Telecommunication Standardization Sector
Study Group 15
Experts Group for ATM Video Coding
(Rapporteur's Group on Part of Q.2/15)

Document AVC-481
March 1993

INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO-IEC/JTC1/SC29/WG11
CODED REPRESENTATION OF PICTURE AND AUDIO INFORMATION

ISO-IEC/JTC1/SC29/WG11/MPEG92/372

Source: M. E. Nilsson, I. Parke, BT Labs
Title: The use of MPEG-2/H.26x for low bit rate applications
Purpose: Discussion and proposal

This document is concerned with the use of MPEG-2/H.26x for low bit rate applications. It is argued that MPEG-2/H.26x should support these applications as it is a generic standard. The limitations of the syntax in this respect are described and some changes are proposed. In particular, it is proposed that true picture skipping is allowed. The proposals retain syntax compatibility with MPEG-1.

1. The scope of MPEG-2/H.26x

The MPEG-2/H.26x standard will be a generic standard. This means it will not be optimised for any particular application and may be used for any number of applications, coding various image formats at a variety of bit rates.

Although the majority of the MPEG-2/H.26x algorithm development work has been concerned with the coding of CCIR 601 resolution images at bit rates of about 4 to 9 Mbit/s, it should be remembered that this does not represent the only type of application for the standard. Another major application of H.26x is video coding for the B-ISDN. This requires error resilience and compatibility with H.261. For conversational services, low bit rate coding is necessary, in particular for calls made between the B-ISDN and the N-ISDN.

2. Syntax implications for low bit rate applications

The effectiveness of syntax elements depends on the relative bit rate used. It is important that syntax elements that are vital to the performance at low bit rates are not 'optimised' out of the syntax for marginal gains in performance at the higher bit rates. Examples of such 'optimisations' include past proposals to remove coded block pattern and macroblock addressing from the syntax.

At present, the Working Draft states that there can be no gaps between slices and that the first and last macroblocks of each slice can not be skipped. This demand on encoders has little effect at high bit rates but unnecessarily limits the performance at low bit rates. When coding at low bit rates, quite often only a few macroblocks of a picture are coded. This restriction therefore enforces the wasteful coding of some macroblocks, at least the first and last of the picture, and many macroblock address bits. The flexible concept of slices is in principle ideal for low bit rate applications, as the cost of large macroblock address can be reduced by simply coding slice headers to absolutely address the few macroblocks that are to be coded. However, this is currently not allowed.

This is considered to be an unnecessary restriction on encoders. It should be possible to skip any macroblock, whether it is the first, last or one following an intra macroblock. It is only necessary to ensure that the decoder knows how to handle skipped macroblocks. This can be achieved by ensuring that all macroblock coding parameters have default values, in much the same way as motion vector and intra DC coefficient predictions are reset to defaults at given points. The acceptance of this proposal would improve coding performance at all bit rates, as it reduces the

number of bits that are needlessly encoded, although the saving would be larger at the lower bit rates.

3. Limiting the generation of coded bits

There are primarily two mechanisms that can be used to limit the number of bits produced when coding at low bit rates: the use of coarse quantization and skipping pictures Optimum picture coding at low bit rates will utilise a combination of both techniques. Coarse quantization and a reduced picture rate will achieve the optimum picture quality for a given bit rate.

A reduced picture rate can be achieved in two ways. Firstly, a constant but reduced rate of source pictures can be selected, for example 15Hz or 10Hz. The second is to maintain a high nominal picture rate of 25Hz or 30Hz and to encode as many of these as the available bandwidth allows. The second method has a distinct advantage over the former in that the rate of coded pictures is adapted to the complexity of the scene being coded: when the source is easy to code, near to full picture rate can be achieved, with the picture rate only dropping when there is significant motion or a scene change.

4. Skipping pictures in MPEG-1

MPEG-1 does not support the skipping of pictures. The minimum frame rate is 24Hz. If an encoder had to reduce the frame rate to reduce the number of bits to be transmitted, it would have to send a picture with limited data: a picture header, a slice header and the first and last macroblocks of the picture, with a rather large macroblock address between them. This is particularly inconvenient.

5. Skipping pictures in MPEG-2/H.26x

Recently in the development of MPEG-2/H.26x the concept of a skipped picture has been suggested. This is a new picture type with no slice data. However, an encoder must still encode the picture header. This still represents a significant cost in terms of wasted bits for low bit rate applications.

The definition of skipped pictures in Test Model 4 is unclear. It states, 'When S-picture, there should be no data for below the slice layer, and the decoder does not take time to display them.'.

The first clause presumably implies that the skipped picture consists of a picture header only and that no slice data is present. The second clause is very unclear, if not wrong. It gives the impression that no display time is associated with the skipped picture. This would imply that the picture following the skipped picture would be displayed immediately after the previous non-skipped picture. This is clearly not possible as it would result in the decoder consuming its bits too quickly. When a skipped picture is detected something must be displayed, as a picture was captured by the camera during this time interval in the encoder. In practice, the previous picture would be repeated.

6. When are skipped pictures displayed?

When an encoder is skipping pictures, it is coding the other pictures with more bits than the average number transmitted/stored in a picture period.

Usually, one picture will be coded with more bits than the average number of bits per picture, the excess will be stored in the encoder buffer and subsequent pictures are skipped until the buffer occupancy returns to normal. The decoder can take either of two approaches to decoding the resulting bitstream. Firstly, it can wait for all of the coded data to arrive, decode it, display the picture, and then repeat it as required to replicate the pictures skipped in the encoder. Alternatively, while it is waiting for the data of the coded picture to arrive, it could repeatedly display the previous decoded picture until it can display the coded picture. After this, it would continue decoding as normal. The advantage of the second approach over the first is that the total

delay through the encoder and decoder buffers is minimised: all coded pictures have low delay except those coded with a large number of bits.

As stated before, MPEG-1 does not allow true picture skipping although it does allow some pictures to be coded with a small number of bits. However, as these pictures do contain real data, they must be decoded and displayed at the right time. This forces the decoder to use the first option outlined above. Low delay operation with this type of skipped picture is therefore not possible.

The position with respect to MPEG-2/H.26x skipped pictures is not clear. As these pictures contain no real data, they do not need to be decoded at the correct time, and can be totally discarded. Is this to be allowed by MPEG? Should it be stated in the Working Draft? Also, if this is to be allowed, what possible meaning could the vbv delay field in a skipped picture have?

7. Proposal to allow real skipping of pictures

It is proposed that the standard should allow the skipping of pictures by including no data at all in the coded bitstream for the skipped pictures, as well as, or instead of, allowing skipped picture headers. H.261 provides this mechanism for skipping pictures.

There is no information in the skipped picture header except that it implicitly informs the decoder that a picture was present in the encoder. It is shown in the appendix that it is possible to deduce which pictures have been skipped by simply observing the temporal references of the pictures in the bitstream. It is also shown that there is very little difference for a decoder to detect and handle picture skipping done in this way and handling a continuously changing value of the M parameter.

8. The meaning of picture rate

The picture rate is currently specified at the sequence layer. The meaning of this field is clear when no pictures are skipped. However, if pictures are totally omitted from the coded bitstream as proposed in the section above, the meaning of the field requires clarification. For example, an encoder may state the picture rate to be 30Hz and then skip every second picture, effectively coding at a rate of 15Hz. It is proposed that text is added to ensure that this field indicates the rate of the source intermediate pictures, rather than the rate at which picture headers occur on average in the coded bitstream. Source intermediate pictures are those input to the encoding process, which are not necessarily those captured by the camera.

Some decoders built for low bit rate applications may have a maximum processing rate that does not allow the full 25Hz or 30Hz to be decoded. Considering the general philosophy that decoders should be able to determine whether they are capable of decoding a bitstream by reading header information rather than by trial and error, it is suggested that an additional field is added to the sequence header. This would indicate the minimum picture interval in the coded bitstream. For example, if this parameter has the value of 1, then no two consecutive source intermediate pictures will be coded. At least the values of 0,1,2 and 3 should be allowed.

9. Conclusion

This document was concerned with the use of MPEG-2/H.26x for low bit rate applications. It was argued that MPEG-2/H.26x should support these applications as it is a generic standard. The limitations of the syntax in this respect were described and some changes were proposed.

It was proposed that syntax elements vital to performance at low bit rates are not 'optimised' out by considering high bit rates only. It was proposed that it should be possible to skip any macroblock in any picture. Finally it was proposed that true picture skipping is allowed, and that an indication of the maximum rate at which pictures are coded is included at the sequence layer.

All the proposals retain syntax compatibility with MPEG-1.

10. Appendix, Identification of skipped pictures

It is shown in the appendix that it is possible to deduce which pictures have been skipped by simply observing the temporal references of the pictures in the bitstream. It is also shown that there is very little difference for a decoder to detect and handle picture skipping done in this way and handling a continuously changing value of the M parameter.

10.1. Skipping an interpolated picture

This section considers how the skipping of a picture that would normally be coded as an interpolated picture would be detected at the decoder.

The diagram below shows and numbers original and coded pictures with M=3. Suppose that the encoder has decided to skip picture 7. It will not write anything into its coded bit buffer during the period when it would normally have coded this picture. Data for picture 11 will therefore come immediately after data for picture 6 in the coded bitstream.

When the decoder finds the temporal reference of picture 11, it will know that data for picture 7 will never arrive: it has missed its opportunity. The decoder therefore knows that a picture has been skipped. It will know this as quickly and as easily as if it had received a skipped picture header for picture 7.

Original	B	B	I	B	B	P	B	-	P	B	B	P	
Pictures	0	1	2	3	4	5	6	7	8	9	10	11	
Coded			I	B	B	P	B	B	P	B	-	P	B	B
Pictures			2	0	1	5	3	4	8	6	7	11	9	10
Decoded				B	B	I	B	B	P	B	-	P	B	B	P
Pictures				0	1	2	3	4	5	6	7	8	9	10	11

10.2. Skipping an intra or predicted picture

This section considers how the skipping of a picture that would normally be coded as an intra or predicted picture would be detected at the decoder.

The diagram below shows and numbers original and coded pictures with M=3. Suppose that the encoder has decided to skip picture 8. It will not write anything into its coded bit buffer during the period when it would normally have coded this picture. Data for picture 6 will therefore come immediately after data for picture 4 in the coded bitstream. The encoder can only use forward prediction to code pictures 6 and 7 and backward prediction to code pictures 9 and 10.

When the decoder finds the temporal reference of picture 6, it will know that an intra or predicted picture has been skipped, or that some change in the GOP parameters has taken place. This is because picture 6 is an interpolated picture which has no future prediction picture. At this time, the decoder does not know which picture has been skipped, but it does not require this information at this time. It should just decode pictures 6 and 7 as normal. It will then receive data for picture 11, from which it can deduce that it must be picture 8 that was missing, as it has now missed its opportunity to be coded.

Original	B	B	I	B	B	P	B	B	-	B	B	P	
Pictures	0	1	2	3	4	5	6	7	8	9	10	11	
Coded			I	B	B	P	B	B	-	B	B	P	B	B
Pictures			2	0	1	5	3	4	8	6	7	11	9	10
Decoded				B	B	I	B	B	P	B	B	-	B	B	P
Pictures				0	1	2	3	4	5	6	7	8	9	10	11