AN MPEG2 COMPLIANT PSNR CONTROLLER FOR A CONSTANT HIGH QUALITY STUDIO ENVIORMENT

Enrico Frumento, Rosa Lancini
CEFRIEL, Politecnico di Milano,
Via Fucini 2, Milano, I-20133, ITALY
Tel. +39 2 23954-209, Fax. +39 2 23954-254
e-mail: frumento@cefriel.it, rosa@cefriel.it

ABSTRACT

This paper proposes a new approach for the quality control of an Mpeg2 source. Actually the VBR coder is able to assure an average constant quality (namely less variable PSNR with respect to the CBR one) but it does not in a contribution application case, where the quality constraint plays an important role and a small variance of quality could be unacceptable. The reproduced quality is mostly relayed on the quantization parameters and we have studied how to control them to assure the same constant PSNR on each frame for a VBR-Mpeg2 profile 4:2:2 source.

1 INTRODUCTION

Nowadays the Mpeg2 [2] is become the most used technology for compression of digital video streams. The signal can be compressed in order to produce a bitstream at Constant Bit Rate (CBR) or at Variable Bit Rate (VBR). In the first case a constant bit-rate is sent to the network, not considering the variation of the quality, in the second case the constraint is to assure a "constant" quality whatever the generated bit rate. Unfortunately this last assert is not assured when high constant quality is required.

Actually the VBR coder is able to assure an average constant quality (namely less variable PSNR with respect to the CBR one) but, it does not in a contribution application case, where the quality constraint plays an important role and a small variance of quality could be unacceptable. The reproduced quality is mostly relayed on the quantization parameters and we have studied how to control them to assure the same constant PSNR on each frame for a VBR-Mpeg2 profile 4:2:2 source.

The Mpeg2 quantization phase is based on two parameters: a scalar quantizer usually named "`mquant`", and a Quantization Matrix (QM). The first parameter sets the "granularity" of the quantization, the second one weights the different coefficients belonging to the block depending on their "frequency" position. As explained in [1] the coding base data unit is the block (8×8 pixels), and each coding parameter could be modified at least on a macroblock basis. Note that due to the particular Mpeg2 quantization method, the lower is the `mquant` the higher is the coded sequence quality. The Mpeg2 standard specifies that the QM could be modified only on a frame by frame base and the scalar quantizer `mquant` on a macroblock base so it is evident that the scalar quantizer is more convenient for a more accurate quality controller.

A standard Mpeg2 coder does not directly store in the stream the quantizer values, but two indexes called `q_s_c` and `q_scale_type` [2] by which the `mquant` is tuned. The decoder uses these two values to find the real quantizer inside a standard bidimensional look-up table: because `q_scale_type` could only be 0 or 1 there are two possible sets of allowed `mquant` values that are function of `q_s_c` [2]. For our particular application `q_s_c` is the parameter that has to be controlled to assure more accurate results. This paper is so organized: section 2 explains the structure of the proposed algorithm and the different proposed solutions. Section 3 presents the comparison results between controlled and uncontrolled sources. For the safe of clarity the obtained results for the VBR case are compared to that obtained by the CBR solution. Finally conclusions and future research are given in section 4.

2 ALGORITHMS STRUCTURE

We propose different optimizations for the same algorithm to the quality control of sources in an VBR-Mpeg2 coder. As said in the previous section, we act on the `q_s_c` parameter to assure a control based on a macroblock dimension. In the Mpeg2 quantization phase we introduce two new steps: the *first* one evaluates an initial estimation of the quantization value q_0, that gives a $PSNR_0$, and the *second* one refines the initial estimation in order to find the final value q_{new} that assures a target quality $PSNR_T$. The overall logical structure is shown in figure 1.

With reference to figure 1 sl, and mb are respectively the current slice [1] and macroblock, *type* is the frame type (Intra or Inter), $PSNR_0$ is the quality obtained

[1] Our coder uses slices that are always a frame's row long and a macroblock large.

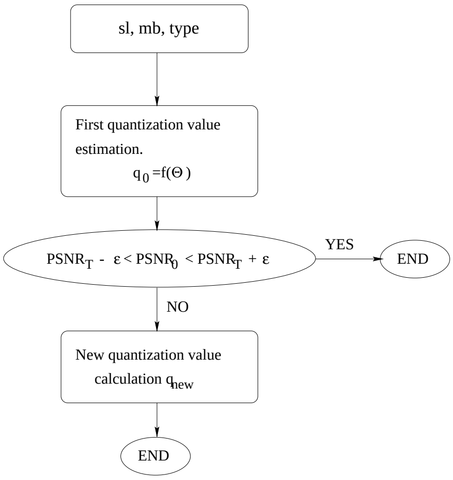

Figure 1: Logical structure of the proposed algorithms.

with the quantizer q_0 and $PSNR_T$ is the quality target. The first step gives an initial quantization value q_0 estimated utilizing the quantization values previously used for the adjacent macroblocks. This is obtained by a function $f(\Phi)$, where Φ is a subset of the previously used quantization factors: $\Phi = \{q :$ neighborhood of macroblock mb, slice sl, frame $fr.\}$.
Then, the macroblock is quantized by the obtained factor q_0 as Mpeg2 coder does. After this step, if the $PSNR_0$ is not close to the target $PSNR_T$ (a tolerance threshold ϵ is given) a new quantization value q_{new} is calculated and then used. This is the core of the procedure, and it is based on a coding error model. After this second phase, the controller does not verify the resulting PSNR any more: the controller loop is done only one time due to the supposed goodness of the errors models and the computational effect constraints. We proposes two different control algorithms to generate q_{new}.
The proposed method adds new complexity to the coder (the further quantization), but the percentage of re-quantization is a function both of the imposed $PSNR_T$ and of the characteristics of the sequence to be coded. The higher is the difference between the uncontrolled quality $PSNR_0$ and the imposed $PSNR_T$, the stronger will be the controller action (more computational effort). The controller performance is directly relayed on the quantization error model and a particular attention is paid to this.

2.1 First quantization value prediction q_0.

With reference to figure 1, the value q_0 that is used for the first quantization is calculated as a function of the set Φ, that is a subset of the quantizations values used for the previously coded macroblocks.
If sl and mb are respectively the current slice and macroblock and $q(sl, mb)$ is the corresponding quantizer, the set Φ used is of this type:

$$\Phi = \{q(sl, mb-1)\ ;\ q(sl-1, mb)\} \tag{1}$$

as in figure 2.

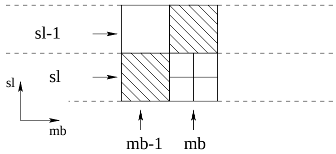

Figure 2: Scheme of the Φ set for the macroblock mb of the slice sl. There are also shown the four blocks of the current macroblock.

The proposed algorithm is shown in figure 3 where $quant(i, j)$ is the quantization factor for the macroblock j-th of the i-th slice, *prev_quant* is equal to the value used for the first macroblock (j=0) of the first slice (i=1) of the last coded frame ($quant(1, 0)$). With reference to figure 3 steps (1) and (2) are necessary to avoid problems at frame bounds.

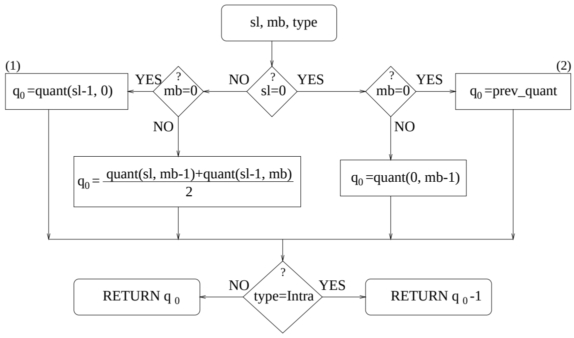

Figure 3: Algorithm scheme for the calculation of the quantization factor q_0.

The type of frame (Intra or Inter) introduces a difference within the procedure. As can be seen in figure 3 the returned q_0 is different depending on the type of the image (Intra or Inter) in fact in the Intra case the q_0 is decreased of one unit. Having a smaller q_0 means to obtain a better quantization, namely a better quality. Being q_0 the only parameter that we can manage in the Intra case (we can not exploit the temporal redundancy) we use it in a more "reduced" way to obtain a good quality and guarantee a comparable quality to the Inter frame. This solution is proved by the alignement of the PSNR values between Intra e Inter macroblocks. This procedure is particularly useful within P and B frames for which could be either Intra and Inter macroblocks.

2.2 Estimation of the new quantization factor q_{new}

As explained in section 2 and with reference to figure 1, after the first quantization value prediction q_0, the obtained $PSNR_0$ value is tested. If it is far from the $PSNR_T$ target value, a new quantization is done with a new estimated quantizer q_{new}. This is the most delicate algorithm's phase, for which we studied two different solutions.

2.2.1 *Variable Increment solution (VI)*

The simplest way to calculate the new quantization factor is to increase/decrease the previously obtained quantizer q_0 by a value that is function of the difference between $PSNR_0$ and $PSNR_T$ as in formula (2). Figure 4 shows the logical structure of the algorithm.

The constant K of the formula (2) is an attenuation factor that is useful both to reduce the algorithm sensibility and as scaling factor. In our case it is equal to 1.5; this value is estimated measuring the controller stability [1,2].

$$\begin{cases} q_{new} = & q_0 + Inc(PSNR_0, PSNR_T) \\ & \text{if } PSNR_0 > PSNR_T \\ q_{new} = & q_0 - Inc(PSNR_0, PSNR_T) \\ & \text{otherwise} \end{cases} \tag{2}$$

$$Inc(\cdot) = round\left(\frac{PSNR_0 - PSNR_T}{K}\right)$$

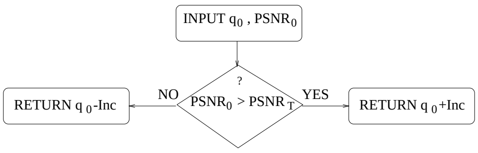

Figure 4: Scheme of the VI algorithm.

In this way, we obtain an adaptive algorithm that finds the correct quantization value (the quantization able to give a constant $PSNR_T$ during the sequence) after few macroblocks. The limited $Inc(\cdot)$ impulsivity is the only drawback of the algorithm when big differences between adjacent macroblocks occur, even if the stability is reached within few macroblocks.

2.2.2 *MSE Analysis solution (MA)*

The second solution moves its attention on the MSE generated by the quantization process, rather than on the PSNR. Analyzing the DCT coefficients quantization process of the Mpeg2 standard, it comes to evidence that a first approximation of the quantization error can be seen as a function of the MSE, like in the formula (3). In that formula, α is a proportional coefficient variable for each macroblock, function of the current macroblock DCT coefficients and of the Mpeg-2 QM; q is the scalar quantizer value:

$$MSE = \alpha \cdot q \cdot \tag{3}$$

With reference to figure 1 it is possible to estimate α after the first quantization q_0, as in formula (4) where MSE_0 is the MSE corresponding to $PSNR_0$:

$$\alpha = \frac{MSE_0}{q_0} \cdot \tag{4}$$

Once the parameter α is known it is possible to find q_{new} as in formula (5), where MSE_T is the MSE corresponding to $PSNR_T$:

$$q_{new} = \frac{MSE_T}{MSE_0} \cdot q_0 \cdot \tag{5}$$

It should be pointed out that this solution returns the real value for q_{new} (that is the real quantization value) instead of the `q_s_c` index as suggested by the Mpeg standard and as done by the previous presented solution (VI), therefore it is necessary a further step able to find the exact standard `mquant` value nearest to that estimated by our algorithm, able to return the corresponding `q_s_c` index. The difference between the nearest allowed `mquant` and that calculated in formula (5), is variable and can lead to unpredictable PSNR fluctuations. Fortunately this problem is more relevant only in the low quality coding case for which the larger differences among the allowed `mquant` values (in the Mpeg2 look-up table) causes bigger differences and so bigger variations of the generated PSNR.

3 SIMULATION RESULTS

In this section, first a comparison study between the two proposed methods (VI and MA) for quality control applied to VBR source, is presented. Further, the best one is compared to a VBR source without quality control and for completeness also to a CBR source that in average offers a similar quality.

The results refer to a 48 frames, 2 sec CCIR-601 sequence obtained joining three standard sequences. The resulting sequence is particularly critical for the PSNR controller, due to the extremely different characteristics of the joined sequences. It is most variable than for usual sequences, even for the presence of two scheme changes in 2 seconds. The structure is the following:

- 20 frames of "Mobile" with spatial frequencies and low movements, no Pan and Zoom;
- 20 frames of "Basket" with relatively low spatial frequencies, high movements, high Pan;
- 8 frames of "Table Tennis" with medium spatial frequencies and movements, high Zoom.

Figure 5 shows the comparative study between the two proposed methods (VI, MA) for a $PSNR_T$ equal to 42 dB (this quality is obtained by an average compression of 36-40 Mbit/s). For high quality environments the two proposed methods give similar performances.

Figure 6 shows a comparative study among either VBR and CBR sources and the controlled VBR one by using the VI solution. In the comparison we used a fixed (12,2) GOP, then the standard sources are coded with a bit-rate for the CBR source (40 Mbit/s) and an *mquant* for the VBR source (em mquant equal to 5) that give an average PSNR equal to that imposed to the controlled coder. As you can see (figure 6) the proposed quality controller is able to assure a constant quality during the sequence, which is not true in the other two cases. Tests

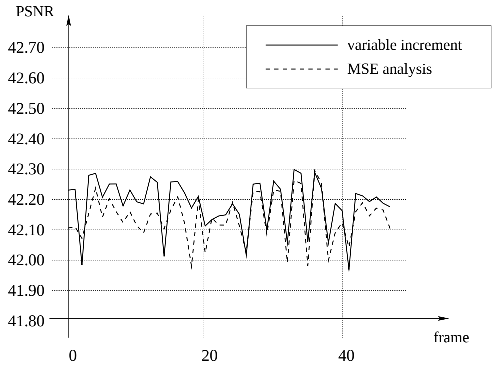

Figure 5: Comparative study between the two proposed methods at $PSNR_T = 42$ dB.

done with more longer sequences (1000 frames, 40 secs) have reported the same performances.
The less variance of the PSNR is more visible in figure 7 where it is shown the macroblock's histogram (distribution) for each PSNR value [2]. The higher is the central peak, the better is the control action.

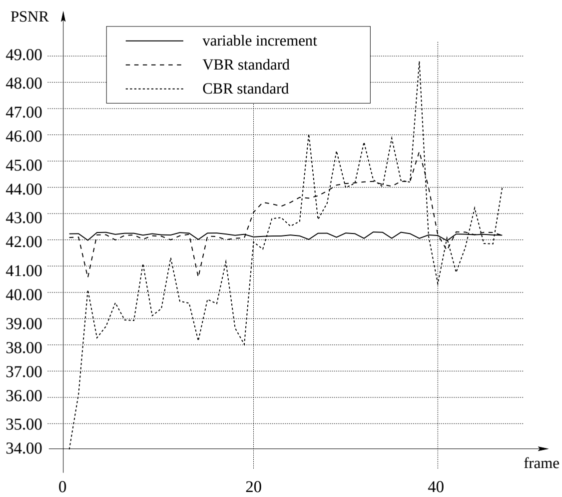

Figure 6: Comparative study among VI solution and VBR, CBR sources with $PSNR_T = 42$ dB.

4 CONCLUSIONS

Two different algorithms for quality control in a VBR source have been studied: VI and MA. We shown that they gave the same results and were able to control and mantaining constant the quality during the sequence coded in the VBR mode. Furthermore we have optimized our solution to reduce the computational effort applying this method to a real-time coder. This method was tested also for lower quality coding, for which not all the possible `mquant` values are allowed. The results have demonstrated our approach competitiveness also in

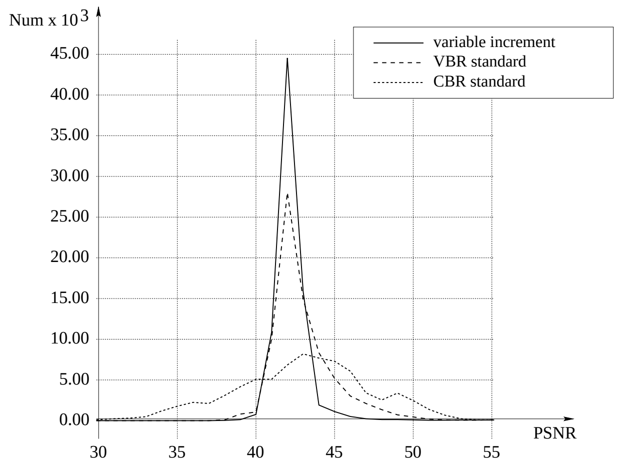

Figure 7: PSNR histogram of the coded macroblocks with 42 dB target.

this case: PSNR is less variable than with normal VBR sources. Future researche is towards the definition of a generic controller able to assure the same controlling preciseness either for high and low quality enviorments.

Acknowledgments

The authors wish to thank R. Zeuli, a postgraduate student at CEFRIEL, who has been involved in the development of this work.

References

[1] D. Le Gall, *Mpeg: a Video Compression Standard for Multimedia Applications*, Communications of the ACM, IEEE, vol. 34, no.4, pp. 46-58, April, 1991

[2] SC29/WG11 ITU-T/SG15, *Generic Coding of Moving Pictures and Associated Audio, ISO/IEC 13818-2*, Committee Draft, May, 1994

[2] This histogram represent the number of macroblocks for each entire PSNR value *val* that have a PSNR belonging to the interval [*val*-0.5,*val*+0.5].