Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Computer Networks 44 (2004) 177–188

COMPUTER NETWORKS

www.elsevier.com/locate/comnet

Study of long-duration MPEG-trace segmentation methods for developing frame-size-based traffic models

Uttam K. Sarkar [a,*,1], S. Ramakrishnan [b], Dilip Sarkar [c,*]

[a] Department of Computer Science and MIS, Indian Institute of Management Calcutta, P.O. Joka, Kolkata 700 104, India
[b] Department of Mathematics, University of Miami, Coral Gables, FL 33124-4250, USA
[c] Department of Computer Science, University of Miami, P.O. Box 248154, Coral Gables, FL 33124-4245, USA

Received 1 March 2002; received in revised form 18 July 2003; accepted 24 July 2003

Responsible Editor: N. Georganas

Abstract

Texture and temporal variations in scenes, and peculiarities of MPEG compression algorithms result in very complex frame-size data sets for any long-duration variable bit rate (VBR) video. A major hurdle in capturing the statistical behavior of such a data trace can be removed by segmentation of all frames into an *appropriate* number of analytically characterizable classes. However, video-trace segmentation techniques, particularly those which also enable preserving periodicity of group of pictures (GOP) in the modeled data, are lacking in the literature.

In this paper, we propose and evaluate few techniques for segmenting frame-size data sets in any long-duration video trace. The proposed techniques partition the *group of pictures* in a video into size-based groups called *shot-classes*. Frames in each shot-class have three data-sets—one each for *intra* (I-), *bi-directional* (B-), and *predictive* (P-) type frames. We have evaluated the performance of the proposed segmentation techniques by modeling each of I-, B-, and P-type frame in each shot-class by a Gamma distribution. Accuracy and usefulness of the proposed segmentation methods in building frame-size traffic models have been evaluated by QQ plots and the leaky-bucket simulation study. The results reveal that one of the segmentation techniques is very effective in characterizing the frame-size data behavior in a long-duration VBR video.

Keywords: Variable bit rate video; Frame-size traffic model; Gamma distribution; MPEG

1. Introduction

Video is expected to be the major traffic source for broadband networks [1,10]. Because of large bandwidth requirement for communication of high-quality uncompressed video it is expected that most, if not all, video will be encoded with MPEG-like data compression techniques [3]. These compression algorithms can provide a very

* Corresponding authors. Tel.: +1-305-284-2256; fax: +1-305-284-2264.

E-mail addresses: uttam@iimcal.ac.in (U.K. Sarkar), ramakris@math.miami.edu (S. Ramakrishnan), sarkar@cs.miami.edu (D. Sarkar).

[1] The author was visiting the Computer Science Department of the University of Miami, Coral Gables, FL 33124, USA on sabbatical when the work was partly done.

1389-1286/$ - see front matter
doi:10.1016/S1389-1286(03)00350-5

high compression ratio while maintaining good quality of decompressed video. However, MPEG-coding provides different amount of compression for different frames [3], and results in *variable bit rate* (VBR) data—known as VBR video. Some deterministic algorithms for computation of burstiness of VBR video have been proposed by Tryfonas and others [18]. Once these algorithms are widely known, they are expected to replace such burstiness measure as leaky-bucket simulation. Texture and temporal scene variations, and peculiarities in the MPEG compressions of intra (I-), bi-directional (B-), and predictive (P-) frames make it extremely difficult to accurately characterize all frame sizes in any long-duration VBR video data set by any single analytical model. This calls for finer characterization of individual I-, B-, and P-type frames and identification of smaller video segments over which each characterization holds good.

Accurate traffic models of VBR video are necessary for prediction of performance of any broadband network during its design and operation. Several traffic models have been proposed in the literature. They include first-order *autoregressive* (AR) [12], *discrete* AR (DAR) [5,9,19], *Markov renewal process* (MRP) [2], MRP *transform-expand-sample* (TES) [11], *finite-state Markov chain* (MC) [1,5,13,19], *Gamma-beta auto-regression* (GBAR) [4], and *group-of-pictures* (GOP) GBAR [3] models. Although traffic for short-duration video clips (those lasting only a few seconds or minutes) can be modeled without segmenting the trace, segmentation is essential for developing traffic models for long-duration videos such as full-length commercial movies. By *long* we mean videos which typically span over a couple of hours or more and would contain a few hundreds of thousands of compressed frames.

1.1. Two categories of traffic model

Video traffic models can be broadly classified into two categories: (i) *data rate models* (DRMs), and (ii) *frame-size models* (FSMs). In a data rate model, only the rate at which data are arriving at a link is generated for performance prediction purpose. Almost all models including AR, DAR, MRP, MRP TES, MC, GBAR fall under this category. These models are good for predicting average packet-loss probability, and ATM buffer over-flowing probability. However, they fail to identify such details as percentages of I, B, and P frames affected.

In a frame-size model, sizes of individual MPEG frames are considered, and hence data rate information can be obtained from the frame-size information. Models reported by Krunz and Tripathi [8], Dawood and Ghanbari [2], and Frey and Nguyen-Quang [3] fall under this category. It is believed that the main obstacle in the development of an accurate frame-size model for a long-duration video is to capture the distributions of I-, B-, and P-type frames in suitably identified segments of the video. The GOP GBAR model attempts to capture overall statistical properties of I, P, and B frames of MPEG movies without capturing segment-level regularity of frame sizes [2]. The model by Krunz and Tripathi [8] assumes that the variation of a scene changes the average size of I frames, but not the sizes of P and B frames. However, results in Dawood and Ghanbari [2] and our analysis (see Table 2 in Section 3.2) show wide variations in P and B frames as well. Also, the sizes of the frames, drawn from log-normal distributions following Krunz et al. [7], are not quite good fits. Dawood and Ghanbari have used one video traffic model for each type of video, and combined them using a finite state machine; at any instant the video model to be used for traffic generation is determined by the current state of the finite state machine. The accuracy of the model reveals the need for segmentation of video trace. We used video-trace segmentation for developing accurate traffic models in [15,17]. This further motivated us to study various video-trace segmentation methods which are reported in this paper. Before reporting the segmentation methods, next we discuss our objectives for video-trace segmentation.

1.2. Segmentation of a long-duration video trace

A long-duration VBR video, say a full-length commercial movie, contains a few hundreds of thousands of frames. For example, Crocodile Dundee, one of the movie traces in our study, has

29,089 group of pictures (GOP). This amounts to about 174,500 MPEG compressed frames in IBBPBB format of a GOP. When these frame sizes in bytes are plotted against the time axis (see [9] for an example) the data behavior appears too noisy and suggests no simple way of a single statistical characterization for the entire video trace. Two major factors contribute to the complexity of this data behavior. First, three different types of frames (known as I, B, and P in MPEG parlance) having different mean and variance characteristics periodically intermingle in the trace. Second, texture and temporal variations in scene changes follow no regular pattern. For example, one scene in a movie may continue for five minutes in the same room with hardly any movement of the actors or of their surroundings resulting in relatively small sized P and B frames. In contrast, the very next outdoor scene may have lots of scene variations over only a few seconds causing large values of B frames for that interval. Consequently, frame-size behavior of a long-duration video cannot be captured well unless such size-based segments or bursts are identified and the size behavior of each frame type in each segment is modeled.

In a video-trace segmentation, the primary concern is how to do a *proper* segmentation of all video frames into an *appropriate* number of analytically characterizable classes using which the traffic model can be developed [16]. This segmentation problem is yet another manifestation of the classical problem of pattern classification for a given modeling purpose, and no *the-best-classification* criterion exists. Classification difficulties lie at two levels. It is not known what would be the *best* number of classes. Worse, for a given number of classes or segments it's unclear what would be *the ideal* criteria to partition all frames into those classes. In the domain of classification of a VBR video data-set, it's necessary to characterize the behavior of each frame type, to capture the duration of subsequent segments in terms of real time, and also to enable preservation of periodicity of frames as they occur in a of group of pictures (GOP). Although Iraqi and Boutaba [6] have shown scene changes can be identified by comparing adjacent GOP sizes and have used the concept for bandwidth allocation of wireless networks, it is not intended for segmentation of video traces. Segmentation techniques fulfilling these requirements are lacking in the literature.

1.3. Outline of the paper

The purpose of this paper is to address the two fundamental difficulties of segmentation from the viewpoint of characterizing frame-size behavior in a VBR video. We address both (i) how to find a good number of segments, and (ii) how to determine the criteria based on which individual frames may be made to belong to those segments. In this paper, we propose and evaluate three such segmentation techniques. Each of these techniques partitions the *group of pictures* in the video into size-based groups called *shot-classes*. Frames in each shot-class have three data-sets—one each for *intra* (I-), *bi-directional* (B-), and *predictive* (P-) type frames. We have evaluated the performance of the proposed segmentation techniques by showing how the behavior of I, B, and P frames in identified segments can be captured by Gamma models, and how intra-segment duration and inter-segment transitions can be modeled by a Markov chain. Applicability of the proposed methods in developing frame-size traffic models have been demonstrated using QQ plots and leaky-bucket simulations. Our results indicate the segmentation technique GIIL (named after Geometrically Increasing frame-size Interval Lengths for shot-classes) proposed in this paper is very effective in capturing the frame-size behavior in a video trace.

Section 2 presents the proposed methods for segmentation of long-duration video-traces. In Section 3, we evaluate the performance of the proposed segmentation methods and study the effect of threshold parameter. To these ends, we showed how the sizes of individual frame types could be modeled under the segmentation schemes and how a variation in threshold affects the accuracy of the model. We also show QQ plots and the leaky bucket simulation results to highlight how the proposed segmentation led to accurate frame-size traffic models as in [17]. Results for the trace of the full-length movie, *Crocodile Dundee* are reported—the results for other videos (see Section 3) being very similar have not been given separately. We discuss our observations and future extensions of our work in Section 4.

2. VBR video segmentation methods

In a VBR video, frames are of different sizes because of composition or content of a picture, and temporal similarity of adjacent pictures. Also, frames from different parts of the video produce different amounts of compressions. Parts or segments of the video with similar compressions are necessary for successful analysis and modeling as shown by Dawood and Ghanbari [2]. This section presents three different techniques of segmenting a long-duration VBR video trace. Of particular emphasis in this study is to observe the sensitivity of the segmentation techniques on the effectiveness of a segmentation-dependent video model. Let $F = F_1, F_2, F_3, \ldots, F_{n_f}$ be the sequence of n_f frames obtained from MPEG encoding of a long-duration video. Since estimation of *data loss rate* (DLR) requires only the size of a frame and not the actual data, a frame for the modeling purpose is represented by its *serial number*, *type* (I, B, or P), and *size* in bytes after the MPEG encoding. While not required by the MPEG standard, the underlying GOP usually follows an (N, M) cyclic format in which the first frame of a GOP is an I frame, every Mth frame is a P frame, and the rest are B frames. For this study it is assumed that a video has been encoded with a single GOP structure. The *size* of a GOP is the sum of the sizes (in bytes) of all N frames in the GOP. We denote successive GOPs in a video trace by $G_1, G_2, \ldots, G_{n_g}$ for our reference in the paper, where n_g denotes the number of GOPs. For long-duration videos like full-length commercial movies n_g will typically be several tens to as high as a few hundreds of thousands.

2.1. Formation of clips

A *clip* of length k, $0 < k \leqslant n_g$, is any consecutive sequence of k GOPs, that is, $G_{i+1}, G_{i+2}, \ldots, G_{i+k}$ for some i, $0 \leqslant i \leqslant (n_g - k)$. We denote successive clips by $C_1, C_2, \ldots$ and a set of clips by C. We use the notation $G_i \in C_j$ to indicate G_i belongs to C_j, $length(C_j)$ to indicate the number of GOPs in C_j, and $size(G_i)$ to mean the sum of sizes in bytes of all frames in GOP G_i.

We use a technique similar to moving averaging and group *similar-size* GOPs to obtain a set of video clips. Sizes of these clips depend on the clustering of similar sized GOPs together—larger the cluster higher the clip size. During clip construction, let the average size of a GOP in the partially formed clip of length k starting with G_{i+1} be, $clip\_avg = (\sum_{l=1}^{k} size(G_{i+l}))/k$. The next GOP, G_{i+k+1}, is included in the current partial clip if the size of G_{i+k+1} does not differ from $clip\_avg$ by more than a user provided *threshold* value. The smaller the value of *threshold*, the smaller the *length* of each clip and consequently, the higher the total number of clips formed. In this paper we have made a detailed study of the effect of the value of *threshold* on the effectiveness of segmentation. We express the threshold value by a *threshold factor* which is a multiple of the average size of B-frame in the original video from which the model is derived. For example, the average B-frame size in the trace of the movie *Crocodile Dundee* is 4445.64 bytes. So, a threshold factor of 2 would mean the actual value of threshold used is $2 \times 4445.82 = 8891.64$ bytes. Normalizing the threshold value with respect to frame-size rather than expressing it as an absolute byte size makes the threshold parameter uniformly applicable to video traces irrespective of their resolutions. Computational results (see Section 3) indicate the choice of threshold is not difficult and a range of good choices, namely [0.5, 2], exists.

2.2. Methods for identification of shot-classes

A *shot-class* of length k, $k \geqslant 1$, is a union of k *distinct*, not necessarily consecutive, clips. We represent shot-classes by $S_1, S_2, \ldots$ and $C_j \in S_i$ denotes all GOPs belonging to clip C_j belong to shot-class S_i. Every clip belongs to one and only one shot-class.

The challenges in constructing the shot-classes are twofold—(i) how many shot-classes (n) to use and (ii) how to partition the entire range of GOP sizes into n such sub-intervals—one for each shot-class so that each of these classes can be modeled accurately. We have performed experimentation with a wide ranging number of shot-classes. It was found that too few (less than 5) shot-classes fail to capture the detailed frame-size variations in different segments of a video essentially eliminating the intended benefits of segmentation. On the other hand, too many (more than 10) classes result in some classes containing too few frames for the classification to be of any statististical significance. Moreover, too many classes give rise to too many modeling parameters affecting the utility of the model in practice. With extensive experimentations on varying n we have observed that 5–10 shot-classes strike a balance and result in good models. Seven shot-classes was found highly satisfactory for all videos we considered. For the study results reported in this paper we have used seven ($n = 7$) shot-classes and three segmentation techniques, namely EIL, ENG, and GIIL, respectively. Our study revealed a simple and intuitively appealing segmentation technique like EIL produces rather poor results compared to more sophisticated technique like GIIL.

In all three segmentation techniques discussed in this paper, a clip is made to belong to a shot-class if the average size of a GOP in this clip falls in the interval for that shot-class. The choice of intervals depends on the segmentation technique as detailed below. The following three segmentation methods have been studied: (1) equal interval lengths for all shot-classes (EIL), (2) equal number of GOPs in all shot-classes (ENG), and (3) geometrically increasing interval lengths for shot classes (GIIL). A relative comparison of these technique exhibits the effect of segmentation in modeling. Since GIIL is the best method we found and would recommend for use we discussed the other two techniques rather briefly. For the purpose of classification we assume that $g_{\min}$ and $g_{\max}$ denote the sizes of the smallest and the largest GOPs in bytes in the whole video, respectively.

2.2.1. Equal interval lengths for all shot-classes (EIL)

In this method, the interval $[g_{\min}, g_{\max}]$ is divided into n intervals of equal length. That is, the ith shot-class, $1 \leqslant i \leqslant n$, corresponds to the interval $[g_{\min} + (i-1)d, g_{\min} + id]$, where $d = (g_{\max} - g_{\min})/n$ is the common difference of the arithmetic progression comprising the boundaries of successive intervals. This method resulted in some shot-classes containing too few GOPs to make any statistical observation on those worth any significance. For example, see Table 1 which shows that out of 29,089 GOPs in the video only 68 and 7 GOPs belonged to shot-class 6 and 7, respectively. The following two methods could eliminate its drawback.

Table 1
Number of GOPs in shot-classes

Shot class	Number of GOP		
	EIL	GIIL	ENG
1	5825	2493	4155
2	16576	2434	4155
3	5076	4521	4155
4	1340	7940	4155
5	197	5915	4155
6	68	3290	4155
7	7	2496	4159

2.2.2. Equal number of GOPs in all shot-classes (ENG)

In this method, the interval $[g_{\min}, g_{\max}]$ is divided into n intervals such that each interval contains the same (or almost the same in case the total number of GOPs is not an integral multiple of n) number of GOPs. The shot-class boundaries in this method are identified by sorting the GOP sizes in non-decreasing order and noting the sizes for boundary values which meet the above requirement. This resulted in a significant improvement over the EIL technique.

However, it was observed a relatively small sub-interval of the entire range of GOP sizes contained a large fraction of all GOPs. This resulted in some of the shot-classes being too close making their inter-class separation less meaningful. Moreover, it was observed that the presence of a few *too small* and *too large* GOPs introduces undesirable biases in any model unless those extreme sizes are treated more like exceptions. The following technique attempted to incorporate remedies for these drawbacks.

2.2.3. Geometrically increasing interval lengths for shot-classes (GIIL)

In this technique, the smallest and the largest 1 percent GOPs are initially set aside as being too extreme. Let the GOP sizes corresponding to these 1 and 99 percentile points be referred to by a and b, respectively. The remaining interval of GOP sizes, namely, $[a, b]$, is partitioned into n sub-intervals. The successive partitioning boundaries of these intervals are made to increase in a geometric progression with a as the first term and $b = ar^n$ as the $(n+1)$th term. This results in n sub-intervals $[a, ar], [ar, ar^2], \ldots, [ar^{n-1}, ar^n]$ where $r = e^{(\ln b - \ln a)/n}$ is the common ratio of the geometric progression. Since a, b, and n are known, r and hence the intervals can be computed easily. The first ($[a, ar]$) and the last ($[ar^{n-1}, ar^n]$) sub-intervals are now redefined as $[g_{\min}, ar]$ and $[ar^{n-1}, g_{\max}]$, respectively to reinclude the extreme GOPs initially kept aside and so no data point gets excluded.

Having partitioned the GOPs into n ($n = 7$ in our study) shot-classes by any one of the partitioning techniques described above, the I-, B-, and P-frames in the GOPs of each shot-class S_i, $1 \leqslant i \leqslant n$, are separated to obtain three sets of frames $S_{i\mathrm{I}}$, $S_{i\mathrm{B}}$, and $S_{i\mathrm{P}}$ denoting the I-, B-, and P-frames in S_i, respectively. Thus, any of these shot-class finding algorithms partitions the clips into n shot-classes eventually separating all frames of the video into $3n$ data sets. The technical strength of a partitioning technique is decided on how accurately these data sets can be modeled by standard statistical distributions, which in turn could be used to build accurate models for the whole video.

3. Evaluation of the segmentation methods

We have presented three segmentation techniques namely, EIL, ENG, and GIIL, respectively. We first evaluate how ENG and GIIL segmentation techniques separate the video frames into well behaved shot-classes. Next, frame sizes of two full-length VBR video—one for each segmentation method—are synthetically generated using the model outlined in Section 3.1. To further validate the effectiveness of the segmentation techniques each synthetically generated VBR video has been compared with the original video. Following standard techniques in the literature, we show QQ-plots, and data-loss observed from simulation of leaky-bucket for these comparisons. Experiments were performed with MPEG-1 traces of full duration commercial movies including Crocodile Dundee, [2] ET [2], Jurassic Park, [3] Starwars [3], Terminator II [3], and The Silence of the Lambs [3] [14]. Most of these movies corresponded to about 2 hours of real time. Results reported in this paper are for the movie Crocodile Dundee. Results for other movies being similar have not been reported separately.

3.1. Markov-modulated gamma-based modeling of full-length video traffic

For the evaluation of the proposed segmentation methods, we briefly recapitulate a modeling technique proposed in [15,17]. In this model, (i) a Markov chain generates the GOP sequence of the synthetic video-trace, and (ii) axis-shifted Gamma distributions are utilized to generate frame sizes. The transition matrix of the Markov chain is obtained from analysis of the segmented video. Thus, for the same video two different segmentations may result in two different transition matrices. The number of states of the Markov chain is same as the number of shot-classes generated during the segmentation process. For this study we used seven shot-classes and obtained twenty one I-, B-, and P-frame-size distributions.

In the model a video *segment* in a shot-class is a *consecutive* sequence of clips belonging to that shot-class. The *length* of a segment is the number of GOPs in the segment. It is known that a Gamma distribution [2], or an exponential distribution [8,15] can be used to estimate the length of a video segment. For this study, we used only exponentially distributed segment lengths. The inter-shot class transition probabilities are approximated by normalized relative frequency of transitions in the original video.

This model used the Markov chain approximated by the transition matrix for inter- and intra-

[2] The authors thank Wu-Chi Feng for the MPEG traces.
[3] The authors thank Oliver Rose for the MPEG traces.

state transitions while generating a trace. Each shot-class corresponded to one state of the underlying 7-state Markov chain. After all frames of a GOP are generated following the Gamma distribution for the frames, the next state is determined by state transition matrix. The process is repeated until the desired number of frames have been generated. This state transition method generates video segments whose lengths are exponentially distributed. The detailed model and schemes for the generation of synthetic video trace are available in [17].

3.2. Cumulative distribution of I-, B-, and P-frames

Figs. 1 and 2 show GIIL and ENG techniques separate the I-, frames of the movie into seven shot-classes. Separations of B-, and P-frame are similar and have not been drawn separately. Table 2 shows the average frames sizes in the shot-classes for GIIL and ENG. Similar nature of spread for I and P frames despite the absolute difference in their frame sizes in Table 2 indicates GOPs with relatively larger I frames tend to have relatively larger P frames as well.

3.3. QQ plots for visual inspection

The *quartile–quartile* (QQ) plot of two data-sets is a well-known visual inspection method for verification of their similarity. In this method, for a

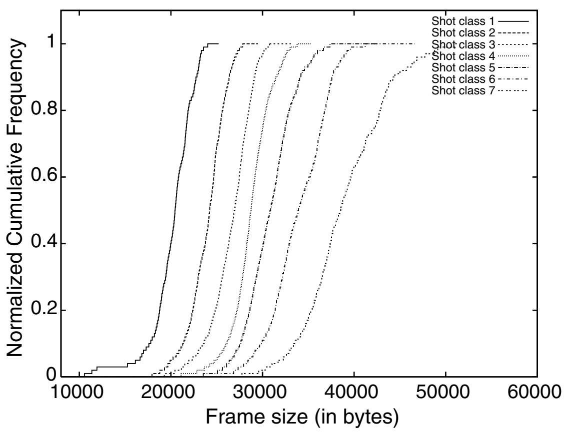

Fig. 1. Separation of I-frames using GIIL (threshold factor = 1).

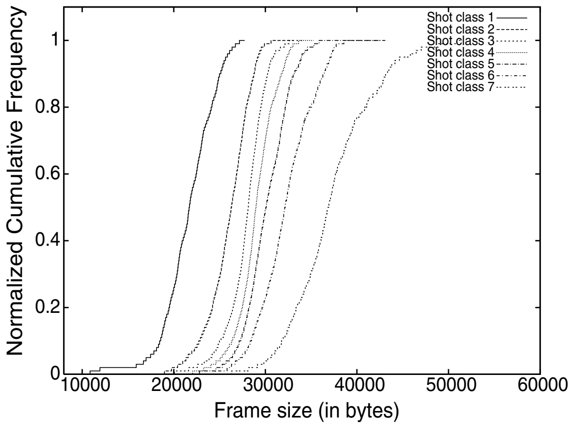

Fig. 2. Separation of I-frames using ENG (threshold factor = 1).

given percentile rank (say, 10%), a pair of values of data (say, $\langle 1293, 1243 \rangle$) from two data-sets is obtained. Usually, several pairs of values are collected for desired range of percentile values, and are plotted. If two data-sets are identical, a straight line described by $y = x$, is obtained. Thus, closer the plot to the line $y = x$, better the similarity between the data-sets. The plots in Figs. 3 and 4 depict the similarity of original VBR video data-set with that synthetically generated using ENG and GIIL with threshold factor of 1. The x- and y-axis of the QQ plot corresponds to frame sizes in the original trace and in the modeled trace, respectively. A near-overlap of the QQ plot with the $y = x$ line suggests the traffic models based on both ENG and based on GIIL partitioning are quite accurate. Of the two methods, GIIL is marginally superior.

3.4. Leaky-bucket simulation for buffer overflow loss observation

A QQ plot depicts *global* similarity of two data-sets. However, if the elements of these two data sets are ordered by frame index, as in case of actual video frames, a QQ plot does not reveal any information about local distributions of the frames. For instance, one dataset may have all the large data values together, but another dataset may have these large and small data values interleaved and yet both may show identical QQ plots. For

Table 2
Average I-, B-, P-frame size in shot-classes

Frame type	Shot class	Average frame size (in bytes)			
		GIIL	Difference	ENG	Difference
I	1	20166.73		21622.92	
	2	23968.30	3801.57	25998.68	4375.76
	3	26583.57	2615.27	27889.70	1891.02
	4	28692.57	2109	28963.41	1073.71
	5	30803.21	2110.64	30147.70	1184.29
	6	33915.01	3111.8	32213.62	2065.92
	7	38915.22	5000.21	37236.78	5023.16
B	1	600.20		793.26	
	2	1217.59	617.39	1930.39	1137.13
	3	2156.49	938.9	2913.41	983.02
	4	3556.16	1399.67	3799.45	886.04
	5	5372.35	1816.19	4830.36	1030.91
	6	7604.58	2232.23	6372.85	1542.49
	7	12055.55	4450.97	10557.03	4184.18
P	1	3204.32		3911.61	
	2	5269.39	2065.07	6809.16	2897.55
	3	7212.75	1943.36	8324.93	1515.77
	4	9158.11	1945.36	9482.27	1157.34
	5	11580.79	2422.68	10852.80	1370.53
	6	14593.84	3013.05	12919.13	2066.33
	7	19786.70	5192.86	18085.89	5166.76

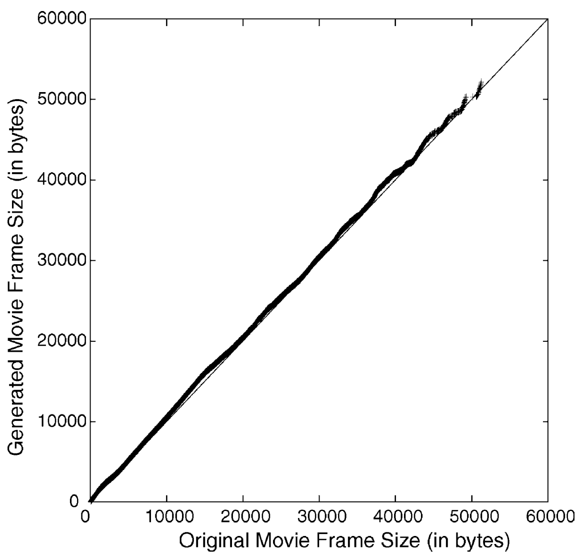

Fig. 3. QQ plot for the whole movie generated using ENG (threshold factor = 1).

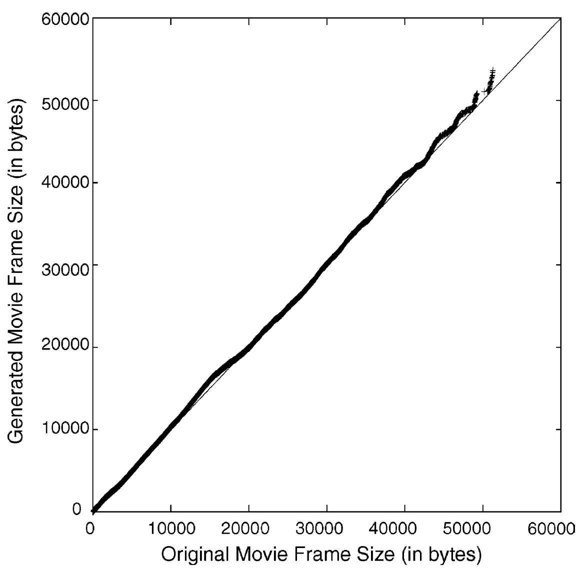

Fig. 4. QQ plot for the whole movie generated using GIIL (threshold factor = 1).

communication of VBR videos over B-ISDN, temporal ordering of the frames plays a critical role in DLR; for a given data transmission-rate, occurrence of long runs of large-size frames (*known as burstiness*) has higher DLR than the absence of them. Hence, temporal burstiness of original VBR video must be preserved in the frame-size data generated by a good model. Most commonly used test (see [18] for algorithmic approaches) for measuring this data size behavior is leaky-bucket simulation where the data is passed through a communication channel having a generic buffer with capacity c, and *drain rate factor* d. For our study, buffer capacity is expressed in terms of mean frame size of VBR source, and is *independent of* d. So, for a 25-frames per second source, $c = 20$ ms corresponds to one half of a mean frame size of the VBR video. The *drain rate factor* d is the ratio of the number of bytes actually drained (that is, transmitted) per second to the average data rate of the incoming VBR video. A drain rate factor of 1 or less would obviously cause too much of data loss and is unlikely to be ever used in practice. On the other hand, a high (say, 4 or more) drain rate factor would cause wastage of channel capacity because of hardly any improvement of video quality beyond such drain rates. Drain rate factors in the range [2,4] are often used in practice. The results included in the paper belong to this range. Algorithmic approaches proposed in [18] may be followed for studying more detailed impact of drain rates and burstiness on loss patterns.

Due to their relatively larger sizes I-frames unlike B- and P-frames are more likely to be affected or lost during transmission. Moreover, the nature of MPEG compression/decompression algorithms suggests the loss of I-frames is far more crucial than the loss of B- or P-frames for the visual quality of received video.

Figs. 5–7 show the percentages of I-frames affected in the original video and also in the synthetic videos generated by the two models. One model uses ENG as its segmentation method and generates a full-length video using the parameters of the original video as discussed in Section 3.1. Results for various values of the threshold factor used in segmentation are shown. The other model

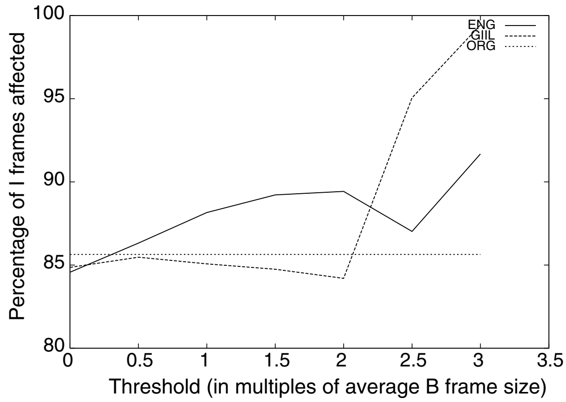

Fig. 5. Effect of threshold on I frame loss (drain rate factor = 2, buffer = 20 ms).

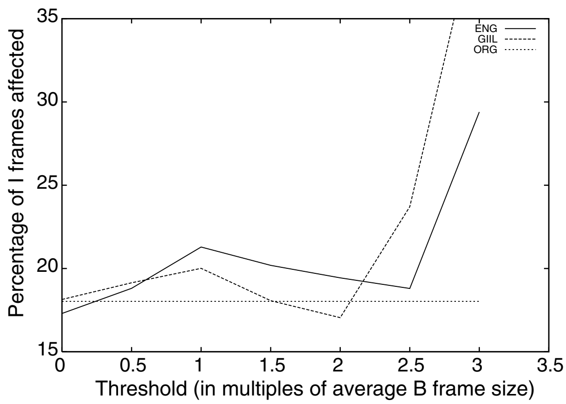

Fig. 6. Effect of threshold on I frame loss (drain rate factor = 3, buffer = 20 ms).

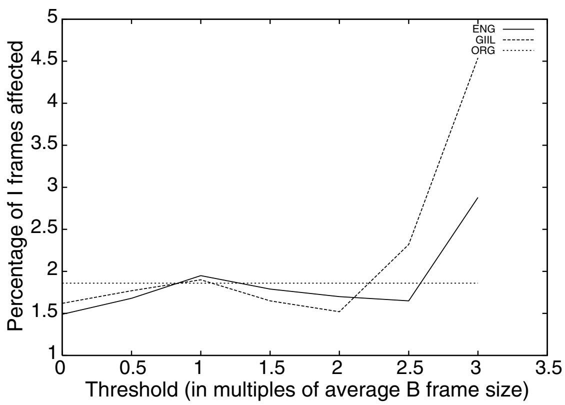

Fig. 7. Effect of threshold on I frame loss (drain rate factor = 4, buffer = 20 ms).

uses GIIL instead of ENG as its segmentation technique. Figs. 8 and 9 show the percentages of all rather than only I-frames affected and data loss. The closer the plot for a segmentation method (labeled ENG and GIIL) to that of the original video (labeled ORG) the better the method. For example, the horizontal line labeled ORG in Fig. 5 shows about 85% of I-frames of the original movie are affected when the full movie is transmitted over a communication channel whose capacity corresponds to a drain rate factor of 2 with 20 ms buffer. This is constant because it has nothing to do with the threshold. The curves for ENG and GIIL show the percentages of I-frames affected for various threshold values used in these segmentation techniques. Figs. 6 and 7 show this comparison for a drain rate factor of 3 and 4, respectively. The plots indicate GIIL method is very good for a wide range of drain rate factors, that is, communication channel capacities. It may be noted we have observed that for a practical drain rate factor, say one in the range 2 to 4, there is hardly any loss of B- or P-frames in either the original or the synthetically generated traces and so these plots haven't been shown separately. Fig. 8, for an example, shows the percentages of all, that is I-, B-, and P-frames taken together, affected for a drain rate factor of 3. Since the movie under consideration has a (6,3) cyclic format, the number of I-frames is $\frac{1}{6}$th of all frames and the comparison of Fig. 8 (showing about 3% overall frames affected) with Fig. 6 (showing about 18% I-frames affected) reveals almost all frames affected are I-frames. Whereas Figs. 5–8 show the percentages of the number of frames *affected*, that is, frames partly or fully lost, Fig. 9 depicts the actual percentages of data loss of the original and the generated movies. It again shows the model using either ENG or GIIL very closely mimics the original movie for a wide range of threshold and GIIL still exhibits its superiority over ENG. All these plots showing the leaky-bucket simulation results indicate there is a wide range, say $[0.5, 2]$, of threshold factors over which GIIL results in very good model of VBR video. So, the choice of a threshold is not that crucial and a value close to the average size of B-frame can be used to get a good model. It may be noted the choice of a threshold decides clip formation which ultimately decides segmentation and Gamma model parameters. So, it's important to choose a workable threshold value. Figs. 5 and 6 indicate the range $[0.5, 2]$ of threshold values for which the model (and hence data loss patterns) is quite accurate and that values beyond 2.5 should not be used.

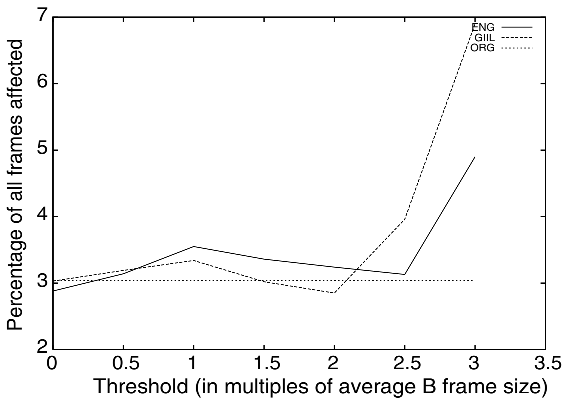

Fig. 8. Effect of threshold on all frames affected (drain rate factor = 3, buffer = 20 ms).

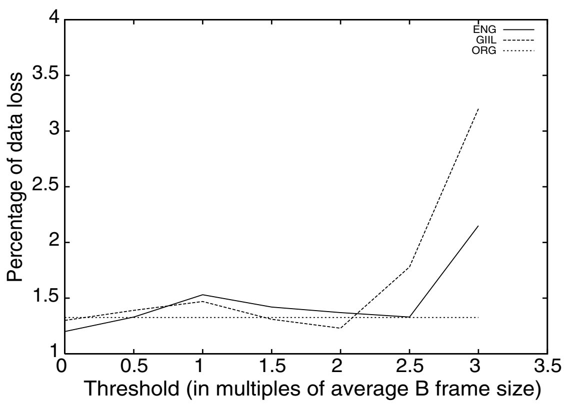

Fig. 9. Effect of threshold on overall data loss (drain rate factor = 3, buffer = 20 ms).

4. Discussion and conclusions

A long-duration VBR video trace such as a commercial movie contains several hundreds of thousands of frames having intricate variations in their sizes. Any successful characterization of this frame-size behavior needs to address how to partition these frames into statistically characterizable segments. The two major issues in segmentation are the selection of the number of segments called shot-classes, and a formulation of criteria following which the frames can be partitioned into those shot-classes so that I-, B-, and P-type frame-size variation in each shot-class can be statistically modeled. Segmentation plays a key role in the development of frame-size-based traffic models used for understanding the effect of data loss during transmission of MPEG-compressed VBR video over broadband links. The limited success of past efforts in frame-size-based modeling of full-length VBR video is due to the complex nature of variations in frame-size data in an unsegmented long-duration trace. Three different compression techniques applied to I-, P-, and B-frames produce different amounts of compression. Moreover, different segments of a lengthy video produce frames of different sizes because of composition or content of picture, and due to temporal similarity of adjacent pictures.

In this paper, we have proposed and evaluated three segmentation methods. We have used QQ plots to show visual similarity of model generated synthetic VBR video data-sets with the data-set of the original video used for constructing the model. Similarity of local burstiness of model-generated and original VBR video traces have been validated using the leaky-bucket simulation technique. The synthetic traces were generated following the proposed segmentation methods described in Section 2. The recommended segmentation technique, named GIIL, uses a user-chosen threshold. The choice of this threshold has been discussed and it was seen a wide range of workable threshold values exists.

The results indicate segmentation of a long-duration video in seven shot classes based on geometrically separated shot-class boundaries is very effective in modeling segment-based frame-size behavior. The findings also strengthen our earlier work on segmentation-based approach towards developing frame-size traffic models [15,17]. The authors are now working towards applying this segmentation approach in modeling multiplexed videos.

References

[1] K. Chandra, A.R. Reibman, Modeling one- and two-layer variable bit rate video, IEEE/ACM Trans. Network. 7 (3) (1999) 398–413.

[2] A.M. Dawood, M. Ghanbari, Content-based MPEG video traffic modeling, IEEE Trans. Multimedia 1 (1) (1999) 77–87.

[3] M. Frey, S. Nguyen-Quang, A gamma-based framework for modeling variable-rate MPEG video sources: The GOP GBAR model, IEEE/ACM Trans. Network. 8 (6) (2000) 710–719.

[4] D.P. Heyman, The GBAR source model for VBR video conferences, IEEE/ACM Trans. Network. 5 (4) (1997) 554–560.

[5] D.P. Heyman, A. Tabatabai, T.V. Lakshman, Statistical analysis and simulation study of video teleconferencing traffic in ATM, IEEE Trans. Circ. Syst. Video Technol. 2 (1992) 49–59.

[6] Y. Iraqi, R. Boutaba, Supporting MPEG video VBR traffic in wireless networks, Comput. Commun. 24 (2001) 1188–1201.

[7] M. Krunz, R. Sass, H. Hughes, Statistical characteristics and multiplexing of MPEG streams, in: Proceedings of the IEEE INFOCOM 1995, 1995, pp. 455–462.

[8] M. Krunz, S.K. Tripathi, Scene-based characterization of VBR MPEG-compressed video traffic, in: Proceedings of the 1997 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, 1997, pp. 192–202.

[9] D.M. Lucantoni, M.F. Neuts, A.R. Reibman, Methods for performance evaluation of VBR video traffic models, IEEE/ACM Trans. Network. 2 (2) (1994) 176–180.

[10] P. Manzoni, P. Cremonesi, G. Serazzi, Workload models of VBR video traffic and their use in resource allocation policies, IEEE/ACM Trans. Network. 7 (3) (1999) 387–397.

[11] B. Melamed, D.E. Pendarakis, Modeling full-length VBR video using markov-renewal-modulated TES models, J. Selected Areas Commun. 16 (5) (1998) 600–611.

[12] M. Nomura, T. Fuji, N. Ohta, Basic characteristics of variable rate video coding in ATM environment, J. Selected Areas Commun. 7 (1989) 752–760.

[13] Q. Ren, H. Kobayashi, Diffusion approximation modeling for markov modulated bursty traffic and its applications to bandwidth allocation in ATM networks, J. Selected Areas Commun. 16 (5) (1998) 679–691.

[14] O. Rose, Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems, in: Proceedings of the 20th Annual Conference on Local Area Networks, 1995, pp. 397–406.

[15] U.K. Sarkar, S. Ramakrishnan, D. Sarkar, Modeling full-length video using markov-modulated gamma-based

framework, in: Proceedings of IEEE Globecom 2001, Multimedia QoS/Video Applications, 2001, pp. 1–5.

[16] U.K. Sarkar, S. Ramakrishnan, D. Sarkar, Segmenting full-length VBR video into shots for modeling with markov-modulated gamma-based framework, in: Internet and Multimedia Systems II, Proceedings of SPIE, vol. 4519, 2001, pp. 191–202.

[17] U.K. Sarkar, S. Ramakrishnan, D. Sarkar, Modeling full-length video using markov-modulated gamma-based framework, IEEE/ACM Trans. Network 11 (4) (2003) 638–649.

[18] C. Tryfonas, A. Varma, S. Varma, Efficient algorithms for computation of the burstiness curve of video sources, in: Proceedings of International Teletraffic Congress (ITC-16) 99, 1999.

[19] J.-L. Wu, Y.-W. Chen, K.-C. Jiang, Two models for variablebit rate MPEG sources, IEICE Trans. Commun. E78-B (1995) 737–745.

Uttam K. Sarkar received his B.Tech., M.Tech., and Ph.D. degrees from the Department of Computer Science and Engineering of the Indian Institute of Technology, Kharagpur. His current research interests include Video Traffic Modeling, Multimedia Systems, and Combinatorial Algorithms. He is presently an Associate Professor at the Indian Institute of Management Calcutta. His e-mail address is uttam@iimcal.ac.in.

Subramanian Ramakrishnan received his B.Stat. (Honours), M.Stat., and Ph.D. degrees in 1974, 1975, and 1982 respectively from the Indian Statistical Institute, Calcutta, India. Since then he has been a faculty member at the University of Miami where he is currently an Associate Professor in the Department of Mathematics. His research interests include the foundations of probability theory and stochastic processes, theory of gambling, and more recently, discrete-time queueing networks and theoretical computer science. His e-mail address is ramakris@math.miami.edu.

Dilip Sarkar (SM'96) received his B.Tech. from the Indian Institute of Technology Kharagpur, M.S. from the Indian Institute of Science Bangalore, and Ph.D. from the University of Central Florida. He is an Associate Professor of Computer Science at the University of Miami, Coral Gables. His research interests include VBR video traffic modeling, multimedia communication over broadband and wireless networks, middleware and Web computing, design and analysis of algorithms, parallel and distributed processing. He is a senior member of the IEEE, a member of IEEE Computer Society and the Association for Computing Machinery. His e-mail address is sarkar@cs.miami.edu.