Video Redundancy Coding in H.263+

Stephan Wenger
Technische Universität Berlin
stewe@cs.tu-berlin.de

ABSTRACT: *The forthcoming new version of ITU-T's advanced video compression recommendation H.263 [1] includes several optional modes for the support of packet networks. This paper gives a brief description of those modes and discusses in detail a new method for temporal error resilience, called Video Redundancy Coding, which is one possible usage of one of the optional modes. In conjunction with spatial error resilience mechanisms, Video Redundancy Coding has been proven to be a superior method for achieving high quality video transmission over non guaranteed QoS packet networks that have packet loss rates as high as 20%, with a minimal additional coding overhead.*

1. INTRODUCTION

One major trend of the multimedia communication research field is the transmission of audio and video signals over non guaranteed QoS networks, especially the Internet. Research activities in this area are often based on the MBONE environment [2]. A high percentage of this research tries to analyze the implications of packet losses to multimedia data streams, especially audio and video data [3], [4]. Since packet losses are very critical to the modern video coding algorithms, which are usually based on frame-to-frame prediction techniques, the video information data path, from the technical point of view, is more critical than the audio data path[1]. This is due to the high compression ratio of modern video compression algorithms that lead to a very small amount of redundancy.

Most of the published research describing mechanisms to minimize the impact of packet losses are based on three different ideas:

- *Avoiding temporal dependencies* of the coded pictures by using only spatial compression techniques, sometimes combined with the subdividing of the picture into small picture segments, and a decision process on the importance of the transmitting of those segments. Even by using the most advanced still image compression techniques, these methods give usually a poorer performance in terms of video quality compared to the algorithms based on temporal prediction techniques, like ITU-T H.263 or the MPEG family standards. Examples for this type of error resilience can be found in tools like nv [5] and vic [6].
- Using *rate control mechanisms* in trying to avoid packet loss on the network. These mechanisms have been proven to improve the packet loss characteristics of a virtual connection [7], [8].
- Using *Forward error correction* mechanisms, which allow the reconstruction of packets by sending error correction information packets (and thus adding redundancy) [9]. These methods are data type independent and consequently can be used with all video and audio coding schemes, including those depending on temporal prediction. Their most important disadvantage, however, is the added latency time.

In this paper, we introduce a fourth scheme, which is to add mechanisms for temporal error resilience into the video coding itself in such a way that the coded video stream is to some extend resilient against packet losses. We do this by adding some redundant information to the video stream, and consequently the method is called Video Redundancy Coding. Video Redundancy Coding (VRC) was introduced to the Advanced Video Expert Group of the ITU-T (now Q 15 in SG 16) and codepoints for the usage were adopted in the forthcoming ITU-T recommendation H.263+, hence allowing the use of VRC in a standardized manner. The basic algorithm, however, can be used in conjunction with any video coding scheme based on temporal prediction techniques, like the MPEG standard family.

2. THE ITU-T RECOMMENDATION H.263+

In the series of ITU-T recommendations, describing the coding of video signals, H.263 is the most innovative one. Currently, a new version of H.263 is about to undergo the final standardization process. This new version is known under its working name of H.263+.

Similar to the four optional modes of H.263, H.263+ offers 16 optional modes, which can be independently chosen by the coder. However, certain restrictions apply in terms of mode combinations. ITU-T is about to publish documents on the recommended mode combinations to allow easier capability negotiations and achieve higher interoperability levels. Most of the optional modes, including all four modes which are present in today's H.263, are designed to allow a tradeoff between computorial complexity and coding performance/picture quality. However, three of the new modes can be used to achieve a high gain in error

[1] This is true even if the fact that the hearing sense is much more critical to information loss than the visual sense is considered.

resilience[2]. These modes will now be briefly introduced.

2. 1. Slice Structured Mode

The Slice Structured Mode was introduced to allow the division of not yet coded YUV pictures into picture segments of any size at a Macroblock granularity. This, especially when used in conjunction with the Independently Segmented Decoding Mode, allows the adaptive choosing of image parts to be coded, which, in turn, helps to adjust the usual packet size of the coded data. Slices can be defined, similar to those in MPEG 1, as a number of Macroblocks in scanning order, or, as a rectangular part of the picture to be coded, which is often more useful in packet loss environments. The obvious advantage of this rectangular slices, compared to the mandatory GOB model, is the possibility of using two dimensional motion vectors together with the Independently Segmented Decoding Mode, even for small slices[3], as described below.

2. 2. Independently Segmented Decoding Mode

This mode enforces both encoder and decoder to treat segment boundaries like picture boundaries, where a segment is defined as either a slice, or one or more consecutive GOBs. Consequently, no data whatsoever is referenced by the decoding process outside of the spatial area of the given segment. Especially, in conjunction with rectangular slices, this mode limits the negative influence of a packet loss of coded data to a small spatial area of the picture, given that the packetization scheme and the usage of slices are optimized with each other.

2. 3. Reference Picture Selection Mode

When using this mode, both encoder and decoder have to use more than one reference picture memory to store the information for inter picture prediction. Hence, P frames can be used for the transmission of the video signal, even in cases where a previously coded picture was not received correctly by the decoder. This mode was originally intended for use together with a back channel. Back channel messages can be used to acknowledge the complete decoding of a picture, or to acknowledge the non-complete decoding of a picture (e. g. due to missing data because of a packet loss) or both types of information. Since more than one resychronization frame can be present at any given time, it is likely that inter picture prediction can continue at the resychronization frame, if a later frame is damaged or is not transmitted.

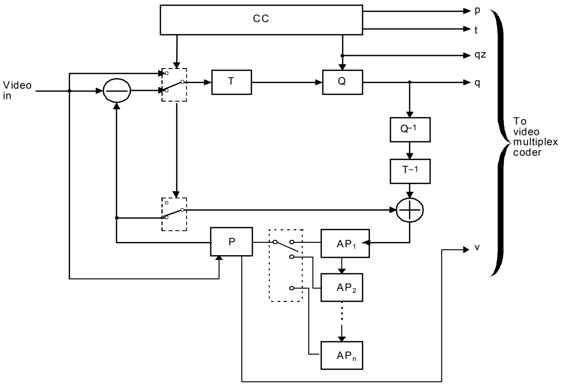
However, the use of a back channel should be avoided, especially in cases of multicast or broadcast transmission. As a result, a submode was defined which allows a pretty high tolerance of frame (i. e. packet-) losses without the use of a back channel. The rest of this paper will describe this method of using the Annex N syntax. The proposed method is especially useful in combination with other coding methods, which are error resilient on packet losses in some way but rely on a correctly transmitted base datastream (Annex O, layered codec, comes in mind, see section 4.2.).

3. VIDEO REDUNDANCY CODING

In this section, the Video Redundancy Coding algorithm is introduced. For this, it is necessary to present the functionality of the Reference Picture Selection mode in more detail.

3. 1. Annex N Coder and Decoder

Figure 1 shows the block diagram of the H.263+ encoder using Annex N:



T	Transform
Q	Quantizer
P	Picture Memory with motion compensated variable delay
AP	Additional Picture Memory
CC	Coding control
p	Flag for INTRA/INTER
t	Flag for transmitted or not
qz	Quantizer indication
q	Quantizing index for transform coefficients
v	Motion vector

Figure 1. H.263+ Encoder using Annex N

The main difference between the usual coding loop and the coding loop using Annex N is the presence of one or more additional reference picture memories (APn). The coder uses the APs in a first-in, first-out strategy to store reference pictures. If a decoder signals, by the means of a back channel, that it was unable to decode a picture, it is possible for the encoder to choose one of its additional pictures as the reference picture for temporal prediction for the next frames. Consequently, temporal prediction mechanisms can be used even in error prone

[2] A fourth mode (Layered Codec, Annex O) can also be used to improve the error resilience characteristics of the coding scheme (see section 4.2).
The old BCH forward error correction mechanism known from H.261 is still present in H.263+ and described in Annex H. Since this mode is not useful in packet environments because it only corrects bit errors, it is not further discussed in this paper.

[3] It should be noticed here that for the usual picture sizes up to CIF a GOB in H.263 consists of exactly one row of Macroblocks. This is different from H.261, where a GOB was formed out of three rows of Macroblocks.

environments, in which it is not sure that the decoder and the coding loop of the encoder always decode the same information and so run synchronously.

3. 2. Back channel usage problems

It is obvious that the Reference Picture Selection mode offers a great amount of error resilience when used together with a back channel. However, for several reasons, the use of a back channel is not always advisable.

- In multipoint or broadcast environments the semantic problem of having 'a' back channel is obvious; what should a decoder do, if some decoders signal the correct decoding of a frame, and others do not?
- On many packet networks, especially the Internet, we have the additional problem of the non realtime (and sometimes non reliable) transmission of back channel data. While these problems are not very difficult to manage if using a sufficiently high number of APs in point-to-point scenarios, they become more or less unsolvable in multipoint or broadcast scenarios, if one wants to achieve useful latency times. Different and unpredictable latency times of the various back channels will not allow the encoder to make the decision to use another reference frame, other than the default early enough to be able to achieve a sufficiently high amount of temporal predictions.

It is out of the scope of this paper to discuss these types of problems (closely tied to the research area of reliable multicast transmission techniques) in further detail. It should now be obvious, however, that a temporal error resilience method without utilizing back channel techniques is useful in many environments, including the Internet/MBONE environment.

3. 3. Using Annex N without a back channel

Video Redundancy Coding (VRC) is a mechanism to achieve temporal error resilience by using more than one reference frame and more than one prediction threads.

The idea of video redundancy coding is to send at least two threads of P frames simultaneously where each of these P-frames depend on the earlier P-frame of the thread but not on any information of the other thread(s). Newly coded pictures will be assigned to the various threads in an interleaved manner.

All threads are started from a so called Sync-Frame (which can, but not must be an I-frame) and end into another Sync-Frame. If one thread is damaged (i. e. because of a packet belonging to a frame in this thread got lost), the other threads remain intact and can be decoded and displayed. This will lead into lower frame rates for the time to the next sync-frame, but avoids the otherwise introduced 'Ghost images' (due to missing P-frames of a H.263 sequence) or the need of a complete resychronization by signaling and later sending an I-frame.

Usually an H.263+ data stream consists of a sequence of P-frames. This makes it necessary to have the completely decoded frame (t-1) ready for decoding frame (t). If parts of the frame (t-1) cannot be decoded, for example because of a packet loss, the complete frame sequence is corrupted and has to be reinitialized by sending an I-frame. By using Annex N of H.263+, it is possible to send more than one independent, but short frame threads, e.g. 5 frames (for more sophisticated systems, the length and the number of such frame threads could be adaptively adjusted according to the network characteristics). Figure 2 shows the dependencies between the various frames in an example of two threads with a length of three frames each. Thread one has the odd frame numbers, thread two the even ones.

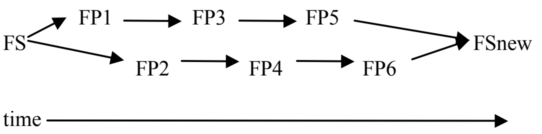


Figure 2: Threads in Video Redundancy Coding

Let us assume that FS is a correctly transmitted frame. The first frame coded after FS will be transmitted as the first P-frame of thread 1, which depends only on FS. The next frame is transmitted as the first P-frame of thread 2 (FP2). Later on, the 'odd' Frames of thread 1 depend only on their predecessor, which is similar for thread two. This scheme can be easily adopted to n threads of length m.

If a packet loss occurs it will occur either in thread 1 or in thread 2. In such a case, all further decoding of the thread in which the packet loss has occurred will be stopped by the decoder. Consequently, assuming two threads, the frame rate will drop to half, but there will be still a moving picture and, more important, at the end of the uncorrupted thread a new sync-frame (FSnew) which will be the starting point for two (or more) new threads. In multicast or broadcast environments, the decoding of all threads may continue in those terminals, which were not effected by the packet loss. In any case, the coder will continue to produce all threads, because there is no means like a back channel to inform it about the loss of one of the threads.

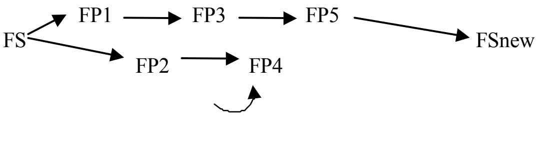


Figure 3: Packet Loss in VRC

3. 4. Four obvious problems

By taking a closer look to the principles of video redundancy coding, four obvious problems can be identified, which may have a negative impact on the coding quality:

(1): Most of the frames are predicted from a frame which is much 'older' than usual. This results in a higher amount of bits necessary to code this frame, because of the longer motion vectors and a higher degree of degradation between the two frames of one thread. The more threads are used, the higher the need will be for additional number of bits. On the other hand, more threads will lead to a higher error resilience. While the problem is obvious, it should be clear that this additional redundancy is part of the price one has to pay when using VRC and so achieving higher error resilience. It is up to the sending system to decide on the number of threads and their length – ideally based on information about the network quality – and thereby achieving a reasonable tradeoff between error resilience and additional bitrate.

(2): It would be desirable that all 'last' frames of all threads (e. g. FP5 and FP6 in the example of Figure 2) independently generate exactly the same sync frame FSnew. It seems to be impossible to completely achieve this goal. It was shown in simulations, however, that it is possible to come close enough to this goal, so that artifacts resulting from the problem are usually not visible, and that SNR measurement shows no significant differences even when not using the 'perfect' thread, from time to time.

(3): (This problem is closely coupled with (2).) When using VRC with n threads, the encoder has to send n frames with the same temporal reference (TR) which all describe the new sync frame FSnew. This leads to an additional amount of bits for a whole coded scene. Again, error resilience vs. bitrate tradeoffs have to be considered by the coder to achieve a sufficient low bitrate while gaining a sufficient high error resilience. In the same manner, if more than one thread 'survives', the decoder would see more than one frame with the same TR, which is forbidden by H.263+. Therefore, the transport stack has to discard packets containing information for the new sync frame except those out of one thread. Usually, the transport stack will use the thread containing the 'perfect' information to generate the new sync frame and discard the other threads, but if the 'perfect' thread is damaged, one other thread has to be used.

(4): What happens, if all threads are destroyed because of packet loss? In such a situation we have the same problem (and possible solutions) as if a packet loss destroys a one-threaded video stream.

4. VIDEO REDUNDANCY CODING IN H.263+

When using VRC in H.263+, it is possible to combine several of the optional modes for achieving a very high amount of error resilience at minimal additional bitrate cost for which various scenarios are possible. In the following, the combined usage of the above introduced optional modes N, K and R of H.263+ is described. This, in conjunction with an RTP payload format optimized for that particular usage, allows a relatively efficient video coding with a very high packet loss error resilience.

4. 1. Using VRC with Independently Decodable Slices

In our experience, the combined usage of the rectangular slices of the *Slice Structured mode*, which should be independently decodable by using the *Independently Segmented Decoding mode* and VRC on the resulting independent slices gives the best performance for medium to low quality Internet connections (with packet loss rates up to 20%).

To use this mode combination, the sending terminal has to choose a subdivision scheme for the frames, which should be the same during the whole connection. The size of the picture parts (called segments) for the independent coding should be chosen in such a way that the complete coded segments fit into one RTP packet whose size is not larger than the physical network size of the underlying network - e. g. in case of Ethernet around 1,500 bytes. This means that for most applications at bitrates below 100 kbit/s and framerates above 10 fps, no subdivision of the frame is necessary at all; however, if higher bitrates, lower framerates or smaller packet sizes are to be used by the specific application and network characteristics, subdivision of the picture is necessary.

We propose to use for this subdivision the rectangular submode of the Slice Structured mode only, because only in this mode it is possible to form rectangular picture segments at an aspect ratio similar to the original picture format (usually 4:3, or 16:9) for which the motion vector coding of H.263+ is optimized. We further propose, that these slices have to be independently decodable to avoid the otherwise possible reference of data outside of the slice. Both goals can be achieved by a combined usage of Annex K and Annex R syntax and semantics.

In cases where the subdivision of the frame that will be coded is necessary, the Video Redundancy Coding method, as described above, will be applied to either the whole frames or to all of the slices independently. This leads to a very high amount of error resilience at minimal additional bitrate and no additional latency time cost.

4. 2. VRC versus Layered Codecs (Annex O)

H.263+ includes an optional Annex O, which describes a layered codec approach. In addition to the base layer, which contains a full H.263+ data stream, three types of enhancement layers may be present as well, which allow different types of scalability:

- The Temporal Scalability enhanced layer introduces the concept of bidirectionally predicted pictures, or B-pictures, known from the MPEG standard family.
- The SNR Scalability enhanced layer permits the coding and sending of the error picture (which is the mismatch of the original picture and the reference picture of the coding loop, introduced by the lossy nature of the coding algorithm). By doing this, the coding error can be minimized.
- The Spatial Scalability enhanced layer allows to send the error picture resulting of a spatial enhancement by a factor of two (e. g. from QCIF to CIF), similar to SNR Scalability.

The reason for the introduction of this mode was mainly to provide better support for heterogeneous network structures. For example, it is possible to send the base layer at a bitrate of 20 kbit/s, QCIF format and 7.5 fps. A spatial scalability layer enhances this base layer to a CIF, 48 kbit/s and 7.5 fps coded picture. In addition to that, a temporal scalability layer can be used for achieving a target bitrate of 112 kbit/s at 15 fps in CIF format. This allows a multipoint connection between systems hooked up by PSTN lines and ISDN lines using one or two B-channels without any need for transcoding of the video data in an MCU or gateway. The sending terminal needs only one coder and only one 2-B-channel network interface at 128 kbit/s to the MCU (including 16 kbit/s audio).

The layered codec approach and Video Redundancy Coding are not only compatible, they complement each other in a nearly perfect way. It is possible, for example, to use VRC on the base layer only, and achieve a minimum QoS regarding the video quality. The higher layers may or may not be supported by VRC. The only enhancement layer, for which the usage of VRC makes no sense, is the temporal scalability layer, because by definition it is not allowed to use B-frames as predictor information for any other data.

5. AN RTP PAYLOAD FORMAT FOR VRC AND H.263+

Since the currently used RTP payload format for H.263 is no more usable for H.263+ because of the introduction of the new optional modes, a new RTP payload format for H.263+ has to be introduced. At the time this paper had to be finished, our proposal for an H.263+ payload format was under discussion in the relevant mailing lists. It will be presented at the München IETF (11.8.97 - 15.8.97). The current status of this payload format will be briefly outlined here, to show how the optional mode combinations for error resilience as described in section 4. 1. can be supported by today's most frequently used packet based network infrastructure, the Internet.

5. 1. Packet types

Three types of payload packets are defined in our H.263+ RTP payload proposal:

- The Frame-packet contains only information regarding VRC, the start bit position and the end bit position of the data stream, plus the data stream itself. It is to be used for the first packet of any coded frame, and consequently always contains the H.263+ Picture Header in the payload. In many situations, frame-packets will be the only packet type needed.
- The Segment-packet is used to transport single segments (usually one rectangular slice) of coded H.263+ data, the Picture Header of the picture the slice belongs to and VRC information. Segment packets are to be used, if the usual size of a coded frame is larger than the advisable size of RTP packets on the particular network, which makes it necessary to subdivide the picture to be coded into slices.
- The Follow-on-packet is a fallback solution for such cases, in which the size of the Frame- or Segment-packet, although carefully chosen, exceeds the maximum advisable packet size. This situation can happen, for example, in case of I-frames. A follow on packet can only be decoded in conjunction with the associated segment- or frame packet. The follow on packet header contains similar information like the frame-header, including the VRC information.

5. 2. VRC information in the payload headers

As stated above, all payload headers of the three packet types contain VRC information. This 8-bit field is necessary, in case of a packet loss, to identify the thread, in which the packet loss has occurred. This is achieved by transmitting a 3 bit number of a thread-id (so allowing up to 8 threads) and a 4 bit cyclic packet-per-thread counter. One bit is transmitted to identify the first packet in each thread, which is used to code the sync frame and allow threads with a variable number of coded pictures.

6. SIMULATION RESULTS

Extensive simulations were made to prove the effectiveness of Video Redundancy Coding. By the time this paper is published, a more detailed report which discusses more scenarios will be available from the author. However, some simulation results will also be given in this paper.

The simulation to be presented here was done using the TELENOR tmn2.0 H.263 reference codec, which was modified to support the mechanisms of annex N, but using a different syntax. A fixed bitrate of 110 kbit/s, a fixed frame rate of 15 fps and the CIF sized sequences ‘Deadline’ and ‘Paris’ (both are ‘talking-head’ sequences) were used. The Simulation was performed with two different configurations of Video Redundancy Coding:

- 2 threads with 5 frames each and
- 3 threads with 3 frames each.

It was assumed that only frame-packets have to be used for this configuration; so the consequence of a packet loss would be the loss of a whole frame. Both bursty on non-bursty packet loss characteristics at packet loss rates of 0%, 1%, 3%, 5% 10% and 20% were used.

The following table shows the subjective quality relative to a packet loss rate of 0%, assuming non-bursty packet loss characteristics:

Packet-loss	no VRC	2:5 VRC	3:3 VRC
0%	perfect	perfect	perfect
1%	good	perfect	perfect
3%	good	perfect	perfect
5%	fair	good	perfect
10%	poor	fair	good
20%	unusable	poor	fair

Table 1: Subjective quality vs. packet loss

Bursty packet loss characteristic would lead to even better performing of the VRC 3:3 method compared to 2:5 VRC.

The additional bitrate generated by VRC is shown in the next table:

Method	rel. Number of bits (generated by the coder)
no VRC	100%
2:5 VRC	118%
3:3 VRC	142%

Table 2: relative number of bits for VRC

If one takes into account that out of the 42% additional bitrate used in VRC 3:3, 20% can get lost by the network without much effecting the picture quality, the additional bitrate does not seem to high. Again, this high amount of error resilience is introduced at no additional latency time cost.

7. SUMMARY

In this paper we introduced a new method for achieving error resilience in packet lossy environments. By referencing not only the previous, but also earlier pictures of a P-frame series, we can achieve high tolerance levels at minimum cost of subjective quality and additional bitrate and no cost of additional latency time at all. The support for Video Redundancy Coding was incorporated to the ITU-T Recommendation H.263+; additional necessary support will be proposed as an RTP payload specification to the IETF.

REFERENCES

[1]: ITU-T Draft H.263, Video Coding for low Bitrate Communication, Version ‘draft 12’, based on the determined version May, 1997

[2]: S. Casner: “Frequently Asked Questions (FAQ) on the Multicast Backbone (MBONE)“, available at ftp://ftp.isi.edu/mbone/faq.txt

[3]: M. Yajnik, J. Kurose, D. Towsley: ”Packet Loss Correlation in the MBONE Multicast Network“, available from {kurose, towsley@cs.umass.edu}

[4]: J. Bolot, T. Turletti: ”Adaptive Error Control for Packet Video in the Internet“, available from the authors turletti@lcs.mit.edu or bolot@sophia.inria.fr

[5]: R. Frederick: ”Experiences with real-time software video compression“, Proc. 6th Packet Video Workshop, Portland, OR, USA, Sept. 1994

[6]: S. McCanne, V. Jacobson, ”Vic: A flexible framework for packet video“, Proc. ACM Multimedia’95, Washington DC, USA, Oct. 1995

[7] S. McCanne, M. Vetterli, ”Receiver-driven layered multicast“, Proc. ACM Sigcomm ’96, Stanford, CA, USA, Sept. 1996

[8] J. Bolot, T. Turletti: ”Experience with control mechanisms for packet video in the Internet“, INRIA report, March 1996

[9] J. Bolot, A. Vega-Garcia ”The Case for FEC-Based Error Control for Packet Audio in the Internet“, to appear in ACM Multimedia Systems