H.264 / MPEG-4 Part 10 White Paper

Overview of H.264

1. Introduction

Broadcast television and home entertainment have been revolutionised by the advent of digital TV and DVD-video. These applications and many more were made possible by the standardisation of video compression technology. The next standard in the MPEG series, MPEG4, is enabling a new generation of internet-based video applications whilst the ITU-T H.263 standard for video compression is now widely used in videoconferencing systems.

MPEG4 (Visual) and H.263 are standards that are based on video compression ("video coding") technology from circa. 1995. The groups responsible for these standards, the Motion Picture Experts Group and the Video Coding Experts Group (MPEG and VCEG) are in the final stages of developing a new standard that promises to significantly outperform MPEG4 and H.263, providing better compression of video images together with a range of features supporting high-quality, low-bitrate streaming video. The history of the new standard, "Advanced Video Coding" (AVC), goes back at least 7 years.

After finalising the original H.263 standard for videotelephony in 1995, the ITU-T Video Coding Experts Group (VCEG) started work on two further development areas: a "short-term" effort to add extra features to H.263 (resulting in Version 2 of the standard) and a "long-term" effort to develop a new standard for low bitrate visual communications. The long-term effort led to the draft "H.26L" standard, offering significantly better video compression efficiency than previous ITU-T standards. In 2001, the ISO Motion Picture Experts Group (MPEG) recognised the potential benefits of H.26L and the Joint Video Team (JVT) was formed, including experts from MPEG and VCEG. JVT's main task is to develop the draft H.26L "model" into a full International Standard. In fact, the outcome will be two identical) standards: ISO MPEG4 Part 10 of MPEG4 and ITU-T H.264. The "official" title of the new standard is Advanced Video Coding (AVC); however, it is widely known by its old working title, H.26L and by its ITU document number, H.264 [1].

2. H.264 CODEC

In common with earlier standards (such as MPEG1, MPEG2 and MPEG4), the H.264 draft standard does not explicitly define a CODEC (enCOder / DECoder pair). Rather, the standard defines the syntax of an encoded video bitstream together with the method of decoding this bitstream. In practice, however, a compliant encoder and decoder are likely to include the functional elements shown in Figure 2-1 and Figure 2-2. Whilst the functions shown in these Figures are likely to be necessary for compliance, there is scope for considerable variation in the structure of the CODEC. The basic functional elements (prediction, transform, quantization, entropy encoding) are little different from previous standards (MPEG1, MPEG2, MPEG4, H.261, H.263); the important changes in H.264 occur in the details of each functional element.

The Encoder (Figure 2-1) includes two dataflow paths, a "forward" path (left to right, shown in blue) and a "reconstruction" path (right to left, shown in magenta). The dataflow path in the Decoder (Figure 2-2) is shown from right to left to illustrate the similarities between Encoder and Decoder.

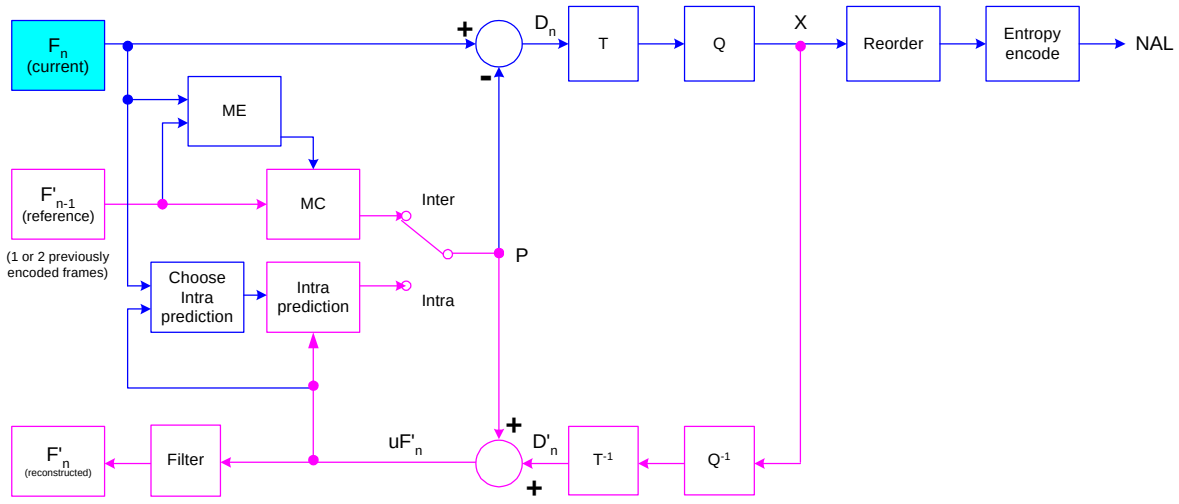

Figure 2-1 AVC Encoder

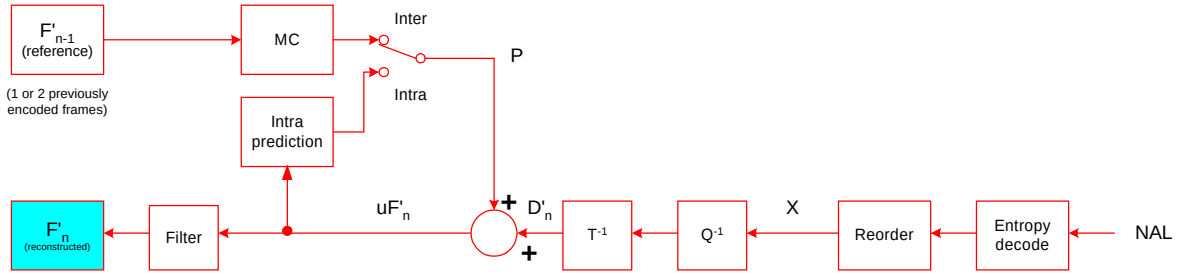

Figure 2-2 AVC Decoder

2.1 Encoder (forward path)

An input frame F_n is presented for encoding. The frame is processed in units of a macroblock (corresponding to 16x16 pixels in the original image). Each macroblock is encoded in **intra** or **inter** mode. In either case, a prediction macroblock P is formed based on a reconstructed frame. In Intra mode, P is formed from samples in the current frame n that have previously encoded, decoded and reconstructed (uF'_n in the Figures; note that the **unfiltered** samples are used to form P). In Inter mode, P is formed by motion-compensated prediction from one or more reference frame(s). In the Figures, the reference frame is shown as the previous encoded frame F'_{n-1} ; however, the predicton for each macroblock may be formed from one or two past or future frames (in time order) that have already been encoded and reconstructed.

The prediction P is subtracted from the current macroblock to produce a residual or difference macroblock D_n. This is transformed (using a block transform) and quantized to give X, a set of quantized transform coefficients. These coefficients are re-ordered and entropy encoded. The entropy-encoded coefficients, together with side information required to decode the macroblock (such as the macroblock prediction mode, quantizer step size, motion vector information describing how the macroblock was motion-compensated, etc) form the compressed bitstream. This is passed to a Network Abstraction Layer (NAL) for transmission or storage.

2.2 Encoder (reconstruction path)

The quantized macroblock coefficients X are decoded in order to reconstruct a frame for encoding of further macroblocks. The coefficients X are re-scaled (Q^{-1}) and inverse transformed (T^{-1}) to produce a difference macroblock D_n'. This is not identical to the original difference macroblock D_n ; the quantization process introduces losses and so D_n' is a distorted version of D_n.

The prediction macroblock P is added to D_n' to create a reconstructed macroblock uF'_n (a distorted version of the original macroblock). A filter is applied to reduce the effects of blocking distortion and reconstructed reference frame is created from a series of macroblocks F'_n.

2.3 Decoder

The decoder receives a compressed bitstream from the NAL. The data elements are entropy decoded and reordered to produce a set of quantized coefficients X. These are rescaled and inverse transformed to give D_n' (this identical to the D_n' shown in the Encoder). Using the header information decoded from the bitstream, the decoder creates a prediction macroblock P, identical to the original prediction P formed in the encoder. P is added to D_n' to produce uF'_n which this is filtered to create the decoded macroblock F'_n.

It should be clear from the Figures and from the discussion above that the purpose of the reconstruction path in the encoder is to ensure that both encoder and decoder use identical reference frames to create the prediction P. If this is not the case, then the predictions P in encoder and decoder will not be identical, leading to an increasing error or "drift" between the encoder and decoder.

3. References

1 ITU-T Rec. H.264 / ISO/IEC 11496-10, "Advanced Video Coding", Final Committee Draft, Document JVT-E022, September 2002