WINDOWED REFERENCE PICTURE SELECTION FOR H.264 TRANSMISSION ERROR RECOVERY

Pat Mulroy, Mike Nilsson

Broadband Applications Research Centre, BT, Adastral Park, Ipswich, UK

ABSTRACT

For conversational real time applications we are concerned with low latency, high efficiency techniques for rapid recovery from transmission errors and in this paper we propose the use of 'windowed' references and standard compliant reference remapping instructions within H.264 to aid in recovery of unreliably transmitted multi-reference inter predicted video. Key benefit of our approach is that the amount of reference picture memory required for our windowed reference picture selection technique is reduced for feedback channels suffering from high round trip time.

Index Terms— windowed RPS, high rtt

1. INTRODUCTION

The strong inter frame dependencies in compressed video mean uncorrected transmission errors can cause long lasting quality degradation. In applications without a feedback channel (e.g. broadcast TV) error resilience can be increased through the use of forward error correction techniques and the addition of regular sync (intra) frames in the video to restore the prediction path. These techniques increase error resilience but at the expense of compression efficiency. In applications with a feedback channel (e.g. VOD) packet retransmission can be a suitable technique. Retransmissions will only be triggered if a loss occurs so overhead is a direct function of loss rate and there need be no unnecessary compression efficiency drop. The approach however can lead to unacceptable latencies particularly for conversational video telephony applications.

2. REFERENCE PICTURE SELECTION

The use of alternative reference frames in the predictive coding loop was first introduced in the standards in an optional annex to H.263 – Annex N called the Reference Picture Selection (RPS) mode and in MPEG-4 as NEWPRED [1][2]. The idea was to make multiple references available at both encoder and decoder so that if feedback from decoder to encoder indicated a reception error both could switch to using a known good reference frame. Inter prediction using an older reference frame would still be more efficient than a complete intra update. Feedback could be in the form of positive or negative acknowledgement – so called ACK or NACK modes.

H.263 – Annex U extended this concept to include multiple reference frames in the predictive coding loop as a general coding efficiency tool and the management of the decoder reference store through the use of reference memory management control operation (MMCO) syntax. This enhanced reference picture selection mode was later subsumed into the latest H.264 video coding standard and is one of the reasons H.264 outperforms earlier standards such as MPEG-2.

3. WINDOWED REFERENCES

To accommodate a high round trip time, that could be many multiples of the frame period, RPS demands a high reference memory storage requirement. Limited reference memory RPS has been studied in [3] but references are restricted in this approach to the most recent set. Our approach extends the protection offered for the same constraint on memory. We propose a time-windowed approach of reference frames with both encoder and decoder maintaining a subset of frames from specific time periods with respect to the current frame. As frames are coded this reference subset can be managed so that there are always both very recent references for good compression efficiency and old references suitable for error recovery with high delay feedback.

In our proposed system we retain the two immediately preceding frames so that in a no packet loss situation multiple recent references are still available and compression efficiency should remain good. In addition, at least one frame that is earlier than the current frame by an amount greater than or equal to the round-trip time (rtt) is retained, so that if notification of a packet loss is received a potential predictor frame is available and an intra update can be avoided. This retention is however subject to an upper limit, and any frame older than this limit is discarded. In order to make it possible to maintain this condition in the steady state it is in general necessary to retain one or more frames that are intermediate in age between the most recent two and the one(s) older than the round-trip time. In this example, it is convenient to refer to the age of a frame: thus, if the current frame is frame n, then the age of frame j is n-j.

Essentially we define three window types ("recent" frames, "intermediate" frames and "old" frames) and a window size *m* being the age range over which references in the intermediate and old windows span. In one

example configuration with four reference buffers, we have the:

- recent window with two frames ages 1 and 2,
- intermediate window with at least one frame and age(s) ranging from 3 to *m*+2 and the
- old window with at least one frame and age(s) ranging from *m*+3 to 2*m*+2.

with *m* chosen in our system such that $rtt \leq m+2$ and *rtt* being the round trip time of our loss feedback mechanism measured in frame periods. In this way we ensure that once we reach steady state with a populated old window we can accommodate a loss feedback message on this channel and we should always have a valid predictor available. The age of the picture in our "old" window will vary from $m+3$ to $2m+2$ in configuration above so choice of *m* will also affect predictor efficiency. At the youngest end of this range correlation should be as high as possible. At the oldest end correlation will be lower but probably still better than intra update. Adding more reference buffers to this system would imply adding more intermediate windows with consecutive age ranges and would give some protection against loss of the feedback message itself.

The rules for the reference buffer management are:
1) Do not remove a frame unless the buffer is full
2) Always keep the two most recent frames, i.e. ages 1, 2
3) Subject to the 2*m*+2 limit, always keep the oldest frame
4) Keep sufficient frames that when the oldest frame ages beyond the "old" window there is still a frame in this "old" window.

This implies that a frame shall not be removed from the buffer if the time difference between the next oldest frame in the buffer and the next youngest frame in the buffer exceeds the size of the old window (i.e. *m*). Where, as here, the intermediate and old windows are the same size, this requires at least one frame always in the intermediate window. The ideal scenario is to end up with the references spread in time in a steady state configuration. One way to ensure this is by working back from the oldest frame and keeping the remaining references evenly spread; (*oldest*-*m*), (*oldest*-*2m*) etc. or closest if these are not available.

So working through an example with a buffer capacity of 4 frames, and *m*=3, the ranges are [1, 2], [3 to 5], [6 to 8]

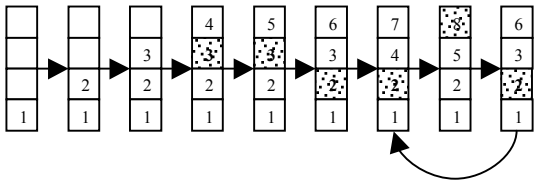

Figure 1 - Windowed reference memory progression

Figure 1 illustrates the progress of the contents of the buffer starting from empty where each number represents the age of a frame in the buffer and the retention criterion is to retain that frame whose age is closest to $n_{MAX} - m$, where n_{MAX} is the age of the oldest frame in the buffer and *m* is, as before, the range of the "old" window. The frames with ages shown as shaded are removed in the next iteration.

4. MEMORY MANAGEMENT CONTROL OPERATIONS

In order to keep both reference buffers at encoder and decoder exactly synchronized it is necessary to transmit frame deletion instructions from the encoder to the decoder. Such messages are sent using the "memory management control operations" defined in H.264. The decoder buffer follows the encoder buffer by acting on these instructions that the encoder has put in the bitstream. To protect against loss of these instructions and to provide for better error concealment at the decoder we also code using Flexible Macroblock Ordering mode in the chessboard formation. This results in four slices per picture and as each NAL is required to carry our "memory management control operations" (MMCO commands) we have extra resiliency built in here. MMCO commands can be used to remove a short or long term picture from the buffer, map a short term to a long term, empty the buffer etc. and multiple MMCO commands can be sent in the same slice (NAL unit).

If the decoder loses a set of MMCO commands, there will be mismatch between the contents of the encoder and decoder buffers. To alleviate this, we propose a modified version in which every slice contains a message informing the decoder as to exactly which frame should be in the decoder buffer at that time. One way of doing this is to make use of the "reference picture list reordering" commands defined by H.264. The way these function is as follows. The encoder has a set of reference frames. It may choose to use one, some or all of these when encoding a frame. Conceptually, these frames are put in a default order, starting with the most recent (largest frame number, ignoring wrap issues, proceeding to the oldest short term frame, followed by the long term frames (if any) in order of increasing long term index. The encoder may optionally include in the bitstream a set of remapping instructions that alter this default order into some other order, known as the remapped order. The main purpose of remapping instructions is to improve the entropy coding efficiency of indices that identify the reference frame to be used for prediction. These remapping instructions can be made use of in the present context for indicating to the decoder which frames (by their frame numbers) should be in its buffer.

The way we do this is for the encoder to insert remapping instructions in every slice (NAL unit) that reference all frames in its buffer (there is not necessarily any changing of the default order), and for the decoder to use these to determine which frame to remove from its buffer when it

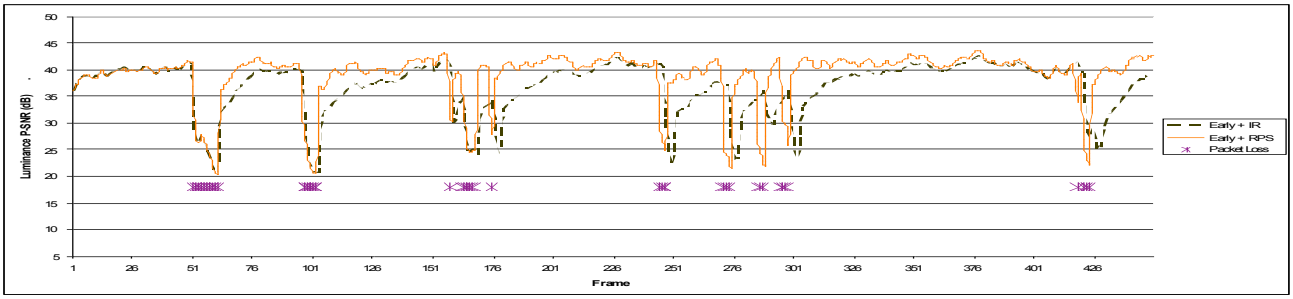

Figure 2 - PSNR Comparison of Intra Refresh and Reference Picture Selection recovery of multi reference H.264 ("Silent voice", QCIF@80 kbps, 12.5 Hz, E=5%, B=10, loss feedback delay <80ms)

otherwise does not have the required knowledge – perhaps because there is no MMCO removal command because the encoder has fewer frames in its buffer, or because an MMCO removal command refers to a frame the decoder does not have. There is also a defined "Decoded reference picture marking repetition" SEI message which could be used to reduce probability of buffer mismatch further.

H.264 also has the concept of long and short term reference picture lists. Long term references bring compression gains particularly with character dialogue type sequences with different cameras switching between views over long timeframes. We decided not to use long term references in our scheme as we would be vulnerable to the loss of the MMCO remapping instruction. Remapping instructions refer to short term references by frame number whereas they refer to long term references by shared indices. It would not be as easy to realize that both buffer sets were mismatched using these long term references. Even with short term references it would still be possible to arrive at a buffer mismatch situation (due to frame number wrap). A fallback solution we adopted here was to calculate an encoded frame CRC and send in-band in a user data SEI message. The decoder was configured to compare this SEI message if received with the locally calculated decoded frame CRC. Any mismatch detection was fed back to the encoder and corrected with intra refresh as a last resort.

5. WINDOWED RPS SYSTEM

Our proposed RPS framework was implemented using H.264 encoding with RTP/UDP as the transport and a NACK based early feedback scheme as detailed in the RTP/AVPF internet draft [4]. This profile was investigated previously by the authors in [5] and found to be an effective error recovery profile especially when combined with RPS. Packet loss in this framework is detected at the RTP layer through RTP sequence number discontinuity or reception timeout. RTCP Transport feedback messages were populated with generic NACK indications and sequence IDs of the lost RTP packet(s).

To simulate burst packet losses in our system a simplified Gilbert-Elliot 2-state packet loss model [6] was used. In this model the network is either in a good "no loss" state where all packets are delivered successfully or in a bad "loss" state where all packets are lost. *Figure 3* below gives the state diagram where $P_L = 1 - 1/B$ and $P_N = E / B(1-E)$ and E and B represent the average packet loss rate and average burst length respectively.

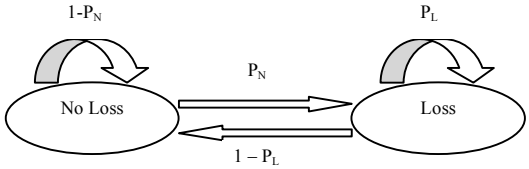

Figure 3 - State diagram of Gilbert-Elliot packet loss model

The parameters used in our simulations were B = 10 and E = 5% giving $P_L = 0.9$ and $P_N = 0.00526$ respectively.

Figure 2 illustrates the benefit in P-SNR terms of the RPS scheme when combined with the early feedback RTP profile. Packet loss bursts of between 1 and 25 packets (up to 2 second bursts) are accommodated by our implementation with much quicker return to good quality frames and an average of 2.5 dB improvement is recorded over the early feedback intra refresh scheme for a run of 1200 frames. The 12 second sequence was looped for these tests.

Figure 5 also shows the subjective improvement over the early intra refresh scheme just after a repair. As the codec still strives to maintain the target bit budget it is forced to quantize the intra frame heavily and the block artifacts are visible. With RPS inter prediction is still possible using the older references.

The benefit of windowed RPS is best shown at higher loss feedback delays. In *Figure 4* and *6* we compare high rtt performance of intra refresh, normal RPS (i.e. configured to hold the 4 most recent frames) and windowed RPS with *m*=5. The 25Hz version of "Silent Voice" was coded for these tests with 4 reference frames and using 4 slice groups per P picture and the same burst packet loss pattern as for *Figure 2*. Packet loss in the decoder was detected by RTP sequence number discontinuity only which meant the loss notification was delayed by the burst length and a fixed delay parameter. In *Figure 4* the feedback delay

varied between 40 and 280ms for each burst. The normal RPS configuration scores marginally better than intra refresh here but is frequently overwhelmed. Windowed RPS with *m*=5 guarantees to have a usable reference for up to 7 frame periods (280ms) and offers the possibility of a usable reference for up to 12 periods (480ms). Results validate this with windowed RPS outperforming both intra refresh and normal RPS over the extended range and no worse at even higher rtt.

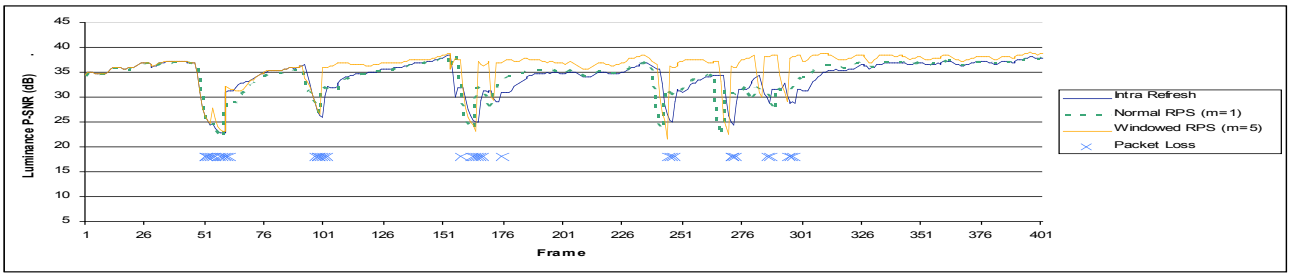

Figure 4 - PSNR Comparison of Intra Refresh, Normal and Windowed RPS at higher loss feedback delay

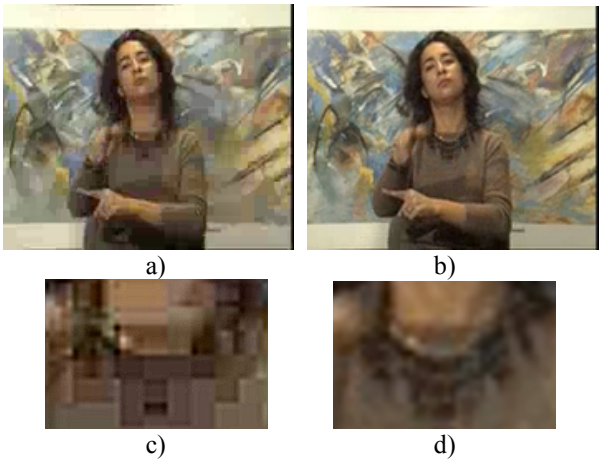

Figure 5 - Decoded frame just following repair using a) Intra refresh (IR) scheme b) RPS scheme. c) enlarged neckline IR frame d) enlarged section from RPS frame

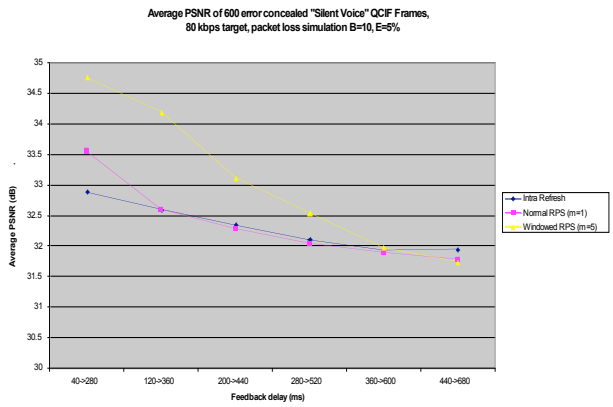

Figure 6 Comparison of Intra Refresh, Normal and Windowed RPS over high loss feedback ranges

6. CONCLUSIONS

This paper detailed an extension to the basic Reference Picture Selection technique using a windowed reference framework within H.264. It has been shown that even in high round trip environments RPS as a technique can be effectively used without requiring a large memory overhead. It has further been shown that a novel interpretation of the "reference picture list reordering" commands in H.264 can be an effective way to synchronize encoder and decoder buffers.

7. REFERENCES

[1] ITU-T/SG15/LBC-96-033, "An error resilience method based on back channel signalling and FEC", Telenor R&D, San Jose, Jan 1996.

[2] S Fukunaga, T Nakai and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures", Proc. IEEE GLOBECOM, vol.3, Nov 1996 pp 1503-1508

[3] Tomita Y, Kimura T, Ichikawa T, "Error resilient modified inter-frame coding system for limited reference picture memories", Picture Coding Symposium. PCS 97, 10-12 Sept. 1997, VDE-Verlag, ITG-Fachberichte, no.143 pp 743-8

[4] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol(RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, July 2006.

[5] P Mulroy, "Application layer QoS for videotelephony", BT Technology Journal Vol. 24 No. 2 April 2006

[6] Wang H S and Moayeri N: "Finite State Markov channel – a useful model for radio communication channels", IEEE Trans. On Vehicular Technol, 44, No 1, pp 163-71 (February 1995)