Combined Wavelet-Domain and Motion-Compensated Video Denoising Based on Video Codec Motion Estimation Methods

Ljubomir Jovanov, *Member, IEEE*, Aleksandra Pižurica, *Member, IEEE*, Stefan Schulte, Peter Schelkens, *Member, IEEE*, Adrian Munteanu, Etienne Kerre, and Wilfried Philips, *Member, IEEE*

Abstract—Integrating video coding and denoising is a novel processing paradigm, bringing mutual benefits to both video processing tools. In this paper, we propose a novel video denoising approach of which the main idea is reusing motion estimation resources from the video coding module for video denoising. In most cases, the motion fields produced by real-time video codecs cannot be directly employed in video denoising, since they, as opposed to noise filters, tolerate errors in the motion field. In order to solve this problem, we propose a novel motion-field filtering step that refines the accuracy of the motion estimates to a degree that is required for denoising. Additionally, a novel temporal filter is proposed that is robust against errors in the estimated motion field. Numerical results demonstrate that the proposed denoising scheme is of low-complexity and compares favorably to the state-of-the-art video denoising methods.

Index Terms—Image enhancement, motion estimation, noise, video coding, video signal processing, wavelet transform.

I. INTRODUCTION

I N MANY applications such as video surveillance or tele-medicine, it would be beneficial to integrate video denoising and video coding as closely related parts of the same video processing chain. Indeed, noise in the video sequences increases image entropy, thereby reducing the effective video compression performance. This problem is largely solved by introducing a noise reduction step prior to encoding.

Wavelet-based video encoders [1]–[3] have proven their advantages, ensuring support for quality, resolution and temporal scalability, while yielding a compression performance on par with that of the state-of-the-art non-scalable H.264-codec. Some of the recent, best performing wavelet-domain video denoisers include [4]–[7]. With the exception of our initial work [8], to our knowledge, there are no other processing paradigms reported in the literature that integrate video denoising and video coding.

Manuscript received September 04, 2007; revised December 13, 2007. First published February 03, 2009; current version published March 20, 2009. This work was supported by the Fund for the Scientific research (FWO) Flanders, Belgium, through Project G.0369.05N. This paper was recommended by Associate Editor X. Li.

L. Jovanov and W. Philips are with the UGent-TELIN-IPI-IBBT, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium (e-mail: ljj@telin.ugent.be).

A. Pižurica is with the UGent-TELIN-IPI-IBBT, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium and also with the Fund for the Scientific Research (FWO), Flanders, Belgium.

S. Schulte and E. Kerre are with the Fuzziness and Uncertainty Modelling Research Group, Ghent University, Krijgslaan 281-S9, B-9000 Gent, Belgium.

P. Schelkens and A. Munteanu are with the Fund for the Scientific Research (FWO), Flanders, Belgium.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2009.2013491



Fig. 1. Motion compensated temporal filtering followed by adaptive spatial wavelet filtering.

A natural step towards integrating video denoising and coding modules is reusing resources, such as motion estimation, of the video codec for the denoiser. Often, the motion fields produced by real-time video codecs do not capture the actual object trajectories, and as such, cannot be applied directly for motion-compensated denoising. In this paper, we extend our initial work [7] and develop an efficient approach that integrates a motion estimator from a video codec into a video denoiser. In particular, we observe that inaccuracies of multiresolution motion estimators such as [9] consist mainly of false motion vectors in background image areas without actual motion. We introduce a motion field filtering step that refines the accuracy of the motion field making it usable for denoising. The essence is that the output of the same motion estimator is used as an input for the coding scheme, and with the proposed filtering step, as an input to the denoiser.

Here we assume that the video sequences are contaminated with the additive white Gaussian noise, with zero mean and known variance σ . The results demonstrate that the proposed approach competes with the best and most recent multiresolution video denoisers, such as [5], [6] and [10]. The proposed denoising approach is based on spatio-temporal filtering [7], which combines wavelet-domain spatial filtering which is preceded by pixel-domain temporal filtering. We propose a novel motion compensated temporal filter, as opposed to the simple pixel-based motion detection technique employed in [7], which brings a significant improvement in the denoising quality. Here we proposed the scheme shown in Fig. 1, where temporal filtering precedes spatial, as opposed to [7]. In this case remaining noise after the temporal filter is spatially non-stationary, which requires adaptivity of the spatial filter to the local noise statistics. Different solutions for this problem are possible, including the use of the wavelet threshold which depend on a local noise variance. Here we extend the fuzzy-logic filter [11], which is attractive for hardware implementation, by making it adaptive to a localy estimated noise variance.

This paper is organized as follows. The proposed approach is presented in Section II, where we introduce first the proposed motion field refinement technique (see Section II-A) and a novel motion-compensated temporal filter (see Section II-B). Thereafter we describe the proposed spatial filter (see Section II-C), which is an extension of [11] for spatially non-stationary noise. Implementation and complexity issues of the method and experimental results are presented and discussed in Section III. Finally, Section IV draws some conclusions of this work.

II. PROPOSED APPROACH

A. Motion Field Refinement Step

Motion estimation algorithms used in real-time video codecs like [9], which is the base of motion estimator used in this paper, very often produce motion vector fields which contain falsely estimated motion vectors in the background, due to use of the reduced number of pixels for MAD calculation like in [12] and performing a search on a reduced number of locations. Other motion estimation algorithm, such as half-pixel motion field estimator defined in MPEG-4 standard used in our work, do not capture the realistic motion fields, because it does not use the neighboring motion vectors to impose a structure on a motion field.

In the following we propose a motion field filtering technique that eliminates spurious motion vectors from the spatial areas in the video frames where no actual motion exists. The basic idea in this algorithm is to compare the MAD between the corresponding blocks with the average MAD, and based on that we decide if motion is present or not. Althought implicit motion filtering is performed in MRST motion estimation algorithm of [9] by comparing MAD values of the candidate motion vectors (including zero motion) from the spatial and temporal neighborhood, wrongly estimated motion vectors still apppear in background. The proposed motion filtering method is particularly effective in surpressing spurious background motion vectors. First we calculate the MAD D between the pixels in the corresponding blocks in the current and the previous frames

$$D_{i,j}^{k} = \frac{1}{N^2} \sum_{m=1}^{N} \sum_{n=1}^{N} \left| d_{m,n}^{k,i,j} - d_{m,n}^{k-1,i,j} \right|$$
(1)

where k is the frame number, i, j are the spatial coordinates of a block, m, n the coordinates of a pixel inside the block, and Nis the block size. We define a threshold for motion detection in the kth frame as follows:

$$\text{THR} = \gamma \frac{1}{N_{bh} N_{bv}} \sum_{i=1}^{N_{bh}} \sum_{j=1}^{N_{bv}} D_{i,j}^k$$
(2)

where γ is a scalar and N_{bh} , N_{bv} are numbers of blocks along horizontal and vertical axes. We found experimentally that the value $\gamma = 0.45$ yields the best results for most of the sequences. The parameters were optimized on the test sequences different than the ones used in the denoising performance comparison of the different tested algorithms. We note that the performance of the algorithm is much less sensitive to the values of γ compared with the approach where a fixed predefined threshold for motion detection is used.

In this filtering step, we decide whether motion exists in each block simply by comparing the absolute block difference with the previously calculated threshold. If the $D_{i,j}^k < THR$, both

motion vector components are set to zero. Otherwise, the motion vector keeps its original value.

B. Motion Compensated Temporal Filter

This section presents a novel motion-compensated recursive temporal denoising filter. Denoising based on motion compensated filtering along the estimated motion trajectory is a very powerful approach, provided that the motion is correctly estimated. On the contrary, this approach can yield very disturbing artifacts at positions where the motion estimates are incorrect. To alleviate this problem, temporal filtering should take into account the reliability of the estimated motion vectors and adapt the amount of smoothing accordingly [4].

The main idea behind the proposed filter is to control switching between weaker and stronger temporal smoothing based on a motion detection variable $m_{i,j}^k$. At positions where no motion was detected $(m_{i,j}^k = 0)$, we apply a standard recursive temporal filter. At moving positions $(m_{i,j}^k = 1)$ we filter as well, but this time along the estimated motion trajectory, and using different filter coefficients. In this way we take into account the possibility that the estimated motion is not perfect and we allow a different degree of temporal smoothing for moving and for non-moving areas. Moreover, we take into account an estimate of the reliability of the estimated motion through prediction errors, as follows.

Define the normalized prediction error $\epsilon_{i,j}^k$, for the block (i,j) in the kth frame as

$$\epsilon_{i,j}^{k} = \frac{\left| d_{i,j}^{k} - d_{i-p,j-q}^{k-1} \right|}{255} \tag{3}$$

where (p,q) are the motion vector components. The absolute difference $|d_{i,j}^k - d_{i-p,j-q}^{k-1}|$ in this expression is divided by its maximum possible value, which is 255 for 8-bit grayscale video data to ensure $0 \le \epsilon \le 1$. In our model, the smaller the prediction error $\epsilon_{i,j}^k$, the more reliable the filtering at the corresponding position along the estimated motion trajectory is. Our main idea behind expressing filtering unreliability through $\epsilon_{i,j}^k$ is to avoid wrong averaging of the different pixel values along the estimated motion trajectory, and hence to avoid motion blur and ghosting artifacts.

The proposed motion compensated filter is

$$\hat{d}_{i,j}^{k} = (1 - m_{i,j}^{k}) \left(\alpha \left(1 + \epsilon_{i,j}^{k} \right) d_{i,j}^{k} + (1 - \alpha) \left(1 - \epsilon_{i,j}^{k} \right) d_{i-p,j-q}^{k-1} \right) + m_{i,j}^{k} \left(\beta \left(1 + \epsilon_{i,j}^{k} \right) d_{i,j}^{k} + (1 - \beta) \left(1 - \epsilon_{i,j}^{k} \right) d_{i-p,j-q}^{k-1} \right)$$
(4)

where α and β are the fixed parameters of the recursive filters in static and moving areas, respectively, and $1 + \epsilon_{i,j}^k$ and $1 - \epsilon_{i,j}^k$ are data driven factors for these parameters. The factor $1 + \epsilon_{i,j}^k$ (taking values between 1 and 2) increases the influence of the current frame pixel value $d_{i,j}^k$ in the case when the normalized prediction error is large $\epsilon_{i,j}^k$ (i.e., when $d_{i-p,j-q}^{k-1}$ differs much from $d_{i,j}^k$). The influence of $d_{i-p,j-q}^{k-1}$ on the filtering result is in this case simultaneously suppressed through $1 - \epsilon_{i,j}^k$ (which is then close to zero). Otherwise, when the prediction error $\epsilon_{i,j}^k$ is small, the factor $1 - \epsilon_{i,j}^k$ is close to 1, enforcing smoothing along the estimated motion trajectory. We optimized the parameters α and β experimentally in the mean squares error sense over a number of different sequences and different noise levels. These parameters are optimized on training set of a video sequences different than the ones used for evaluation of denoising performance of the algorithm. Training set is chosen in a such a way to contain most often types of motion that can be present in video sequences (global motion, zoom, movement of the foreground object with the fixed background). The resulting optimized parameter values are $\alpha = 0.45$ and $\beta = 0.85$.

C. Spatial Filter

Similar to [7], we combine the temporal filter with a wavelet domain spatial filter. Aiming at low complexity and a hardwarefriendly solution, we start from the fuzzy filter of [11], which is a fuzzy-logic version of the spatially adaptive ProbShrink from [13]. This filter applies to each wavelet coefficient a shrinkage factor, which is a function of two measurements: the coefficient magnitude w_l and a local spatial activity indicator (LSAI) z_l , i.e., $\hat{y} = \gamma(w_l, z_l)w_l$.

It was shown experimentally in [11] that this FuzzyShrink method yields the same PSNR performance as ProbShrink. We propose the modification of the FuzzyShrink method to make it adaptive to spatially non-stationary noise by estimating σ locally, since the noise after temporal filtering has non-uniform variance. We use 16 × 16 overlapping windows and shift these in steps of 8 pixels along each direction. For each window we use Donoho's wavelet domain median estimator [14], i.e., the noise variance is estimated as the median absolute deviation of the wavelet coefficients from the highest frequency subband, divided by 0.6745.

The proposed temporal-spatial denoising scheme, depicted in Fig. 1 performs motion-compensated filtering followed by the spatial filtering.

III. RESULTS

In this Section we first analyze the performance of the proposed motion field refinement algorithm and the performance of the proposed denoising schemes in comparison with related recently reported ones.

A. Motion Field Refinement Algorithm

To evaluate the effect of the proposed motion field refinement step, we compare the mean squared error (MSE) in the motion compensated frame, obtained using the estimated motion field with and without the motion field refinement step.

The mean square error of the motion field is defined as

$$MSE = \frac{1}{N_x N_y} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left(d_{i,j}^k - d_{i-v_x,j-v_y}^{k-1} \right)^2$$
(5)

where v_x and v_y are the components of the motion vector and $N_x \times N_y$ is the image size.

We observe that the MSE of the motion compensation decreases for the most of the test sequences, which proves the effectivenes of the motion field filtering step.

Also we evaluate the effect of the proposed motion field refinement step, by comparing the average PSNR of the denoised sequence, obtained using the estimated motion field with and without the motion field refinement step. The average PSNR of

TABLE I Average PSNRS of Denoised Sequences With and Without Motion Field Filtering Step

Sequence	Without MV filtering	With MV filtering
"Salesman"	31.21	32.57
"Miss America"	34.79	36.43
"Tennis"	29.44	30.23
"Flower Garden"	27.63	27.58



Fig. 2. Motion field of the 15th frame of the "Chair" sequence, before and after filtering.

the denoised sequences obtained with and without the proposed motion estimation refinement technique for several sequences are given in Table I.

Filtered motion field also appears to correspond better to the real motion visually, as shown in Fig. 2, where the motion fields prior and after motion-field refinement are depicted for a frame from the "Chair" sequence. One observes that, as expected, the algorithm sets the motion vectors to zero in the smooth areas where no actual motion exists.

B. Denoising Results

In this section, we present the results of the proposed methods using non-decimated wavelet transform, implemented with the algorithm à trous [15], with 3 orientations per scale, and with 2 decomposition levels. The wavelet function used here is the Daubechies least asymmetrical wavelet (symmlet) with eight vanishing moments.

We used four standard test sequences: "Miss America", "Salesman", "Tennis" and "Flower Garden", with additive white Gaussian noise of standard deviation $\sigma = 10$, 15, and 20. The resolutions of the test sequences are 352×240 for the "Salesman" and "Tennis" sequence and 352×288 for the others. The PSNR values obtained with the proposed methods are compared against those obtained with the SEQWT method of [7], and with a recent wavelet domain spatio-temporal (WST)



Fig. 3. Quantitative performance comparison between the proposed method, method with MPEG-4 motion estimation and two reference methods SEQWT [7] and WST [6] for the test sequences corrupted with $\sigma = 15$: (a) "Flower Garden"; (b) "Salesman"; (c) "Tennis"; and (d) "Miss America".

method from [6] and with a recent iterative scheme of [10]. The comparison to [10] will be shown later in this Section, only in terms of PSNR values averaged per test sequence, since per-frame PSNR results were not available for this reference method. The results in Fig. 3 are similar or sometimes better than the corresponding PSNR plots in [5].

The plotted PSNR values shown in Fig. 3 contain: (a) SEQWT-method [7]; (b) the WST-method [6]; (c) proposed temporal-spatial scheme; and (d) proposed scheme with half pixel interpolation where spatial filtering precedes temporal. The method of [6] offers very good results in both visual and numerical (PSNR) sense.

The proposed temporal-spatial scheme depicted in Fig. 1 performs best for all test sequences, with PSNR improvements ranging from 0.5–1.4 dB as compared to the SEQWT algorithm. The improvement over the WST method of [6] is around 1 dB for all sequences except "*Miss America*" where both algorithms show similar performance. The main reason for such results is that the algorithm of [6] slightly tends to degrade the textures in the image, while preserving well static image edges. Since the "*Miss America*" sequence contains less textures than the other test sequences, the performance differences between the proposed algorithm and that of [6] are reduced.

The results show that the proposed methods also outperform the reference methods in terms of visual quality, especially in the sequences which contain more dynamics. Examples of denoised video frames are shown in Figs. 4 and 5. Moreover the low frequency noise components that are hardly perceived in a single frame shown, but appear in the processed video sequence are



Fig. 4. Illustration of the proposed method with: (a) the 5th noise-free frame of the "Flower Garden" sequence; (b) the same frame corrupted with additive Gaussian noise with $\sigma = 15$; (c) the frame after filtering with WST filter; and (d) the frame after filtering with the best proposed approach.

much better suppressed by the new method. Fig. 4 shows that the new method preserves textures much better than the WST method of [6], which is particularly visible on the stem of the tree and the details of the roofs. A similar conclusion is valid for the "Salesman" sequence shown enlarged in Fig. 5. The new method introduces less motion blur, which can be observed on the hand of the salesman and less spatial blur in the static areas, which is visible on the books on the shelf.



Fig. 5. Part of the 31th frame of the "Salesman" sequence with: (a) the original; (b) the noisy sequence with $\sigma = 15$; (c) the WST algorithm [1]; and (d) the proposed method.

We also compare our algorithm with the recent denoising iterative scheme of [10] where only average PSNRs per test sequence were reported. For the "Flower Garden" sequence, with input PSNR of 28 dB, the average PSNR of our denoising method is 30.8 dB, which is approximately 0.5 dB less than reported in [10]. For the "Salesman" sequence, with input PSNR of 28 dB our method has PSNR of 34.61 dB while the reported value in [10] is 35.13 dB. For the same sequence with a larger amount of noise, with input PSNR of 24 dB, the performance loss of our method compared to that reported in [25] is less than 0.1 dB. It is important to notice that the method in [10] is much more complex. For example, the reported processing time of this reference method is approximately 1 min and 40 s per frame for the frame size 384×288 on a powerful 8×3 GHz processor. The processing time of our method, for the frame resolution of 352×288 , and on a much slower processor (one core on a working frequency of 2.4 MHz) is 11 ms per frame. Moreover, the methods [10] requires up to 7 frames to be stored, while our method uses only current and previous frame.

IV. CONCLUSION

The main idea presented in this paper is reusing of motion estimation resources from video codecs for video denoising, resulting in a low-complexity method, which is easily integrable into existing video codecs. The core of the method are two components: a novel motion field filtering step and a novel recursive temporal filter with appropriately defined reliability of the estimated motion field. The results show that this low-complexity scheme compares favorably with recent related video denoising methods and that it is competitive even with much more complex recent approaches.

REFERENCES

- J. Y. Tham, S. Ranganath, and A. A. Kassim, "Highly scalable waveletbased video codec for very low bit-rate environment," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 1, pp. 12–27, Jan. 1998.
- [2] J. Barbarien, Y. Andreopoulos, A. Munteanu, P. Schelkens, and J. Cornelis, "Motion vector coding for in-band motion compensated temporal filtering," presented at the IEEE Int. Conf. Image Process. (ICIP), Barcelona, Spain, 2003.
- [3] L. Thornton, J. Soraghan, and M. Chakraborty, "Perceptually weighted wavelet-based codec for handheld videophone applications," *Electron. Lett.*, vol. 38, pp. 839–841, 2002.
- [4] V. Zlokolica, A. Pižurica, and W. Philips, "Wavelet-domain video denoising based on reliability measures," *IEEE Trans. Circuits Syst. for Video Technol.*, vol. 7, no. 3, pp. 477–488, Mar. 2006.
- [5] S. M. Mahbubur Rahman, M. Omair Ahmad, and M. Swamy, "Video denoising based on inter-frame statistical modeling of wavelet coefficients," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 2, pp. 187–198, Feb. 2007.
- [6] E. J. Balster, Y. F. Zheng, and R. L. Ewing, "Combined spatial and temporal domain wavelet shrinkage algorithm for video denoising," *IEEE Trans. Circuits Syst. for Video Technol.*, vol. 16, no. 2, pp. 220–230, Feb. 2006.
- [7] A. Pižurica, V. Zlokolica, and W. Philips, "Noise reduction in video sequences using wavelet-domain and temporal filtering," presented at the SPIE Conf. "Wavelet Applications in Industrial Processing", Providence, RI, 2003.
- [8] L. Jovanov, A. Pižurica, V. Zlokolica, S. Schulte, E. Kerre, and W. Philips, "Combined wavelet domain and motion compensated filtering compliant with video codecs," presented at the ICASSP, Honolulu, HI, 2007.
- [9] J. Chalidabhongse and C.-C. J. Kuo, "Fast motion vector estimation using multiresolution-spatio-temporal correlations," *IEEE Trans. Circuits Syst. for Video Technol.*, vol. 16, no. 8, pp. 993–1007, Aug. 1997.
- cuits Syst. for Video Technol., vol. 16, no. 8, pp. 993–1007, Aug. 1997.
 [10] J. Boulanger, C. Kervrann, and P. Bouthemy, "Space-time adaptation for patch-based image sequence restoration," *IEEE Trans. Pattern Analysis Machine Intell.*, vol. 29, no. 6, pp. 1096–1102, Dec. 2007.
- [11] S. Schulte, B. Huysmans, A. Pižurica, E. Kerre, and W. Philips, "A new fuzzy-based wavelet shrinkage image denoising technique," presented at the Adv. Concepts for Intell. Vision Syst. (Acivs), Antwerp, Belgium, 2006.
- [12] B. Liu and A. Zaccarin, "New fast algorithms for the estimation of block motion vectors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 148–157, Mar. 1993.
- [13] A. Pižurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 654–665, Mar. 2006.
- [14] D. L. Donoho, "De-noising by soft thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
- [15] S. Mallat, A Wavelet Tour of Signal Processing. London, U.K.: Academic, 1998.