Full Interactive MPEG-based Wireless Video Streaming

Kostas E. Psannis
Department of Computer Science and Engineering
Nagoya Institute of Technology, Japan
Kostas.Psannis@nitech.ac.jp

Yutaka Ishibashi
Department of Computer Science and Engineering
Nagoya Institute of Technology, Japan
ishibasi@nitech.ac.jp

***Abstract* – Despite much research in the field of mobile media, delivery of real time interactive video over noisy wireless channels remains a challenging problem. A major limitation in wireless networks is that mobile users must rely on a battery with a limited supply of energy. Effectively utilizing this energy is a key consideration in the design of wireless networks. In this paper, we jointly consider source coding and dynamic allocation of the physical layer communication resources in an efficient method. The proposed efficient approach achieves acceptable visual quality using minimum transmission energy, while satisfying delay constraints. Mechanism for controlling the interactive streams is also presented and its effectiveness is assessed through extensive simulations.**

***Index Terms* - Wireless Video, MPEG, Interactive Operations,**

I. INTRODUCTION

The imminent arrival of third generation wireless networks will dramatically increase the viability of mobile media communications. In particular, it will make it possible to hold video conferencing sessions using portable devices. On the other hand, the hostile and highly variable channel conditions common to mobile communications makes the realization of this goal very difficult Transmitting media over unreliable networks, such as IP or cellular networks has been an active area of research. Work in this area has focused on various error resilience and error concealment techniques for minimizing the effects of losses. These techniques attempt to encode the video sequence in ways that minimize the distortion at the receiver, given a statistical characterization of the channel error [1], [2]. At the physical layer, communication over wireless channels has also received considerable attention. Many of the physical layer techniques that have been considered for wireless channels can be classified as dynamic resource allocation techniques. These techniques can dynamically allocate communication resources such as transmission power and rate over the time. The point in these techniques is on performance measures such as transmission power and rate over the time. A mobile device is too big and heavy. The source of the problem is mainly the battery. About 60% of the size and the weight of a mobile device is due to the battery. Moreover the battery is exhausted too quickly. A laptop can only be used for a few hours without a connection to main outlet. A bigger battery can be employed, but this is not practical [3]-[5]. Although the capacity of the battery is increasing every year, more energy the same weight/size, the improvement is marginal compared to the additional energy requirements for new functionalities. In addition is that the mobile device may change dramatically in the short term as well in the long term. Wireless communications has two quite different characteristics compared to communications via fixed wires. First, the energy resources of a battery powered mobile require an energy efficient operation. Second, the dynamic environment of a mobile causes large quality variations of the transport medium and requires continuously monitoring of the Quality of Services (QoS) of the wireless link. A wireless link is far less stable than a fixed link, because the environment of a wireless link changes over the time. A lot of different factors influence the quality of the wireless link. Some factors are technology dependent or implementation dependent, while other factors are determined by the environment [6]. The Quality of Services (QoS) of a wireless link may be expressed in different ways, e.g.

- The perceived quality(frame loss, delay voice, resolution on image)
- Support Interactivity
- The cost
- The security

Most definitions of QoS consider the perceived quality and the supported interactive operations. Interactive access to video content over wireless or wire networks is generally defined as a program or service controlled by the user and which can affect the content itself, the presentation manner of the content, or the presentation order of the content [7]. Users, instead of being restricted to passive viewing of the video, have the ability to interact with it affecting the viewing schedule supported by a full range of interactive functions. This potentially offers the desired multilevel QoS to the user [8], [9]. In recent years several techniques for supporting interactivity for MPEG code video streaming applications have been devised. In [10], [11] interactive functions are supported by dropping parts of the original MPEG-2 video stream. Typically, this is performed once the video sequence is compressed and aims to reduce the transport and decoding requirements. These approaches introduce visual discontinuities during the interactive mode, due to the missing video information. Alternatively interactive functions can also be supported using separate copies of the movie that are encoded at lower quality of the normal playback copy [12], [13]. In these cases, there is no significant degradation in the visual quality. However, the number of pre-stored copies of the movie limits the speed-up granularity. Other conventional schemes that support interactive functions require that frames are displayed at a rate much higher than the normal playback (for example 90 fps [14]), or involves downloading parts of the video data in a player device located at the customer premises so that the customer can view without further intervention form the network [15]. In the latter case the downloading can be done prior to viewing. Previous work in this area has concentrated to support limited interactive functions (see [16] for recent survey). Note that none of the methods mentioned above fully address the problem of supporting interactive operations with minimum additional resources at the load of the server/network bandwidth and decoder complexity. In addition they support limited interactive functions. In this paper we propose a very efficient technique for supporting full-range of interactive operations in an MPEG video streaming over wireless network. The corresponding interactive version is obtained by encoding every Nth (i.e. uncompressed) frame of the original movie as a sequence of I- P(M)- frames video using different GOP pattern of the normal version. We analyze the effects of performing interactive modes by formulating a mathematical model in order to establish the constraints of interactivity. Mechanism for controlling the interactive streams is also investigated to assess the overheads required in supporting interactive operations for wireless applications. Moreover, the implementation issues are also discussed on an MPEG-4 video. The paper is organized as follows. Section II analyses the impact on supporting interactivity. In Section III the preprocessing steps required to support full interactive operations with minimum additional resources are detailed. The proposed algorithm for controlling the interactive streams over the wireless network is presented and its effectiveness is assessed through extensive simulations in Section IV. Finally conclusions are discussed in Section V.

II. IMPACT ON SUPPORTING INTERACTIVITY

The video streams are compressed using MPEG-4 video coding and are stored in the server. The clients can view the video while the video is being streamed over the network. The implementation of full interactive operations with MPEG-4 coded video is not a trivial task. MPEG-4 video compression is based on motion compensated predictive coding with an I-B-P structure. Assume that I_i is the starting point of the fast playback mode and F-S (Frames-Skipped) is the number of skipped.

Consider the case in which the current frame is P_3 and frame P_9 is the following frame to be displayed during the Fast Playback for F-S factor equals to 6. Actually there is no need to send any of the B-frames because for the given F-S factor the displayed P-frames can be decoded only from the preceding -I or -P frames. Fig.1 (c) depicts the number of required frames to be transmitted in order to display correctly the P-frames. It is logical to assume that the start point of the Fast Forward mode is an I-frame. This is due to the fact that the I- frames will not cause an unpleasant effect in viewing because they are decoded independently. For simplicity but without loss of any

$$I_o \ B_1 \ B_2 \ \mathbf{P}_3 \ B_4 \ B_5 \ I_6 \ B_7 \ B_8 \ \mathbf{P}_9$$

(a) normal play

$$B_1 \ B_2 \ B_4 \ B_5 \ B_7 \ B_8 \ B_{10} \ B_{11} ..$$

(b) redundant frames

$$P_3 \ I_6 \ P_9 \ I_{12} \ P_{15} \ I_{18} \ P_{21} \ I_{24} \ P_{27}$$

(c) transmitted frames

Fig. 1. Fast Forward mode for F-S=6

generality it can be assumed that $F-S \le N$, where N is the distance between two successive I frames, defining a "Group of Pictures" (GoPs). N can be defined as follows [16].

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

M is the distance between two successive P frames (usually set to 3) and a is nonnegative constant $(a \ge 0)$. The MPEG bitstream used for simulation is the "Mobile" sequence with 180 frames, which was encoded at 2Mbps, with a frame rate of 30fps. The Group of Pictures format was N=15 and M=3.Fig. 2 depicts the average bit rates required for sending the video stream with respect to different F-S factors. Fig. 3 shows the numbers of discarded frames from the decoder, every decoding pattern for different F-S factors

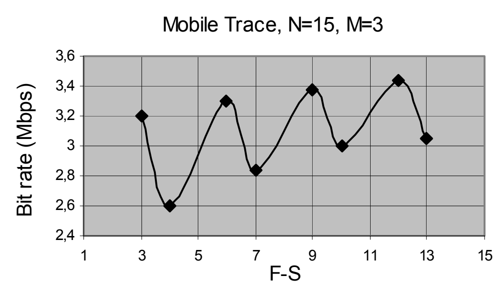

Fig. 2. Average bit rates with respect to different *(F-S)* factors.

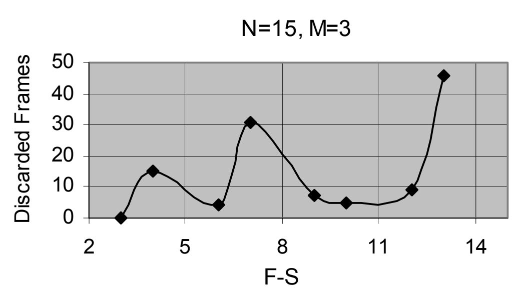

Fig. 3. Number of discarded frames as function of F-S factor.

III. PREPROCESSING OF VIDEO MOVIE

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. Assume that I- frame is always the start point of interactive mode. Since I- frames are decoded independently, switching from normal play to interactive mode and vice versa can been done very efficiently. 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. This is because it freezes the content of the I-type frames, reducing the visual discontinuities. Moreover P(Marionette) frames are produced between successive I-frames in order to maintain the same frame of normal play and achieve full interactive operations at variable speeds. Note that P(Marionette) frames contain only codes for not-coded macroblocks and are represented by a fixed pattern .To improve the marketability of video streaming applications, the client should interact with the content of the presentation deciding the viewing schedule with the full range of interactive functions. The full interactive functions can be supported as follows.

Fast Forward /Rewind (FF/FR) is an operation in which the client browses the presentation in the forward/backward direction with normal sequence of pictures. This function is supported by abstracting all the I-type frames of the original (uncompressed) movie in the forward/backward direction and encoding each frame as a sequence of I-P(M) frames.

Jump Forward/Backward (JF/JB) is an operation in which the client jumps to a target time of the presentation in the forward/backward direction without normal sequence of pictures. Therefore the users jump directly to a particular video location. Jump Forward/ Backward operation is supported by skipping forward/backward some I-type frames of the original (uncompressed) movie and encoding each of the remaining frames as a sequence of I-P(M) frames.

Rewind operations can be supported by abstracting all the I-type frames of the original (uncompressed) movie in reverse order and generate P(M) frames as many as P- and B in a GoPs of normal playback ($N = N_{interactive}$).

Slow Down/Reverse (SD/SR) is an operation in which the video sequence is presented forward/backward with a lower playback rate. This function can be supported by abstracting all the I-type frames of the original uncompressed movie in the forward/backward order and generate P(M) frames as many as P- and B-frames in a recording ratio (RR) of normal playback ($N_{interactive} = RR$).It is useful to derive a closed-form formula to show the number of the supported speedups of the proposed method. The speedups can be computed as follows.

$$SpeedUps = \frac{\dfrac{RR\_InteractiveMode}{N_{interactive}}}{\dfrac{RR\_NormalPlay}{N}}$$

where, $RR = RecordingRato = decoded\_fps$

To minimize the complexity at the decoder module the frame rate during the interactive mode the ($RR\_InteractiveMode$) must be the same as the one during the normal playback ($RR\_NormalPlay$). If the decoder consumes data at higher or lower rate than the one specified it would result in slight hiccups at the client end. This phenomenon will occur in any system where the server's production rate differs from the consumption rate of the decoder. Either the decoder will eventually starve or overrun its buffers. Hence, the speedups can be derived as follows.

$$SpeedUps = \begin{cases} \dfrac{N}{N_{interactive}}, \quad S_I = 0 \quad (FF) \\ S_I x\left(\dfrac{N}{N_{interactive}} - \dfrac{1}{\Omega}\right) + \dfrac{N}{N_{interactive}} \ , 1 \le S_I < \dfrac{TF}{N}, (JF) \end{cases}$$

where

S_I is the number of skipped sequential I-type frames and

$$\Omega = \frac{RR\_NormalPlay}{N}$$

Fig. 4 depicts the number of supported speedups (SpUps) as a function of $N_{interactive}$ for various numbers of skipped I-type frames.

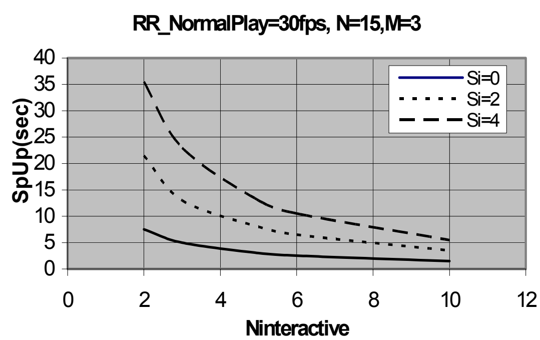

Fig. 4. Relative increase in the speedups as a function of $N_{interactive}$ for various number of skipped I-frames (S_I).

The graph in Fig. 4 shows that variable speedups can be achieved during the Fast Forward (FF) mode by increasing/decreasing the number of P (M) frames ($N_{interactive}$). On the other hand, variable speedups in the Jump-Forward (JF) mode depends on the number of P(M) frames between successive I-frames and the number of skipped I-type frames (S_I) of the normal stream.

IV. WIRELESS INTERACTIVE STREAMING

A major limitation of wireless communication is that mobile users depend on a battery with limited energy supply. Efficient utilizing this energy is a key consideration in designing wireless networks. We consider a system where video in encoded using MPEG-4 encoder [17]. The MPEG-4 video encoder takes raw of intra frames and produces interactive video packets streams, with I- P(M) frames. These video frames are buffered and then transmitted over the wireless network. The transmitter can dynamically allocate communications resources at the physical layer to each frame in order to meet the delay constraints and ensure reliable transmission. Note that conventional re-transmission techniques at the transmitter are not used in most real time applications because they can not accommodate the delay requirements of the applications [18]. At the receiver, the incoming interactive video frames are stored in the decoder buffer. Final the decoder reads video frames from this buffer and displays the interactive video sequence.

A. Proposed Algorithm

Our goal is to assign to each intra Macro block(MB) a choice of coding mode and quantization step size at the source coding level (MPEG-4 encoder) and a channel rate and transmission power at the physical layer (Transmitter) in order to obtain acceptable video quality during the interactive mode. For a intra-MB video packet, i, of size B(i) bits that is transmitted at rate R(i) we denote the delay as

$$Delay(i) = \omega(i) + \frac{B(i)}{R(i)}$$

where

$$\omega(i) = Delay(i-1) - T_{MB} > 0$$

is the amount of time the intra packet is must wait in the buffer before transmission and depends on the delay on the previous MB, i-1. T_{MB}

$$\frac{B(i)}{R(i)}$$

is the transmission delay for MB, i. This is the amount of time it takes to transmit $B(i)$ bits at a channel of rate $R(i)$ bits per second. We assume that each interactive video packet is sent over a slowly varying wireless channel. We model the channel over which the ith intra packet is sent as a band limited additive white Gaussian noise channel with fixed gain $h(i)^{1/2}$. The gain is assumed to be known at the transmitter and the receiver. The desired transmission rate for the ith intra packet is R(i). We assume that the required transmission power is the minimum power such that the channel over which this intra packet is sent has capacity R(i),

$$P(h(i), R(i)) = \frac{N_0 W}{h(i)}(2^{\frac{R(i)}{W}} - 1)$$

W is the bandwidth of the channel and N_0 is the power spectral density of the noise. We consider a wireless channel with bandwidth $W = 500KHz$ and additive white Gaussian noise with variance $N_0 W = 0.4$ Thus the amount of energy [19] required to transmit intra packet i of size B(i) bits at a rate of R(i) bits per second can be expressed as

$$E(i) = P((h(i), R(i))\frac{B(i)}{R(i)}$$

The problem addressed is that of transmitted a sequence of I- P(M) frames using the minimum amount of energy subject to quality and the delay constraints imposed by the network application Intra MB i is coded using a quantizer q(i) from a set of allowable quantizers Q=(2,4,6,8,10,12,14,16,18,20,22,24,26,28,30) resulting in video packet of B(i) bits with distortion D(i). This intra packet is transmitted at rate R(i) bits per second chosen from a set of allowable channel rate R=(100Kbps, 200Kbps, 300Kbps, 400Kbps).

Our goal is to proper select a quantizer and q(i) and channel rate R(i) for each intra MB in order to minimize the energy required to transmit the intra packet subject to both distortion constraint and delay per intra MB constraint. Let nMB be the number of Macroblocks in an intra frame and i the MB index. We consider an optimization model [19] as follows

$$\min E_L^{q(i),R(i)}\left(\sum_{i=0}^{nMB-1} E(i)\right) \quad (1)$$

Subject to

$$E_L\left(\sum_{i=0}^{nMB-1} E(i)\right) \le D_T$$

$$Delay(i) \le T_{max}$$

Lagrange multiplier can be applied λ>0 to solved the following relaxed problem.

$$\min E_L^{q(i),R(i)}\left(\sum_{i=0}^{nMB-1}[E(i) + \lambda D(i)]\right) \quad (2)$$

Subject to

$$Delay(i) \le T_{max}$$

This relaxed problem can be solved using techniques from Dynamic Programming [20]

B. Reference Scheme

In order to illustrate the advantage of the proposed algorithm we consider a system where source coding decisions are made to minimize the expected disruption at the receiver subject only to the bit budget constraint. Thus

$$min\, E_L^{q(i)}\left(\sum_{i=0}^{nMB-1} D(i)\right) \qquad (3)$$

Subject to

$$\left(\sum_{i=0}^{nMB-1} B(i) \le B_T\right)$$

Therefore the solution approach presented above can also used to solve Eq.3.

The following relaxed problem is solved.

$$min\, E_L^{q(i)}\left(\sum_{i=0}^{nMB-1} D(i)\right) + \lambda_2 \mathrm{B}(k) \qquad (4)$$

Deterministic Dynamic Programming [20], can also be used to solve Eq.4

C. Experiments

There are two types of criteria that can be used for the evaluation of video quality; subjective and objective. It is difficult to do subjective rating because it is not mathematically repeatable. For this reason we measure the visual quality of the interactive mode using the Peak Signal-to-Noise Ratio (PSNR). We use the PSNR of the Y-component of a decoded frame. The MPEG bitstream used for simulation is the Motor Race (Mobile) sequence with 180 frames, which was encoded at 2Mbps, with a frame rate of 30fps. The Group of Pictures format was N=15 and M=3. The number of I-P(M) frames in the interactive mode can be computed as follows

$$TF_{I,P(M)}^{Interactive} = I_{number} \times N_{ineractive}$$

where,

- $I_{number} = \dfrac{TF}{N}$ is the number of I-frames
- $(N_{\text{interactive}}, M_{\text{interactive}})$ are the new re-encoding parameters.

Hence, for TF=180, (N=15,M=3) and $(N_{\text{interactive}} = 5, M_{\text{interactive}} = 1)$,

$$TF_{I,P(M)}^{Interactive} = 60$$

Figure 5 shows the PSNR plot per frame obtained with the proposed algorithm and the reference scheme during the interactive mode. The proposed algorithm yields and advantage of 1,2 db on average.

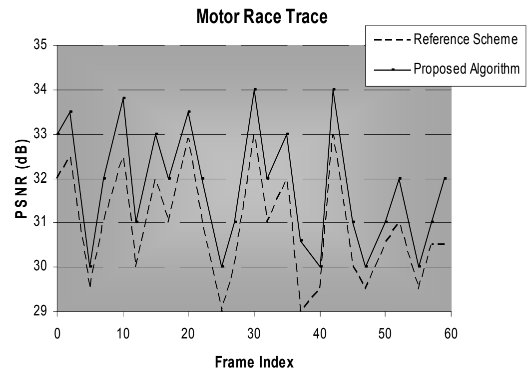

Fig. 5. PSNR as a function of frames in the interactive mode

The absolute values of PSNR do not convey the advantage of the 1st proposed algorithm compared to 2nd proposed algorithm (reference scheme). For this purpose, we compute the PSNR values for the 60 frames when re-encoding is done without any constraints and uses these values as a reference. For each frame, we compute the difference between its reference PSNR value and the PSNR value resulting from each of the two algorithms. These differences are plotted in Fig. 6 of a segment of the interactive mode. In the same figure a large value indicates a large deviation from the reference PSNR, and thus higher quality.

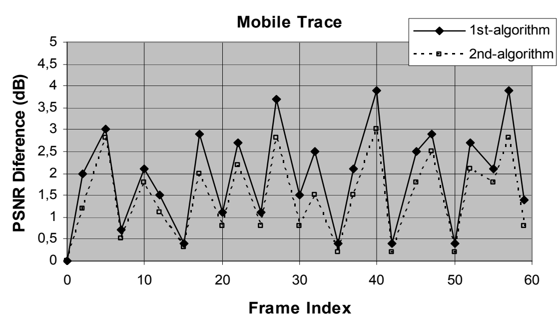

Fig. 6. Difference in PSNR between constrained and unconstrained encoding

Figure 7 depicts the PSNR values for motor race clip (mobile trace_ complex motion) as a function of the number of bits of the interactive bit stream.

It shows both the proposed algorithm and the reference scheme. We can see from this figure that as the bit-budget increases the difference between the reference scheme and algorithm is increased. Clearly the proposed algorithm achieves better visual quality than the reference scheme, but at expense of more encoding complexity.

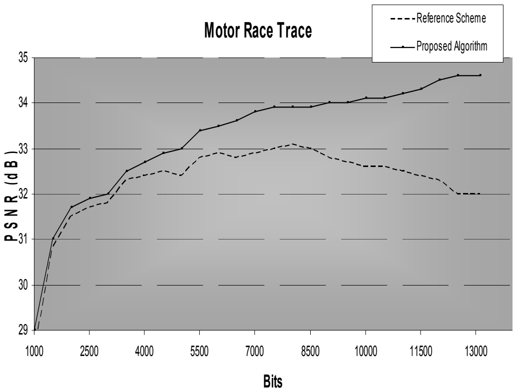

Fig. 7. PSNR as a function of the bit-budget of the interactive bit stream

V. CONCLUSIONS

In this paper, we investigated the constraints of implementing an MPEG coded video streaming system, which supports full interactive services. In order to overcome these additional resources we proposed the use of multiple differently encoded version of each video sequence. Each one of the differently coded sequences is obtained by encoding every N^{th} frame of the original (uncompressed) sequence. This allows for support of fully interactive operations at variable speedups. All versions of the original sequence are stored in the system's server. Mechanism for controlling the interactive streams is also investigated to assess the overheads required in supporting interactive operations for wireless applications. Clearly the proposed very efficient approach achieves full interactive operations with acceptable visual quality. Our future work includes developing a multilevel Quality of Services framework for streaming interactive video over wireless network. In addition more experiments will be carried out with different type of motion of MPEG-based video clips in order to further validate the performance of the proposed method.

REFERENCES

[1] D. -P. Wu, Y. -T. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha: Streaming Video over the Internet: Approaches and Directions, IEEE Trans on Circuits and Systems for Video Technology, Feb 2001, pp 1-20.

[2] Luna, L. P. Kondi, A. K. Katsaggelos, : Maximizing User Utility in Video Streaming Applications, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 2, February 2003, pp. 141-148.

[3] N. Cranley, L. Fiard and L. Murphy,: Quality of Service for Streamed Multimedia over the Internet, Proc. Irish Signals and Systems Conference 2000, Dublin, Ireland, June 2000

[4] G. J. Conklin, G. S. Greenbaum, K. O. Lillevold, A. F. Lippman, and Y. A. Reznik, : Video Coding for streaming media delivery on the Internet, IEEE Trans. Circuits Syst. Video Technol., Mar 2001, pp 269-281.

[5] Y. Eisenberg, T. N. Pappas, R. Berry, A. K. Katsaggelos, "Minimizing transmission energy in wireless video communications," Proceedings of IEEE International Conference on Image Processing, Thesaloniki, Greece, pp. 958-961, September 2001.

[6] S. Appadwedula, M. Goel, N Shanbhag, D. Jones and K. Ramchandran : Total System Energy minimization for wireless image transmission, Journal of VLSI Signal Processing, 27:99-117,2001

[7] Yutaka Ishibashi, Kazutoshi Nishimura, Hirotaka Nakano: Multiple-access Visual Search for Digital Video-on Demand Services, Technical Report of IEICE, IE92-96, Dec 1992.

[8] Kostas E Psannis and Yutaka Ishibashi.: Support Full Interactive Services For Educational Media Content, submitted to ACM Multimedia Systems journal, special issue on Multimedia Systems Technologies for Educational Tools, May 2005

[9] .Kostas E Psannis, Yutaka Ishibashi, Marios Hadjinicolaou. : A Novel Method for Supporting Full Interactive Media Stream over IP Networks, International Journal on Graphics, Vision and Image Processing, pp 25-32

[10]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, 1994.pp 594-605..

[11]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.

[12].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

[13]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.

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

[15] Kostas E Psannis and Marios Hadjinicolaou. : Transmitting Additional Data of MPEG-2 Compressed Video to Support Interactive Operations, In: Proc. IEEE ISIMP01,Hong Kong, May 2001,pp 308-311.

[16]Kostas E Psannis, Marios Hadjinicolaou: MPEG based Interactive Video Streaming: A Review, WSEAS Transaction on Communication, vol.2, April 2004, pp 113-120

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

[18]R Talluri: Error resilient video coding in the ISO MPEG-4 standard, IEEE Communications Magazine, June 1998

[19]C. E. Luna, Y. Eisenberg, R. Berry, T. N. Pappas, A. K. Katsaggelos, "Joint source coding and data rate adaptation for energy efficient wireless video streaming," IEEE Journal on Selected Areas in Communications, Special Issue on Recent Advances in Wireless Multimedia, vol. 21, no. 10, pp. 1710-1720, December 2003.

[20]D. P Bertsekas: Dynamic Programming and Optional Control, vol 1, Athena Scientific Belmont, MA, 1995.