# Layered H.264 video transmission with hierarchical QAM

M. M. Ghandi, M. Ghanbari

*Electronic Systems Engineering Department, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK*

---

## Abstract

In multimedia communication systems, channel bandwidth and probability of error are the two main limitations that affect the quality of service. Therefore, in applications such as video over mobile networks, a video codec should cope with the erroneous situations of the channel as well as the bandwidth limitation. There are several techniques to make a video bitstream robust to the channel error. Layered video coding in conjunction with unequal error protection is a common method that provides error resilience. Hierarchical quadratic amplitude modulation (HQAM) is an efficient method that provides the unequal priority control for communication channels without adding any redundancy to the transmitted data. To generate the layers in the H.264/AVC video coding standard, the Data Partitioning technique has been included. Scalability is another efficient layering technique that is not entirely supported in the current specification of H.264/AVC. This paper proposes an SNR scalable scheme to adapt to the H.264/AVC codec. It contains new features that make it more efficient than the previous SNR scalable codecs. In this paper applying unequal error protection by hierarchical coding to both scalable and data partitioned bitstreams is analyzed. Simulation results show that the scalable scheme is more successful in conjunction with unequal error protection.

*Key words:* Scalable video coding, Error resilience, Unequal priority control

---

## 1 Introduction

In recent wireless multimedia communication systems, bandwidth is still a limiting factor. Hence, video compression is a crucial part of the applications

---

*Email addresses:* `mahdi@essex.ac.uk` (M. M. Ghandi), `ghan@essex.ac.uk` (M. Ghanbari).

such as video over mobile networks. The H.264/AVC video coding standard [1], proposed by the joint video team (JVT) of ITU-T and ISO/IEC, achieves a significant improvement in the compression efficiency over the other existing standards [2–4]. This makes H.264 a serious contender for all future multimedia applications.

On the other hand, due to the error-prone nature of communication channels there is a need for useful operation of robust video coding algorithms. In H.264 a number of features have been added to increase its robustness to channel errors, i.e. slice structure, intra update, data partitioning (DP) and so on [5]. DP is an effective layering technique that partitions the compressed data into separate units of different importance. Therefore, it can be considered as an efficient priority classification tool for unequal error protection (UEP).

Scalability is also considered as another powerful layering technique suitable for UEP [6,7]. It is an important feature of the recent video codecs [8–10] and has many other applications in video streaming over wired and wireless communication channels, such as in video on demand, wireless video for multi-party conversational services and so on [11]. Scalability is to partition a video bitstream into layers so that the base layer is an independent bitstream. The base layer is decodable into a video sequence with reduced quality (SNR) or spatial/temporal resolution. The enhancement layers provide additional data necessary for video reproduction with higher quality, spatial or temporal resolution. The current specification of the H.264 standard [1] supports only temporal scalability. However, other scalable approaches (SNR and Spatial) have been considered as important tools that should be supported by the standard [12].

There are numerous proposals for scalability to be added into the H.264 standard. The proposed scalability schemes in [13] and [14], use only one Motion Estimation (ME) stage and so they are not flexible to efficiently use layers of data in coding of the enhancement layer. In [15] a spatial scalability scheme is proposed that has a separate ME for the enhancement layer, but the method does not efficiently exploit the available motion data of the base layer. In [16] the efficiency of the proposed wavelet based SNR scalability method strongly depends on the contents of the sequence.

In this paper, a SNR scalable scheme based on the H.264/AVC standard is proposed. In the base layer, the coefficients are quantized with a coarse quantization step (determined by QPB), and in the enhancement layer a finer quantization (determined by QPE) is used. The encoder employs an independent motion estimation for the enhancement layer with a new concept of direct prediction. If the base layer is well protected against the channel error, a minimum picture quality (quality of the base layer) without drift would be guarantied. Unequal error protection is realized via hierarchical quadratic

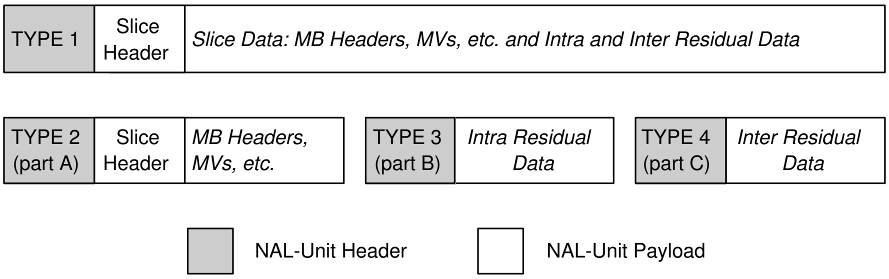


Fig. 1. A non-IDR slice placed in NAL units. Data partitioning is disabled (top) and enabled (bottom).

amplitude modulation (HQAM) which is a simple unequal priority control technique without introducing any additional bits (overhead). This is well suited for wireless multimedia systems especially when it is combined with channel coding techniques [17,18].

The remainder of the paper is organized as follows. In Section 2 a brief review of H.264 network abstraction layer and its data partitioning is presented. Section 3 describes details of the proposed SNR scalable codec. The hierarchical QAM for applying unequal priority control is briefly discussed in section 4. A simple error concealment is described in section 5 and simulation results are demonstrated in section 6. Finally, section 7 concludes the paper.

## 2 H.264 network abstraction layer and its layered coding

In the H.264/AVC standard, adaptation of the video coding layer (VCL) output to a variety of communication channels is done through the network abstraction layer (NAL). The NAL Layer in fact facilitates the delivery of the H.264 VCL data to the underlying transport layers such as RTP/IP, H.32X and MPEG-2 transport systems. Each NAL unit can be considered as a packet that contains an integer number of bytes including a header and a payload (see Fig. 1). The header specifies the NAL unit type and the payload contains the related data.

Each video frame can be divided into several slices; each containing a flexible number of macroblocks (MBs). In each slice the arithmetic coder is aligned and the predictions are reset. Therefore, they can be considered as resynchronization markers that prevent error propagation into the next slice. In H.264, each slice is placed in one or more separate NAL units that are independently decodable. The slices of an IDR frame (i.e. a frame with all intra slices which resets inter predictions) are located in a type 5 NAL unit, while those belonging to a non-IDR picture are placed in the NAL units of type 1 to 4 depending on the data partitioning (DP) mode.

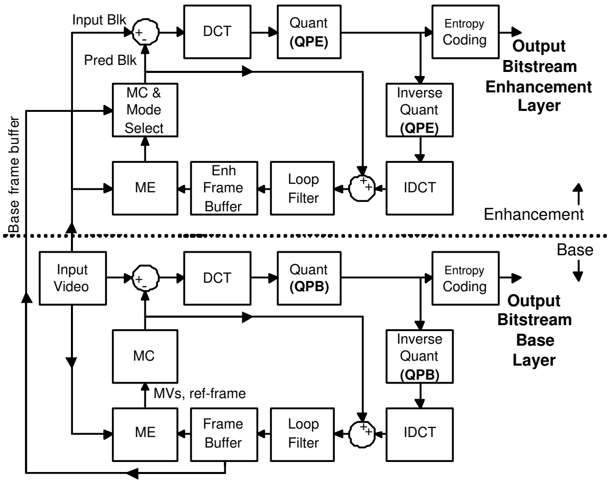


Fig. 2. The block diagram of the proposed scalable coder.

DP is an important and efficient way of making a video bitstream more robust. It is conceived based on the fact that by bringing the more important parts of the video data (such as headers, MVs and addressing data) ahead of the non-important data, the side effect of channel error can be significantly reduced [11]. In H.264, when DP is enabled, every slice is divided into three separate partitions (Fig. 1) and each partition is accommodated in a particular NAL unit. Therefore, DP can be used as an efficient method of separating the data with different importance and facilitates the layered protection of contents (with hierarchical QAM or by other means). In the H.264 standard, DP is currently the only tool that provides layering in the same temporal resolution and we intend to extend this layering capability to SNR scalability.

## 3 Proposed H.264/AVC SNR scalable codec

The block diagram of the proposed SNR scalable encoder is illustrated in Fig. 2. The base layer is exactly the same as the standard non-scalable encoder (bottom part of the figure) that is: the input blocks are first predicted and the difference is transformed, quantized and entropy coded. The coded pictures are then reconstructed and stored in the picture buffer for the prediction of the future pictures. For coding the enhancement layer (upper part of the Fig. 2), prediction is made among the decoded pictures of the base and enhancement layers. In the following, selecting the best prediction mode for coding an enhancement block is explained.

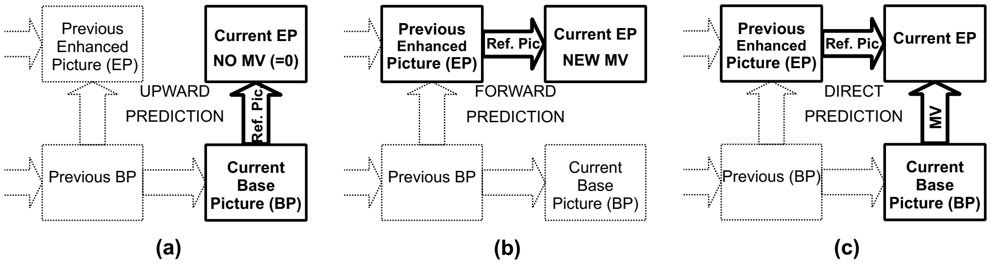


Fig. 3. Enhancement prediction modes; (a): Upward, (b): Forward and (c): Direct mode.

### *3.1 Enhancement layer prediction modes*

Every inter coded block in the enhancement layer is firstly predicted by one of the upward, direct or forward prediction modes shown in Fig. 3. In the upward mode no additional MV is sent for the block and the reference picture is the base layer reconstructed picture (Fig. 3-a). This mode is especially useful when the base layer MB is finely coded and there is a small residual data to be coded in the enhancement layer. Furthermore, this mode prevents error propagation and picture drift in case data of the enhancement layer are corrupted. Note that due to unequal error protection, safe reception of the base layer data is more guarantied than that of the enhancement layer. Additionally, in case an MB at the base layer should be intra coded (in order to prevent temporal error propagation), the enhancement layer would be forced to select only upward mode.

In the forward mode an enhancement block is coded with reference to previous enhanced pictures with a new set of motion vectors (Fig. 3-b). This mode is often selected when the quality difference between the base and the enhancement layer is large (base layer is coarsely coded). However, due to correlations between the base and the enhancement layers, coding of enhancement MVs is modified (described in section 3.2).

In the direct mode similar to the upward mode no additional MV is sent, but the motion compensated prediction of the previous enhanced picture is made using the corresponding base layer MVs. Since the projected moving objects in the base and the enhancement layers are at the same direction, there is a high correlation between the base and the enhancement MVs. Any discrepancy between them is mainly due to false motion estimation resulting from quantization distortions. Our simulations show that 20 to 50 per cent of the enhancement layer MVs (depending on the picture content and the quality difference between the base and the enhancement layers), closely resemble the base layer MVs. By avoiding duplicate transmission of MVs, the compression efficiency of the enhancement layer is improved. Another advantage of selecting this mode is that the base layer MVs are typically better protected than the enhancement layer ones and so the direct mode has a better immunity to channel errors than the forward mode.

As well as the above inter modes, an enhancement MB can be coded in an intra mode which is selected very rarely since it is not compression efficient. Furthermore, a MB can be skipped and in this case no information is sent for that MB. In our encoder, a Lagrangian optimizer selects the proper mode for every block. Using the selected modes, motion compensation, block transformation and quantization are carried out. The resulting MB headers, MVs and the residual data are coded. For layered transmission we have slightly modified the coding methods and the bitstream structure which are described in the following sections.

### *3.2 Coding of Motion Vectors*

In the the standard H.264 codec (as well as the base layer of our codec), motion vectors are predictively coded from the MVs of the neighboring blocks (by calculating a prediction MV, PMV). The difference between the original MV and PMV (i.e. MVD) is calculated and entropy coded. In the enhancement layer of our layered codec, only a block in the forward mode has new MVs to send. To calculate its PMV, however, the block may be surrounded by neighbors that do not have enhancement MVs (i.e. are in modes other than forward). In these conditions, we use the corresponding base layer MVs of the neighbors for motion vector prediction. This will produce a better MVP and eventually improves coding efficiency of the enhancement MVs.

In the CABAC (Context based Adaptive Binary Arithmetic Coding) mode of H.264, the signs and the absolute values of the MVD are coded separately. For coding the sign of an MVD in the base layer, since it is almost equally likely to be negative or positive, equal probability model is used. However, the motion vectors of the enhancement layer ($MV_e$) are normally at the same direction of the corresponding base layer MVs ($MV_b$), and hence the sign of their MVD would have high similarities. Hence, from the estimated sign of the base $MVD_b$ four different context models for the sign of the enhancement $MVD_e$ can be derived (two for horizontal and two for vertical MVD). The context index (from 0 to 3) of the $MVD_e$ sign is defined as:

$$MvdSignCtxIdx = 2 \times hv + sign(MV_b(hv) - PMV(hv)) \tag{1}$$

where $hv$ is 0 for horizontal and 1 for vertical and $sign(x) = (x < 0)?1 : 0$. Each index corresponds to a different probability table and the tables adaptively evolve during the encoding procedure. We should mention that in this work we only consider the CABAC mode of H.264 (which is more efficient), and adaptation to the other coding mode (CAVLC) is the subject of the further works.

### *3.3 The enhancement layer bitstream structure*

To incorporate the proposed method in the H.264 structure, the P-Picture syntax for the enhancement layer has been modified. To address the prediction modes, we define a reference picture identifier *ref-idx* (for each $16 \times 16$, $16 \times 8$, $8 \times 16$ or $8 \times 8$ block), which is set to zero for forward mode, 1 for upward and 2 for the direct mode. In the case of multireference modes, higher reference indices point to other reference pictures.

When an $8 \times 8$ block is in the forward mode, a syntax called *sub-mb-type* is sent to determine the subpartitioning mode (the subdivision mode to $8 \times 8$, $8 \times 4$, $4 \times 8$ or $4 \times 4$ subblocks), and so to specify the number of different MVs for that block. However, in the upward mode, since all MVs are zero, there is no need to further subdivide the $8 \times 8$ block and hence to send *sub-mb-type*. In the direct mode, the *sub-mb-type* is not sent either and the number of different MVs and the subpartitioning mode is extracted from its corresponding base layer block. Note that even when a $16 \times 16$, $16 \times 8$ or $8 \times 16$ partition is in the direct mode, it can have more than one pair of MVs (and also more than one reference picture) depending on the base layer MB division modes (and its selected reference pictures). In the modified structure, since *ref-idx* determines the prediction mode and in some modes there is no need to send *sub-mb-type* and MVs, *ref-idx* is sent ahead of *sub-mb-type* to prevent sending unnecessary data (whereas in the standard H.264 codec, *ref-idx* comes after *sub-mb-type*). One may modify the bitstream syntax to another structure. For example add more possible modes in the MB-type header and *sub-mb-type* instead of changing the semantic of *ref-idx* and it could be more sensible. However, this may have no significant impact on the coding efficiency which is the aim of our simulation.

Another amendment to the scalable bitstream is that every enhancement layer slice is placed immediately after its corresponding base layer slice in a separate NAL unit as shown in Fig. 4. The enhancement layer NAL units have types of 24 to 28 depending on the data-partitioning mode. Note that these values of NAL unit types are unused in the current standard. By this method of NAL unit distribution, duplicate sending of slice headers is prevented and a number of bits are saved, especially when there are several slices in a picture. Furthermore, to apply unequal priority control, the underlying layer needs only to read the NAL unit type. The NAL units of the base layer move to the high priority buffer and the enhancement NAL units are placed in the low priority buffer. The low and high priority buffers are then transmitted

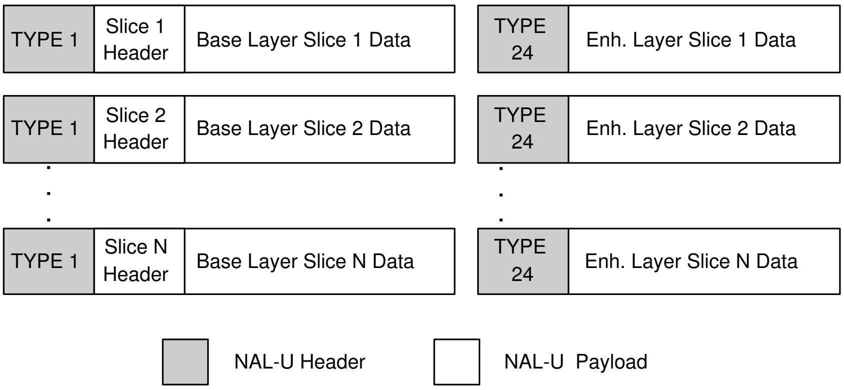


Fig. 4. The scalable bitstream NAL unit distribution, when DP is disabled for both layers.

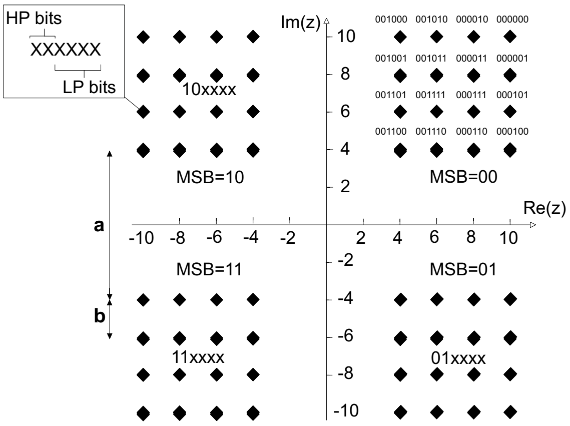


Fig. 5. A non-uniform 64-QAM mapping with $\alpha = a/b = 4$.

with the proper protections. The unequal error protection approach through a modulation system that is used in this paper is discussed in the next section.

## 4 Hierarchical QAM

M-tuple quadratic amplitude modulation (M-QAM) is an efficient modulation mode that achieves additional compression by assigning more than one bit to each transmission symbol. In this paper for conciseness, we only examine 64-QAM. In the 64-QAM constellation (Fig. 5), among the six bits of each symbol, as a result of the Gray bit distribution of the points, the two most significant bits (MSB) have the same value in each quadrant. Therefore, these two bits are more distinguishable and so have lower bit error rates than the four least significant bits (LSB)[19].

Let us define the distance between quadrants by $a$ and the distance between the points within a quadrant by $b$, as shown in Fig. 5. By mapping the high

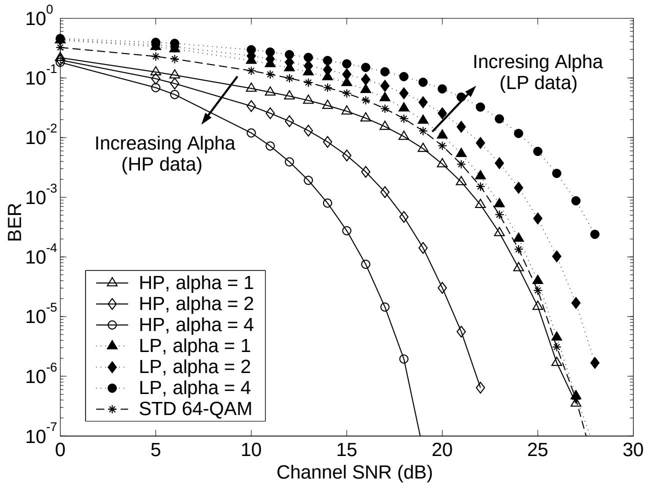


Fig. 6. Bit Error Rate (BER) vs channel SNR (for a Gaussian channel) for High Priority (HP) and Low Priority (LP) data using hierarchical 64-QAM with $\alpha$=1,2 an 4 and without hierarchical mode (STD).

priority (HP) and the low priority (LP) data to the MSB and LSB bits as shown in the figure, then by varying the ratio of $\alpha = a/b$ one can distribute the transmitter power between the HP and LP data. Higher portion of the transmitter power means better protection against the channel errors than the lower portion of the power. This is a simple unequal error protection to a layered coded video, without introducing any redundancy to the modulated data.

Figure 6 shows the average bit error rate (BER) of the HP and LP data in a hierarchical 64-QAM with $\alpha$ values of 1, 2 and 4 in an additive white gaussian channel for random generated data. It can be seen that for $\alpha = 1$, the BER of the high and low priority parts become closer to each other. However, as the value of $\alpha$ increases, the difference between HP and LP becomes greater. Note that hierarchical QAM (HQAM) does not provide any error protection like channel coding. As a result, according to the diagram, high channel SNR is required for an acceptable BER. What is actually offered by HQAM is the unequal priority control to the different parts of the data. To achieve good quality of service in low channel SNR values, one may add some channel coding techniques to the current system and as a result the diagram of the Fig. 6 would simply shift to lower SNR values [18].

## 5 Error concealment

Typically, the compressed video data are protected against channel errors. However, some of the macroblocks (MBs) may still be received in errors. In the

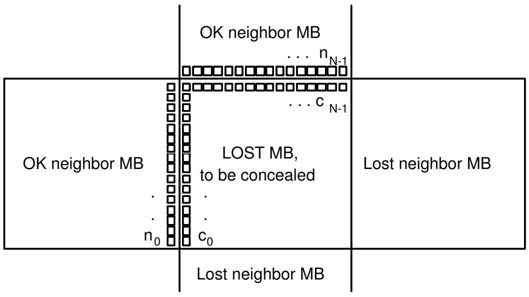


Fig. 7. Boundary pixels of a lost MB.

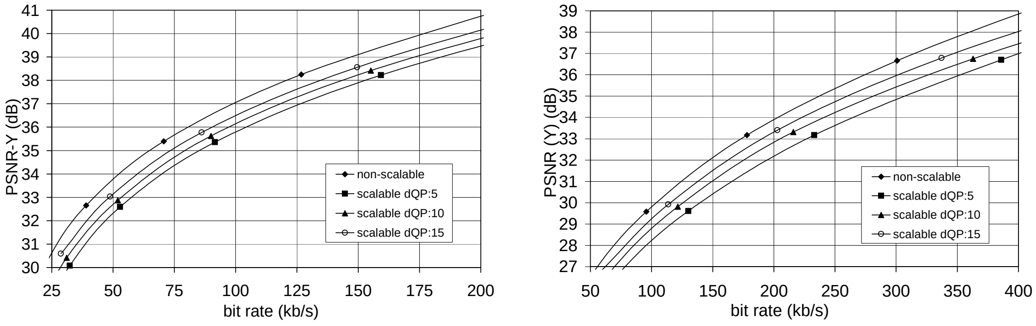
decoder, these erroneous MBs can be concealed using the correctly received data to minimize their visual artifacts. In our decoder, for the base layer, the error concealment method of [20] is used. In this method, every missed MB in an inter frame is replaced by a motion compensated block from the previous frame by an estimated motion vector (MV). It is selected among the surrounding MVs, that result in a minimum block boundary discontinuity (see Fig. 7) which is defined as:

$$D_e = \frac{1}{N} \sum_{i=0}^{N-1} |c_i - n_i| \tag{2}$$

where $c_i$ and $n_i$ are the boundary pixels of the correctly received neighboring MBs and the substituted pixels, respectively. $N$ is equal to $16 \times k$ with $k = 1, 2, 3$ or 4 as the number of correctly received neighboring MBs.

### *5.1 Error concealment in the enhancement layer*

The best choice for concealing a lost enhancement MB would be the one that has been used for its prediction, which is also unknown to the decoder due to loss. Therefore, the concealment choices should be made among: 1) a motion compensated block of the previous enhanced picture by estimating the corresponding MV (forward), 2) the corresponding base layer block (upward), and 3) the motion compensated block using the corresponding base MV (direct). We can examine these three options and the one that results in the lowest boundary discontinuity ($D_e$) would be selected to replace the missing block. This discontinuity is calculated only based on the enhancement data. However, the base layer information can also be used for more accurate error concealment. This is the subject of our further research.



(a)

(b)

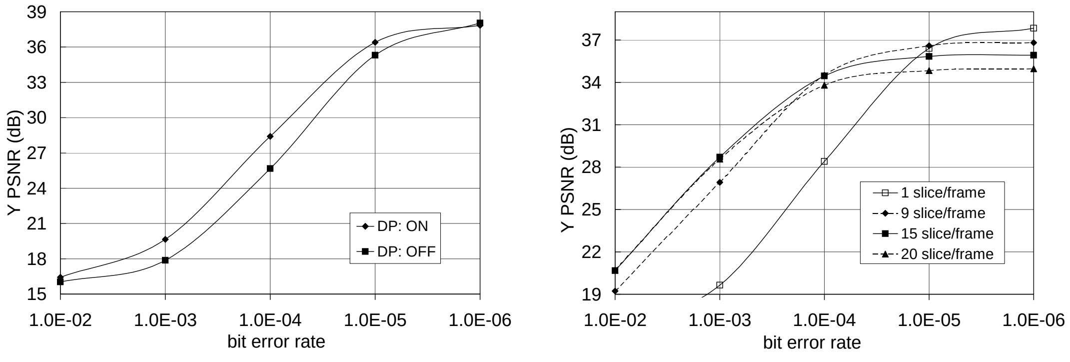
Fig. 8. R-D performance of the scalable codec in three different values of QPB-QPE (dQP) and the non-scalable one, QCIF@10Hz, (a): Foreman 133 frames and (b): Mobile 100 frames.

## 6 Simulation results

### *6.1 R-D efficiency of the SNR scalable codec in a noise free environment*

The encoder and decoder of the proposed SNR scalable codec have been implemented in software based on the standard JVT codec software version 7.3. It has been tested in different coding scenarios for different test video sequences. The Rate-Distortion performance of the proposed method in an error free environment is illustrated in Fig. 8. The results are shown for two different video sequences: "Foreman" and "Mobile and calendar", both QCIF at 10 frames/sec. The encoded bitstreams have one intra frame followed by the remaining sequence in P-pictures. The CABAC mode is enabled for all tests and the prediction (reference) picture is confined to previous frame (current base for upward and direct predictions).

For every sequence, the SNR scalable codec was tested in three different values of dQP (i.e. the difference between the quantization parameters of the base and the enhancement layers). It can be seen that when more bits are assigned to the enhancement layer (larger dQP) the efficiency of the scalable codec is better and is closer to the non-scalable one. The reason is that in SNR scalability, what is actually coded in the enhancement layer as the residual data, is the quantization distortion of the base layer [11]. Therefore, when dQP is small, the distortion coefficients are normally smaller than the enhancement quantization step size and would not be coded (i.e. requantized to zero). Hence, the enhancement layer does not improve the picture quality noticeably while wasting a reasonable amount of addressing data. At high dQPs, however, the quantizer step size of the base layer is large and so the second layer efficiently codes any residual base layer quantization distortion.



(a) (b)

Fig. 9. PSNR vs. bit error rate for standard (non-scalable) bitstreams, Foreman QCIF@10Hz 100 KBits/Sec, (a): when DP is enabled and disabled (1 slice/frame). (b): DP enabled, with different slice/frame values.

### *6.2 Error resilience evaluation*

We have arranged a series of tests to evaluate the robustness of the scalable and non-scalable bitstreams against the channel error. The resilience of the bitstreams are first presented without unequal error protection (UEP) for the data partitioned bitstream, and then the UEP performance for both SNR scalable, and data partitioned bitstreams is presented.

To simulate the channel errors, an Elliot-Gilbert two level error model [11] has been used to introduce bit errors into the bitstreams. We assume that there is a perfect error detection at the decoder and so when the first error occurs in a slice, the rest of the slice will be marked as corrupted and the decoder waits to decode form the next slice. The error detection, however, could be done in the underlying layers with some parity bits or channel decoding. The source decoder also is capable of detecting the errors with an acceptable accuracy [21,22]. Finally, to assess the quality of the decoded video, the corrupted parts of the pictures are concealed using the method explained in section 5.

The average luminance PSNR of the Foreman coded bitstream with different bit error rates is illustrated in Fig. 9. The results are the average of 30 simulation runs, and the bit rates of the bitstreams were set to 100 kb/s, using our Lagrangian rate controller [23]. The sequence lengths were limited to 33 frames assuming that after this number of frames an intra frame would resynchronize the encoder and the decoder. From Fig. 9-a it can be observed that enabling the DP which is simple with negligible overhead (and hence, small quality degradation) has significantly improved the resilience of the bitstream to channel errors.

Fig. 9-b shows the quality of the decoded erroneous video bitstream of the same test sequence with four different number of slices per frame. It is obvious that by inserting more resynchronization headers (i.e. more slices per frame) due to the larger overhead, the output quality in the error free situations has been degraded. However, at higher bit error rates, the more sliced bitstreams have significantly better performances. It should be noted that when the number of slices/frame is more than a specific amount (in this test 15), no further improvement is achieved. For all the remaining experiments we chose 9 slices per frame which is a reasonable value with an acceptable overhead.

### *6.3 Unequal priority control using hierarchical QAM*

To apply the unequal priority control we have chosen 64-HQAM with $\alpha$ equal to 2 and 4. The Foreman sequence was coded at 200 kb/s. This rate was chosen in order the bit rate of the partition A (DPA) of the data partitioned bitstream to be about 33 percent of the total bit rate and so it is considered as the high priority data for the hierarchical modes. Note that for 64-HQAM of Fig. 5, out of 6 bits 2 bits are assigned to the DPA (or base layer in scalability), and hence DPA (or base layer) bit rate is 33 percent of the total bit rate. Here, the NAL units of type A (MVs and addressing data) are mapped to the high priority (HP) bits of HQAM and NAL units of type B and C (residual data) are mapped to the low priority (LP) bits. Note that in the DP mode, there is no provision to influence the bit-rate ratio of the individual data partitions, whereas in SNR scalability the HP to LP ratio (the base and the enhancement layer) is simply adjustable. Therefore, to have a fair comparison between the SNR scalability and DP we just selected a bit rate that naturally has the proper ratio for DP.

Figure 10 shows the resulting luminance PSNR for DP mode, where it is clear that the bitstream with unequal priority control has much better quality in the noisy situations where the SNR of the channel is low. At high SNR values, however, the unequally protected bitstreams have lower quality since the other partitions of the data (DPB and DPC) have poor error protections.

Figure 11 shows the same experiment but with an SNR scalable bitstream. The scalable bitstream has 200 kb/s bit rate and the base layer bit rate is controlled to have 33 percent of the total bit rate using the Lagrangian optimized rate controller [23]. The number of slices/frame is 9 but the DP for both layers were disabled. Form the figure it can be clearly seen that the scalable codec in conjunction with unequal error protection has a dramatic improvement in the bitstream robustness in low channel SNR values.

Figure 12 shows the results of Fig. 10 and Fig. 11 together to compare the

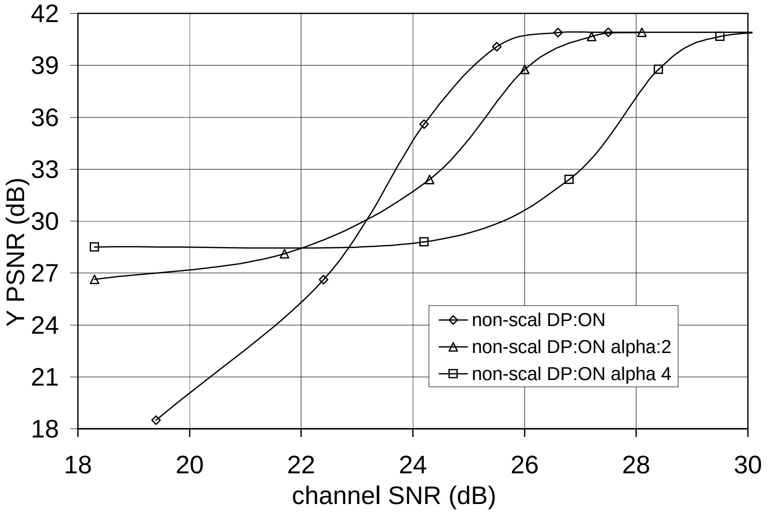


Fig. 10. Luma PSNR vs channel SNR, for a non-scalable, data-partitioned bitstream in non-hierarchical 64-QAM mode and in hierarchical mode when Data Part A is considered as high priority data, with $\alpha = 2$ and 4.

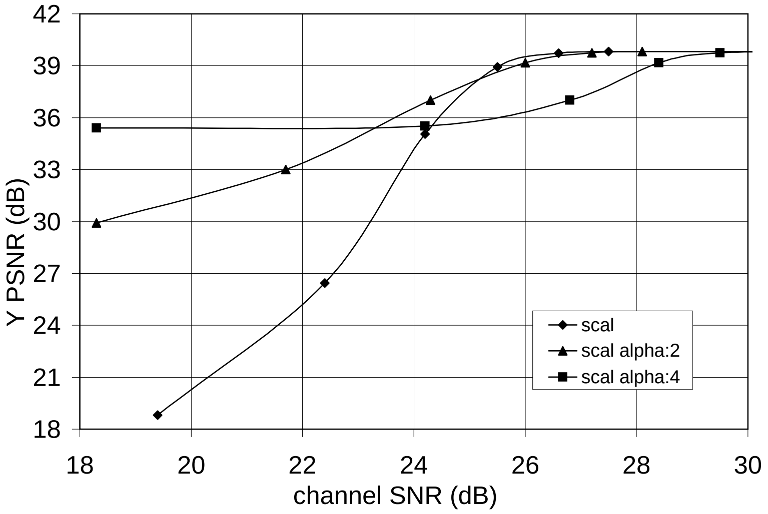


Fig. 11. Luma PSNR vs channel SNR, for an SNR scalable bitstream in non-hierarchical 64-QAM mode and in hierarchical mode when the base layer is considered as high priority data, with a= 2 and 4.

SNR scalability and DP efficiencies. It can be observed that when the channel is error free (high SNR) the scalable bitstream has a lower PSNR (in this test around 1 dB) compared to the data-partitioned one. However, as the SNR of the channel degrades, the unequally protected scalable bitstream has up to 7 dB improvement over the data-partitioned one. It should be added that in this particular test, the DPA bit rate (in the most efficient mode of the encoder) was 33 percent of the bit rate whereas in general, adjusting the DPA bit rate may significantly decrease the encoder efficiency. However, in the scalable encoder we are always able to control the base layer bit rate flexibly

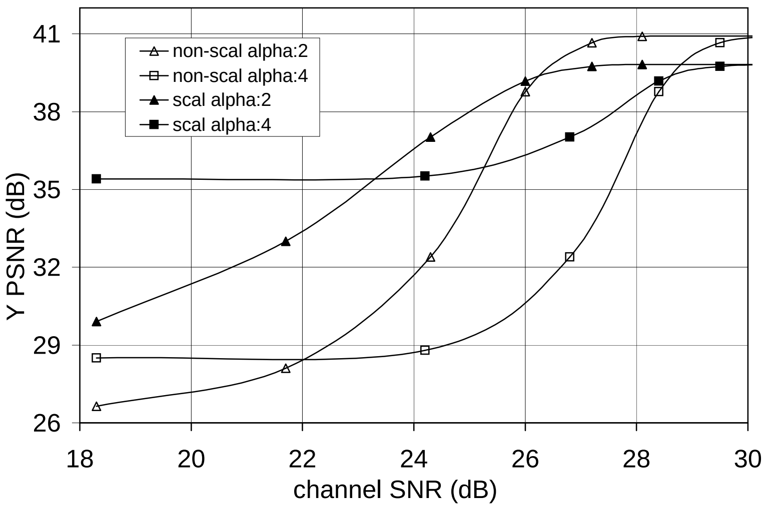


Fig. 12. The comparison of applying unequal priority control on a non-scalable (data-partitioned) and an SNR scalable bitstream.

with limited efficiency degradation as shown in Fig. 8.

## 7 Conclusions

We have proposed an SNR scalable codec based on the H.264/AVC video coding standard. A new set of prediction modes is proposed for the enhancement layer as well as the required modifications to coding and bitstream structure. The scalable codec has an acceptable R-D performance especially when more bit budget is allocated to the enhancement layer. Unequal error protection using hierarchical QAM is applied to the scalable bitstream as well as the non-scalable one with data partitioning. The simulation results confirm that the scalable bitstream in conjunction with unequal error protection is more robust to the channel errors than the data-partitioned one. In addition, the scalable coder provides more flexibility in distributing the bit rate over layers, making it more suitable for unequal error protection by hierarchical QAM.

## 8 Acknowledgement

This project is supported by the engineering and physical sciences research council (EPSRC) of the UK. The authors would like to thank Mr. Pronsak Raibroycharoen and Mr. Bashar Barmada for their valuable contributions.

## References

[1] ITU-T and ISO/IEC JTC1, Advanced video coding for generic audiovisual services, ITU-T Recommendation H.264 ISO/IEC 14496-10 AVC, 2003.

[2] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, G. J. Sullivan, Rate-constrained coder control and comparison of video coding standards, IEEE Trans. Circuits Syst. Video Technol., vol. 13, July 2003, pp. 688-703.

[3] D. Alfonso, D. Bagni, D. Pau, A. Chimienti, A performance analysis of H.264 video coding standard. Proceedings, Int. Conf. on Picture Coding Symposium, Saint-Malo, France, April 2003.

[4] A. Joch, F. Kossentini, P. Nasiopoulos, A Performance analysis of the ITU-T draft H.26L video coding standard. Proceedings, Int. Packet Video Workshop, Pittsburgh, PA, April 2002.

[5] T. Stockhammer, M. M. Hannuksela, T. Wiegand, H.264/AVC in wireless environment. IEEE Trans. on Circuits and Systems for Video Technology, Vol. 13, No. 7, July 2003.

[6] Y. Wang, Q. F. Zhu, Error control and concealment for video communication, a review. Proceedings, IEEE, Vol. 86, 974-997, May 1998.

[7] M. Gallant, F. Kossentini, Rate-Distortion optimized layered coding with unequal error protection for robust internet video. IEEE Trans. on Circuits and Systems for Video Technology, Vol. 11, No. 3, March 2001.

[8] ISO/IEC MPEG, Generic coding of moving pictures and associated audio information, ISO/IEC 13818-2, Video (MPEG-2), May 1996.

[9] International Telecommunication Union (ITU), Recommendation H.263, video coding for low bit rate communications, Feb 1998.

[10] ISO/IEC MPEG, Coding of audio-visual objects - Part 2: Visual. ISO/IEC 14 496-2 (MPEG-4 Visual Version 1), Apr. 1999.

[11] M. Ghanbari, Standard Codecs: image compression to advanced video coding. IEE Telecommunications Series, 2003.

[12] MPEG Video Subgroup Chair, Registered responses to the call of proposals on scalable video coding, ISO/IEC JTC1/SC29/WG11/M10569.

[13] T. Halabach, T. R. Fischer, SNR scalability by transform coefficient refinement for block-based video coding. Proceedings, Conf. Visual Commun. and Image Processing, Lugano, Switzerland, July 2003.

[14] Y. He, F. Wu, S. Li, Y. Zhong, S. Yang, H.26L-based fine granularity scalable video coding. IEEE int. symposium on circuits and systems, Vol. 4, pp. 548-551, Phoenix, USA, May 2002.

[15] L. Blaszak, M. Domanski, S. Mackowiak, Spatio-Temporal scalability in AVC codecs. ISO/IEC JTC1/SC29/WG11, Pattaya, March 2003.

[16] H. Schwarz, D. Marpe, T. Wiegand, SNR-scalable extension of H.264/AVC. ISO/IEC JTC1/SC29/WG11, JVT-I032d1, San Diego, September 2003.

[17] European Broadcasting Union, Digital Video Boradcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television, ETSI EN 300 744 v1.4.1, 2001.

[18] B Barmada, E V Jones, Adaptive mapping and priority assignment for OFDM, Proceedings, IEE conference of 3G mobile communication technologies, London, 2002.

[19] L. Hanzo, W. Webb, T. Keller, Single and multi carrier quadrature amplitude modulation: principles and applications for personal communications, WLANs and broadcasting, John Wiley and Sons, 2000.

[20] Ye-Kui Wang, M.M. Hannuksela, V. Varsa, A. Hourunranta, M. Gabbouj, The error concealment feature in the H.26L test model. Proceedings, IEEE Int. Conf. on Image Processing (ICIP), 2002.

[21] P. Zhou, Y. He, A fragile watermark error detection scheme for JVT, Proceedings, Int. Symposium on Circuits and Systems (ISCAS), Vol. 2, May 2003, pp. II-956-II-958

[22] H. C. Shyu, J. J. Leou, Detection and concealment of transmission errors in MPEG-2 images-a genetic algorithm approach. IEEE Trans. on Circuits and Systems for Video Tech., Vol. 9, Issue. 6, Sept. 1999, pp. 937-948

[23] M. M. Ghandi, M. Ghanbari, A Lagrangian optimized rate control algorithm for the H.264/AVC encoder, Proceedings, IEEE Int. Conf. on Image Processing (ICIP), 2004.