5th
SASTech
Iran | Mashhad
May 12 - 17 | 2011
5th Symposium on Advances in Science & Technology
RESEARCH
Tech
Communication
Conference
E-mail

Robust H.264 Video Streaming Under Erroneous Transmission Conditions over GPRS Wireless Networks

Mehdi Jafari
Islamic Azad University, Kerman Branch,
Department of electronic engineering, mjafari@iauk.ac.ir
Shohreh Kasaei
Sharif University of Technology
Department of Computer Engineering, skasaei@sharif.edu
Solmaz Jafari
Academy for scientific Research, Kerman Branch
Paper Reference Number:
Name of the Presenter: Mehdi Jafari

Abstract

Channel errors have a very detrimental effect on the perceptual video quality. Despite the research done in the field of wireless multimedia, delivery of real-time interactive video over noisy wireless channels is still a challenge for researchers. This paper presents a method for improving the quality of video transport using H.264/AVC over wireless networks that is the prioritization of different parts of I- and P-frames. The effectiveness of the technique is demonstrated by examining its performance when the transport of the prioritized video streams can be accomplished using packet switching technology over the enhanced *general packet radio service* (GPRS) access network infrastructure.

Key Words: GPRS, H.264/AVC, Video Streaming, Wireless Networks.

1. Introduction

The 2.5/3G personal communications systems such as the GPRS/EGPRS networks are becoming a reality now. This new generation is being designed to be able to provide data and multimedia services. In addition to conventional voice communication services (provided by the second generation *global mobile system* (GSM) [1] networks), the 2.5/3G mobile networks will support a high data rate transmission that will enable the support of a wide range of real-time mobile multimedia services (including combinations of video, speech/audio and data/text streams) with the *quality of service* (QoS) control. With 2.5/3 G, mobile users will have the ability to remotely connect to the Internet while retaining access to all its facilities (such as e-mail and Web browsing sessions). Mobile terminals will be enabled to access remote websites and multimedia-rich databases with entering the Web browsers of these terminals. All mentioned services require a real-time transmission of video data over fixed and mobile networks with varying bandwidth and error

rate characteristics. Due to the huge bandwidth requirements of raw video signals, they must be compressed before transmission in order to optimize the required bandwidth to provide a multimedia service.

Image and video coding technology has witnessed an evolution, from the first generation canonical pixel-based coders to the second-generation segmentation-, fractal-, and model-based coders to the most recent third-generation content-based coders. Both the ITU and *international organization for standardization* (ISO) [2] have released standards for still image and video coding algorithms that employ waveform-based compression techniques to trade-off between the compression efficiency and the quality of reconstructed signals. After the release of the first still image coding standard (namely, JPEG in 1991), ITU recommended the standardization of its first video compression algorithm (namely, ITU H.261) for low bit rate communications over the ISDN, in 1993. Intensive work has since been carried out to develop improved versions of this ITU standard, and this has culminated in a number of video coding standards (namely, MPEG-1 (1991) for audiovisual data storage on CD-ROM, MPEG-2 (or ITU-T H.262, 1995) for HDTV applications, and ITU H.263 (1998) for very low bit rate communications over the PSTN networks). The newest video coding standard is developed by the joint of video teams of ISO/IEC MPEG and ITU_T VCEG as the international standard 14496-10 (MPEG-4 part 10) advanced video coding standard (AVC) [3,4].

Standards	MPEG-4/ASP	H.263/HLP	MPEG-2
H.264/AVC	38.62%	48.80%	64.46%
MPEG-4/ASP	----	16.65%	42.95%
H.263/HLP	------	-----	30.615

Today H.264/AVC has been very successful in many applications including digital media storage, video streaming, TV and so on. H.264/AVC has gained more and more attention; mainly due to its high coding efficiency, minor increase in decoder complexity compared to existing standards, adaptation to delay constraints, error robustness, and network friendliness [1, 2]. Table 1[3] and figure 1[4] show the performance comparisons using MPEG-2, MPEG-4 (ASP), and H.264/AVC.

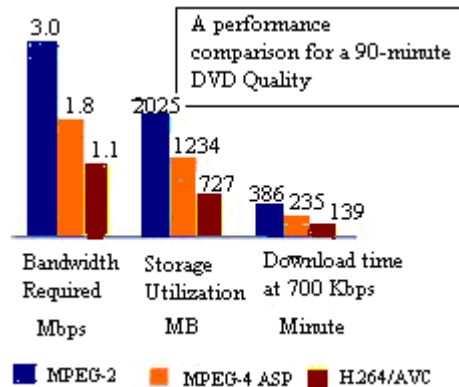Table 1: Average bit-rate reduction compared to prior coding schemes.

Figure 1: Performance comparison of different video coding standards.

To achieve an outstanding coding performance, H.264/AVC employs several powerful coding techniques such as directional prediction of intra-coded blocks, inter-prediction with variable block-size motion compensation, multi-reference frame motion estimation, motion vectors with quarter-pel accuracy, in-loop deblocking filter, 4×4 integer transform, and the forth. In this paper, for error resilience over error-prone wireless communication channels, a new encoding is made using of H.264/AVC [3] [4]. Since the H.264/AVC coded video data are highly sensitive to information loss and channel bit errors, the decoded video quality is bound to suffer dramatically at high channel bit error ratios. This quality degradation is unacceptable when no error control mechanism is employed to protect coded video data against the hostility of error-prone environments. In certain cases, these streams are required to travel across a number of asymmetric networks until they get to their final destinations. Consequently, the coded video bit streams have to be transmitted in the form of packets whose structure and size depend on the underlying transport protocols. During transmission, these packets and the enclosed video payloads are exposed to channel errors and excessive delays; and hence to information loss. Lost packets impair the reconstructed picture quality if the video decoder does not take any action to remedy the resulting information loss. The effects of a bit error on the decoded video quality can be categorized into three different classes as follows.

I) A single bit error on one video parameter does not have any influence on segments of video data other than the damaged parameter itself. In other words, the error is limited in this case to a single *macro block* (MB) that does not take part in any further prediction processes. (One example of this category is encountered when an error hits a fixed-length intra DC coefficient of a certain MB which is not used in the coder motion prediction process.) Since the affected MB is not used in any subsequent prediction, the damage will be localized and confined only to the affected MB. This kind of error is the least destructive to the QoS.

II) The second type of error is more problematic because it inflicts an accumulative damage in both time and space due to the prediction process. (One example of this category is encountered when the prediction residual of motion vectors is sent; and hence the bit errors in the motion code words propagate until the end of frame.) Moreover, the error propagates to subsequent inter coded frames due to the temporal dependency induced by the motion compensation process. This effect can be mitigated if the actual *motion vectors* (MVs) are encoded instead of the prediction residual.

III) The worst effect of bit errors occurs when the synchronization is lost and the decoder is no longer able to figure out that the received information belongs to which part of the frame. This category of error is caused by the bit rate variability characteristics. When the decoder detects an error in a variable length code word, it skips all the forthcoming bits (regardless of their correctness) when searching for the first error-free synch word to recover the state of synchronization.

Since real-time video transmissions are sensitive to time delays, the issue of re-transmitting the erroneous video data is totally ruled out. Therefore, other forms of error control strategy must be employed to mitigate the effects of errors inflicted on coded video streams during transmission. Different error control strategies are as follows:

- *Concealment techniques*: These techniques are decoder-based and incur no changes on the transport technologies employed. Moreover they do not place any redundancy on the compressed video streams and thus referred to as zero-redundancy or error concealment techniques.

- *Resilience techniques*: These error control schemes operate at the encoder and apply a variety of techniques to enhance the robustness of compressed video data to channel errors.

- Packetization techniques: This error control mechanism operates at the transport level and tries to optimize the packet structure of coded video frames in terms of their error performance as well as channel throughput.

This paper is focused on the two error effects, part I and II, and presents some error resilience techniques based on *unequal error protection* (UEP) for H.264/AVC coding over the GPRS mobile

networks. The rest of the paper is organized as follows. Layering syntax of H.264/AVC is briefly introduced in Section 2. Proposed data partitioning for INTRA- and INTER-frames are described in section 3 and 4 respectively. Experimental results are given in section 5 and finally Section 6 concludes the paper.

2. Layering Syntax for ITU-T H.264/AVC Video Coding Standard

In common with earlier standards, H.264/AVC does not define the encoder, but defines the syntax of an encoded video bit stream together with the method of decoding the bit stream [16, 17]. To achieve an outstanding coding performance, H.264/AVC employs several powerful coding techniques such as directional prediction of intra-coded blocks, inter-prediction with variable block-size motion compensation, multi-reference frame motion estimation, motion vectors with quarter-pel accuracy, in-loop deblocking filter, 4×4 integer transform, and the forth. The improvement in coding performance comes mainly from the intra- and inter-prediction. The Intra- and Inter- prediction error is more problematic because it inflicts an accumulative damage in both time and space due to the prediction process. (One example of this category is encountered when the prediction residual and motion vectors are sent; and hence the bit errors in the motion vector propagate until the end of frame.) Moreover, the error propagates to subsequent inter coded frames due to the temporal dependency induced by the motion compensation process. The H.264 codec consists of two main layers: the video coding layer (VCL) and the network abstraction layer (NAL). VCL provides the core compressed video contents and produces the coded data. NAL performs packetization of the coded bitstream and supports delivery over various types of network. For layered transmission of the coded video, H.264 supports data partitioning. In this method, the bitstream is divided according to importance into a number of partitions. The first partition carries the most important data (i.e. addressing, Intra- and Inter-Prediction vectors), and the other partitions contain less important information (i.e. residual data). In H.264 when using data partitioning, each slice is divided into three partitions (three NAL units). NAL-A is the most important unit, followed by NAL-B and finally NAL-C.
Ideally for transmission, each NAL unit should be protected according to its importance. In this paper NAL-A will be considered as the first part, and NAL-B & NAL-C (denoted as NAL-BC) will be grouped in the second part.

Intra-Prediction Mode Decision

Intra-prediction is based on the observation that adjacent macroblocks tend to have similar properties. Prediction may be formed for each 4×4 luma block (I4MB), 16×16 luma MB (I16MB), and 8×8 chroma block. For prediction of 4×4 luminance blocks, the 9 directional modes consist of a DC prediction (Mode 2) and 8 directional modes (labeled 0, 1, 3, 4, 5, 6, 7, and 8) as shown in Figure 2(a). In Figure 2(b), the block (values of pixels "a" to "p") is to be predicted using A to Q pixel values.
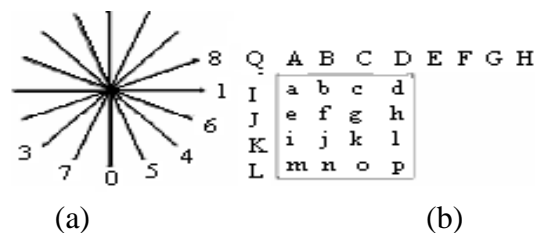

(a)   (b)

Figure 2: (a) Intra-prediction modes for 4×4 luminance blocks. (b) Labeling of prediction samples.

The DC prediction (mode 2) is useful for those blocks with little or no local activities, the other modes (1-8) may only be used if all required prediction samples are available. For regions with less spatial details (*i.e.,* flat regions), H.264/AVC supports 16×16 intra-coding; in which one of four prediction modes (DC, vertical,

horizontal, and planar) is chosen for prediction of the entire luminance components of the macroblock as shown in figure 3.
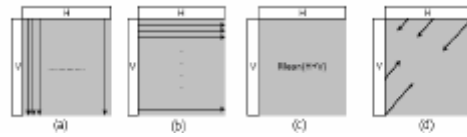

Figure 3: Intra 16×16 prediction modes. (a) Mode 0(vertical). (b) Mode 1(Horizontal). (C) Mode 2(DC). (d) Mode 3(plane).

H.264/AVC supports four chroma prediction modes for 8×8 chrominance blocks, similar to that of the I16MB prediction, except that the order of mode numbers is different: DC (Mode 0), horizontal (Mode 1), vertical (Mode 2), and plane (Mode 3). This paper selects NAL-A for I4MB and NAL-BC for I16MB to reduce error propagation in H.264.

Inter-Prediction Mode Decision

Inter-prediction is based on using motion estimation and compensation to take advantage of temporal redundancies that exist between successive frames. The important differences from earlier standards include the support for a range of block sizes (down to 4×4), multiple reference frames, and fine sub-pixel motion vectors (1/4 pixel in the luma component).

H.264/AVC supports motion compensation block sizes ranging from 16×16 to 4×4 luminance samples with many options between the two. The luminance component of each macroblock (16×16 samples) may be split up in 4 ways as 16×16, 16×8, 8×16 or 8×8. If the 8×8 mode is chosen, each of the four 8×8 macroblock partitions within the macroblock may be split in a further 4 ways as 8×8, 8×4, 4×8 or 4×4. The four macroblock type sizes and four macroblock subtype sizes are shown in figure 5. These partitions and sub-partitions give rise to a large number of possible combinations within each macroblock.
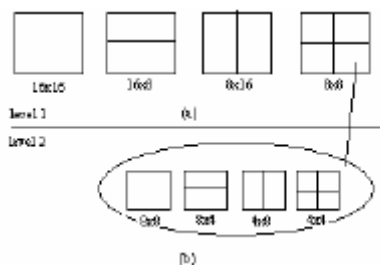

Figure 4: Variable block size in H.264, (a) sizes for a MB type, (b) sizes for a sub MB type.

A separate motion vector is required for each partition or sub-partition. Each motion vector must be coded and transmitted; in addition, the choice of partition(s) must be encoded in the compressed bitstream. Choosing a large partition size (*e.g.,* 16×16, 16×8, 8×16) means that a small number of bits are required to indicate the choice of motion vector(s) and the type of partition; however, the motion compensated residual may contain a significant amount of energy in frame areas with high details. Choosing a small partition size (*e.g.,* 8×4, 4×4, etc.) may give a lower energy residual after motion compensation, but requires a larger number of bits to signal the motion vectors and the choice of partition(s). The choice of partition size therefore has a significant impact on compression (a small partition size may be beneficial for detailed areas).

H.264/AVC as an enhanced reference picture selection as H.263++ enables efficient coding by allowing an encoder to select (for motion compensation purposes) among a large number of pictures that have been decoded and stored in the decoder. This paper presents a method for improving the quality of video transport using H.264/AVC over wireless networks that is the prioritization of different parts of Intra- and inter-prediction data.

3. Robust Transmission of Intra Prediction Data

Intra prediction treats a video frame as a still image without any temporal prediction employed. In this case, the frames are coded without any reference to video information in previous frames. In intra frame (I-frame) coding mode, all MBs of a frame are intra coded. Therefore, an INTRA frame could be used to refresh the picture quality after a certain number of frames have been INTER coded. Moreover, INTRA frames could be used as a trade-off between the bit rate and the error robustness. For robust video communications, the Intra coded blocks must be optimally coded. Protecting the INTRADC coefficients of an I-frame with a convolutional coder does not make an I-frame fully resilient to channel errors. A bit error that corrupts one of the variable-length codes of an I-frame leads to the loss o synchronization even when INTRADC coefficients are protected with FEC techniques. To reduce the vulnerability of I-frames to channel errors that lead to loss of synchronization, all the fixed-length INTRADC coefficients must be transmitted using NAL-A. For a QCIF-size sequence, the fixed length section of the I-frame contains 594 INTRADC coefficients, 16-bit long each with half-rate convolutional coding. If an error is still detected in the fixed-length segment of the I-frame, the decoder preserves its synchronization and goes to the next 16-bit protected INTRADC coefficient in the I-frame. However, if an error is detected in one of the VLC words in the variable-length section, the decoder sets to zero all the AC coefficients following the position of error. This error resilience scheme is used in this paper that gives a noticeable improvement to the subjective and objective quality of the I-frame, as indicated by figure 4 and 5, respectively.

Figure 5: Frame 71 of the walking person sequence encoded at 64 kbit/s and transmitted over a GPRS channel with random error distribution and BER=10^{-4} : (a) ordinary H.264, (b) Robust I-frame coding.
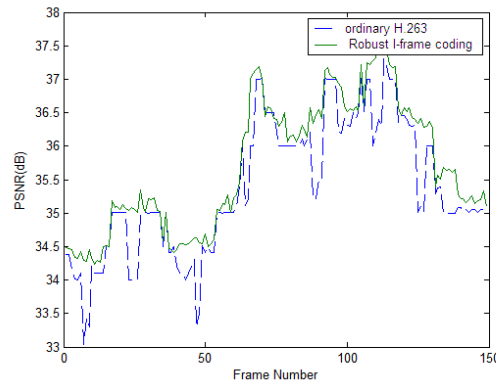
Figure 6: Y-PSNR values for 150 frames of the walking person sequence encoded with ordinary H.264 and robust I-frame coding, BER=10^{-4}.

4. INTER FRAMES *CODING USING UNEQUAL ERROR PROTECTION OVERE GPRS*

A. Sensitivity to Errors of H.264/AVC Video Parameters

In a video sequence, adjacent frames could be strongly correlated. This temporal correlation could be exploited to achieve higher compression efficiency. Exploiting the correlation could be accomplished by coding only the difference between a frame and its reference. In most cases, the reference frame used for prediction is the previous frame in the sequence. The resulting difference image is called the residual image or the prediction error. This coding mode is called INTER frame or predicted frame (P-frame) coding.

INTER coding mode uses the block matching (BM) motion estimation process where each MB in the currently processed frame is compared to MBs that lie in the previous reconstructed frame within a search window of user-defined size. The displacement vector between the current MB and its best match 16*16 matrix in the previous reconstructed frame is called the motion vector (MV) and is represented by a vertical and horizontal component.

Motion prediction process in block-based video coding algorithms makes the compressed bit stream more sensitive to errors and information loss. For instance, a bit error in differential MV of a MB or one of its candidate MV predictors can lead to incorrect reconstruction of the MB. The reason is that the decoder becomes unable to compensate for the motion of the currently processed MB with respect to its best-match matrix in the reference frame. An erroneous MV leads to incorrect reconstruction of its corresponding MB and other MBs, whose MV depend on the erroneous MV as a candidate predictor. This MV dependency is the main reason for video quality degradation in both spatial and temporal domains. This effect can be mitigated if the actual MV is Transmitted using NAL_A. encoded instead of the prediction residual. In Figure 7, for *walking person* sequence encoded at 30 Kbit/s, the quality of reconstructed frames using MVD is compared with that of actual MV value.
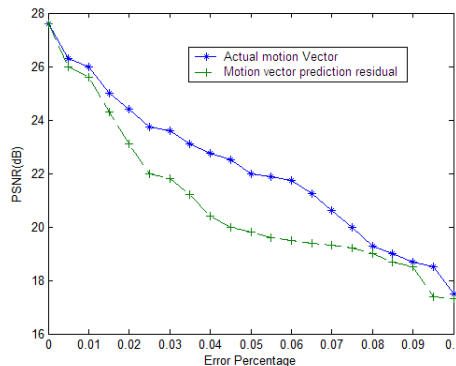

Figure 7: PSNR values at different error rates with and without motion vector prediction for walking person sequence.

Motion-compensated residual values such as Frame intensity values are coded as texture. The basic tools for coding the texture are similar to the MPEG-1 and MPEG-2. Thus, they are transformed using the H transform, and then scalar quantized and entropy coded by a CAVLC or CABAC. Usually, the majority data of each packet is made up of texture information; the loss of which causes much less distortion at decoder than if motion or header data is lost. It is obvious that MVs and synchronization words are more sensitive to errors than texture data. Here, *walking person* and Stefan sequences are used to analyze the error sensitivity of data in the first and second partitions of an H.264 video packet. At the decoder, a simple error concealment technique sets both MVs and texture blocks of the concealed I-frames to zero, while it copies the same MB from the previous frame to the current error-concealed I-frames. Figure 8 shows that while texture errors can be concealed with reasonable efficacy, the concealment of motion data results in reconstructed frames that contain a high degree of distortion. The subjective results shown in Figure 8 confirm the above mentioned error sensitivities. These errors are demonstrated by the PSNR values of Figure 9. Corruption of texture produces little effect in terms of visible distortion until the bit stream is subjected to high error rates.
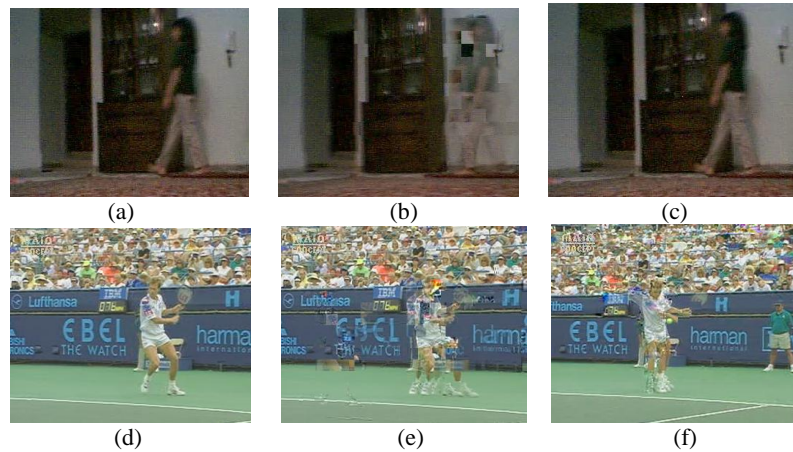
Figure 8: (a), (d), Error free sequences. (b), (e), Motion data,
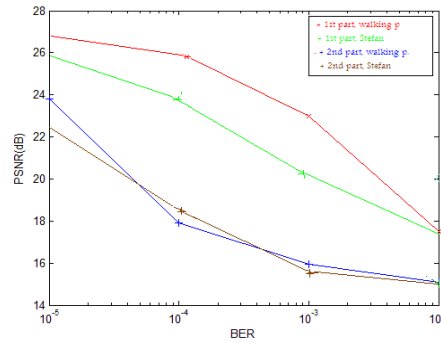(c), (f), texture data, all corrupted at BER= 5×10^{-4} .

Figure 9: Sensitivity to errors of H.263 video parameters generated by *walking person* and Stefan sequences, with corruption of the first and second partition.

B. General Packet Radio Service:

The 2G mobile cellular networks, namely GSM, do not provide sufficient capabilities for routing of packet data. In order to support packet data transmission and allow the operator to offer efficient radio access to external IP-based networks (such as the internet and corporate intranets), the *general packet radio service* (GPRS) has been developed by the *European telecommunication standards institute* (ETIS) and added to the GSM. The GPRS permits packet mode data transmission and reception based on IP technology. For data packet transmission in the GPRS network, at the time the session is set up, the mobile terminal is identified by an IP address assigned to it either permanently or dynamically. In addition to the packet structure, the QoS of video communications over the future mobile networks depends on a number of other parameters, namely the available throughput and the employed channel coding schemes. For example, the GPRS data is transmitted over the *packet data traffic channel* (PDTCH) after being error-protected using one of four possible channel protection schemes, namely CS-1, CS-2, CS-3, and CS-4. Table 1 shows the data rates per timeslot for each of the four channel protection schemes. The first three coding schemes use convolutional codes and block check sequences of different strengths to produce different protection rates. The CS-2 and CS-3 use punctured versions of the CS-1 code, thereby allowing for a greater user payload at the expense of reduced performance in error-prone environments. However, the CS-4 only provides error detection functionality and is therefore not suitable for video transmission purposes. The four channel coding schemes supported by the GPRS can be used to offer different levels of protection for separate video streams.

TABLE 2
GPRS CHANNEL CODING SCHEMES

Code Scheme	Radio block rate	Data rate/slot payload (bits)	(kbit/s)

CS1	1/2	181	8.0
CS2	2/3	268	12.35
CS3	3/4	312	14.25
CS4	1	428	20.35

In this work, the stream prioritization scheme described above was tested on simulated GPRS mobile access channel, so that each stream will be transported over a different mobile bearer channel as offered by the underlying network.

C. H.264 Data Prioritization partitioning: Proposed Method

Since real-time video transmissions are sensitive to time delays, the issue of re-transmitting the erroneous video data is impractical. Therefore, other forms of error control strategies must be employed to mitigate the effects of errors inflicted on coded video streams during transmission. Some of these error control schemes employ data recovery techniques that enable decoders to conceal the effects of errors by predicting the lost (or corrupted) video data from previously reconstructed error-free information. These techniques are decoder-based and incur no changes on the employed transport technologies. More ever, they do not merge any redundancy on compressed video streams and thus are referred to as zero redundancy error concealment techniques. Concealment at the decoder is based on exploiting the spatial and temporal data redundancy to obtain an estimate of the lost texture data. The efficiency of error concealment depends on the amount of redundancy in texture and compressed bit streams that are not removed by the source encoder. To be more specific, error resilience for compressed video can be achieved through the addition of suitable transport and error concealment methods, as outlined in the system block diagram shown in Figure 10.

Another error control scheme used for H.264 allows multi-layered video coding. The compressed bit stream in the video sequence consists of a number of layers; namely the NAL-A and a number of enhancement layers such as NAL-BC. The base layer contains information that is essential for texture reconstruction, while the enhancement layers contribute to improve the perceptual quality at the expense of additional overhead bits. The compression ratio of enhancement layers is a compromise between the coding efficiency and video quality. If the channel can handle high bit rates, then more enhancement layers can be accommodated to improve the output quality. Conversely, in situations such as congestion of network links, only the base layer is transmitted to avoid traffic explosion and to guarantee the maximum possible video quality. In addition to its scalability benefits, the layered video coding has inherent error-resilience benefits; particularly when the base layer can be transmitted with higher priority and the enhancement layers with lower priority. The layered video coding is usually accompanied by the use of unequal error protection to enable the high-priority base layer to achieve a guaranteed QoS and the enhancement layers to produce quality refinement. This approach is known as layered coding with transport prioritization, and is used extensively to facilitate error resilience in video transport systems.

A similar method to improve the quality of video transport over networks is the prioritization of different parts of the video bit stream by sending data as different separate streams. In this work we use data partitioning prioritization for H.263 robust I-frame, which the error resilience is improved for video bit stream by sending the data as two separate streams. This enables the video encoder to demand the network to send the data using channels with different priorities. This allows the encoder to allocate a higher priority to the more error-sensitive data and transmit it over higher quality and more reliable channels. This Section describes the proposed prioritization method using H.263 over a GPRS channel model. In this paper, in addition to prioritization error control, a simple error concealment technique sets both the MVs and the texture blocks of the concealed inter frames to zero, while it copies the same MB from the previous frame to the current error-concealed intra frames.

The simulated scenario is shown in Figure 11, in which the video frame is H.263 encoded and separated into two segments, with high- and low-priority. Then, the video data are produced by the encoder and the RTP/UDP/IP protocols are encapsulated to them. They are then passed through different schemes of wireless GPRS links and sends to the end system. The NAL-A and NAL-BC partitions are streamed using the CS-1 and CS-3, respectively. At the end system the two different radio bearer channels are decoded and combined to reconstruct the data.

The comparison is made among partitioning the proposed method using CS-1, CS-3, and single streaming over CS-2. For real-time operation, video packets are encapsulated into RTP packets and transmitted using the UDP transport protocol. Figure 12 shows the PSNR values obtained for *walking person* sequence coded with H.264 and sent over GPRS using the prioritized video transport technique described above. Here, the two data-partitioned output streams are generated using *mobile multimedia systems* and H.264 verification model software. They are subsequently encapsulated into RTP packets for real-time transmission. The CS-1 is used to protect the high-priority video stream and CS-3 is used to protect the low-priority stream for the prioritized video transport of partitioned video data. However, only the CS-2 is used to protect the single stream output with no prioritization. The PSNR values (see Figure 12) show that the

prioritization of video streams for UEP protection and transport over two GPRS radio bearer channels offers a better objective performance than the single stream case with EEP and no transport prioritization. Figure 13 shows the subjective quality resulted from frame 71, for single and prioritized stream transmissions.
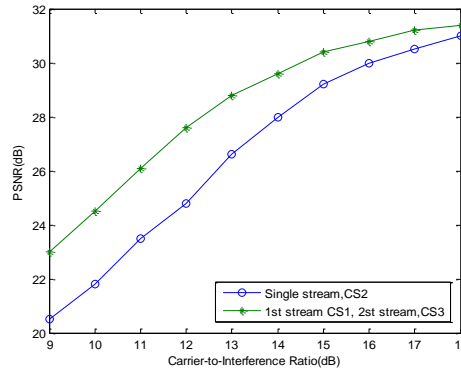

Figure 10: PSNR for *walking person* sequence encoded with H.263 translated over a GPRS channel using both the single stream and prioritized stream transport mechanisms.

Figure 11: Frame 71 of *walking person*. (a) Single stream transmission. (b) Prioritized stream transmission.

5. SIMULATION RESULTS

H.264 standard is used for illustration. Also the stream prioritization scheme described above was tested on simulated GPRS mobile access channel, so that each stream will be transported over a different mobile bearer channel as offered by the underlying network.

Also a variable to fixed length code system solution is introduced, that avoids the errors propagation due to synchronization loss in the payload. In typical video sequences, and in H.264 standard, the H transform coefficients represent about 80% of the bit stream. They are sensitive to synchronization loss because most of them are CAVLC encoded. In the proposed solution, all these information bits are no more sensitive to synchronization loss. A typical video sequence called "Stefan" is used for the simulation results(figure 15). An error on one bit on a non null frequency DCT coefficient, noted AC, is created in the H.264binary stream and in the proposed solution binary stream. The error is on the same position and the same codeword in both cases for comparison. The error becomes local: only the codeword that is corrupted by the erroneous bit is corrupted, and the remaining bit stream is still correct. So, there is no more error propagation due to CAVLC loss of synchronization. With the new solution, the video bit stream can be transmitted using less protected transmission schemes which have higher data rates. The video quality is controlled since the number of bits used to encode CAVLC codewords into fixed sized codeword is based on the probability of occurrence of the codewords. This adaptive system reduces the needed data rate for the transmission for similar image quality.

The PSNR values shown in Figure 16, demonstrates the effectiveness of the proposed error resilience tools in improving the quality of a video service over the GPRS. As shown in these figures, data prioritization partitioning using robust I-frame with synchronization has the best performance.
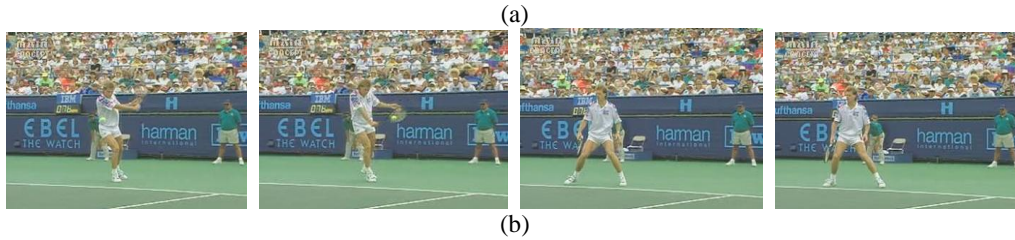
(a)

(b)

Figure 12: (a) An error on one bit of one AC coefficient is created in the standard H263 bit stream. (b) The same error is created in the new solution H.263 bit stream. The error is not visible.
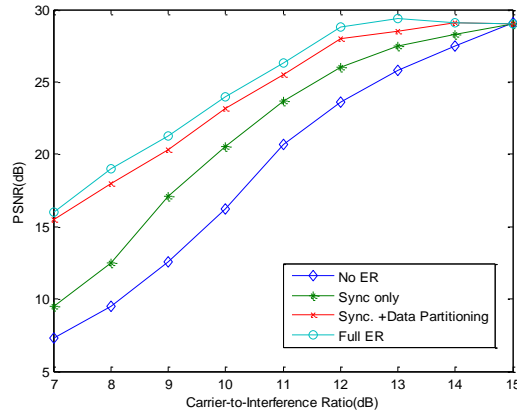

Figure 13: PSNR values for robust I-frame video transmission over GPRS, for Stefan sequence.

VI. CONCLUSION

In this paper, we proposed a prioritized streaming technique for video, I- and P-frame, translation over simulated GPRS mobile networks. Also a variable to fixed length code system is introduced that avoids the errors propagation due to synchronization loss in the payload. Subjective and objective results have been analyzed. The results have shown that the proposed method provides considerable improvement in QoS, when transmitting image sequences over GPRS mobile channels.

REFERENCES

[1] *Overall description of GPRS radio interface*, 3GPP TS 43.064, V4.0.0, 01-2001.

[2] *ITU-T Recommendation H.263*, 03-96.

[3] *ITU-T Recommendation H.263 (H263+)*, 02-98.

[4] *H26L Test model*, ITU-T Standardization Sector, Study Group 16,VCEG, june 2001.

[5] Takishima, Wada, Murakami, "Reversible Variable Length Codes*"*, *IEEE Trans. on Comm.*, Feb.-March-April 1995   volume: 43, issue: 2, part: 3, page(s): 158 –162.

[6] Itoh, Cheung, "Universal Variable Length Code for DCT Coding", *Proceedings of IEEE International Conference on Image Processing*, 2000, page(s): 940 -943 vol.1.

[7] Wen, Villasenor, "Reversible Variable Length Codes for Efficient and Robust Image and Video Coding*"*, *IEEE Conference Record of the Thirty-First Asilomar Signals, Systems & Computers*, 1997, volume: 2, page(s): 973 –979.

[8] Hang Nguyen and J. Brouet and P. Duhamel, Robust and adaptive transmission of compressed video streams over EGPRS, 2004, January, Proceedings of CCNC'04 (IEEE Consumer Communications and Networking Conference), Las Vegas, USA.

[9] R. Garg, M. Jindal, M. chauhan, "Statistics Based Fast Intra Mode Detection" , Proc. of SPIE Vol. 5960, pp 2085-2091, 2005.

[10] B. Jeon and J.lee, "Fast Mode Decision for H.264", ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT 10th Meeting, Waikoloa, Hawaii, December 2003.

[11] C. Yang, L. PO, W. Lam,   " A Fast H.264 Intra-Prediction Algorithm Using Macroblock Properties", ICIP, pp. 461-464, 2004.

[12] B. Meng, O. C. AU, C. Wong, H. Lam,   " Efficient Intra-Prediction Algorithm in H.264", IEEE, pp. 837-840, 2003.

[13] C. Tseng , H. Wang, J. Yang,   " Improved and   Fast Algorithms for Intra-4×4 Mode Decision in H.264/AVC", IEEE, pp. 2128-2131, 2004.

[14] C. Hsu, M. Ho, J. Hong, " An Efficient Algorithm for Intra- prediction in H.264", IEEE, pp. 35-36, 2006.

[15] Joint Video Team (JVT) , reference software JM7.1, http://bs//hhi.de/~suehring/tml/download/JM7.1.zip.

[16] M. Jafari, S. Kasaei, "An Efficient Intra-Prediction b Mode Decision Algorithm for H.263 to H.264 bTranscoding", 10th IEEE international conference on computer systems and applications, page 1082-1089, march 2006.

[17] M. Jafari,s. Kasaei," Prioritisation of data partitioned MPEG4 video over GPRS/EGPRS mobile networks", Asian internet engineering conference (AINTEC), Tailand, pp. 68-82, December 2005.

[18] I. Richardson, "H.264/MPEG-4 part 10 white paper", available:
http://www.vcodex.fsnet.co.uk/resources.html.

[19] C. Kim, H.-H. Shih, and C.-C. J. Kuo, "Feature-based intra- Prediction mode decision for h.264, " in *Proc. IEEE International Conference on Image Processing*, 2004.

[20] Yu-wen Huang, Bing-Yu Hsieh, Shao-Yi Chien, Shyh-yih Ma, and Liang-Gee Chen, "Analysis complexity reduction of multiple reference frames motion estimation in H.264/AVC",February 2006.

[21] G. Sullivan, "Recommended simulation common conditions For H.26L coding efficiency experiments on low resolution Progressive scan ource material," presented at the 14[th]VCEG- N81 Meeting, Santa Barbara, CA, Sep. 2001.

[22] Jiang Gang-yi, Li Shi-ping,Yu Mei, Li Fu-cui, " An efficient Fast mode selection for intra-prediction ", IEEE Int. Workshop VLSI Design & video Tech. , China, pp. 357-360, May 28-30, 2005.

[23] Yin, P., Tourapis , H.C., toupis, A.M., and boyce, J.: 'Fast mode decision and motion estimation for JVT/H.264'. Proc. Int. Conf. on Image Processing, Barcelona, Spain, 2003,Vol.3,pp. 14-17.

[24] Lim, K.P., Wu, S.,Wu, D.j.Rahardja, S., Lin, X., Pan, f., and Li, Z.G.:' Fast inte mode selection'. 9th Joint video Team (JVT) Mtg, San Diego, CA, USA, September 2003.

[25] H. Kim, Y. Altunbasak, "Fast Mode Decision for Inter Prediction in H.264," ICIP 2004, Singapore, 2004.

[26] Y.H. Kim, J.W. Yoo, S.W. Lee, J. Shin, "Adaptive mode decision for H.264 encoder," in Electronic Letters, Vol.40 No.19, September, 2004.

[27] A. C. Yu, G. Martin, "Advanced block size selection algorithm for inter frame coding in H.264/MPEG-4 AVC," in ICIP 2004, Singapore, 2004.

[28] Q. Dai, D. Zhu, R . Ding, "Fast mode decision for inter prediction in H.264," in ICIP 2004, Singapore, 2004.

[29] M.Jafari, S.Kasaei, 'fast intra and inter prediction mode Decision in H.264 advanced video coding, may 2008, IJCSNS.