Noise Reduction Techniques for Video Coding Applications

G. Cortelazzo, G. A. Mian, R. Parolari, L. Sbaiz
Dipartimento di Elettronica e Informatica
Padova, Italy

Abstract

Granular noise is visually bothersome with respect to high quality rendition, and it is a well known source of visual artifacts for coding algorithms. This work considers the application of nonlinear filtering techniques to granular noise reduction and their effects on video coding. The performance of the examined techniques on coded images is documented both in terms of visual rendition, and in terms of objective indexes. The obtained coding quality improvement makes the proposed method rather adviseable.

1 Introduction

Camera noise is a disturbance intrinsic to camera operation since it is generated by the light sensors and the amplification circuits. Not only is it a disturbing presence both with respect to high quality rendition but it can also generate artifacts when the noisy images are coded using a block based coder, such as MPEG2 or H263. In this case the presence of noise may cause mainly three drawbacks. First, the coder will tend to randomly choose between "compensated" and "non-compensated" blocks on uniform regions. As a consequence, the "blocking effect," typical of DCT, coders is increased. Second, the motion field of the compensated blocks is more irregular and this increases the number of bits required to code the motion vectors. Third, the DCT coefficients are altered by the presence of the broadband noise and, as a consequence, the coding gain decreases.

For these reasons, noise reduction before video coding is highly desirable. Linear filtering techniques are not suitable to this task because they would necessarily damage the details. Furthermore, the statistical characteristics of camera noise show that it is not of gaussian nature, hence simple averaging procedures are not optimal for its treatment [1, 5].

This work examines the impact on coding of three prefilters characterized by the use of different *non linear* filters applied to the pixels of a spatially adaptive region. These filters have been considered for their simplicity and their effectiveness demonstrated by the experimental results.

2 Proposed techniques

Camera noise is especially visible on uniform areas [1]. In these areas it is relatively easy to distinguish between signal and noise, therefore appropriate actions can be taken. Clearly, the noise is present also on detailed areas, not differently than in the rest of the scene. However, in these parts its removal without any damage to the detail is impossible, owing it to the lack of a plausible signal model. Luckily, granular noise is masked by the detail and its presence on finely textured regions is less noticeable.

These considerations suggest one to filter only the uniform regions, of either small size (as regions in between details) or of larger size. In this respect, camera noise reduction poses a double task, namely the recognition of the uniform areas (where the signal can be safely assumed constant) and an appropriate noise removal action.

Let $W(\mathbf{n}, N_1, N_2)$ denote a window of a field of dimension N_1 and N_2 centered at pixel $\mathbf{n} = (n_1, n_2)$ and $y(\mathbf{n})$ the luminance value on the noisy image of pixel $\mathbf{n}$. With the first filter proposed, denoted as technique 1 in the following, the output $v(\mathbf{n})$ is obtained as

$$v(\mathbf{n}) = \begin{cases} y(\mathbf{n}) & \text{if } D(\mathbf{n}) > T(\mathbf{n}) \\ \text{median}\{y(\mathbf{k}),\ \mathbf{k} \in W(\mathbf{n}, N_1, N_2)\} & \text{if } D(\mathbf{n}) \leq T(\mathbf{n}) \end{cases} \tag{1}$$

where $D(\mathbf{n})$ is a measure of the activity of the block $W(\mathbf{n}, N_1, N_2)$ and $T(\mathbf{n})$ is a local estimation of the noise variance.

The threshold $T(\mathbf{n})$ is adapted to the luminance values $y(\mathbf{n})$ according to the following rule

$$T(\mathbf{n}) = \begin{cases} T_1 & \text{if } y(\mathbf{n}) < 100 \\ T_2 & \text{if } 100 \leq y(\mathbf{n}) \leq 150 \\ T_3 & \text{if } y(\mathbf{n}) > 150 \end{cases} \tag{2}$$

where $T_1 = 20$, $T_2 = 10$ and $T_3 = 3$ (the digital luminance values range from 16 to 234). Rule (2) is motivated by a number of reasons. A first reason is the observed dependence of the granular noise variance σ_n^2 from the luminance values [1]. Such a dependence can be approximated by $\sigma_n^2 = \sigma^2 / y(\mathbf{n})^{1.1}$, where σ^2 is the noise variance before gamma-correction. The pice-wise constant approximation of a continuous dependence of $T(\mathbf{n})$ from the values of $y(\mathbf{n})$, given by (2) was experimentally found satisfactory in this context. A second motivation is the limited visibility of granular noise at high luminance values.

The measure of activity $D(\mathbf{n})$ is computed by ordering the luminance values of the pixels of the window $W(\mathbf{n}, N_1, N_2)\}$ into the sequence $p(1)$, $p(2), \ldots,$ $p(N_1 N_2)$ and defining $D(\mathbf{n}) = p(N_1 N_2 - 1) - p(2)$.

The idea behind (1) is clearly the desire of eliminating the transients near edges and of not damaging finely textured regions, typically associated to high $D(\mathbf{n})$ values.

It should be noted that large $N_1 \times N_2$ values would improve noise smoothing on uniform areas, but would also leave the noise unaffected in between details. The best trade-off was obtained by a 5×3 window. The fact that N_1 is different than N_2 is due to the interlaced nature of the signal.

The on/off filtering of technique 1, although simple to implement and quite appropriate in the transitions between uniform regions of different luminance, presents undesirable artifacts on the regions of moderate scene activity. In these regions the artifacts originated by the passages between the two operation modes become visible, especially if compared with the results of the other techniques examined in the following.

The following adaption of the technique originally proposed in [2], hereafter referred to as technique 2, can potentially overcome such a problem. An adaptive centered weighted median (ACWM) filter is used, the output is computed as

$$v(\mathbf{n}) = \text{median}\{y(\mathbf{k}),\ \mathbf{k} \in W(\mathbf{n}, N_1, N_2);\ 2M(\mathbf{n}) \text{ copies of } y(\mathbf{n})\} \tag{3}$$

where $M(\mathbf{n})$ is an integer computed by means of an adaption rule. Expression (3) indicates that $v(\mathbf{n})$ is the median of the luminance values of the window pixels and $2M(\mathbf{n})$ copies of $y(\mathbf{n})$. Hence input sample $y(\mathbf{n})$ is weighted $2M(\mathbf{n})$ times more than the other samples of $W(\mathbf{n}, N_1, N_2)$ with respect to the median.

It is important to note that (3) can be simply computed [2] as

$$v(\mathbf{n}) = \text{median}\{p(L + 1 - M(\mathbf{n})), y(\mathbf{n}), p(L + 1 + M(\mathbf{n}))\} \tag{4}$$

where L is such that $2L + 1 = N_1N_2$ and $p(i)$, $i = 1, 2, \ldots, N$ denotes the i-th ordered luminance value of the pixels of window $W(\mathbf{n}, N_1, N_2)$. Expression (4) makes clear that if $M(\mathbf{n}) = L$, $v(\mathbf{n}) = y(\mathbf{n})$, that is, the ACWM filter reduces to the identity operator, instead if $M(\mathbf{n}) = 0$, the ACWM filter becomes a simple median filter (such a fact is immediate from (3)). Obviously the use of the identity operator is well suited to the detailed areas, and the median is suited to the uniform areas. As $M(\mathbf{n})$ varies between 0 and L, the ACWM filter output continuosly varies between the above two extremal situations. Such a remarkable capability is directly controlled by the value of $M(\mathbf{n})$. The rule for chosing $M(\mathbf{n})$ at location $\mathbf{n}$ is

$$M(\mathbf{n}) = \lfloor LR(\mathbf{n}) \rfloor \tag{5}$$

with

$$R(\mathbf{n}) = \begin{cases} 1 - \frac{T(\mathbf{n})}{\sigma_S^2(\mathbf{n})} & \text{if } \sigma_S^2(\mathbf{n}) > T(\mathbf{n}) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\lfloor \cdot \rfloor$ denotes the integer part of the argument, $\sigma_S^2(\mathbf{n})$ is the signal variance on $W(\mathbf{n}, N_1, N_2)$ and $T(\mathbf{n})$ is the threshold determined from (2). According to (5) and (6), $M(\mathbf{n})$ assumes integer values between 0 and L, as $R(\mathbf{n})$ varies between 0 and 1.

The attempt of suiting the filtering action to the local characteristics of the image can be more effectively pursued by locally varying also the shape of the filter support as described in [3, 4, 5]. The last considered technique (technique 3) uses the approach of [4] for granular noise reduction. The method prescribes the computation of average $m_i(\mathbf{n})$ and variance $\sigma_i^2(\mathbf{n})$ along the four principal directions of a square window $W(\mathbf{n}, M, M)$ centered at the input pixel. The pixels entering the computation belong to one-pixel thick stripes lying along each direction. The minimum activity direction associated to the minimum variance $\sigma^2(\mathbf{n}) = \min_i \sigma_i^2(\mathbf{n})$ is determined. The output is obtained by an ACWM filter whose support is chosen to be the L_1 pixels thick stripe oriented along the minimum activity direction. The value L_1 is adaptive and it is determined by the rule

$$L_1 = 2\lfloor (1 - \alpha)L \rfloor + 1 \tag{7}$$

	Tech. 1	Tech. 2	Tech. 3
PSNR noisy seq.,whole image	32.7	32.7	32.7
PSNR filtered seq.,whole image	32.8	33.4	33.5
# of filtered pixels %	2.4 %	11.4 %	14.2 %
PSNR noisy seq., filtered region only	34.7	30.2	30.7
PSNR filtered seq., filtered region only	41.0	35.2	35.4

Table 1: Noise attenuation of the proposed filter

where $L = (M^2 - 1)/2$ and α is an anisotropy parameter determined as

$$\alpha = \frac{\max_i\{m_i\} - \min_i\{m_i\}}{\max_i\{m_i\} + \min_i\{m_i\} + \varepsilon} \tag{8}$$

where ε is a small positive quantity, preventing division by zero.

3 Experimental Results

To evaluate the performance of the proposed prefilters, in particular the noise attenuation in uniform regions, it is necessary to identify the noise on the input sequence. As a preliminary step, we used a computer generated noise and we added it to Y, U and V components of the test sequence "Calendar", supposed "noise free". The granular noise was generated according to the statistical description given in [1] assuming a variance of 35.

In table 1 the noise attenuation of the proposed filters is compared. The results are expressed as values of PSNR (Peak Signal to Noise Ratio), that is

$$\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\mathrm{MSE}}$$

where MSE is the mean squared error with respect to the noise free image. In the second line of table 1 the PSNR of the filtered images is reported while on line three the percentage of filtered pixels is given. As it could be expected, the anisotropic technique acts on the highest number of pixels and gives the best results in terms of noise attenuation. Technique 2 gives values that are slightly smaller with respect to those of technique 3. On the other hand, technique 1 allows only a little attenuation of the added noise and acts on the smallest number of pixels. In the third and fourth line of table 1 the PSNR for the noisy and the filtered sequences is given, this time computed only *on the filtered regions*. It can be seen that all the three techniques give a gain of PSNR of about 5 dB even if this gain is relative to regions of different area.

The impact on coding of the use of the proposed filters is evaluated using an MPEG2 coder at 4 Mbit/s on 96 frames of the sequence. In table 2 some average results are reported.

As it can be expected the filtered sequence allows to improve the coding steps with respect to the noisy sequence. Namely, the number of motion compensated blocks is increased reducing the blocking effects at low bit rates.

	Orig.	Noisy	Tech. 1	Tech. 2	Tech. 3
# of intra blocks (P)	4.67	13.0	12.83	8.83	8.5
# of intra blocks (B)	0.19	1.25	1.62	0.76	0.75
# of bits for MV (P)	13635.0	17787.2	17810.2	17361.2	17384.5
# of bits for MV (B)	25779.3	31333.1	31420.7	30525.6	30719.8
Coding gain (I) dB	9.451	8.880	8.887	8.970	8.982
Coding gain (P) dB	0.860	0.602	0.604	0.636	0.641
Coding gain (B) dB	0.537	0.357	0.359	0.377	0.379
PSNR whole image	28.12	27.84	27.85	27.97	27.98
PSNR noisy seq., fil. reg.			34.64	33.47	33.22
PSNR fil. seq, fil. reg.			34.88	34.19	33.85

Table 2: Coding results for the original, the noisy and the filtered sequences

Moreover, the number of bits used to code the motion vectors decreases for the filtered sequences since the motion field is more regular. Also, the prefilters are effective to obtain a higher coding gain and PSNR. Last but not least, the visual quality improvement is apparent and superior to what could be expected on the basis of PSNR. Indeed, these quantities are computed on the data of the whole sequence while the human visual system is more sensitive to noise on the uniform regions where the filters really act.

The previous results, based on synthetic additive noise, have been confirmed on the real sequences: “Voiture,” “Renata,” “Table tennis,” and “Flower garden.” In this case subjective tests were carried out according to the CCIR 500-3 recommendation and clearly showed the effectiveness of the noise reduction techniques proposed. In the figures 1 and 2 two examples for the sequence “Renata” and “Voiture” are shown. For each picture the top half was obtained coding the original sequence at 4 Mbit/s while the bottom half was obtained applying the anisotropic technique before the coding stage. It is apparent that the details obtained from the prefiltered sequence show less blocking effect on the uniform regions with respect to the original sequence confirming the effectiveness of the proposed methods.

References

[1] G. Cortelazzo, G. A. Mian, R. Parolari, “Statistical characteristics of granular camera noise,” *IEEE Trans. on Circuits and Systems for Video Technology,* vol. 4, no. 6, pp. 536-543, Dec. 1994.

[2] Sung-Jea Koo, Yong Hoon Lee, “Nonlinear spatio-temporal noise suppression techniques with application in image sequence processing,” *Proceedings of IEEE ISCAS*, pp. 662-665, Singapore, 1991.

[3] P. Zamperoni, “An automatic low-level segmentation procedure for remote sensing images,” *Multidimensional Systems and Signal Processing*, 3, pp. 29-44, 1992.

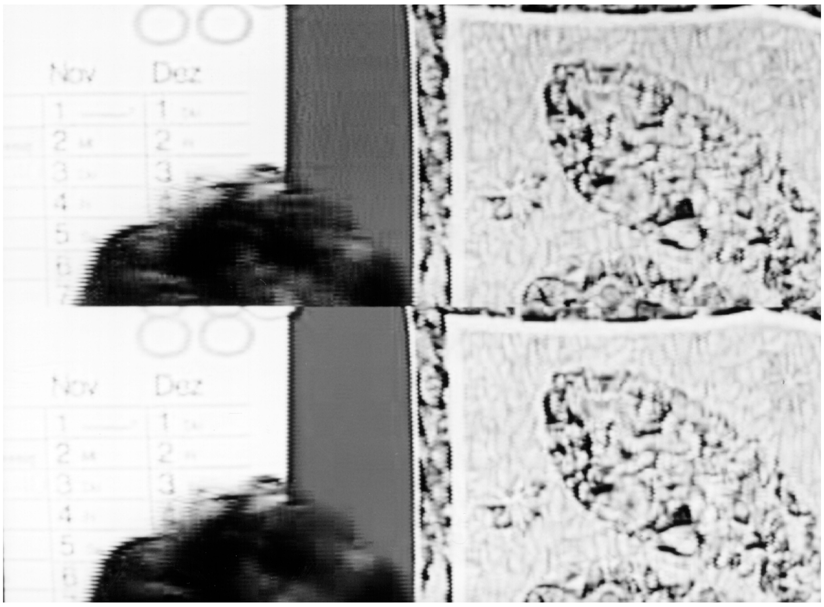

Figure 1: A region of the sequence "Renata" coded at 4 Mbit/s (top half). The same region coded using the anisotropic prefilter (bottom half).

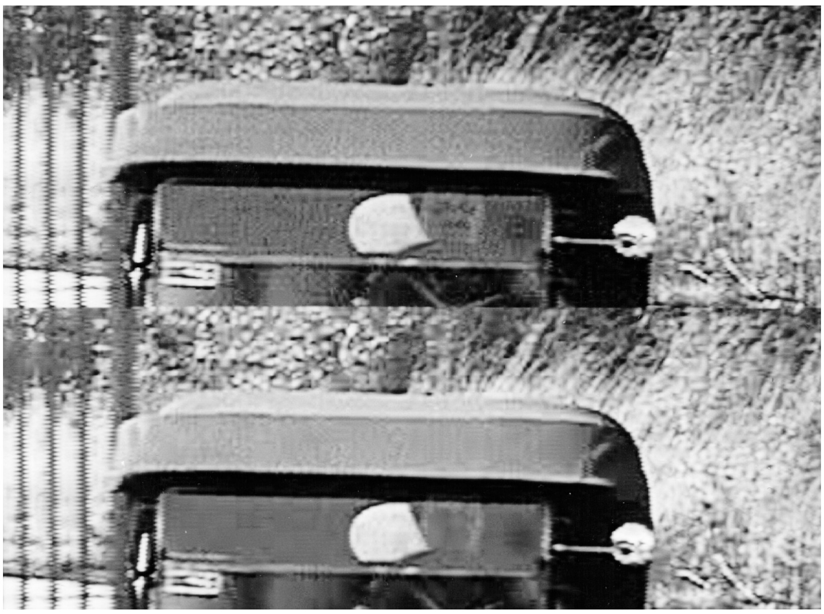

Figure 2: A region of the sequence "Voiture" coded at 4 Mbit/s (top half). The same region coded using the anisotropic prefilter (bottom half).

[4] P. Zamperoni, "Adaptive rank-order filters for image processing based on local anisotropy measures," *Digital Signal Processing*, 2, 174-182, 1992.

[5] G. Cortelazzo, F. Guindani, G. A. Mian, R. Picco, "Anisotropic filtering for granular noise reduction", *Proc. IEEE Workshop on Nonlinear Digital Signal Processing*, 5.1 (2.1-4), January 1993, Tampere (Finland).