MPEG-based Interactive Video Streaming: A Review

Kostas E Psannis and Marios G Hadjinicolaou
Dept of Electronic & Computer Engineering
Brunel University, Uxbridge, Middlesex, UB8 3PH,UK

Abstract: - With the establishment of MPEG video coding standards, it is expected that many video sequences will be encoded in MPEG formats. However, the implementation of videocassette recording (VCR) functionalities with the MPEG coded video presents some technical challenges that have not yet been well resolved. There are three different rates the video can flow from the disk of the server to the client display; retrieval rate, network transmission rate and consumption rate. The contribution of this paper is to evaluate the various proposed mechanisms for supporting interactive operations according to the above requirements.

Key-Words: - Multimedia Communication-Interactive Operations- VCR control

1 Introduction

A video streaming system should be capable of delivering concurrent video streams to a large number of users. The realization of such a system presents several challenges, such as the high storage-capacity and throughput in the video server and the high bandwidth in the network to delivery large number of video streams. With the rapid progress of video compression technologies, magnetic storage subsystems and broadband networking, these problems are being solved and video streaming applications are becoming increasingly popular [1].

To improve the marketability of video streaming services and accelerate their wide-scale deployment, these services must support user interactivity at affordable cost. Attempts to make video interactive are almost as old as video itself. Interactive video has different meanings for different people. Some people see the choice from many different channels already as interactivity. On the other hand some other would require the selections of a local video store to be available in real time, on-demand, through a broadband connection. It is generally believed that interactive program is a program or service in which the content itself, or the presentation manner of the content or even the presentation order of the content can be affected by the viewer. The client should interact with the content of the presentation deciding the viewing schedule with the full range of interactive functions. This gives the desired multilevel QoS to the client. Full range of interactive functions include play/resume, stop, pause, Jump Forward (JF)/ Jump Backward (JB), Fast Forward (FF)/ Fast Rewind FR), Slow down (SD), and Slow Reverse (SR), Rewind [2].

The difficulty of supporting interactivity varies form one interactive function to another. A stop or pause followed by resume are relatively easy to support since they don't require more bandwidth than what is already allocated for normal playback. On the other hand, Fast Scanning {(FF), (FR), (JF), (JB)) involves displaying frames at multiple times the normal rate. Transporting and decoding frames at such rate is prohibitively expensive and is not feasible with today's hardware decoders. In addition, the implementation of the full VCR functionality with the MPEG coded video in not a trivial task. MPEG video compression is based on motion compensated predictive coding [3]-[5]. The MPEG structure allows straightforward realization of the forward normal play, but imposes several constraints to other trick interactive mode. Several approaches have been proposed to support interactivity in a VOD system. Interactive functions can be supported by dropping parts of the original MPEG-2 video stream [6]-[8]. Typically, dropping is performed after compression and aims to reduce the transport and decoding requirements without causing significant degradation in video quality. Alternatively interactive functions can also be supported using separate copies of the movie that are encoded at lower quality of the normal playback copy [9], [10]. Other conventional scheme that support interactive functions displays frames at rate

Corresponding Author: K E Psannis
Email: kpsannis@hotmail.com

much higher than the normal playback, for example 90 fps [11]. Moreover full interactive functions can be supported by storing multiple differently encoded versions of the video stream at the server. A normal version is used for normal playback, while several other versions are used for Fast/Slow/Jump Forward (Rewind)/Rewind at variable speedups. Full interactive functions are produced by encoding every N^{th} frame as a sequence of I-P(M) frames[12],[13].

In this work, we present various methods for supporting interactivity. Specifically, we analyze the impact of interactivity on the server load, network bandwidth and decoder complexity respectively. The rest of the paper is organized as follows. In Section 2 describes the MPEG structure. . Section 3 presents the requirements for interactivity. Section 4 analyses the various proposed methods for supporting interactivity. Section 5 summarizes the various methods. Finally, conclusions are given in section 6.

2. MPEG Video Coding

An MPEG video stream comprises intra-frames (I), predicted frames (P), and interpolated frames (B) According to MPEG coding standards, I-frames are coded such that they are independent of any other frames in the sequence; P-frames are coded using motion estimation and each one has a dependency on the preceding I- or P- frame; finally the coding of B- frames depends on the two "anchor" frames - the preceding I/P frame and the following I/P frame. An MPEG coded video sequence is typically partitioned into small intervals called GoP (Group of Pictures). Two parameters, M and N, describe the succession of I- P- B- frames, where

N is the distance between two successive I frames, defining a "group of pictures" (GoP).

M is the distance between a P- frame and the immediately proceeding I- or P- frame. M is usually set to 3.

N can be defined as follows

$$N = \begin{cases} \alpha \times M, & M > 0,\ \alpha. > 0, \quad I - P - B \ frames \\ N = \alpha & M = 1,\ \alpha. > 0, \quad I - P \ frames \\ N > 0, & M = 0, \quad I - \ frames \\ M, & N = M > 0, \quad I - B \text{ frames} \end{cases}$$

The video during the normal mode is coded with all I-P-B frames in order to achieve high compression ratios for the transport over the network with minimum bandwidth resources. To enable continuous playback at the Set Top Box (STB), MPEG-2 frames are transmitted over the network according to the decoding order. This is because of the interpolative nature of B-frames. Hence, the decoding (transmission) order of MPEG-2 sequence is different from its temporal (display playback) order. An example of the temporal and transmission orders of the normal mode is given below (Figure.1)

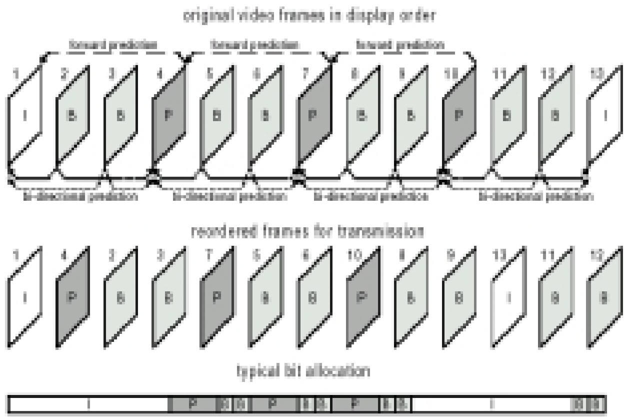

Figure 1: A GoPs structure (N=12, M=3) with typical bit allocation per frame

3 Requirements of Interactivity

There are three different rates at which a video can flow from the disk of the server to the client display, which should be taken into consideration while analyzing interactive functions. First, the video data has a retrieval rate going from the server's disk array to the main memory. Second, the data has a network transmission rate going from the server's memory to the client's memory. Lastly, the client decoder has a consumption rate at which it displays data from the client's memory.

3.1 Server load

A GoP is used as the retrieval block; normally stored contiguously on disk. Assuming that every I/O cycle, 1 GOP (15 frames) is retrieved, the disk executes one rotation, seek and transfer. If four different frames from three GoPs (12 frames) and three frames form the next GOP are retrieved during each cycle, the disk must perform four rotations, seeks and retrievals for every 15 frames which increases the overhead (load) on the server to process Fast Play (F-Play) request

3.2 Network Bandwidth

Since the network bandwidth usually is the highest concern, the video during the normal version is coded with all I-P-B frames in order to achieve high compression ratios for the transport over the network with minimum bandwidth resources. In addition it is desirable to support interactive operations with minimum requirement in the network bandwidth.

3.3 Client Resources

Client resources refer to the memory and CPU requirements that are needed to process and decode a received frame. It can be seen from normal forward mode, it is necessary to keep exactly two decompressed frames in the memory buffer for decoding a frame that references these two frames. Since decompressed frames are of the same size, buffer requires space for two decompressed frames to do the decoding for normal play.

4 Simulation Results

In order to determine the additional resources required for the various proposed schemes to support interactivity computer simulation can be used. For our experiment we used an MPEG-2 encoder developed by the Software Simulation Group (SSG) [14]. We encoded 180 frames of the football clip. The Group of Pictures (GoPs) format was N=15 and M=3. Therefore we had a larger I- or P- frame, followed by two smaller B-frames. Figure 2 depicts the bit rate allocation for the football sequence

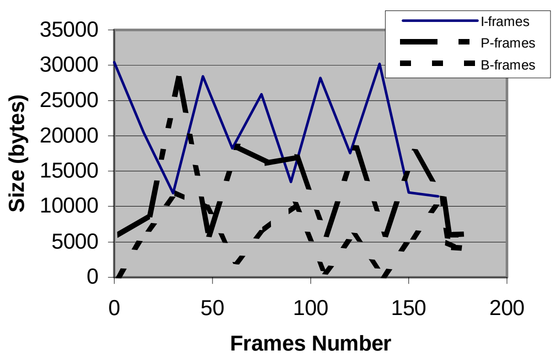

Figure 2: Bit rate allocation for the football sequence

According to Figure 2 the average bandwidth during the normal playback is given by

$$30\text{fps} \times Average\,(\text{IPB})\text{Size} \times 8\, {}^{bits}\!/_{byte} = 1.98MBps$$

where,

$$Average(\text{IPB})\text{Size} = \frac{\text{I}_{\text{average}}}{\text{N}} + P_{average}\,\text{x}\left(\frac{1}{\text{M}} - \frac{1}{N}\right) + B_{average}\text{x}\left(1 - \frac{1}{M}\right)$$

4.1 Transmitting frames at multiple rates [11]

Interactive operations can be supported by transmitting frames at multiple rates the normal frame rate. Assume that a video is transmitted to a client with a frame rate of 30 frames per second (fps) during normal playback. In the increased playback rate approach, for a speedup of four, the server would deliver120fps. Figure 3 depicts the supported speedups as a function of transmitted frames.

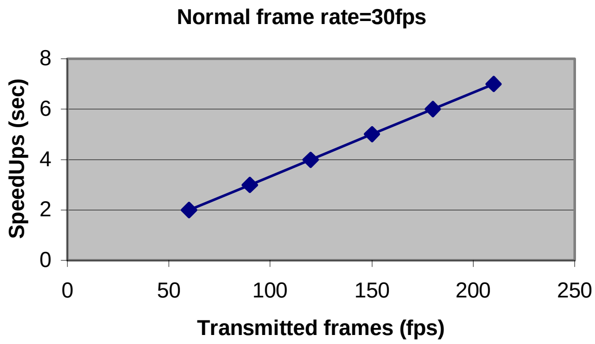

Figure 3: Variability in the supported speedups

The main drawback of this method is that the hardware decoder must be able to display the video using the new playback rate. It is the responsibility of the receiver to do an intelligent thing and play the stream k-times the normal playback. This might be very difficult, because it requires k-times more decoding power. The receiver could perhaps filter out some of the stream and decode only parts relevant to Fast playback operation. It is worth mentioning that currently, there are no hardware or software MPEG-2 decoders that can run faster than 30fps.

4.2 Send only I - P frames [7]

This method sends only the I- and P-frames to support Fast Forward operations. Transmitting only the I- and P- frames of each GoPs a limited number of speedups can be achieved. The main problem with this method is that it increases the load on the network (the frame rate remains constant). During the normal playback, the entire sequence is retrieved from the disk and sent to the client where the frames are reordered and presented to the decoder. Assuming a playback rate of 27fps (frames per second), 3 GoPs are sent to the client every second. During the Fast Forward the fist 3 I-frames ($I_1\, I_{10}\, I_{29}$) and 6 P-frames ($P_4\, P_7\, P_{13}\, P_{16}\, P_{19}\, P_{22}\, P_{25}$)

are sent to the client. This is done for 3 consecutive sequences. So instead of sending 3 GoPs (27 frames) consisting of I-, P- and B frames during each cycle as in normal playback, the I- and P- frames from 3 sequences (9 GoPs) are sent. This implies that 27fps are sent. Figure 4 shows the increase (%) in the network bandwidth for different GoPs length (N) using typical characteristics of MPEG-2 compressed video (football sequence).

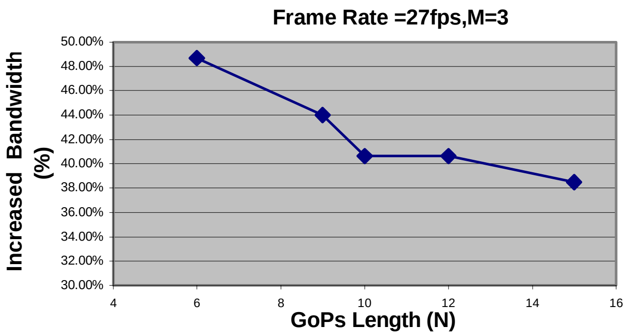

Figure 4:The increased percentage of the network bandwidth as a function of N.

The graph in Figure 4 shows that there is not a linear increase (%) in the network bandwidth during the Fast Forward (FF) mode as a function of N.

4.3 GOP-Skipping [8]

The GoP-Skipping method maintains the same playback frame rate and skips entire GoPs. This method can achieve variable speedups. Figure 5 shows the variable number of speedups as a function of skipped GoPs for N=15 and frame rate 30 fps.

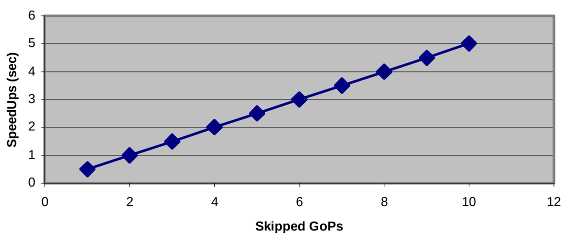

Figure 5:Number of speedups as a function of skipped GoPs

One limitation of this approach is that only independently decode-able GoPs can be used. If open GoPs are used, some extra processing must occur at the client-site increasing the requirements of the decoder complexity.

4.4 Partial GoPs-skipping [6]

Another approach is to partial skip GoPs rather than skip full GoPs. Like the skipping segment method, the same playback rate is maintained at the client. The bit rate required to be sent over the network will be increased, as well as the resources at the server. Figures 6 shows the increase in the bit rate as a function of the supported speedups for GoPs Length N=15 and M=3 using typical characteristic of MPEG-2 compressed video.

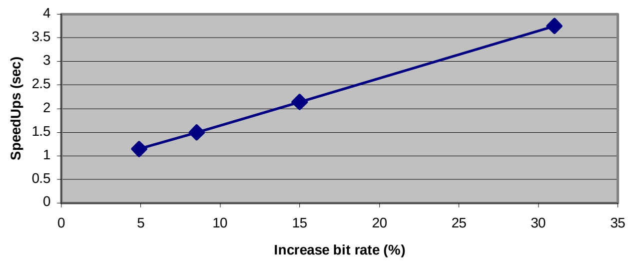

Figure 6:Relative increase in the speedups as a function of the transmitted frames.

4.5 Skipping row of frames [9]

This approach is based on encoding separate copies of the movie to be used for interactive operations in a Video On Demand (VOD) system. Each copy is generated by skipping rows of frames before compression. The server maintains multiple different encoded versions of each movie. The normal version is used for normal-speed playback. The other versions, which are referred to as the scan version, are used for Fast Scanning (FS) operations. Each scan version is used to support both Fast Forward Scanning (FFS) and Backward Fast Scanning (BFS). For a given speedup factor, the corresponding scan version is obtained by encoding a subset of the raw uncompressed frames of the original movie at a sampling rate of 1-to-s, where s is the skip factor. Generating separate copies for Fast Scanning operations comes at the expense of extra storage of the server and some variability in the quality of the motion picture during the Fast Scanning (FS) periods. Figure 7 depicts the percentage of the increase in the storage overhead as a function of N using typical characteristics of MPEG-2 compressed video (football sequence) with skip factor $s = 4$ and $S_P = S_B = 0$. For n scan versions (variable speedups) with skip factors

$s_1\ s_2 ....s_n$, the relative increase in the storage requirement is given by $\frac{\sum_i W_{scan}(s)}{W_{normal}}$.

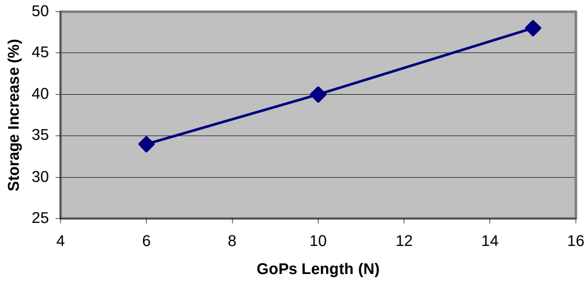

Figure 7: Relative increase of the storage as a function of GoPs Length (N).

4.6 Alternative special file [10]

In this method a special file is created specifically for use in Fast Forward/ Fast Rewind mode and multi-resolution viewing. The encoding of the special file can be done in two ways. If the original uncompressed file is available, then every n-th frame is encoded for a speedup of n. If the original file does not exist, the compressed stream is first uncompressed and every nth frame is then re-compressed. The alternative file can be created based on the motion within the original file. . During periods of little motion, large numbers of frames are skipped from the original file. During periods of high motion, lower numbers of frames are skipped. Figure 8 depicts the increase in the storage at the server as a function of GoPs (N), using typical characteristics of MPEG-2 compressed video with the frame format of the special file $I\ B\ B\ B\ B\ I$ and encoding every third frame of the original file.

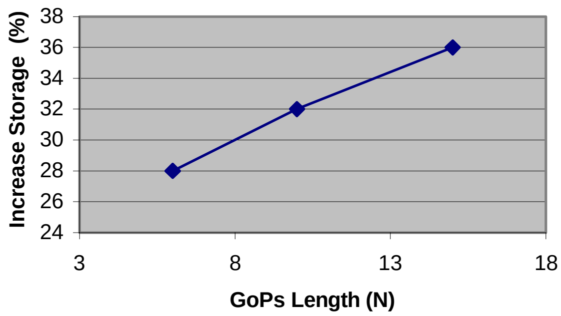

Figure 8: Relative increase of the storage as a function of N.

4.7 Generate P (Marionette) frames [12], [13]

To support interactive functions, the server maintains multiple, different encoded versions of each movie. One version, which is referred to as the normal version is used for normal-speed playback. The other versions are referred to as interactive versions. Each interactive version is used to support Fast/Jump Forward/Backward Slow Down/Reverse and Reverse at a variable speedup. The server switches between the various versions depending on the requested interactive function. Note that only one version is transmitted at a given instant time. The corresponding interactive version is obtained by encoding every N-th (i.e., uncompressed) frame of the original movie as a sequence of I- P(Marionette) - frames $(N_{interactive} = variable, M_{interactive} = 1)$. Effectively this results in repeating the previous I-frame in the decoder, enhancing the visual quality during the interactive mode. Two rate control algorithms have been proposed in order to bound the size of interactive stream. Note that this method supports full range of interactive functions with minimum additional resources. Generate separate copies for interactive operations come at the expense of minimum extra storage at the server. Figure 9 depicts the relative storage overhead of the interactive version as a function of SpUp. For SpUp>5, the storage overhead of the interactive version is not more than 10% of the storage requirement of the normal version.

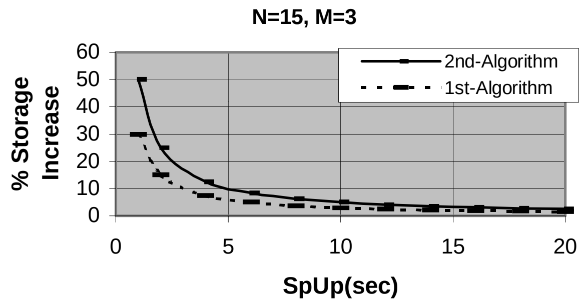

Figure 9: Storage increase of the interactive version as a function of SpUp.

Note that for n-interactive versions the relative increase in the storage overhead at the server is given by

$$\frac{\sum_{i=1}^{n} W_{ExtraStorage}(N_{interactive})}{W_{normal}}$$

5. Summary

Table 1 summarizes the various aspects of the proposed methods.

Table 1: Summary of the methods

Method	Additional Resources	Speedups	Implementation
Transmitting frames at multiple times	Dramatically Increased Server, net , decoder.	Any	Unknown
Send only I – P Frames	Significant Network Bandwidth	Limited	Possible problems with Fast Reverse
GOP Skipping	No increase if 1GoP/disk	Any	Problem with open GoPs
Partial GoP skipping	Increase Load server /network	Any	Possible problems with Fast Reverse
Skipping raw of frames	Significant extra storage at the server	1 per file	Increase decoder complexity for FR
Alternative special File	Extra Storage at the server	1 per file	Variable consumption rate at the decoder
Generate P(Marionette) Frames	Minimum Storage at the server	Variable	Feasible

9 Conclusions

In this paper, we presented a review of various proposed methods for supporting interactivity. Extensive computer simulations have be used in order to determine the additional resources at the server load, network bandwidth and decoder complexity during the interactive mode. The outcome of the evaluation shows that only the "generate P(Marionette) frames" method support full interactivity with minimum additional overheads.

References:

[1] C-W. Lin, J. Youn, J. Zhou, M-T. Sun, "MPEG video streaming with VCR functionality", IEEE Trans on Circuits and Systems for Video Technology, vol 11 , no 1, Feb 2001, pp 415-425

[2] Miranda Ko, Irene Koo, "An Overview of Interactive Video On Demand", technical paper, University of British Columbia, Dec 1996

[3]Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5Mbits/s, (MPEG-1), Oct. 1993

[4]Generic Coding of Moving Pictures and Associated Audio. ISO/IEC 13818-2, (MPEG-2), Nov 1993.

[5]Coding of Moving Pictures and Associated Audio.MPEG98/W21994, (MPEG-4), Mar.1998.

[6] Banu Ozden, Alexandros Biliris, Rajeen Rastogi, Avi Silberschatz. "A Low-Cost storage Server for Movie on Demand Databases". In: Proc. of the 20th VLD Conference, Santiago, Chile, 1994.pp 594-605

[7] HJ Chen, A. Krishnamurthy, TDC Little, and D. Venkatesh, "A Scalable Video-on Demand Service for the Provision of VCR-Like Functions," Proceedings IEEE Multimedia, 1995, p. 65-71.

[8] M-S. Chen, D.D. Kandlur, and P.S. Yin, "Support for fully interactive playout in a disk-array-based video server". In Proc of Second International Conference on Multimedia, Oct 1994, pp 391-398.

[9] Marwan Krunz, George Apostolopoulos. "Efficient Support for interactive scanning operations in MPEG-based video on video on demand". Multimedia Systems, vol.8, no.1, Jan. 2000, pp.20-36

[10] Michael Vernick, Chitra Venkatramani, Tzi-Cher Chinueh "Adventure in Building the Stony Brook Video Server". In Proc ACM Multimedia, Boston, Nov 1996, pp 287-295.

[11] Jayanta K Dey-Sircar, James D.Salehi, James F Kurose, Don Towsley. "Providing VCR Capabilities in Large-Scale Video Servers" In: Proc. ACM Multimedia Conference, Oct 1994, San Francisco, CA, pp 25-32

[12] Kostas Psannis, Marios Hadjinicolaou, Argy Krikelis "Full Interactive Functions in MPEG-based Video on Demand Systems", In Mastorakis, G. Antoniou Editors, Recent Advances in Circuits, Systems and Signal Processing, 2002, Published by WSEAS Press, pp 419-424.

[13] Kostas Psannis, Marios Hadjinicolaou and Argy Krikelis "Multimedia Streaming of Full Interactive Content Over IP Networks", submitted to IEEE Trans on Circuits and Systems for Video Technology (under review)

[14] MPEG Software Simulation Group,"MPEG-2 Encoder/ Decoder Version 1.2" ftp://ftp.mpeg.org/pub/mpeg/mssg