CONSTANT BIT RATE LIVE VIDEO TRANSMISSION USING FRAME RESIZING [+]

الارسال الحي لاشارات الفديو ذات معدل التدفق الثابت بواسطة تغيير حجم المقاطع

Abdulrahman Ikram Siddiq [*]

ABSTRACT:

The paper studies the use of frame resizing by zooming as a compression technique to achieve Constant Bit Rate (CBR) live video transmission. Most video compression techniques provide a relatively constant level of decoded video quality, but, at the expense of Variable Bit Rate (VBR) outputs. This will result in an inefficient usage of the available bandwidth, and the decoder has to buffer a suitable amount of data before it can play the received stream. This results in a variable overhead delay depending on the variation in the bit rate. Conversely, a CBR technique can efficiently occupy the available bandwidth with constant overhead delay and buffer size, but, at the expense of variable decoded video quality.

The Peak Signal to Noise Ratio (PSNR) is used in the computer simulation tests to measure video quality. The tests show that variation of video quality in a CBR system employing frame resizing is still acceptable ($PSNR_{av}$>25 dB) for resizing factors greater than 40%. Moreover, the tests show that the resizing technique, when compared to conventional video compression techniques, is more immune to additive Gaussian noise, since it involves an interpolation process which averages out part of the added noise.

المستخلص:

يهتم هذا البحث بدراسة امكانية استعمال تقنية تكبير و تصغير الصور لاداء وظيفة ضغط البيانات في انظمة الارسال الحي للفيديو ذات سرعة الارسال الثابتة. ان معظم انظمة ارسال الفيديو توفر امكانية الحصول على مستوى ثابت نسبيا لجودة الفيديو المستلم ولكن ذلك يكون على حساب سرعة ارسال متغيرة. مما يؤدي الى انخفاض كفاءة استخدام عرض النطاق الترددي المتوفر لقناة الاتصال و ضرورة خزن كمية كبيرة من الصور في ذاكرة المستلم المؤقتة قبل ان يكون الفيديو المستلم قابلا للعرض. بينما عندما تكون سرعة الارسال ثابتة فتكون كفاءة استخدام القناة عالية و حجم الذاكرة المؤقتة صغيرا و لكن الثمن هو مستوى جودة متغير.

تم في هذا البحث تنفيذ محاكاة بواسطة الحاسوب لنظام بث و استلام فيديو حيث اعتمد PSNR كمقياس لجودة الفيديو المستلم. اظهرت الاختبارات ان نظام البث المقترح يوفر مستوى مقبولا لجودة الفيديو المستلم (PSNR >25 dB) لمعامل تصغير يصل الى ٤٠% من الحجم الاصلي. و ان النظام المقترح يمتلك حصانة للضوضاء المضاف في قناة الاتصال اكبر من الانظمة القياسية المشابهة. بالاضافة لما تقدم، فان بساطة النظام المقترح مقارنة بالانظمة الاخرى تجعله مناسبا للاستعمال في تطبيقات البث الحي للفيديو.

EYWORDS : **Video transmission, Constant bit rate, Video compression.**

INTRODUCTION:

[+] Received on 10/3/2009 ,Accepted on 22/6/2010 .

[*] Lect .Technical College / Kirkuk

Recently, the subject of live digital video streaming over computer networks like the Internet, has received a significant research effort. This is due to the growing demands on services like steaming live satellite channel programs, video chat, and video conferencing on the Internet [1].

Video streaming systems rely on compression algorithms to minimize storage, transmission bandwidth, and hence the overall cost [2]. The choice of video coding and encoding parameters affects the coded bit rate, quality of the decoded video sequence, and the computational complexity of the video codec. At the same time, practical limits determined by the processor and the transmission environment put constraints on the bit rate and image quality that may be achieved. Therefore, it is important to control the video coding process in order to maximize compression performance while remaining within the practical constraints of the transmission and processing.

There are many approaches to achieve video compression, such as Motion JPEG (M-JPEG), MPEG 1, 2, 4, etc, H.261, H.265, DV25, … etc [2]. They are based on combinations of basic compression techniques like:

- Variable length coding such as Huffman and arithmetic coding
- Run Length Encoding (RLE)
- Quantization
- Scaling
- Transformation
- Temporal compression
- Object detection and motion compression, etc.

An important aspect of video compression is the bit rate mode that is used. In most video compression systems, it is possible to select if the bit rate should run in CBR mode or Variable Bit Rate (VBR) mode [3,4]. The optimal selection depends on the application and available communication infrastructure. With a limited link bandwidth, the preferred mode is normally CBR, since this mode generates a constant and predefined bit rate. The disadvantage is that image quality will vary and while it will remain relatively high when there is no motion in the image scene, the quality will significantly decrease with increased motion [3].

On the other hand, the VBR mode will maintain a high, if so defined, image quality regardless of motion or no motion in the image scene. This is often desirable in security and surveillance applications when there is a need for high quality, especially if there is motion in the picture [2]. Even though, in VBR systems, a maximum bit rate is defined and the available link bandwidth for such a system needs to have a suitable capacity. However, there are times, during the video transmission session, at which the actual bit rate slows down under its maximum value resulting in bandwidth wasting, especially in circuit switched networks.

There is no problem with VBR in cases such as saving the compressed video in a file for playback on a computer. But, if this file is to be available for video streaming, it can be a problem for the receiver to download and simultaneously display a transmission over the network that seems to slow down and speed up throughout the broadcast. For these applications it is better to have a CBR, where there is a constant amount of data being transmitted regardless of what the original video looked like.

There are many techniques to regulate the VBR. The simplest of these is to use a buffer memory to accumulate the data and then transmitting it at a CBR [5]. The size of this buffer memory must be large enough to maintain a CBR output. The main two disadvantages are the delay and added hardware complexity.

In the literature, a lot of work is done to optimize the parameters like the bit rate, delay, buffering, quality, etc, of standard video compressors to suit specific applications [6]. But the point is that these standard video compressors involve relatively complicated algorithms making optimizing their parameters to suit many applications not successful always. Other works aim to enhance the reconstructed video quality allowing certain amounts of drawbacks in other parameters such as system complexity and delay. This aim is achieved by employing different error correction schemes like in [7,8,9,10]. In addition to this, motion compensation is used [11,12] resulting in more system complexity.

Specifically, a video compressor to be used for live video streaming has to have the following basic features:

1. Simple and hence fast.
2. Produce a CBR for efficient link bandwidth occupation and constant overhead and buffer size.
3. Maintain acceptable amount of loss in decoded video quality.

In this paper, only frame resizing is used to compress video sequences due to its simplicity relative to standard video compressors and its ability to produce CBR outputs. The work investigates the amount of quality loss at different resizing factors to determine the margin of acceptable operation. Moreover, the immunity of the proposed video compressor to additive noise is evaluated and compared to that of the standard video compressor M-JPEG.

MODEL OF THE SYSTEM:

Figure 1 shows the model of the CBR video transmission system based on frame resizing (VTFR). The system consists of the video compression, communication channel, and video decompression blocks.

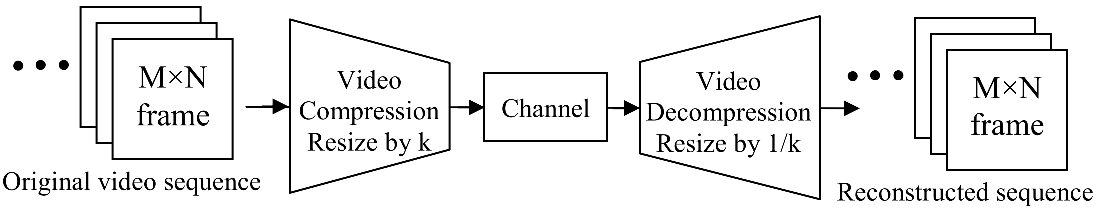

Figure 1 Model of the proposed video transmission system

At the transmitter side, the video compression block compresses each individual frame in the original video sequence to a smaller size using a zooming-out technique. Therefore, the compression factor, *k*, for all of the frames of the video sequence is the same, and it is defined as

$$K = \frac{\text{size of the compressed frame}}{\text{original size of the frame}} \tag{1}$$

Next, the compressed frames pass the communication channel. They are affected by the filtered additive Gaussian noise introduced by the channel. Then, at the receiver, each individual compressed frame is enlarged using the bilinear interpolation image zooming algorithm, with an enlargement factor of $1/k$ to restore the original frame size.

However, the two main sources of error in this system are:

1. The lossy compression technique.
2. The effect of the communication channel.

These will affect the quality of the received video sequence. A suitable measure for the quality of the decoded video stream is the Peak Signal to Noise Ratio (PSNR) defined in [2] as

$$\mathrm{PSNR} = 20\log\frac{255}{\sqrt{\frac{1}{M\times N}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(f(x,y)-\hat{f}(x,y))^2}} \tag{2}$$

where
the constant 255 is the maximum pixel value in a frame
M×N are the dimensions of an uncompressed frame
$f(x, y)$ is an uncompressed frame
$\hat{f}(x, y)$ is a reconstructed frame

The PSNR is to be calculated for each individual reconstructed (received) frame to show the degradation in its quality with respect to its original version.

COMPUTER SIMULATION AND RESULTS:

The performance of the video transmission system based on frame resizing is evaluated by using the Matlab software package. A gray-level video file containing 1000 frames is used in the tests. After removing the header of the video file, each frame is extracted as a (1024×850)-pixel bit map digital image to be manipulated individually by the system.

The performance of the tested VTFR system is compared to that of a video transmission system based on the well-known image compressor JPEG. When the JPEG technique is used to compress every frame of a video sequence individually, it is known as Motion-JPEG (M-JPEG). The M-JPEG is selected to be used in the tests as a reference system for performance comparison because it is similar to the VTFR in that they deal with every frame of the video file separately, unlike other standard video compressors such as MPEGs which process multiple frames simultaneously.

Recall the definition of the compression factor, *k*, from eq.(1), the value of *k* in the M-JPEG system is variable depending on how much the JPEG is able to compress a specific frame. This is clear from Fig. 2, which shows a plot of *k* for the M-JPEG system when it is used to compress the 1000-frame test video sequence. The average value of *k* for this case is 0.334.

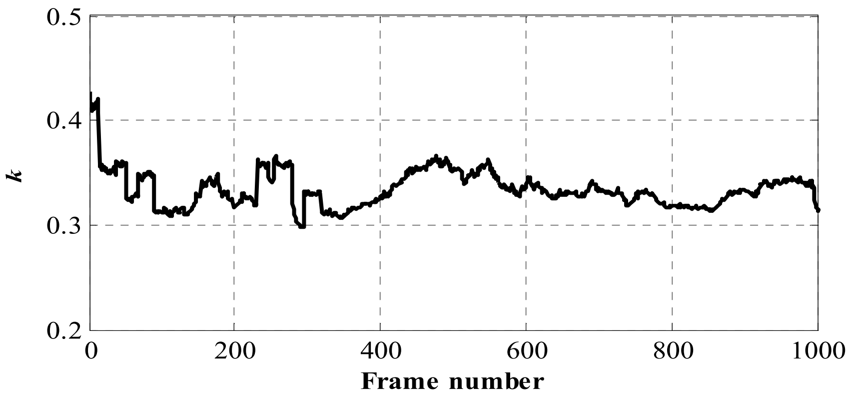

Figure 2 M-JPEG compression factor

On the other hand, the compression factor of the frame resizing technique is a predetermined constant-value parameter (eg. 0.2, 0.4, 0.6, …etc). This is because frame compression is achieved by ignoring a number of pixels from that frame such that a specific compressed size is resulted. However, the decompression process is based on interpolating the pixels of the received compressed frame to calculate estimates of the missing pixels to reconstruct the original frame. In the simulations, the bi-cubic interpolation process is used.

As a result, the VTFR system produces a CBR output leading to efficient usage of the available bandwidth. The system requires a relatively small buffer memory enough to accommodate a single frame. Moreover, the compression and decompression processes are simple and do not require gathering any kind of statistical information nor use code look-up tables leading to small file headers and constant overhead delay.

Conversely, the M-JPEG system produces a VBR output leading to inefficient bandwidth utilization. The M-JPEG system needs to buffer many frames in order to regulate the output bit rate. Moreover, the compression and decompression processes are relatively complex. They involve many stages such as transformation, quantization, RLE, and variable length coding which needs gathering statistical information on the data to be compressed, and storing look-up tables leading to long and variable length headers and overhead delay.

Another important parameter in video transmission is the quality of the reconstructed video stream. Figure 3 shows the PSNR calculated for each individual frame of the 1000-frame video file used in the tests. This is repeated for the M-JPEG system and the VTFR system with k=0.2, 0.4, 0.6, and 0.8. In order to observe the effect of the compression and decompression processes on the quality of the reconstructed video stream, a perfect channel and an infinite Signal to Noise Ratio (SNR) are assumed in the tests of Fig. 3. This figure shows that the best (highest) relative quality is achieved by the M-JPEG system and then by the VTFR system with k=0.8, 0.6, 0.4 and the worst quality is for k=0.2. The interpretation is that frame resizing involves a certain amount of data loss, which is inversely proportional with the value of k. That is, the greatest loss of data occurs for k=0.2 resulting in the worst quality.

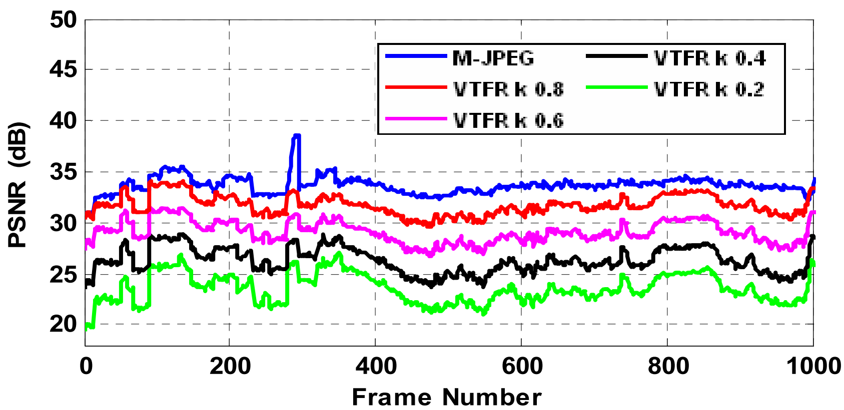

Figure 3 PSNR for ideal channel

The average value of the PSNRs plotted in Fig. 3 are shown in Table 1, where

$$PSNR_{av} = \frac{1}{1000}\sum_{i=1}^{1000} PSNR_i \tag{3}$$

where $PSNR_i$ is the PSNR of the i^{th} frame in the tested video file.

Table 1 $PSNR_{av}$ for infinite SNR

Compression scheme	M-JPEG	Resizing			
		k=0.8	k=0.6	k=0.4	k=0.2
$PSNR_{av}$ (dB)	33.674	32.623	29.99	26.797	23.737

However, by playing back these decompressed videos, it is concluded that the margin value of the PSNR over which the video quality is visually acceptable is about 25 dB. Therefore, the M-JPEG and the VTFR with *k*=0.8, 0.6, and 0.4 can achieve acceptable video quality. Among these, the VTFR with *k*=0.4 is the most suitable from the link bandwidth efficiency point of view.

Next, the effect of Additive Gaussian Noise (AGN) on the quality of the decompressed video sequence is investigated. The tests of Fig. 3 are all repeated three more times, once for a specific value of SNR. Figures 4, 5, and 6 show the PSNR resulted from these tests for SNR=20, 30, and 40 dB, respectively. For SNR of 20 and 30 dB (Figs. 4 and 5), all of the tested systems suffer from a great degradation in PSNR.

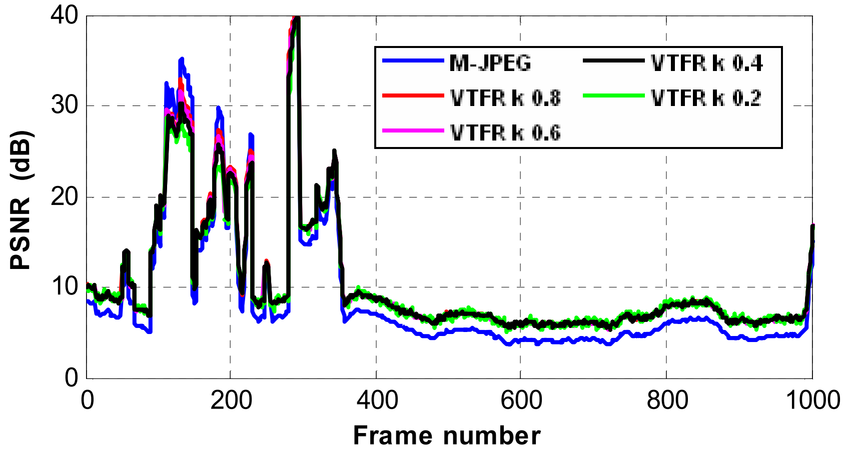

Figure 4 PSNR for SNR=20 dB

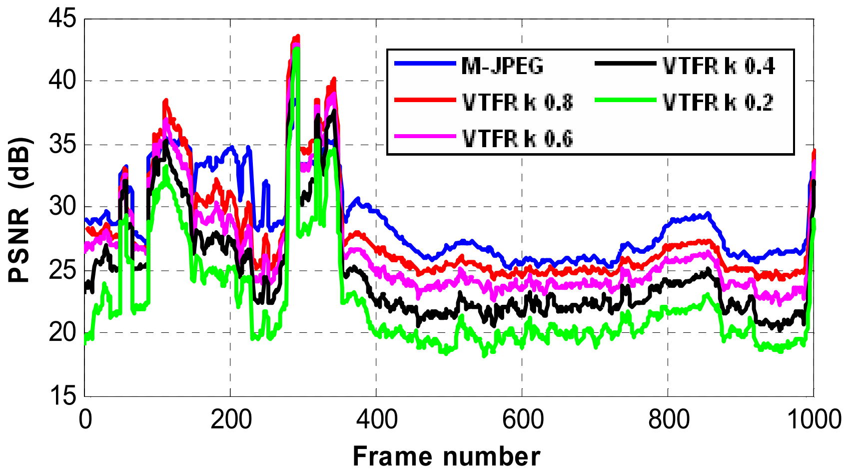

Figure 5 PSNR for SNR=30 dB

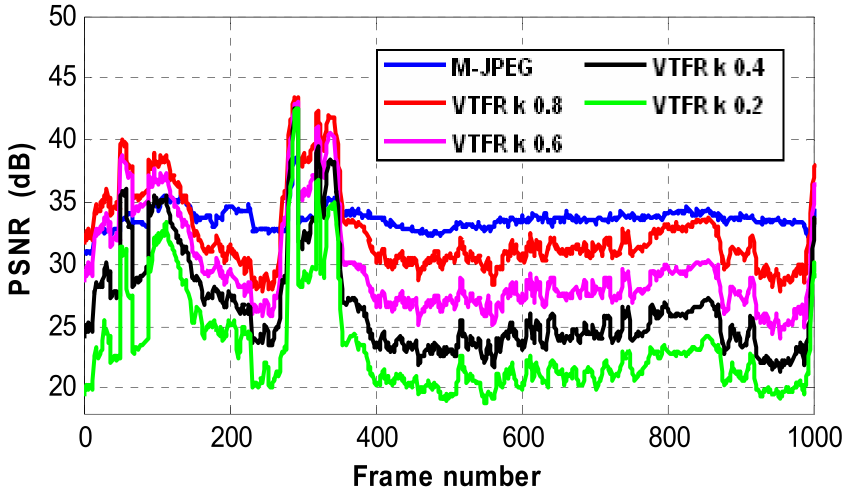

Figure 6 PSNR for SNR=40 dB

Tables 2 and 3 give the amount of degradation in $PSNR_{av}$ for each test case with respect to the case of no AGN which is given in Table 1.

Table 2 PSNRav at SNR=20 dB

Compression scheme	M-JPEG	Resizing			
		k=0.8	*k*=0.6	*k*=0.4	*k*=0.2
$PSNR_{av}$ (dB)	8.967	10.522	10.437	10.402	10.305
Degradation in $PSNR_{av}$ w.r.t. SNR→∞	24.707	22.101	19.553	16.395	13.432

Table 3 $PSNR_{av}$ at SNR=30 dB

Compression scheme	M-JPEG	Resizing			
		k=0.8	*k*=0.6	*k*=0.4	*k*=0.2
$PSNR_{av}$ (dB)	26.922	27.729	25.518	24.737	22.367
Degradation in $PSNR_{av}$ w.r.t. SNR→∞	6.753	4.893	4.473	2.069	1.369

From these tables it is noted that the degradation in $PSNR_{av}$ of the M-JPEG is always greater than those for the resizing technique for all tested values of *k*. Therefore, the VTFR technique is more immune to AGN as compared to M-JPEG tested under the same conditions. The reason is that the resizing technique decompresses the frames of the video sequence using interpolation, which averages out part of the added noise. On the other hand, an M-JPEG coded video corrupted with AGN is decoded to wrong pixel values causing more quality degradation with respect to frame resizing which does not involve code mapping.

Moreover, Fig. 6 shows the case when the effect of noise is low, i. e., SNR=40 dB. As given in Table 4, the degradation in $PSNR_{av}$ of the M-JPEG system is about 1 dB but it is much smaller in the VTFR system especially for *k*=0.8, which is about 0.1 dB, and it is about 0.3 dB for *k*=0.4.

Table 4 $PSNR_{av}$ at SNR=40 dB

Compression scheme	M-JPEG	Resizing			
		k=0.8	*k*=0.6	*k*=0.4	*k*=0.2
$PSNR_{av}$ (dB)	32.653	32.521	29.669	26.486	23.255
Degradation in $PSNR_{av}$ w.r.t. $SNR\rightarrow\infty$	1.021	0.102	0.321	0.311	0.482

The time required by the VTFR system (with k=0.4) to reconstruct a single received frame is calculated using the *'CPUTIME'* in Matlab, to be 31.3 ms. This means that within the time of one second, about 32 frames can be reconstructed which is greater than 26 or 30 which are the standard numbers of displaying frames per second in video sequences.

The tests and results presented so far show that the most suitable value of *k* to be used in the VTFR system is 0.4. This is because it is the smallest value of *k* that gives acceptable degradation in the $PSNR_{av}$ of the reconstructed video file. However, although the average compression factor of the M-JPEG is 0.334, the VTFR with *k*=0.4 is more immune to AGN and has all the advantages of CBR video transmission.

CONCLUSIONS:

In this paper a CBR video transmission system is proposed to be implemented using a simple frame resizing technique, VTFR. Having a CBR output, the proposed system can efficiently occupy the available bandwidth and it is much simpler than the available standard video compressors. It requires a much smaller buffer memory and it does not involve transformation, gathering any kind of statistical information nor the use of code look-up tables. Therefore, the compressed sequence has a smaller header and a predefined constant overhead delay, making it possible and easy to receive and simultaneously display a live video broadcast. However, all of these advantages are at the expense of variable received video quality.

Extensive computer simulation tests show that the VTFR with a resizing factor of 0.4 can achieve a performance with an acceptable amount of $PSNR_{av}$ (>25 dB). Whereas an M-JPEG system, tested under the same conditions, produces a VBR output with an average compression factor of 0.334 and a slightly better reconstructed video quality. Then, the VTFR with k=0.4 is more suitable from the link bandwidth efficiency point of view. Moreover, the tests are repeated under different SNR conditions. The results show that the VTFR with k=0.4 is more immune to additive Gaussian noise than the tested M-JPEG. Therefore, it can be concluded that the proposed VTFR is suitable to be used in real-time video transmitted over band-limited links exposed to additive noise.

REFERENCES:

1- A. K. Ramasubramonian and J. W. Woods, "Video Multicast using Network Coding," *Proc. SPIE VCIP*, Jan. 2009.

2- Iain E. G. Richardson, *Video Codec Design,* John Wiley & Sons, Ltd, 2002, ISBNs: 0-471-48553-5 (Hardback); 0-470-84783-2 (Electronic).

3- R. Kurceren and J. W. Modestino, "An End-to-End Performance Comparison Methodology for ATM Transport of CBR and VBR Encoded Digital Video" *Packet Video Workshop* , Sardinia, Italy, May 2000.

4- William Stallings, *Data and Computer Communications*, Pearson, Prentice Hall, 2004. ISBN: 0-13-183311-1.

5- I. V. Bajic, O. Tickoo, A. Balan, S. Kalyanaraman, and J. W. Woods, "Integrated end-to-end buffer management and congestion control for scalable video communications," *Proceedings of IEEE ICIP 2003*, Barcelona, Spain, September 2003.

6- Chang, Ray-I., Chen, Meng Chang, Ho, Jan-Ming, Ko, Ming-Tat, "Optimization of stored VBR video transmission on a CBR channel", *Proceedings of SPIE* Vol. 3231, p. 382-392, Performance and Control of Network Systems, Hisashi Kobayashi; Eds., October 1997.

7- Y. Shan, I. V. Bajic, J. W. Woods, and S. Kalyanaraman, "Scalable Video Streaming with Fine Grain Adaptive Forward Error Correction," submitted to *IEEE CSVT*, revised Dec. 2008.

8- I. V. Bajic and J. W. Woods, "Error-Concealment for Scalable Motion-Compensated Subband/Wavelet Video Coders", *IEEE Trans. Video Technology*, Vol. 17, p.508-514, April 2007.

9- Sehoon Yea, William A. Pearlman, "On Scalable Lossless Video Coding Based on Sub-Pixel Accurate MCTF", *Visual Communications and Image Processing 2006*, *SPIE/IS&T Conf. on Electronic Imaging, Proc. SPIE* Vol. 6077, Jan. 2006.

10- M. Farshchian and W. A. Pearlman, "Real-time Video Transmission over MIMO OFDM Channels Using Space-Time Block Codes ," *Conference on Information Sciences and Systems*, Mar. 2006.

11- R. A. Cohen and J. W. Woods, "Resolution Scalable Motion-Compensated JPEG-2000," *Proc. 15th Int'l. Conf. on DSP (DSP2007)*, Cardiff, Wales, UK, July 2007.

12- A. Golwelkar and J. W. Woods, "Motion-compensated Temporal Filtering and Motion Vector Coding using Biorthogonal Filters", *IEEE Transactions on Video Technology*, 2005.