Proceedings of ACIVS 2002 (Advanced Concepts for Intelligent Vision Systems), Ghent, Belgium, September 9-11, 2002

ADAPTIVE TEMPORAL FILTERING FOR CFA VIDEO SEQUENCES

[1]*Angelo Bosco*, [1]*Massimo Mancuso*, [1]*Sebastiano Battiato and* [1]*Giuseppe Spampinato*

[1]Angelo.Bosco@st.com
[1]STMicroelectronics, AST Catania Lab, Stradale Primosole, 50 – 95121 Catania, Italy

ABSTRACT

This paper describes a new approach for video sequences noise reduction that directly processes raw data frames acquired by an image sensor. The proposed noise reduction filter works on CFA-(Color Filter Array) Bayer matrixed video data instead of the canonical *YUV* format. Working on CFA raw frames allows saving resources in terms of time and space; this is particularly relevant for real time processing. Noise level is continuously monitored in order to modify the filter strength adaptively. Experiments demonstrate the effectiveness of the proposed method.

1. INTRODUCTION

Quality enhancement algorithms for still pictures and video are demanded to boost digital capturing devices [1][2][3][4][5]. Speed-up of the processing time is also a key factor. The noise removal process is an essential element of the video-processing pipeline; it aims to increase pictures quality usually by means of adaptive recursive temporal filters. A video sequence is constituted by a succession of frames that are temporally correlated. To properly remove noise, a video-filter must be able to use both spatial and temporal data. In frame areas where motion occurs, the noise reduction algorithm should give higher importance to spatial information, neglecting or decreasing the weight of pixels provided by adjacent frames. Generally, temporal video noise reduction filters work on image sequences in *YUV* space and can be partitioned into four main categories [6]:

i) Non Motion Compensated Spatio-Temporal Filters
ii) Motion Compensated Spatio-Temporal Filters
iii) Non Motion Compensated Temporal Filters
iv) Motion Compensated Temporal Filters

The proposed filter fits in the category of *Non Motion Compensated Spatio-Temporal Filters*. The avoidance of motion compensation is reasoned by high performance and real time processing constraints. No two consecutive video-frames are exactly the same, even in the absence of motion. Pixel values fluctuate because of the noise that usually originates from the digital sensor. Filters coping with noisy video sequences, cannot process every single frame

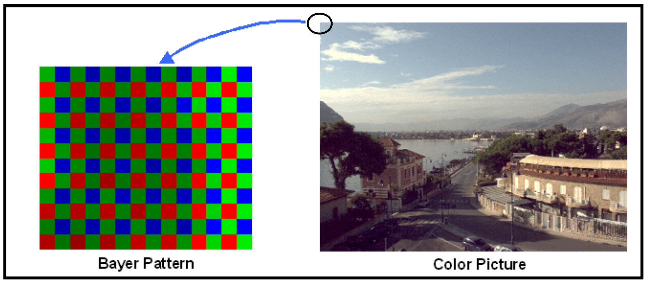

Figure 1. Bayer Pattern from a color interpolated frame.

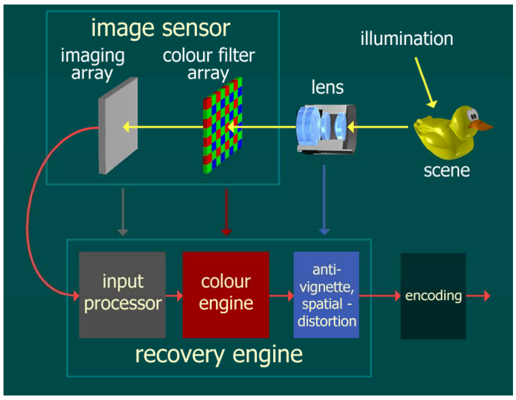

Figure 2. Image Processing Pipeline.

independently; by doing so, they would generate artifacts, as the residual noise at each pixel location would oscillate from frame to frame. The resulting video sequence would be affected by flickering when played-back. Quality enhancement of video images degraded by noise requires that temporal correlation has to be exploited. The following sections describe a video-filter capable to remove noise from a video sequence composed of raw data frames acquired by Bayer matrixed image sensors [7]. These are grayscale sensors covered by Color Filter Array (CFA) to enable color sensitivity (Fig.1 and 2). Using a CFA, each cell of the sensor array is sensible to only one color component. The final color image is obtained by means of a color reconstruction algorithm that combines the color information of neighboring pixels [1][8][9][10]. Instead of processing the usual YUV video format, the filter is posi-

tioned at the beginning of the image-processing pipeline. At this stage, the frame color information is available but not yet interpolated from the Bayer pattern. The other image processing pipeline algorithms will make use of denoised data, improving system performances and image quality[11][12].
Figure 1 shows a single frame and a part of its corresponding Bayer pattern. A typical image-processing pipeline is illustrated in Fig.2. The paper is organized as follows. Section 2 illustrates the proposed filter. Section 3 explains how the effective filtering process takes place. In section 4 the noise level estimation process is described. In section 5, experimental results are presented and discussed. Finally, a conclusion section closes the paper.

2. SPATIO-TEMPORAL FILTER

Two square 5x5 working windows constitute the proposed spatio-temporal filter support; these masks exploit information from the current and the previous filtered frame. Depending on the color of the current pixel, either green or red/blue, the proper pixels of the same color are selected from the working windows, discarding the others [11]. Moreover, the number of pixels involved in the noise reduction process may be restricted. This limitation is due to the artifacts (e.g. motion blurring) that may appear when motion is present in the scene. If the area under processing contains motion, a temporal filter must decrease the information provided by frames farthest from the current one. The proposed filtering process is based on Duncan Filtering (*DF*) as shown in [13]. This filter deals with video sequences affected by zero mean additive white gaussian noise (AWGN) of the form (1), see [14]:

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad (1)$$

In such a statistical distribution, the following rules are valid:

i) 68% of the samples fall in the range [$\mu-\sigma$, $\mu+\sigma$]
ii) 95% of the samples fall in the range [$\mu-2\sigma$, $\mu+2\sigma$]
iii) 99.7% of the samples fall in the range [$\mu-3\sigma$, $\mu+3\sigma$]

These percentages prove that a relatively high number of pixels values belong to the tails of the gaussian distribution noise curve. The *DF* method ensures that even if the working windows include pixels whose value is completely out of range, the noise reduction process continues to work properly. Out-of-range pixels are automatically excluded from being processed. The filter is capable to correctly handle the case in which the current pixel or one or

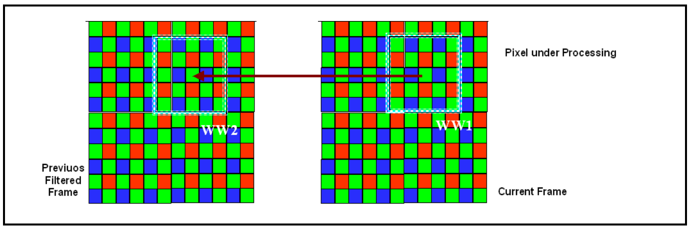

Figure 3. Previous and current frame with corresponding working windows (WW2, WW1).

more of the surrounding ones are outliers. These pixel values can appear because of the noise superimposed on the video sequence, e.g. salt and pepper noise that turns some pixels black or white with some low probability. The *DF* method is specifically designed to address the aforementioned issues and provides a robust solution.
The non-motion compensated approach requires taking some action as to avoid motion artifacts. To correctly filter a video sequence, one should track the moving objects from frame to frame. When filtering a specific pixel in the current frame, a motion compensated temporal filter seeks for the pixel in the previous frame that exactly matches the one under processing; but the current pixel and the matching pixel might be located at different coordinates. Motion vectors are used to align the moving parts in the sequence so that temporal recursive filtering can correctly use the information from two (or more) frames. However, the heavy computations for the determination of motion vectors must be avoided if a low cost implementation is demanded. An alternative solution to motion compensation is motion detection. Specifically, a ghost-tail detection [15] and a motion detection routine have been designed in order to lessen the importance of the adjacent frame if motion occurs. Spatial filtering generally reduces image details, but on moving objects the degradation is not perceptible as for stills. Let WW1 be the filter mask in the current frame and WW2 the filter mask in the previous frame (Fig.3). The mask WW2 must be centered on the same pixel in which WW1 is centered; but, if motion occurred between frames, this assumption may not be true because WW2 might be centered on a pixel that represents a different object or the background. The absence of motion compensation does not allow locating the correct position of WW2 in the previous frame that matches WW1 in the current frame. Under the assumption that the luminance changes monotonically in the video sequence, the ghost-tail detection routine performs the differences between the pixels of the filter masks in the two frames. If these differences have all the same sign, either positive or negative, then a possible artifact is detected: in the filtered video the moving objects will leave a ghost-tail while moving. Similarly, to enforce the previous ghost-artifact detection routine, a second motion detection block computes the SAD (Sum Of The Absolute Differences) between the working windows.

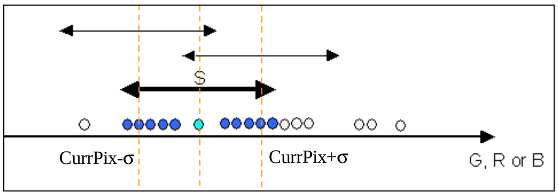

Figure 4. Duncan Filtering principle.

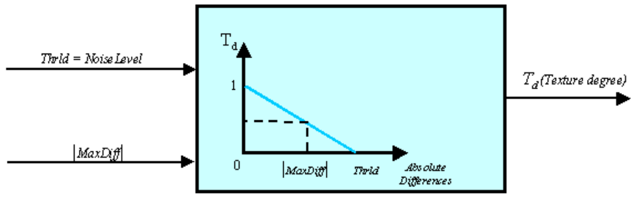

Figure 5. Texture-degree analyzer.

If the SAD is greater than a predefined threshold, then motion is detected. When the artifact-detecting routines provide a positive response, the filter switches to spatial mode only or, eventually, it lowers the weight of the previous frame. To avoid that the SAD value is too sensitive to gray-scale values, the pixel values can be quantized. The motion detection block can be further refined avoiding that scene illumination changes may cause different lighting conditions of two frames to be misinterpreted as motion. To solve this problem, a variation of the SAD method can be adopted [16]. Specifically, the absolute difference between the working windows is computed. The average value of the resulting difference frame is determined. The average value is then subtracted from the difference frame. The SAD of the resulting mean reduced frame is finally computed. This strategy compensates for different illumination conditions in two adjacent frames. If the computed SAD is greater then a predefined experimentally determined threshold then more importance is given to the current working window WW1, whilst WW2 is weighted with less or no importance.

3. DUNCAN FILTERING

The effective noise removal process is based on *Duncan Filtering* (*DF*) [13]. The filter uses two working windows; one in the current frame, the other in the previous filtered one. Depending on the color of the current pixel, either green or red/blue, the appropriate working window masks are chosen. Some issues may come up after the construction of the working windows. First of all, it is possible that a few out-of-range pixel values are introduced; furthermore it is important to choose pixels that are part of the same object to which the current pixel belongs. The *DF* method is expressly intended to properly treat the aforementioned problems. Let *CurrPix*, be the value of the current pixel under processing; the filter also considers the values *CurrPix*+σ and *CurrPix*-σ, thus it takes into account what happens if the current pixel value oscillates from its current value up to reach the extreme values that it can assume when the noise standard deviation is considered. In other words, the filter is capable to handle the case in which the current pixel is also a noisy value. Three ranges, whose wideness S is directly related to σ are considered (Fig. 4). Higher σ values yield wider S ranges. One range is centered upon the current pixel and the other two at distance +/- σ from it. The range that maximizes the number of pixels is selected. Depending on the range being selected, a weighting coefficient α is determined for every pixel in the range. Finally, the filtered value is a weighted average given by (2):

$$FVal = \frac{\sum_i \alpha_i X_i}{\sum_i \alpha_i} \qquad (2)$$

where X_i represents the value of the pixel indexed with i in the filter mask; α_i are weighting coefficients.

4. NOISE LEVEL ESTIMATION

Estimation of the noise level is essential in order to control the degree of filtering. *Duncan Filtering* is strictly dependent on the standard deviation of the noise that is superimposed on the video sequence. This parameter commonly denoted with σ, controls the filter strength. The choice to tune the filter power may be left to the user that, directly or not, may set the σ value. However, an overestimation or underestimation of this value will lead to poor image quality results. Therefore, it is valuable to have an automatic tool for the estimation of the noise standard deviation [17]. Higher σ yields stronger filtering. To determine the σ value, the flat areas of a frame must be detected; in these areas the fluctuations of pixel values are supposed to be caused exclusively by random noise. A texture analyzer block performs the identification of flat areas; it inspects the local characteristics of a frame and computes a coefficient for each image portion that represents an estimation of its texture-degree. When the coefficient is close to 1, the area is almost uniform, conversely, when it is far from 1 and approaches zero, the area must be considered increasingly textured. In order to run the texture analyzer, a *S*patial *N*oise *L*evel (*SNL*) estimation is also required. This is necessary, as we do not want local spatial noise to interfere with the texture degree estimation process. The Texture-Degree analyzer block (Fig.5) takes as input the previous

SNL estimation and the maximum of the absolute differences between the current pixel and its neighbors. The threshold *Thrld* is computed according to the *SNL* [5][11]. Finally the block outputs the texture-degree coefficient T_d that determines if the area under processing is suitable to retrieve statistical information about noise. We start with an arbitrary estimation of the spatial noise level *SNL* (e.g. *SNL* = 0); then the estimation is updated according to (3):

$$SNL = f(T_d(t-1), MaxDiff, SNL(t-2)) \quad (3)$$

Using this texture measurement method, the flat areas of the image can be identified and in these areas the variance or any other form of *energy* can be computed.
The standard deviation of a region *R* in a frame *f(x,y)*, is estimated according to equation (4):

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{(x,y)\in R}\left(f(x,y) - average\right)^2} \quad (4)$$

Where *N* is the number of samples in the region *R* and the *average* is computed over the same region. The value of σ sets the filter strength to the appropriate level. However, the randomness of the superimposed noise could generate variance values having an oscillatory behavior; this would generate flickering when the video is played-back. To avoid this annoying artifact, the noise level σ is averaged between the current and the previous computed σ values. The use of an averaged σ will smooth the temporal filtering process and flickering will be avoided (Fig.9). The energy is computed over the flat areas detected by the texture analyzer block. The average energy accumulated during the frame analysis regulates the filter strength.

5. EXPERIMENTAL RESULTS

The noisy Bayer pattern and the previous filtered one constitute the input frames of the algorithm. The successive algorithms of the graphic pipeline will process denoised data, increasing the performances and the quality (Fig. 6). The effectiveness of the texture analyzer is depicted in Fig. 7. The visual quality of the filter is illustrated in Figg. 8 and 12.The methodology used to test and assess the quality of the filtering process can be easily understood by looking at Fig. 10. A sample YUV 4:2:0 original color sequence was converted in the RGB color space. The Bayer pattern of each frame was generated, producing a sequence of noisy free bayer frames. A noise generator was used to add artificial noise to each bayer frame. Finally the clean color sequence, along with its noisy and filtered counterparts was generated. Peak to signal Noise Ratio (PSNR) was used as a measure to assess picture quality. The experiments show

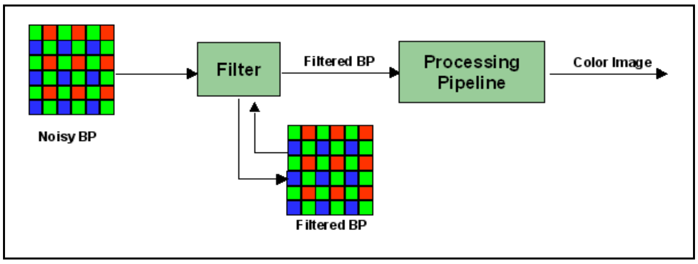

Figure 6. Filtering process overall scheme.

Figure 7. On the left: input Bayer pattern. On the right: bright areas represent the uniform regions detected by the texture analyzer. These regions are suitable to retrieve statistical information about noise.

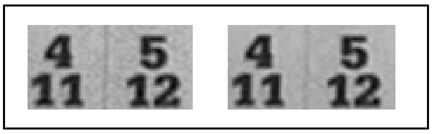

Figure 8. A particular from a noisy (left) and a denoised frame (right).

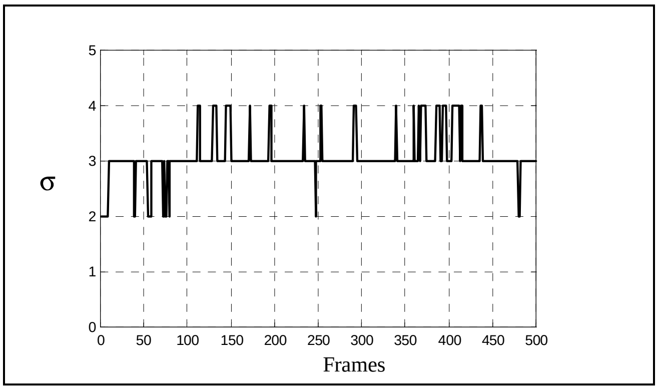

Figure 9. Detected noise standard deviation in a sample sequence composed of 500 frames.

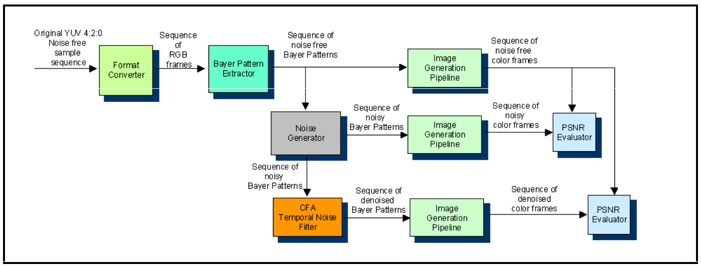

Figure 10. Quality testing methodology.

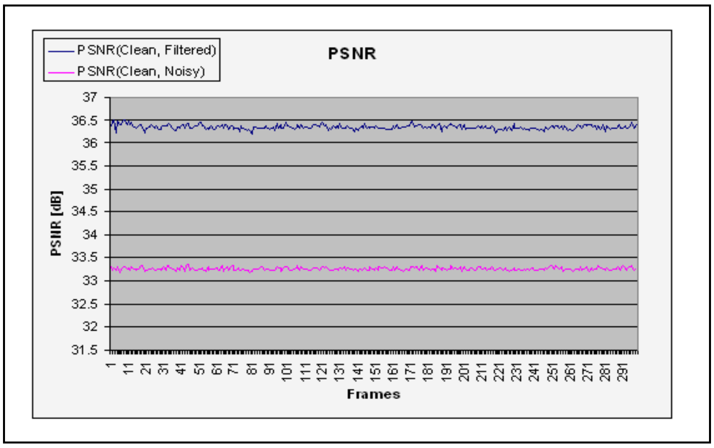

Figure 11. Measured PSNR between clean and noise sequence (bottom line), and between clean and filtered sequence (upper line). No color artifacts are introduced after processing.

that the filtered sequence exhibits a better PSNR, typically the gain is about 3dB (Fig.11).

6. CONCLUSIONS

CFA video sequences can be denoised by means of the proposed method. The input consists of two frames (the noisy frame and the previous filtered one) and outputs a denoised Bayer matrixed frame. Motion compensation is not performed in order to increase real time performances; in alternative, the filter exploits a series of strategies that include a ghost-tail discover mechanism, which detects possible artifacts due to motion in the scene, a motion detection routine based on the computation of SAD and a modified version of the SAD that compensates for different illumination conditions occurring between successive frames. A noise estimator identifies the flat areas in a frame and uses them to determine the statistical properties of the random additive noise; this block controls the degree of filtering depending on the characteristics of the superimposed noise. The filtering process is based on the *DF* method, which is robust with respect to out of range values (e.g. salt and pepper noise) and that further enforces the

(a)

(b)

Figure 12. One frame from a test sequence. (a) Original frame contaminated with AWGN with standard deviation σ=5. (b) Corresponding filtered frame.

working windows to contain pixels of the same object to which the current pixel belongs. The algorithm is suitable for real time processing of video sequences acquired by typical consumer devices. Future works include comparison with other kinds of filtering (e.g. weighted order statistics).

REFERENCES

[1] S. Battiato, M. Mancuso, "*An Introduction to the Digital Still Camera Technology"*, ST Journal of System Research - Special Issue on *Image Processing for Digital Still Camera*, Vol. 2, No.2, December 2001.

[2] S. Battiato, A. Castorina, M. Mancuso, "*High Dynamic Range Imaging for Digital Still Camera"*, In Proceedings of SPIE *Electronic Imaging 2002* - Sensors, Cameras, and Applications for Digital Photography IV - San Jose' CA USA, January 2002.

[3] A. Bruna, M. Mancuso, "*JPEG Compression Factor Control: A New Algorithm"*, in Proceedings of IEEE ICCE2001, pp.206-207, Los Angeles, June 2001.

[4] S. Battiato, M. Mancuso, A. Bosco, M. Guarnera - "*Psychovisual and Statistical Optimization of Quantization Tables for DCT Compression Engines*" - In IEEE Proceedings of International Conference on Image Analysis and Processing ICIAP 2001 - Palermo, Italy, pp. 602-606 - September 2001

[5] A. Bosco - *"Adaptive Image Denoising On Bayer Pattern"* - ST Journal of System Research. Vol.2, No.2, Dec.2001.

[6] J.Brailean, R.Kleihorst, S.Efstratiadis, A. Katsaggelos, R. Lagendijk, "*Noise Reduction Filters for Dynamic Image Sequence: A Review*", Proceedings of the IEEE, Vol.83, No.9, Sept. 1995.

[7] B.E. Bayer, "*Color Imaging Array*", U.S. Patent 3,971,065-1976.

[8] R. Kimmel, "*Demosaicing image Reconstruction from color CCD samples", IEEE Transaction on Image Processing,* Vol. 7, No. 3, 1999.

[9] R. Ramanath, "*Interpolation methods for the Bayer color Array*", *MS Thesis, North Carolina State University, 2000.*

[10] H. J. Trussel, "*Mathematics for Demosaicing*", To appear in *IEEE Transanction on Image Processing,* 2001.

[11] A. Bosco, M. Mancuso, "*Adaptive Filtering For Image Denoising*", in IEEE Proceedings of ICCE2001, pp.208-209, June 2001.

[12] O. Kalevo, H.Rantanen, "*Noise Reduction Techniques for Bayer-Matrix Images*", In Proceedings of SPIE *Electronic Imaging 2002* - Sensors, Cameras, and Applications for Digital Photography IV - San Jose' CA USA, January 2002.

[13] A. M. Borneo, L. Salinari, D. Sirtori, "*An Innovative Adaptive Noise Reduction Filter for Moving Pictures Based on Modified Duncan Range Test*", ST Journal of System Research, Vol.1, No.1, 2001.

[14] G.Cortellazzo, G.A.Mian, R.Parolari, "*Statistical Characteristics of Granular Camera Noise*", IEEE Transactions On Circuits and Systems For Video Technology, Vol.4, No.6, December 1994.

[15] S.Inamori,S.Yamauchi,K.Fukuhara, "*A Method of Noise Reduction on Image Processing*", IEEE Transactions on Consumer Electronics, Vol.39, No.4, 1993.

[16] Ching-Kai Huang, Tsuhan Chen, "*Motion Activated Video Surveillance Using TI DSP*".DSPS FEST '99, Houston Texas, 1999.

[17] S.I.Olsem, "*Noise Variance Estimation in Images*" in Proceedings of the 8th Scandinavian Conference on Image Analysis, Troms∅, Norway 1993.