm model shows a higher deviation at high bitrates of high complexity sequences, e.g., Fig. 1(c). This is because these sequences tend to have more bitplanes, which influences the linear bitrate model accuracy. More importantly is the assumption that zero-quantized coefficients contribute the majority of distortion does not hold in high complexity sequences with low quantization steps. Furthermore, the uniform quantizers (Eq. (6)) approximation is rough, and may not be applicable to FGS coders as proved in [6].



(a) Akiyo, frame 122 (b) Foreman, frame 105 (c) Mobile, frame 104

Fig. 1. R-D curves for four models applied to three video sequences of different complexities.



(a) Akiyo (b) Foreman (c) Mobile

Fig. 2. The average absolute error of the R-D models across all frames of three video sequences with different complexities.

In Fig. 2, we present more global results in the format of average absolute error in each frame. Across all complexity classes and frame types, the generalized Gaussian model constantly produces errors larger than 5dB. This implies that this model is not useful in practice. The square root model works better in high complexity sequences than low complexity ones. On the other hand, the logarithm model works better in low complexity ones. We see the square root model slightly outperforms the logarithm model in Fig. 2(c), and expect to see a larger margin in sequences with even higher complexity. Fig. 2(a) indicates that the logarithm model significantly surpasses the square root model, which is universally true in all low complexity sequences we have tested.

IV. Conclusion

We have analyzed and experimentally compared three R-D models proposed for FGS-encoded video sequences: the square root, logarithm, and generalized Gaussian models. We find that the generalized Gaussian model fails to provide reasonable accuracy. Our results indicate that the logarithm model is more accurate than the square root model in low complexity sequences, while the square root model is more accurate in high complexity sequences. Our findings provide streaming applications guidelines to choose the appropriate R-D model based on sequence complexity. For example, a video conference system may choose the logarithm model, while a high-complexity sports program may perform better with the square root model.

References

[1] 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.

[2] C. Hsu and M. Hefeeda, "On the accuracy and complexity of rate-distortion models for FGS-encoded video sequences," Simon Fraser University, Tech. Rep. TR 2006-12, May 2006.

[3] M. Dai and D. Loguinov, "Analysis of rate-distortion functions and congestion control in scalable Internet video streaming," in *Proc. of ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'03)*, Monterey CA, June 2003.

[4] M. Dai, D. Loguinov, and H. Radha, "Rate-distortion modeling of scalable video coders," in *Proc. of IEEE International Conference on Image Processing (ICIP'04)*, Singapore, October 2004.

[5] Z. He and S. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 11, pp. 970–982, November 2002.

[6] J. Sun, W. Gao, D. Zhao, and Q. Huang, "Statistical model, analysis and approximation of rate-distortion function in MPEG-4 FGS videos," in *Proc. of SPIE International Conference on Visual Communication and Image Processing (VCIP'05)*, Beijing, China, July 2005.

[7] "MPEG-4 Visual reference software," February 2004, ISO/IEC 14496.

[8] "Web Page of Video Traces Research Group," Arizona State University, http://trace.eas.asu.edu/yuv/index.html.

[9] "Web Page of Center for Image Processing Research," Rensselaer Polytechnic Institute, http://www.cipr.rpi.edu/resource/sequences.

[10] D. Adjeroh and M. Lee, "Scene-adaptive transform domain video partitioning," *IEEE Transactions on Multimedia*, vol. 6, no. 1, pp. 58–69, February 2004.