NONLINEAR PREPROCESSING FOR MPEG–2 VIDEO SEQUENCES

F. Gottardo, J. Mir , G.Parladori

Advanced Technologies Laboratory
via Trento, 30
I-20059 Vimercate (MI), Italy

G. L. Sicuranza

D.E.E.I., University of Trieste
via A. Valerio 10
I-34127 Trieste, Italy

ABSTRACT

In this paper a new block-based technique for image prefiltering is proposed, in order to overcome some limits of the MPEG–2 encoding process at low bitrates. The presented scheme performs a suitable classification of the blocks in the images based on motion estimation taking into account the characteristics of the human visual system. The information on the motion intensity, extracted from each image, is used to locally vary the filtering parameters of a nonlinear adaptive low-pass filter.

1. INTRODUCTION

The use of compression algorithms at low bitrate often introduces annoying artifacts. This applies also to the MPEG–2 algorithm, where the encoding process typically reduces the visual quality of image sequences, especially when the bitrate decreases below $6 \div 8$ Mbit/s. Furthermore, the presence of noise at the encoder input can significantly impact on the performance of the coding operation, resulting in a lower quality of the compressed signal. It is possible to overcome this limit of the encoding process with a suitable preprocessing of the input image sequence, in order to reduce the bitrate at a given picture quality or improve the picture quality at a given bitrate. This is the only way to enhance the system performances when using existing chip sets that do not allow any optimization of the encoder.

Prefiltering is defined as a set of operations which makes a video sequence more easily compressible by smoothing image details which are perceptually not relevant. The inclusion of the preprocessing allows the encoder to produce pictures of subjectively higher quality, given a desired bitrate.

The recent literature shows some proposal of video preprocessing schemes ([1], [2]), in which algorithms for prefiltering of video digital sequences have been developed and it has been proved that significant gains in the image quality can be obtained with preprocessing. In this paper we exploit and extend some ideas proposed in [1]. The presented technique performs a suitable classification of different images based on motion estimation, taking into account the characteristics of the Human Visual System (HVS). Our main assumptions are that the sensibility of the HVS is a decreasing function of the object velocity and that the HVS tends to accept a slight blurring distortion in place of the blockyness artifact. Therefore, it is possible to eliminate, with a low-pass filtering, as much high frequency information as possible in motion areas avoiding coding artifacts without compromising the visual image quality.

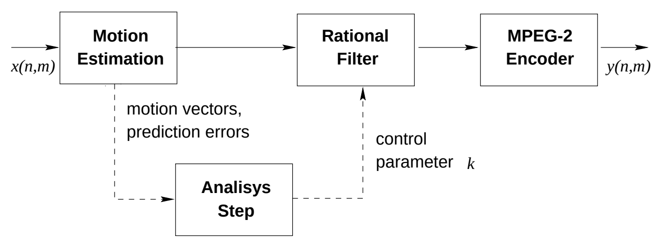

Figure 1: A simple scheme of the system with the preprocessing step and the MPEG–2 encoder. The video sequences used in this work are in the 4:2:2 interlaced 25 Hz digital video format (the images have dimensions of 576×720 pixels)

The motion information extracted from the image is used to control a nonlinear adaptive filter, based on the rational filter proposed in [3]. For a given image the filtering parameters are locally modulated according to the motion intensity. Maximum information reduction is applied on image areas with fast movements, while only a very light low pass action takes place in the stationary areas. Furthermore, the rational filter is capable of biasing its behavior in order to achieve good performances both in uniform areas, where linear smoothing is needed, and in textured zones, where nonlinear and directional filtering is required.

This paper is organized as follows: in Section 2 the motion estimation phase and a detailed description of the rational filter are presented. The image analysis and the filter control are described in Section 3 (refer to the blocks of the scheme in Figure 1). The simulation results are reported in Section 4.

2. MOTION ESTIMATION AND RATIONAL FILTER

The first stage of our scheme is the motion estimation. It is necessary to separate stationary from moving sub-areas of the image. A new algorithm for motion estimation proposed in [4] is implemented to significantly reduce the computational complexity of this step. The result of the motion estimation is the set of motion vectors and prediction errors of the image. Then, the blocks are grouped into sets of homogeneous motion characteristics. Either the motion estimation and the next operations are always made on single fields of the images.

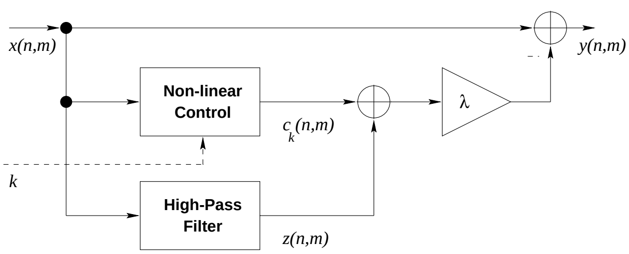

Figure 2: A block diagram of the used rational filter

The behavior of a 1-D rational filter is described by the following expression:

$$y(n) = x(n) - \lambda \cdot c_k(n) \cdot z(n) \qquad (1)$$

where $x(n)$ is the input signal, $z(n)$ is the high frequency component of the input signal obtained from a simple high pass filter, $y(n)$ is the output signal, λ is a positive parameter and $c_k(n)$ is a control function, depending on a parameter k, expressed as a rational operator:

$$c_k(n) = \frac{k}{k + s^2(n) + \epsilon} \qquad (2)$$

In this equation $s(n)$ is a detail sensor, obtained from an edge extractor applied to the input signal $x(n)$.

Let us examine the filter behavior for different positive values of the k parameter:

- $k \simeq 0$: in Eq.(1), $y(n) \simeq x(n)$ and the filter has no effect. Observe that the constant ϵ has the only role of avoiding that a 0/0 condition be reached when $k = 0$ and $s(n) = 0$. In the realization of the operator we can neglect it, as the case $k = 0$ has no practical interest.
- $k \rightarrow \infty$: the proposed filter becomes a simple linear low-pass filter.
- For *intermediate* values of k, the high-frequency component $z(n)$ is weighted by the term $k/[s^2(n) + k]$; if $s(n) \simeq 0$ (in the uniform areas of the image), then the overall behavior is again a linear low-pass filter; if $s(n) \gg 0$ (i.e., its square is not neglectable with respect to k), $c_k(n) \simeq 0$ and again $y(n) \simeq x(n)$. This means the $s(n)$ term perceives the presence of a detail and accordingly reduces the smoothing effect of the operator.

Therefore, the rational filter can be interpreted as a linear filter whose coefficients are modulated by the edge-sensitive component.

It is easy to extend this concept to two dimensions. The 2-dimensional form (Figure 2) of the rational filter is:

$$y(n,m) = x(n,m) - \lambda \cdot \frac{k \cdot z(n,m)}{k + s^2(n,m) + \epsilon} \qquad (3)$$

To obtain the high frequency component $z(n,m)$ of the image, we have implemented the high-pass filter as a Laplacian filter:

$$LP = \begin{vmatrix} -0.7 & -1 & -0.7 \\ -1 & 6.8 & -1 \\ -0.7 & -1 & -0.7 \end{vmatrix}$$

The edge-sensor $s(n,m)$ in the denominator of Eq.(2) is obtained from a Sobel edge extractor. Its presence makes the rational filter very robust to noise [5], while the ability to preserve image details is slightly reduced. In the extreme case $k \rightarrow \infty$, we obtain the following low-pass filter, depending on λ :

$$LP_{eq} = \begin{vmatrix} 0.7 \cdot \lambda & \lambda & 0.7 \cdot \lambda \\ \lambda & 1 - 6.8 \cdot \lambda & \lambda \\ 0.7 \cdot \lambda & \lambda & 0.7 \cdot \lambda \end{vmatrix}$$

It is easy to show that, if $0 < \lambda \leq 10/96$ in the Eq.(3), the filter response is monotonic along the horizontal and vertical directions, while if $10/96 < \lambda \leq 1/8$ it is monotonic only along diagonal directions[1]. In our computer simulations we have chosen λ always between 0.08 and 0.12.

3. IMAGE ANALYSIS AND FILTER CONTROL

A preliminary image analysis is applied to each field extracted from the frames of the sequence. In our system, a block-based segmentation of the fields is used to operate consistently to the encoding process and to obtain a computationally simple control of the filtering action. For each block B_{ij} in the current field, the suitable value k_{ij} of the control parameter k of the rational filter (see Eq. 3) is selected, according to the prediction errors (MAE,

[1] To obtain this conditions, it is sufficient to impose e.g. $H_{LP_{eq}}(e^{j\pi}, e^{j0}) = 0$ for the horizontal and vertical monotony and $H_{LP_{eq}}(e^{j\pi}, e^{j\pi}) = 0$ for the diagonal monotony.

Mean Absolute Error) and the macro-block's motion vectors ($\overline{mv}_{ij} = (m_{ij,x}, m_{ij,y})$, where i is the row index of the block in the field and j is the column index): therefore, in this way it is possible to control the amount of smoothness imposed on the blocks of the image.

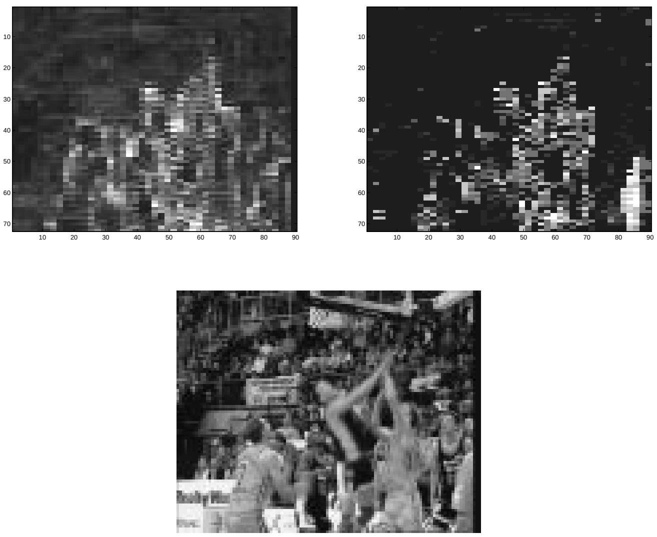

Figure 3: Representation of the MAE matrix (upper left) and of the mi matrix (upper right) for the original frame 125 of the sequence *Basket*

For each field at the time n, the motion estimation is performed either on the homologous and the non-homologous field in the frame at the time $n+1$.
Two *motion indexes* are then computed:

$$mi_{ij}^{hom} = |m_{ij,x}^{hom}| + |m_{ij,y}^{hom}|$$

$$mi_{ij}^{nhom} = |m_{ij,x}^{nhom}| + |m_{ij,y}^{nhom}|$$

which will be used instead of the motion vectors. The dimensions of the search area for the motion estimation in the reference field of 288×720 pixels are 32 pixels in horizontal direction and 16 pixels in vertical direction. Therefore the motion indexes are included in the range $0 \div 24$. We then select the MAE_{ij} for the block B_{ij} in the current field according to the following equation:

$$MAE_{ij} = \min(MAE_{ij}^{hom}, MAE_{ij}^{nhom})$$

and consequently mi_{ij}^{hom} or mi_{ij}^{nhom} is chosen for that block, in order to generate two matrices of motion indexes (mi) and prediction errors (MAE), which will be used in the subsequent operations (Figure 3). Then, either the MAE matrix and mi matrix are normalized with respect to their maximum values, assuming that the minimum value for both matrices is 0.

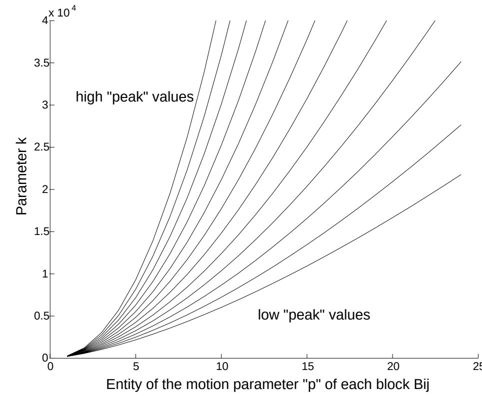

Figure 4: The set of parabolas for the determination of k_{ij} parameters

We must also emphasize that all the choices of the values, made in the remaining of this section, are based on a heuristic approach.

The first operation on the two data matrices obtained from the motion estimation consists of a twelve-interval partition of the MAE and mi values distributions; then, two histograms are computed based on the occurrence of the values in each interval. The two histograms peaks, $peak_{MAE}$ and $peak_{mi}$, are then evaluated and combined in one index:

$$peak = \alpha \cdot peak_{MAE} + (1-\alpha) \cdot peak_{mi} \tag{4}$$

where the parameter α in the Eq.(4) is set to 0.6. The index *peak* is used to choose one of the curves in Figure 4, which have the general form $y = a \cdot x^b$. From the selected curve, the parameters k_{ij} of all the blocks in the current field are obtained, according to the value of a new index p_{ij}, introduced in the following equations. The MAE values scaled in the range $0 \div 24$, i.e. mae_{ij}, are used, together with the motion indexes mi_{ij}, to determine for each block the global index p_{ij}:

$$p_{ij} = \beta \cdot mae_{ij} + (1-\beta) \cdot mi_{ij} \tag{5}$$

where $\beta = 0.4$. Given the index p_{ij}, we are able to choose the most suitable k_{ij}:

$$k_{ij} = a_{field} \cdot p_{ij}^{b_{field}} \tag{6}$$

where a_{field} and b_{field} are the parameters of the curve in Figure 4, selected according to the *peak* value in Eq. (4). From this equation, the parameter of the rational filter for each block of the current field is calculated.

An additional slightly low pass filtering action of the matrix k_{ij} is introduced to avoid step variations of the filter parameters. A k_{ij} matrix and its filtered version are shown in Figure 5.
In order to increase the temporal coherence of the filtering and therefore improve the quality of the processed images, reducing the flickering effect, we have introduced another

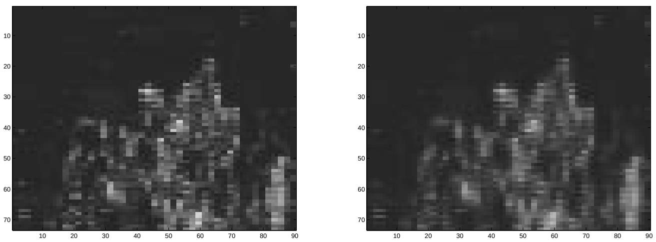

Figure 5: From left: original and lowpass filtered k_{ij} matrix for the frame 125 of the sequence *Basket*

operation on the parameters. At the time n, the parameter *peak* and each k_{ij} are computed taking into account those at the time $n-1$. The used rule is (Figure 6):

$$g_n = \eta \cdot c_n + (1-\eta) \cdot g_{n-1}$$

where g_n and g_{n-1} indicate the parameters *peak* and k_{ij} used at the times n and $n-1$ respectively, while c_n is the corresponding parameter at the time n computed from Eqs. (4÷6). The simulation results have confirmed that the best value of the constant η is included in the interval $0.6 \div 0.8$. In case of frequent scene changes, this control regularizes the filtering action.

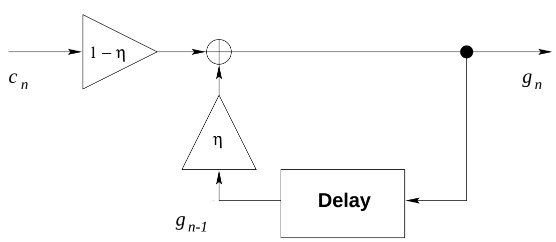

Figure 6: A block scheme of the *IIR* system for the memory of the system

Moreover, an offset to the calculated k_{ij} value is added, in order to slightly filter also the static image areas. In this way, the filter acts on still images as a denoiser. The offset is set by default to 200, but it is variable according to the coder's rate, allowing a flexible preprocessing configuration. Usual values of k_{ij} in moving areas vary from $5 \cdot 10^2$ to $2.8 \cdot 10^4$.

At this point, the k_{ij} calculation is completed. From each 8×8 block of the current field, the gradient and the high frequency component are extracted (Figure 2): then, each block B_{ij} is filtered by the rational operator according to Eq. (3). The parameter λ is usually set equal to 0.12.

If a block has very high values of motion index and MAE, and therefore of p_{ij}, then it is filtered again. The blocks filtered twice are those characterized by a p_{ij} value higher than the mean value of the p_{ij} distribution in the field. In the second filtering, only the parameter λ is varied and set equal to 0.09.

4. COMPUTER SIMULATIONS

In this section, we report the simulation results obtained with the MPEG–2 encoder having GOP dimension equal to 12, and fixed bitrate of 4 Mbit/s. We have tested our preprocessing system on 48 frames of the sequences *Basket*, *Dogs* and *Rafting*.

Figure 7: Original and prefiltered images of the sequence *Dogs*, after 4 Mbit/s MPEG–2 encoding process

The objective quality of the preprocessed sequences after the encoding process is estimated in terms of PSNR of the output images with respect to the original and filtered images and compared to the PSNR of the original coded sequences. We present only some graphs, in which the PSNR referring to the 4 Mbit/s encoding of the sequence *Basket* is reported: in Figure 8, the PSNR of preprocessed and coded images with respect to the filtered images is compared to the PSNR of the original coded images. We can observe that the gain in the case of preprocessed images is $1.3 \div 1.8$ dB on the average: this means that the encoding process is more efficient with preprocessed images. The PSNR of the prefiltered images with respect to the original images is obviously lower than the PSNR of the original images (Figure 9), but this fact does not correspond to a reduction of the visual quality. On the contrary, the preprocessed and coded sequences have often higher subjective quality than the originals.

It is worth noting that, by simple real–time visual comparison of the filtered and original sequences before the en-

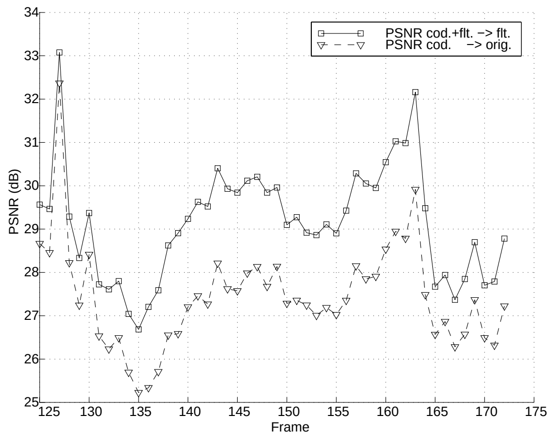

Figure 8: PSNR of the output images with respect to the original (dotted line) and filtered images for the sequence *Basket* (4 Mbit/s)

coding process, one can hardly notice the smoothing effect in the preprocessed images: only in the scenes with high speed moving objects or in areas of high motion complexity, some small differences are perceptible by direct comparison of the filtered and original frames. Moreover, the preprocessing improves the coding performance in terms of picture quality: the blockyness is noticeably reduced particularly for 4 Mbit/s (see the Figure 7) and also a lot of annoying artifacts, such as flickering effects, are almost completely eliminated, while the details and the edges in the static image areas as well as the sharpness are preserved due to the action of the rational filter.

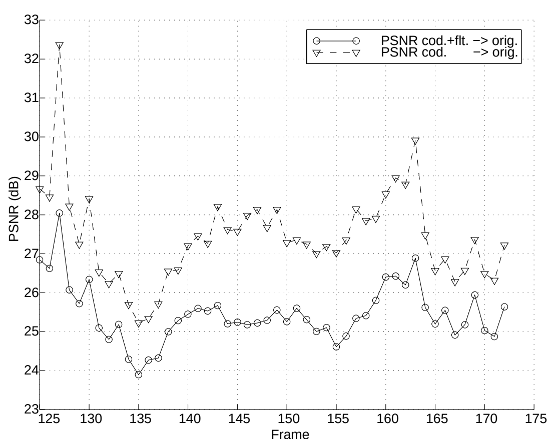

Figure 9: PSNR of the output images with respect to the original images for the sequence *Basket* (4 Mbit/s)

5. CONCLUSIONS

In this work, we have described a preprocessing system for MPEG–2 encoders, based on the classification of the blocks in the picture according to the motion estimation and on the use of a nonlinear edge–preserving filter, called "rational" filter. Computer simulations have shown that a subjective improvement of the picture quality can be obtained after the encoding process at low bitrates: the smoothing effect is present only in moving areas and is not perceptible in the real–time vision, while the edges of the static objects are preserved and the coding artifacts are strongly reduced.

Acknowledgements

This work has been partially supported by the European Project ESPRIT 20229 Noblesse.

6. REFERENCES

[1] N. Vasconcelos, F. Dufaux, "Pre and Post-Filtering for Low Bitrate Video Coding". In *Proceedings of the International Conference on Image Processing ICIP,* October 1997.

[2] L. Yan, "Noise Reduction for MPEG Type of Codec". In *Proceedings of the International Conference on Acoustical Speech Signal Processing ICASSP,* 1994.

[3] G. Ramponi, "A Rational Edge-Preserving Smoother". In *Proceedings of the International Conference on Image Processing ICIP,* Washington, October 1995.

[4] S. Cucchi, D. Grechi, "New Feature Based Fast Algorithm for Motion Estimation: Decimated Integral Projection". In *Proceedings of the International Conference on Information Communications and Signal Processing ICICS,* Singapore, September 1997.

[5] G. Ramponi, G. L. Sicuranza, "Sobel-laplacian Image Sharpening for Noisy Data". In *Proceedings of Digital Signal Processing DSP95,* Cyprus, June 1995.