Wavelet-Domain Video Denoising Based on Reliability Measures

Vladimir Zlokolica, *Member, IEEE*, Aleksandra Pižurica, *Member, IEEE*, and Wilfried Philips, *Member, IEEE*

***Abstract*—This paper proposes a novel video denoising method based on nondecimated wavelet band filtering. In the proposed method, motion estimation and adaptive recursive temporal filtering are performed in a closed loop, followed by an intra-frame spatially adaptive filter. All processing occurs in the wavelet domain.**

The paper introduces new wavelet-based motion reliability measures. We make a difference between motion reliability per orientation and reliability per wavelet band. These two reliability measures are employed in different stages of the proposed denoising scheme. The reliability per orientation (horizontal and vertical) measure is used in the proposed motion estimation scheme while the reliability of the estimated motion vectors (MVs) per wavelet band is utilized for subsequent adaptive temporal and spatial filtering. We propose a novel cost function for motion estimation which takes into account the spatial orientation of image structures and their motion matching values. Our motion estimation approach is a novel wavelet-domain three-step scheme, where the refinement of MVs in each step is determined based on the proposed motion reliabilities per orientation. The temporal filtering is performed separately in each wavelet band along the estimated motion trajectory and the parameters of the temporal filter depend on the motion reliabilities per wavelet band. The final spatial filtering step employs an adaptive smoothing of wavelet coefficients that yields a stronger filtering at the positions where the temporal filter was less effective.

The results on various grayscale sequences demonstrate that the proposed filter outperforms several state-of-the-art filters visually (as judged by a small test panel) as well as in terms of peak signal-to-noise ratio.

***Index Terms*—Motion estimation, video denoising, wavelets.**

I. Introduction

VIDEO sequences are often distorted by noise during acquisition or transmission. Certain noise sources are located in camera hardware, becoming activated under poor lighting conditions. Other noise sources are due to transmission over analogue channels. Noise reduction is required in many applications, e.g., for visual improvement in video surveillance, television and medical imaging; as a preprocessing step for further analysis of video sequences (tracking, object recognition, etc.) or for video coding. In many video applications, the noise can be well approximated by the additive white Gaussian model [1], which we consider in this paper.

Manuscript received May 12, 2005; revised March 21, 2006. The work of A. Pižurica was supported in part by the Fund for Scientific Research (FWO), Flanders, Belgium. This paper was recommended by Associate Editor C. Guillemot.

The authors are with the Department for Telecommunications and Information Processing (TELIN), Ghent University, B-9000 Ghent, Belgium (e-mail: vzlokoli@telin.UGent.be; philips@telin.UGent.be; sanja@telin.UGent.be).

Digital Object Identifier 10.1109/TCSVT.2006.879994

Recently, a number of video denoising methods have been proposed, e.g., [2]–[15]. A thorough review of classical noise reduction algorithms for digital image sequences is presented in [13].

Motion-compensated denoising techniques attempt to better exploit the considerable temporal redundancy in video by temporally smoothing pixel values along their estimated motion trajectories [7], [9], [16]. Techniques such as these do not have the potential disadvantage of reducing the spatial resolution of the input sequence and can even improve it. Furthermore, *time-recursive* implementations are efficient in terms of low computational cost.

It is, however, often impossible or impractical to establish the exact temporal correspondence between consecutive frames *for all pixels* (e.g., because of occlusion, inaccurate motion estimates or computational complexity restrictions). When motion estimation fails, motion-compensated temporal denoising can produce disturbing artifacts. This is especially true for recursive techniques where errors propagate through the sequence. One solution to this problem is to reduce the amount of temporal filtering where no accurate MVs are found [9]. Even so, certain artifacts and/or noise remain and hence additional spatial filtering is desirable. In the case where the spatial filter is applied after the temporal one, the input noise is generally nonstationary and is therefore more difficult to remove [10].

The spatio-temporal filters based on the above principles are either nonseparable ("fully 3-D") [2]–[6], or separable ("2-D + 1-D") [7]–[14]. Moreover, separable filters come in three variants: "spatial-first" filters, where spatial filtering is performed before temporal filtering [12]–[14]; "temporal-first" filters, where the order is reversed [7], [10], [11]; and "combined" filters where two filters are applied in parallel with their outputs combined (usually through weighted averaging) [9], [13], [14].

Full 3-D solutions often have large memory requirements and can introduce a significant time delay because 3-D wavelet transforms often imply processing several future frames before the current one. This is undesirable in interactive applications such as infrared camera-assisted driving or video-conferencing.

Spatial-first techniques facilitate subsequent motion estimation; in these schemes motion estimation can be quite simple and yet robust against noise. However, the spatial denoising may introduce some local "ringing" or blurring artifacts, particularly at high noise levels.

The *combined* spatio-temporal solution is potentially more advantageous due to the possibility of the joint optimization of the spatial and temporal filter performance. Nevertheless, in order to optimize the performance of the combined spatio-temporal filter, one has to estimate the correct (spatially varying)

weight for each filter (spatial and temporal). This is often difficult task which leads to a nonunique solution [9], [13], [14].

In this paper we adopt the temporal-first approach in order to minimize both spatial blurring and artifacts such as "ringing." In order to make our temporal filter efficient we develop a novel robust motion estimation method. We assume that in the vicinity of important image structures robust motion estimation is usually possible, even without spatial filtering. Although the reliability of the motion estimates decreases to some extent at higher noise levels, in the vicinity of important image discontinuities the estimated MVs are sufficiently accurate for the proposed filtering scheme. Moreover, when motion reliability lowers, the amount of temporal filtering decreases and hence the spatial filter has to deal with higher noise levels.[1]

In the proposed motion estimation approach we make use of "image discontinuities" which describe discontinuities such as edges, corners, peaks, lines, etc. These discontinuities are represented by a set (group) of large wavelet coefficient magnitudes, with significantly higher magnitude than the ones representing noisy flat image regions. Specifically, we perform motion matching (aligning) of the image discontinuities in a wavelet band of specific orientation as follows. At the positions where significant image discontinuities are present we search best motion match (in correspondence to the previous frame) in a direction perpendicular to the orientation of the wavelet band. In such a manner we obtain "reliable" motion estimates which are robust against noise. On the other hand, in uniform areas (characterized by none-significant wavelet coefficient values) we assume that reliable motion estimate can not be obtained. Hence, in this case we look for the best MV from the spatio-temporal neighborhood, which is assumed to be sufficiently reliable.

In [17], we proposed a separable approach for motion estimation where both horizontal and vertical MV components were estimated separately by minimizing two mutually independent cost functions. For estimating each vector component one MV was estimated and the corresponding MV component was taken. Namely, the estimated MV was built out of two MVs by taking the corresponding vector component from each of the two. However, this approach yields less accurate motion estimates in cases where true motion is in a diagonal direction. In this paper, we define a *joint* cost function which depends on two directional cost functions, which are weighted according to the estimated motion orientation reliabilities.

In the proposed approach, we define the reliability of the MVs in a novel way, for each direction (in the case of motion estimation) and for each wavelet band (in the case of motion compensation). Based on the estimated MVs and the corresponding reliability per wavelet band, we perform adaptive temporal filtering within each wavelet band. The temporal filtering consists of recursive adaptive smoothing along the estimated motion trajectories, where the level of filtering is proportional to the estimated reliability of the motion estimates in the corresponding wavelet band.

[1]A spatial filter could be introduced as a preprocessing step for motion estimation, which could improve motion estimation reliability at higher noise levels. Based on our experiments, we do not expect that the end filtering result would be significantly improved to justify the introduction of this additional filtering step.

Because the reliability of the estimated MVs in general varies from one place to another, the level of temporal filtering (noise suppression) varies from place to place as well. Consequently, the noise remaining after the temporal filter is nonstationary. Since existing spatially adaptive filters usually assume stationary noise, we apply a novel spatially adaptive filter that efficiently removes nonstationary noise. The proposed filter applies weighted averaging of the wavelet coefficients within a 2-D sliding window, where the degree of spatial smoothing is influenced by the amount of preceding temporal filtering.

We have processed different grayscale sequences with our algorithm and have compared its performance with several state-of-the-art filters. The evaluation of the results was done in terms of peak signal-to-noise ratio (PSNR) and visual quality, judged by a six-person panel. From a PSNR point of view, the new filter behaves better than the reference filters in most cases (the average improvement is 1 dB and usually more). The proposed algorithm was found to be the best by the panel in 90% of test cases in terms of overall quality and in 87% and 55% of test cases in terms of noise reduction, and *least* visible artifacts, respectively.

The paper is organized as follows. Section II reviews existing motion estimation techniques. Section III presents the proposed spatio-temporal filter along with the motion estimation algorithm, and Section III-A introduces the concept of "reliability" for motion estimates in respect to specific motion direction and to each wavelet band separately. Section III-B describes the new algorithm for motion estimation, Section III-C the proposed temporal recursive filter and Section III-D our spatial filtering technique. We present experimental results in Section IV and conclude the paper in Section V.

II. Motion Estimation Techniques for Video Denoising

Classical single resolution motion estimation and compensation techniques use block-matching [1], [18]. This assumes that motion is translational and locally stationary. In general, block matching techniques can be classified according to: 1) the block size which can be fixed [19] or variable [18], [20]; 2) the search strategy (e.g., hierarchical [21], [22] or three-step approach [23]); and 3) matching criteria (e.g., maximum cross correlation or minimum error). The MV $\mathbf{v}$ is determined by minimizing a certain *cost* function in a confined search area

$$\mathbf{v}(s) = \arg\min_{\mathbf{v}} [\mathrm{cost}(s, \mathbf{v})] \tag{1}$$

where $\mathrm{cost}(s, \mathbf{v})$ is a linear or nonlinear function of the pixel values from the sth block of the current frame and those from the corresponding block in the previous frame, displaced by the MV $\mathbf{v}$. The most common cost function is the mean absolute difference (MAD).

Ambiguity in motion estimation usually arises when no spatial image discontinuities (structures) exist within the block or when the motion model is too simple to describe real motion. MV fields estimated from only two frames (the so-called "displaced frame difference") often provide locally optimal estimates but cannot guarantee that the vector field resembles the

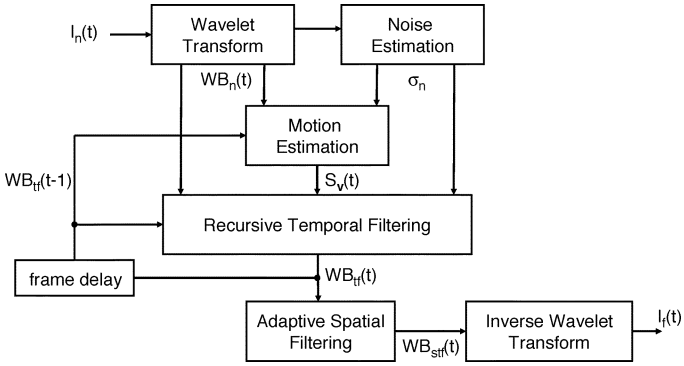

Fig. 1. General framework description of the proposed algorithm. $I_n(t)$: input noisy frame; $I_f(t)$: spatio-temporally filtered frame; $\mathrm{WB}_n(t)$: noisy wavelet band; $\mathrm{WB}_{\mathrm{tf}}(t)$: temporally filtered wavelet band; $\mathrm{WB}_{\mathrm{stf}}(t)$: spatio-temporally filtered wavelet band; σ_n: standard deviation of Gaussian noise; t: temporal coordinate; $S_{\mathbf{v}}(t)$: set of estimated MVs $\mathbf{v}$ for frame t.

true object motion [1] in the scene. This can present a significant problem for video conversion and denoising. In [24]–[26], a recursive "true-motion" estimator was proposed to avoid ambiguity and reduce the computation time.

Multiresolution motion estimation [21], [27]–[32] increases the accuracy of the estimated MVs in comparison with single scale solutions. Moreover, the multiresolution approach significantly reduces the computation time and has the potential to yield smoother MV fields [21]. The basic approach is first to estimate rough MVs at coarse scales and then refine them using information from finer scales. In [31], a multiresolution scheme which exploits spatio-temporal correlation between MVs was proposed for video coding purposes.

The wavelet domain motion estimation methods presented in [21], [28]–[30], and [33] first decompose all frames using a decimated (critically sampled) 2-D wavelet transform and subsequently exploit the temporal correspondence between the wavelet bands from two consecutive video frames. However, the accuracy of the motion estimation based on a critically sampled wavelet transform is limited because of its shift variant nature. It has been shown that the use of a shift-invariant wavelet decomposition improves the accuracy of motion estimation and enables better temporal filtering along the motion trajectories [34]. Efficient solutions in this respect include cycle spinning [29], [30] and the use of nondecimated wavelet transform [35].

III. Proposed Video Denoising Method

A general description of the proposed video denoising algorithm is presented in Fig. 1. Three important steps are the following.

- *Motion estimation*: The proposed approach estimates a *single* MV field for all wavelet bands but *different* MV reliabilities in each band. In the motion estimation, we make use of spatial image discontinuities, represented by wavelet coefficients, in combination with spatio-temporal correlations of the MVs, within a recursive scheme.
- *Motion compensation*: Temporal filtering in each wavelet band is recursively performed along the estimated motion trajectories. The amount of filtering is tuned to the estimated reliability of the corresponding MVs within a certain block of wavelet coefficients.
- *The adaptive spatial filtering scheme* suppresses the remaining (nonstationary) noise. The proposed spatial filter performs adaptive averaging of the wavelet coefficients within a 2-D sliding window in such a way that the reliability of the preceding temporal filter influences the degree to which the spatial smoothing is applied.

The proposed method uses a nondecimated wavelet transform implemented with the à trous algorithm [35]. We apply a two-dimensional (spatial) wavelet transform to each video frame and denote *wavelet bands* (WB) of this spatial wavelet transform by $\mathrm{WB} = \mathrm{LL}, \mathrm{LH}, \mathrm{HL}, \mathrm{HH}$ for the low-pass (approximation), horizontal, vertical, and diagonal orientation bands, respectively. We use a subscript to denote the noisy or denoised band as follows. WB_n: noisy band; $\mathrm{WB}_{\mathrm{tf}}$: temporally filtered; and $\mathrm{WB}_{\mathrm{stf}}$: spatio-temporally filtered band. Additionally, we denote the spatial position as $\mathbf{r} = (x, y)$ and frame index (time) as t. The decomposition level is denoted by a superscript (l), where $l = 1, \ldots, N$ (1 denotes the finest scale and N the coarsest).

As can be seen from Fig. 1, the noisy input frame $I_n(\mathbf{r}, t)$ is first decomposed into wavelet bands $\mathrm{WB}_n^{(l)}(\mathbf{r}, t)$. Using the noisy wavelet bands $\mathrm{WB}_n^{(l)}(t)$ from the current frame and temporally filtered wavelet bands $\mathrm{WB}_{\mathrm{tf}}^{(l)}(t-1)$ from the previous time recursion, and taking into account the estimated standard deviation (σ_n) of the Gaussian noise, we perform motion estimation. Subsequently, we apply a recursive temporal filter (Section III-C) on the noisy wavelet bands $\mathrm{WB}_n^{(l)}(t)$ along the estimated motion trajectory, using the corresponding $\mathrm{WB}_{\mathrm{tf}}^{(l)}(t-1)$ band. The temporally filtered wavelet bands $\mathrm{WB}_{\mathrm{tf}}^{(l)}(t)$ are further subjected to spatial filtering. Finally, the inverse wavelet transform yields the denoised video sequence $I_f(t)$.[2]

We define the MAD for each block s in the wavelet band $\mathrm{WB}^{(l)}(\mathbf{r}, t)$, as follows:

$$\mathrm{MAD}_{\mathrm{WB}}^{(l)}(s, t, \mathbf{v}) = \frac{1}{N} \sum_{\mathbf{r} \in B_s} \left| \mathrm{WB}_n^{(l)}(\mathbf{r}, t) - \mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r} - \mathbf{v}, t-1) \right| \quad (2)$$

where s denotes the index of a block within the current frame, t the current frame index, and $\mathbf{v}$ a MV. B_s represents a set of $N = N_x \times N_y$ spatial positions belonging to the given block s, where $N_x = 8$ and $N_y = 8$ represent the number of rows and columns in the block, respectively.

A. Reliability of MV Estimates

We introduce the idea of reliability of the MVs in respect to each motion orientation (horizontal and vertical) and to each wavelet band, separately. The reliability of the MV *per orientation* is used in the proposed motion estimation approach (which employs three-step search scheme) for coordinating the refinement of the initial MV in each step (Section III-B). Additionally, we define the reliability of the finally estimated MVs *per*

[2]The estimation of the Gaussian noise level is performed using a spatial-gradient approach [36], based on evaluating the distribution of spatial gradient magnitudes.

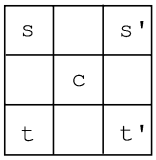

Fig. 2. Position of spatial and temporal motion block neighbors. s: spatially neighboring block; t: temporally neighboring block; c: central block.

wavelet band to determine the appropriate *amount* of temporal filtering (Section III-C), along the estimated motion trajectory.

We define the horizontal θ_H and vertical θ_V "per orientation" reliabilities of the MV $\mathbf{v}$, as follows:

$$\theta_H(s,t,\sigma_n,\mathbf{v}) = \frac{\sigma_n}{1+\sum_{l=1}^{N} d_l \mathrm{MAD}_{\mathrm{HL}}^{(l)}(s,t,\mathbf{v})}$$
$$\theta_V(s,t,\sigma_n,\mathbf{v}) = \frac{\sigma_n}{1+\sum_{l=1}^{N} d_l \mathrm{MAD}_{\mathrm{LH}}^{(l)}(s,t,\mathbf{v})} \quad (3)$$

with parameters d_l $(l=1,\ldots,N)$ denoting the weights associated with the corresponding wavelet decomposition scale l, where $d_1 = d_2 = \ldots = d_N$ and $\sum_{l=1}^{N} d_l = 1$.

Analogously, we define the "per wavelet band" reliability $(\mathrm{WB}^{(l)})$ of the estimated MV $\mathbf{v}$, for temporal filtering, as follows:

$$\theta_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}) = \frac{\sigma_n}{1+\mathrm{MAD}_{\mathrm{WB}}^{(l)}(s,t,\mathbf{v})}. \quad (4)$$

The defined motion reliabilities per orientation in (3) are expressed as the ratio of the standard deviation of noise σ_n and the sum of MADs for the perpendicularly oriented wavelet bands from different scales. On the other hand, the defined reliability per wavelet band in (4) is expressed as the ratio of the σ_n and the MAD of the corresponding wavelet band. These reliabilities are expected to be around 1 in the case of a *reliable* MV estimate (MAD values are close to σ_n) and lower in the opposite case (MAD values are significantly higher than σ_n). More specifically, θ_H, θ_V and $\theta_{\mathrm{WB}}^{(l)}$ are the smallest (close to zero) in poorly motion matched (spatially) structured areas.

B. Motion Estimation

We propose a new block-based motion estimation approach which employs a three-step search [1], [18], [23], within the wavelet domain, operating in a spatio-temporal recursive manner. In the first step the initial MV is fixed to either a previously computed MV from the spatio-temporal neighborhood (Fig. 2) or a zero MV. Subsequently, the chosen initial MV is refined in the three steps. The proposed motion estimation approach produces a single MV field which is used for the adaptive temporal filtering for all wavelet bands.

In the proposed method, the refinement of the initial MV is based on motion-matching image discontinuities (represented by groups of significant wavelet coefficients) and defined reliabilities *per orientation* (3). Specifically, the motion matching value of the image discontinuities is determined as the MAD (2) for the tested MV and corresponding wavelet band. By assuming that the motion estimation is most reliable on significant image discontinuities (producing largest wavelet coefficient magnitudes) and the least reliable in uniform areas (characterized by none-significant wavelet coefficient values), we distinguish three possible cases in our approach.

- *A low motion matching value for image discontinuities*. This implies perfect motion-alignment. In this case we consider that we have found a perfect motion match and that no further refinements of the initial MV are necessary.
- *A low motion matching value in uniform image areas*. In this case we consider that the motion cannot be adequately estimated. Thus, we must rather rely on (already obtained) reliable spatio-temporally neighboring MVs and prevent further refinements (changes) of the chosen initial MV.
- *A high motion matching value for image discontinuities*. This implies poor motion alignment. In this case, we consider that the initial MV is far from the optimal motion match and that refinement is necessary. However, this is not a problem since in this case reliable motion estimation (matching) is possible.

The reason for choosing the specific spatio-temporal neighborhood in Fig. 2, with four causal spatial neighbors and nine temporal neighbors, is the reduced computation complexity. Using two spatial and two temporal MV predictions, lying on two perpendicular axes, we can still take into account the majority of the motion object borders [24].[3]

1) Wavelet-Domain Three-Step Method: As opposed to the method of [31] which estimates first MV field at the roughest scale (lowest resolution) and then in the following steps (at higher resolution scales) refines the MV field, in our method we take into account information at each step from all resolution scales simultaneously. Moreover, we refine the initial MVs in a superior manner; in contrast to the method of [31] which refines the MV field in terms of the minimal MAD values of the low-pass image representations, we refine the MV field by minimizing (a newly defined) cost measure. The cost measure depends on the motion matching values of the horizontal and vertical image discontinuities, which are weighted according to the reliabilities *per orientation* (horizontal and vertical) of the initial MV.

In our approach the search area in each step of the algorithm is confined to a $(N_x/2^{j-1}) \times (N_y/2^{j-1})$ block where $j=1,2,3$ for the first, second, and third step, respectively. For each step j of the algorithm, we define initial MVs $\mathbf{v}_i^{(j)}$ and MV corrections $\mathbf{v}_c^{(j)}$ where $\mathbf{v}_i^{(j)}$ is fixed and $\mathbf{v}_c^{(j)}$ varies. The vectors $\mathbf{v}_i^{(j)} + \mathbf{v}_c^{(j)}$ are then tested in order to find the best matching vector $\mathbf{v}_b^{(j)}$ for step j. The best matching vector is determined by minimizing a cost function (which we define later), as follows:

$$\mathbf{v}_b^{(j)}(s,t) = \arg \min_{\mathbf{v}_c^{(j)} \in S_j} \mathrm{cost}\left(s,t,\vartheta_H,\vartheta_V,\mathbf{v}_i^{(j)},\mathbf{v}_c^{(j)}\right) \quad (5)$$

where S_j represents the set of allowed $\mathbf{v}_c^{(j)}$s. The best matching MV $\mathbf{v}_b^{(j)}$ at step three $(j=3)$ is defined as the final estimated MV $\mathbf{v}_{bf}$. The initial MV at each step $j>1$ is the best estimated

[3]This enables efficient implementation of the concept which combines the consistent velocity field of a recursive process with a fast step response, as required at the contours of moving objects [24].

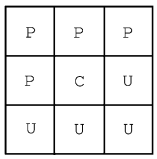

Fig. 3. The 2-D $-$ (3×3)-sliding window: spatially processed neighboring coefficients $P = \mathrm{WB}_{\mathrm{stf}}^{(l)}(\mathbf{r}, t)$, currently spatially processed coefficient $C = \mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r}_c, t)$, spatially unprocessed neighboring coefficients $U = \mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r}, t)$; $\mathbf{r}_c$ is the central spatial position of the sliding window.

MV from the preceding step, i.e., $\mathbf{v}_i^{(j)} = \mathbf{v}_b^{(j-1)}$. In the first step $(j = 1)$ the initial MV is the best matching MV among the tested MVs from a spatio-temporal neighborhood (Fig. 2). Specifically, we define the initial MV at step 1, $\mathbf{v}_i^{(1)}$ as the prior initial MV candidate $\mathbf{v}_{pi}$ that minimizes

$$\mathbf{v}_i^{(1)}(s,t) = \arg\min_{\mathbf{v}_{pi} \in U} \left(\mathrm{MAD}_{\mathrm{LL}}^{(N)}(s,t,\mathbf{v}_{pi}) + P(\mathbf{v}_{pi}) \right) \quad (6)$$

where $\mathbf{v}_{pi}$ denotes the prior initial MV candidate and $P(\mathbf{v}_{pi})$ represents a penalty for the corresponding vector $\mathbf{v}_{pi}$, with which we introduce prior knowledge. If the penalty $P(\mathbf{v}_{pi})$ is smaller, the initial MV candidate $\mathbf{v}_{pi}$ is more likely to be chosen as the initial MV $\mathbf{v}_i^{(1)}$ for the first step of the motion estimation approach.

The prior initial MV candidates $\mathbf{v}_{pi}$ belong to set $U = \{\mathbf{0}, \mathbf{s}, \mathbf{s}', \mathbf{t}, \mathbf{t}'\}$ shown in Fig. 2. This candidate set includes the zero MV $(\mathbf{0})$ and the MVs from two neighboring blocks within the current frame $(\mathbf{s}, \mathbf{s}')$ and from two neighboring blocks in the previous frame $(\mathbf{t}, \mathbf{t}')$. The zero MV $(\mathbf{0})$ is used for a reinitialization of the MV search (estimation) and the spatio-temporal neighboring vectors $(\mathbf{s}, \mathbf{s}', \mathbf{t}, \mathbf{t}')$ are used to enable spatio-temporal recursiveness in the motion estimation approach.

By assigning the smallest penalty to $\mathbf{v}_{pi} = \mathbf{0}$, we increase the sensitivity of the motion estimation to sudden scene changes or the appearance of small image parts (this concerns the accuracy of the motion estimation). Hence, the penalty for $\mathbf{v}_{pi}$ being equal to either of the four spatio-temporal neighboring vectors (Fig. 2) should be sufficiently large to reinitialize the MV search in case of sudden scene changes. However, the penalty value should not be too big either in order to enable spatio-temporal recursiveness in the motion estimation and consequently enforce smoothness (consistency) of the MV field. In our experiments, we use $P(\mathbf{0}) = 0$ and $P(\mathbf{v}_{pi}) = 2.5$; this constant was experimentally optimized in terms of maximal MV consistency and accuracy.

In step 1, we consider the candidate MV corrections $\mathbf{v}_c^{(1)}$ with horizontal component $v_{c_x}^{(1)}$ and vertical component $v_{c_y}^{(1)}$ in the set $\{-8, -4, 0, 4, 8\}$. They are added to the $\mathbf{v}_i^{(1)}$ and tested in order to find the best matching MV $\mathbf{v}_b^{(1)}$. In the second step, $\mathbf{v}_i^{(2)} = \mathbf{v}_b^{(1)}$ and the candidate MV corrections $\mathbf{v}_c^{(2)}$ can have horizontal $v_{c_x}^{(2)}$ and vertical $v_{c_y}^{(2)}$ components within the set $\{-4, -2, 0, 2, 4\}$. Finally, in the third step, $\mathbf{v}_i^{(3)} = \mathbf{v}_b^{(2)}$ and the candidate MV correction $\mathbf{v}_c^{(3)}$ components $v_{c_x}^{(3)}$ and $v_{c_y}^{(3)}$ belong to the set $\{-2, -1, 0, 1, 2\}$.

2) Cost Function: We define a novel cost function for motion estimation, consisting of horizontal and vertical components, where each component is weighted by the estimated reliability measures with respect to the corresponding initial MV component

$$\begin{aligned} \mathrm{cost}(s,t,\vartheta_H,\vartheta_V,\mathbf{v}_i,\mathbf{v}_c) = {} & k_x(s,t,\vartheta_H,v_{c_x})\mathrm{cost}_x(s,t,\mathbf{v}_i+\mathbf{v}_c) \\ & + k_y(s,t,\vartheta_V,v_{c_y})\mathrm{cost}_y(s,t,\mathbf{v}_i+\mathbf{v}_c) \end{aligned} \quad (7)$$

where $\mathrm{cost}_x(s,t,\mathbf{v})$ and $\mathrm{cost}_y(s,t,\mathbf{v})$ are separate cost functions for horizontal and vertical motion, respectively, defined as

$$\begin{aligned} \mathrm{cost}_x(s,t,\mathbf{v}) &= \mathrm{MAD}_{\mathrm{LL}}^{(N)}(s,t,\mathbf{v}) + \sum_{l=1}^{N} \mathrm{MAD}_{\mathrm{HL}}^{(l)}(s,t,\mathbf{v}) \\ \mathrm{cost}_y(s,t,\mathbf{v}) &= \mathrm{MAD}_{\mathrm{LL}}^{(N)}(s,t,\mathbf{v}) + \sum_{l=1}^{N} \mathrm{MAD}_{\mathrm{LH}}^{(l)}(s,t,\mathbf{v}). \end{aligned} \quad (8)$$

The multiplicative penalties k_x and k_y in the cost function (7) are defined in terms of the motion reliabilities, as follows:

$$\begin{aligned} k_x\left(s,t,\vartheta_H,v_{c_x}^{(j)}\right) &= C_1 + C_2 \frac{\left|v_{c_x}^{(j)}\right|}{2^{N-j}} \vartheta_H^2 \\ k_y\left(s,t,\vartheta_V,v_{c_y}^{(j)}\right) &= C_1 + C_2 \frac{\left|v_{c_y}^{(j)}\right|}{2^{N-j}} \vartheta_V^2 \end{aligned} \quad (9)$$

where the constants C_1 and C_2 are optimized in order to obtain a noise robust and smooth MV field. We have experimentally found the following optimal parameter values: $C_1 = 1$, $C_2 = 1.45$. The values of the constants C_1 and C_2 are fixed in all three steps and the correction MV components ($|v_{c_x}^{(j)}|$ and $|v_{c_y}^{(j)}|$) are normalized with their maximum amplitudes (2^{N-j}) for j step of the proposed algorithm.

The $\mathrm{cost}_x(s,t,\mathbf{v})$ and $\mathrm{cost}_y(s,t,\mathbf{v})$ functions in (8) represent motion (block) matching measures of the horizontal and vertical image structures, respectively. Image structures (discontinuities) appear stronger in a specific wavelet band depending on their orientation and are used to perform motion matching in a direction perpendicular to their discontinuity orientation. For example, motion matching of a group of significant wavelet coefficients in horizontally oriented wavelet bands enables reliable (noise robust) motion estimation in a vertical direction. This is because, in the latter case, there will be significant difference in motion matching value between well and poorly aligned horizontal image discontinuities. Hence, the ambiguity concerning the optimal vertical MV component decreases as well as the sensitivity to noise. The same idea holds for the vertically oriented image structures and the horizontal MV component.

The $\mathrm{MAD}_{\mathrm{LL}}^{(N)}$ component is added to the directional cost functions (8) in order to increase MV field consistency and accuracy in the case where orientation reliabilities (horizontal and vertical) significantly differ. In that case, without the $\mathrm{MAD}_{\mathrm{LL}}^{(N)}$ component, the final cost function in (7) would essentially depend on motion matching for only *one* MV component (the one for which the initial MV produces lower reliability per orientation). This can introduce ambiguity in the motion matching because the information about the motion matching concerning the other motion component (the one for which the initial MV produces higher reliability per orientation) is lost. This often occurs in motion matching of corner-like image structures when one

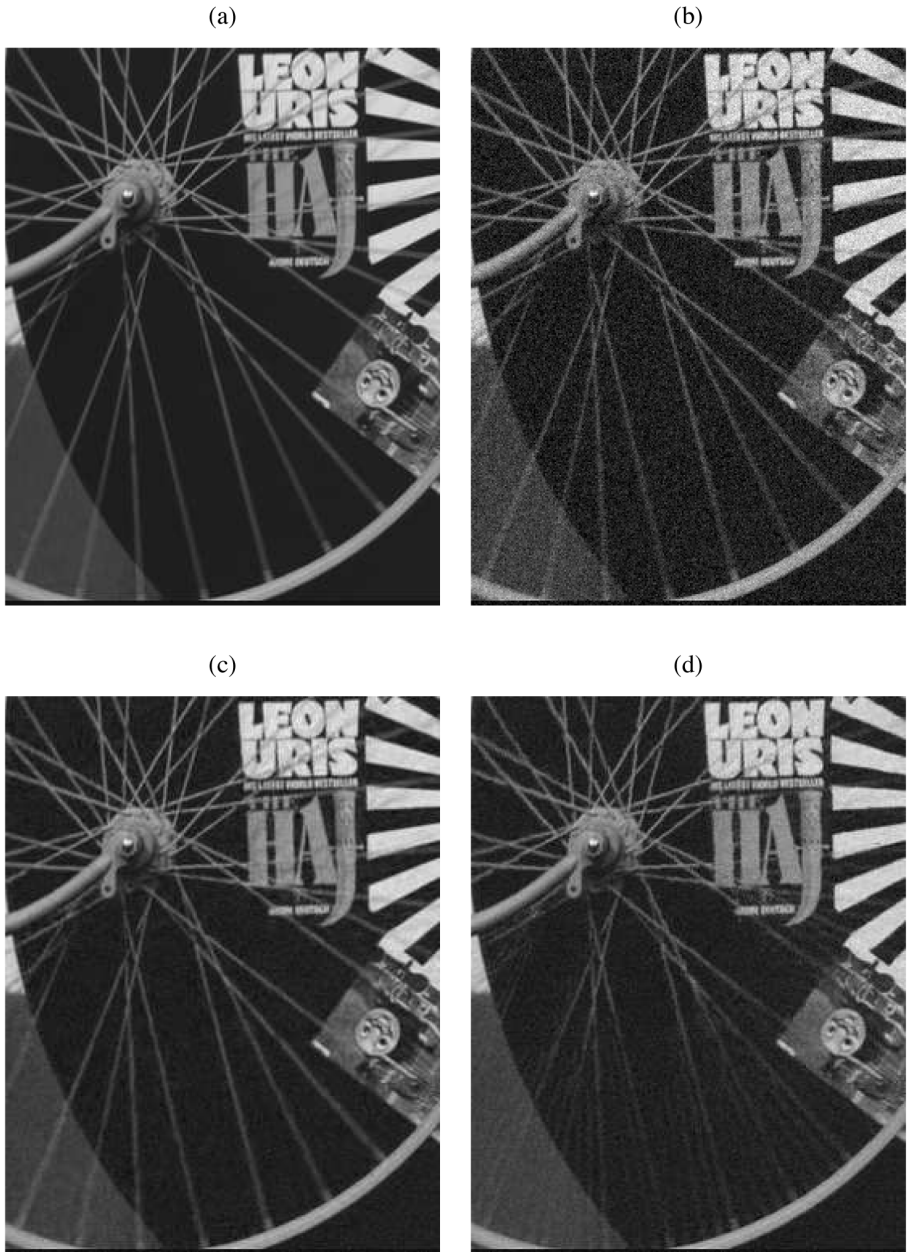

Fig. 4. Results for the 29th frame of "Bicycle" sequence with added Gaussian noise ($\sigma_n = 15$), processed by (c) WRTF filter and (d) 3RDS filter [16]. (a) Original image frame. (b) Noisy image frame.

MV component has (already) "converged" to the optimal one and the other not. By incorporating $\mathrm{MAD}_{\mathrm{LL}}^{(N)}$ in the directional cost functions (8) we include the information about the motion matching concerning both MV components. We use the $\mathrm{LL}^{(N)}$ wavelet band for this purpose because it has a higher SNR than the $\mathrm{HH}^{(l)}$ wavelet band, for example, and contains information concerning both vertical and horizontal image structures.

In the proposed expressions, the penalties k_x and k_y defined in (9) incorporate information concerning the motion matching error of vertically and horizontally oriented image discontinuities for the initial MV, respectively. The smaller the matching error, the higher the reliability of a motion match in a direction perpendicular to the orientation of the corresponding image discontinuity, meaning that the corresponding initial MV component is closer to the optimal one. As a result, the corresponding penalty (k_x or k_y) will proportionally increase for the corresponding nonzero correction MV component ($v_{cx}^{(j)}$ or $v_{cy}^{(j)}$). In such a manner, in case of higher reliabilities we assign more weight to the corresponding cost function [cost_x or cost_y in (8)] for the tested nonzero correction MVs. Consequently, the probability that the initial MV (component) is significantly changed is then reduced.

In the case where the reliabilities ϑ_H and/or ϑ_V are small the corresponding penalties k_x and/or k_y are nearly independent of $\mathbf{v}_c^{(j)}$ and reduce to the constant C_1.[4] Hence, the cost function in (7) will depend only on the corresponding cost functions (cost_x and/or cost_y) and as a result fewer restrictions on the tested (correction) MVs will be applied. This is justified because we assume that in case of image structured areas motion estimation can be reliably determined.

[4]This happens when the motion block belongs to a structured image area and the initial MV is far from optimal.

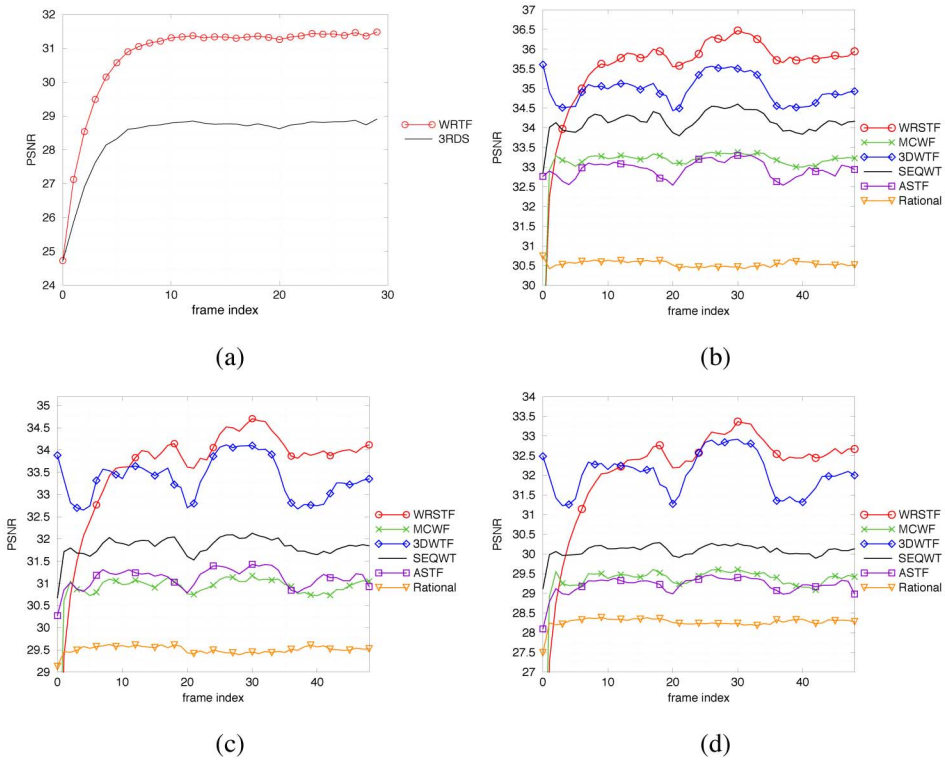

Fig. 5. PSNR versus frame index. (a) "Bicycle" sequence with added Gaussian noise, $\sigma_n = 15$. (b) "Salesman" sequence with added Gaussian noise, $\sigma_n = 10$. (c) "Salesman" sequence with added Gaussian noise, $\sigma_n = 15$. (d) "Salesman" sequence with added Gaussian noise, $\sigma_n = 20$. (Color version available online at: http://ieeexplore.ieee.org.)

We additionally note that the cost function in (7) is a nonlinear function of the MAD values from the corresponding wavelet bands. As a result, the estimated (single) MV field does not represent an averaged MV field of the corresponding wavelet bands (such as in [30]) but rather a nonlinear, structure-oriented combination of the MVs.

C. Recursive Temporal Filtering (RTF)

In this section we propose a new wavelet domain recursive temporal filtering (RTF) scheme which filters a video sequence along the estimated motion trajectories and adapts the amount of smoothing to the estimated reliability of the MVs. Specifically, recursive adaptive temporal filtering is performed separately in each noisy (nonprocessed) wavelet band $\mathrm{WB}_n^{(l)}(\mathbf{r},t)$ as follows:

$$\mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r},t) = \alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)\mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1) + \left(1-\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)\right)\mathrm{WB}_n^{(l)}(\mathbf{r},t) \quad (10)$$

where $\mathrm{WB}_{\mathrm{tf}}^{(l)}$ stands for the temporally processed wavelet band at scale l. The weighting factor $\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)$ controls the amount of filtering for each wavelet band $(\mathrm{WB}^{(l)})$ in the following way:

$$\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b) = b_{\mathrm{WB}}^{(l)}\left(\vartheta_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)\right)^2 \quad (11)$$

with $\vartheta_{\mathrm{WB}}^{(l)}$ the motion *reliability* per wavelet band WB (Section III-A) of the MV $\mathbf{v}_b$ and $b_{\mathrm{WB}}^{(l)}$ a normalizing parameter that we experimentally optimize in terms of the mean squared error. Specifically, in our two-scale decomposition implementation we determine the parameters for the first (finest) scale as $b_{\mathrm{WB}}^{(1)} = 0.9$, for the second scale as $b_{\mathrm{WB}}^{(2)} = 0.95$ and $b_{\mathrm{LL}}^{(2)} = 1.25$ for the low-pass (approximation) band (roughest scale). We have chosen the quadratic dependency in (11) because our experiments showed that it introduces less temporal blur and artifacts than e.g., a linear model which does not respond well to MV miss-matches. In addition, we have also tested higher degree models and found that the quadratic dependency model indeed provides the best results in terms of maximal PSNR of the denoised sequence.

The amount of temporal filtering applied in each wavelet band is of crucial importance. Not only will the filtered band be used for filtering future frames but for future motion estimation too. Therefore, if no reliable MV can be found for a certain block, we filter less. In such a manner, we avoid the propagation of any artifacts through the processed sequence.

Note that the values of $\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)$ are confined to [0,1] where $\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b) = 0$ means no filtering at all and $\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b) = 1$ means full filtering, i.e., the current noisy wavelet coefficient is replaced by the corresponding filtered coefficient from the previous frame $(\mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r},t) = \mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1))$. However, in the latter case a problem may occur when $\mathrm{WB}_{\mathrm{tf}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1)$ has not been sufficiently filtered and hence a noisy wavelet coefficient

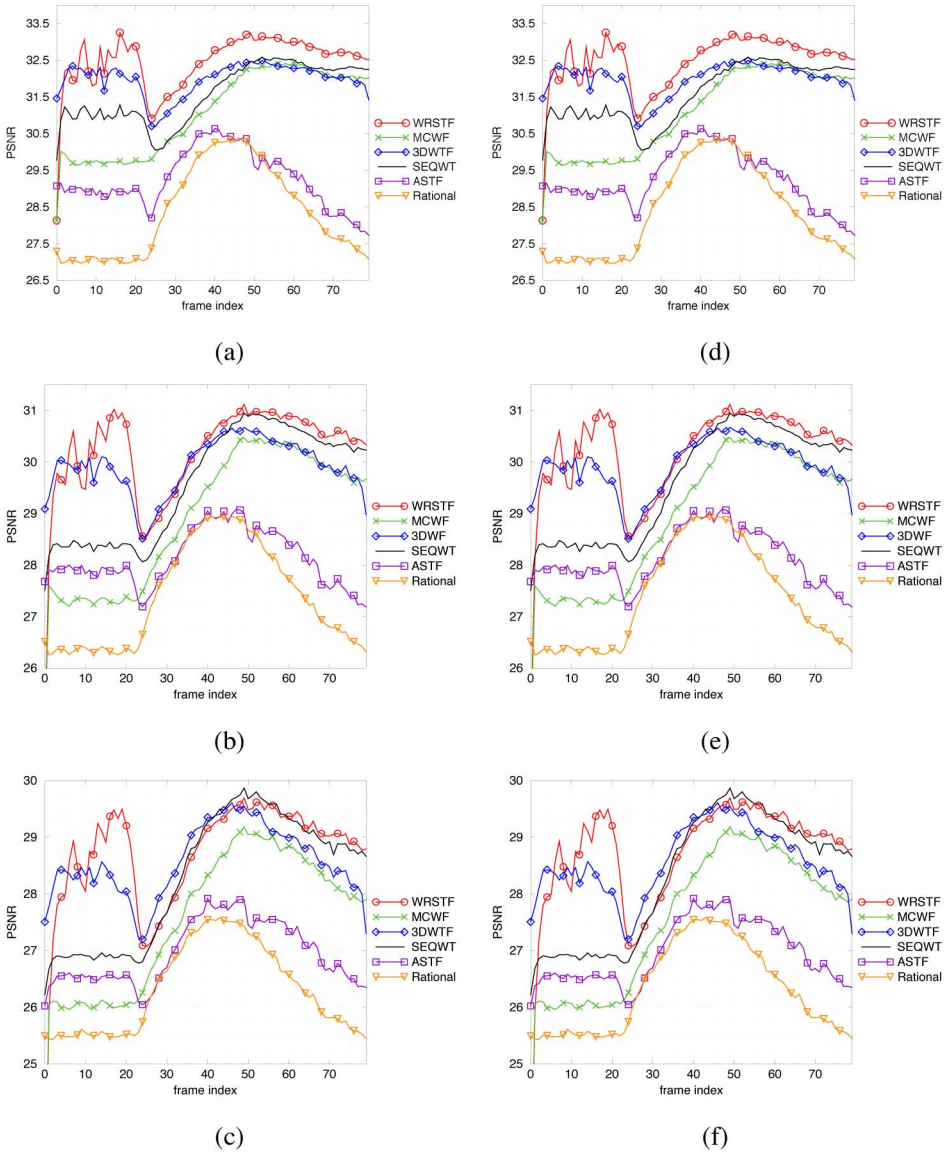

Fig. 6. PSNR versus frame index. (a) "Tennis" sequence with added Gaussian noise, $\sigma_n = 10$. (b) "Tennis" sequence with added Gaussian noise, $\sigma_n = 15$. (c) "Tennis" sequence with added Gaussian noise $\sigma_n = 20$. (d) "Flower Garden" sequence with added Gaussian noise, $\sigma_n = 10$. (e) "Flower Garden" sequence with added Gaussian noise, $\sigma_n = 15$. (f) "Flower Garden" sequence with added Gaussian noise $\sigma_n = 20$. (Color version available online at: http://ieeexplore.ieee.org.)

will propagate through the sequence. To solve this problem we update $\alpha_{\mathrm{WB}}^{(l)}(s,t,\sigma_n,\mathbf{v}_b)$ defined in (11) with a *correction* function, as follows:

$$\alpha_{\mathrm{WB}}^{(l)\,*}(\mathbf{r},t) = \frac{\alpha_{\mathrm{WB}}^{(l)}(\mathbf{r},t)\left(1+\alpha_{\mathrm{WB}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1)\right)}{2} \quad (12)$$

and we use α^* instead of α. Note that the correction function (12) aims at reducing the amount of filtering in the current time-recursion when the amount of filtering in the previous frame is relatively low. In the case where $\alpha_{\mathrm{WB}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1) \approx 0$, we have $\alpha_{\mathrm{WB}}^{(l)\,*}(\mathbf{r},t) \approx 0.5\alpha_{\mathrm{WB}}^{(l)}(\mathbf{r},t)$; this is a reasonable choice since it means that in the case where the wavelet coefficient from both the current and previous frame are noisy, simple averaging is performed. On the other hand, when $\alpha_{\mathrm{WB}}^{(l)}(\mathbf{r}-\mathbf{v}_b,t-1) \approx 1$, we have $\alpha_{\mathrm{WB}}^{(l)\,*}(\mathbf{r},t) \approx \alpha_{\mathrm{WB}}^{(l)}(\mathbf{r},t)$. Furthermore, we apply full filtering $(\alpha = 1)$ in the case where at least two time-recursions with reliable MVs have been applied in the last two frames.

Because of the imperfections of the motion estimation process, due to various difficulties such as occlusion or an imperfect motion estimation model, the temporal filter still leaves some noise behind. This remaining noise is nonstationary because of the varying amount of filtering applied for different spatial positions $\mathbf{r}$. The nonstationary noise introduced by the recursive adaptive temporal filter is removed by the spatial filter proposed in Section III-D.

D. Adaptive Spatial Filtering

Spatial filtering is especially useful at higher noise levels, but even for lower noise levels it can significantly improve video quality. In order not to reduce the resolution of the input image sequence, one has to adapt filtering to the spatial details, i.e.,

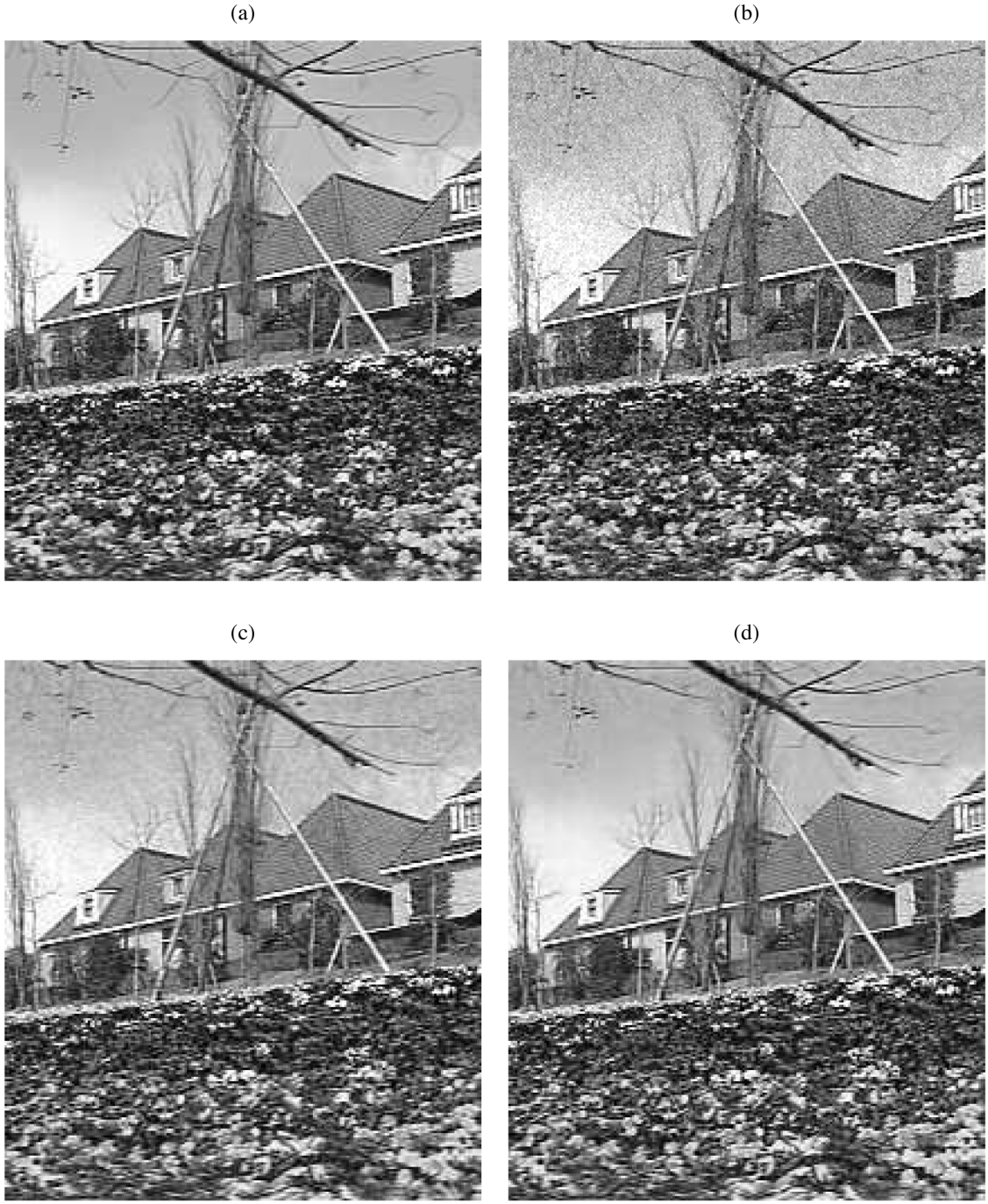

Fig. 7. Results for the 75th frame of the processed "Flower Garden" sequence with added Gaussian noise, $\sigma_n = 15$, by (c) the 3DWTF algorithm, and (d) the WRSTF algorithm. (a) Original image frame. (b) Noisy image frame.

take into account the (local and/or global) statistical distribution of the image such as in [12]. In our proposed scheme, where the temporal filter precedes the spatial one, the spatial filter has to deal with *nonstationary noise*, since the amount of temporal smoothing varies from one spatial position to another (according to the reliability of motion estimates). In certain rare cases where the MVs are not estimated properly, the temporal filter may introduce artifacts. In the literature a number of spatial adaptive methods have been proposed for spatio-stationary noise [12], [37]–[41]; their performance in the case of nonspatio-stationary noise is decreased and strongly depends on the spatial adaption to the local noise variance.

We propose a low-complexity method for spatial denoising of temporally filtered frames corrupted by nonstationary noise. The proposed spatial filter is an extension of our filter [6], [17], which adaptively averages the wavelet coefficients within a 2-D sliding window in such a way that the reliability of the motion estimates, i.e., the amount of preceding temporal filtering, in-

Fig. 8. Results for the 40th frame of the processed "Salesman" sequence with added Gaussian noise, $\sigma_n = 20$, by (a) the 3DWTF algorithm of Selesnick, and (b) the proposed WRSTF algorithm.

fluences the degree of spatial smoothing for the corresponding wavelet coefficients.[5]

Let $\delta(\mathbf{r}_c)$ denote the 3×3 neighborhood surrounding the central pixel $\mathbf{r}_c$ depicted in Fig. 3. Note that this neighborhood includes certain coefficients that are already spatio-temporally filtered $\mathrm{WB}^{(l)}_{\mathrm{stf}}(\mathbf{r}, t)$ and others that are only temporally filtered $\mathrm{WB}^{(l)}_{\mathrm{tf}}(\mathbf{r}, t)$. The proposed spatial filter is

$$\mathrm{WB}^{(l)}_{\mathrm{stf}}(\mathbf{r}_c, t) = \frac{\sum_{\mathbf{r}\in\delta(\mathbf{r_c})} w^{(l)}(\mathbf{r}, t)\mathrm{WB}^{(l)}_{f}(\mathbf{r}, t)}{\sum_{\mathbf{r}\in\delta(\mathbf{r})} w^{(l)}(\mathbf{r}, t)} \tag{13}$$

where the subscripts $f \in \{tf, stf\}$ denote temporally or spatio-temporally filtered coefficients, depending on their spatial position in the neighborhood, shown in Fig. 3. The weighting coefficients $w^{(l)}(\mathbf{r})$ are defined as follows:

$$w^{(l)}(\mathbf{r}, t) = \begin{cases} 0, & \text{if } \left|\mathrm{WB}^{(l)}_{\mathrm{tf}}(\mathbf{r}_c, t) - \mathrm{WB}^{(l)}_{f}(\mathbf{r}, t)\right| > k_m T \\ 1, & \text{otherwise} \end{cases} \tag{14}$$

where the threshold $T = \mathrm{MAD}^{(l)}_{\mathrm{WB}}(s, t, \mathbf{v}_b)$. The parameter k_m optimizes the performance of the spatial filter and ranges from $k_m = 0.75$ to $k_m = 1.25$ (in our implementation we have fixed it to $k_m = 1$, which for most sequences gives the best visual result). Apart from a particular sequence, the optimal k_m value can also depend on the noise model; in our case we have only considered Gaussian noise. Hence, the lower the MAD for the corresponding wavelet band $\mathrm{WB}^{(l)}$ and block s, the less we will average. In other words, the more the temporal filter reduces the noise, the weaker the spatial filtering that will be applied. This agrees with our goal; the proposed spatial filter is intended to suppress the remaining noise without seriously reducing the resolution of the input image sequence.

[5]We consider the averaging of wavelet coefficients in a small spatial neighborhood as one approach of a spatially adaptive soft-thresholding technique.

IV. Experimental Results

In our experiments we used 12 different grayscale sequences: "Salesman," "Miss America," "Bicycle," "Trevor," "Tennis," "Flower Garden," "Bus," "Mobile," "Chair," "Deadline," "Foreman," "Renata," and "Cargate." We added artificial Gaussian noise of the following standard deviation values: $\sigma_n = 5, 10, 15, 20, 25, 30$ and processed the sequences with different filters.[6]

For performance comparison, we use four wavelet-based methods: 1) the spatio-temporal bivariate nonseparable 3-D wavelet thresholding in a dual-tree complex wavelet representation of [4] but with a signal adaptive threshold of [39]. For a fair comparison with the best available methods, Prof. I. Selesnick kindly provided the results of their latest video denoising algorithm, 3DWTF; 2) the adaptive spatio-temporal filter (ASTF) of [14]; 3) the multiclass wavelet spatio-temporal filter (MCWF) of [6]; 4) the sequential wavelet domain and temporal filtering (SEQWT) of [12]; and two spatial domain filters; 5) the rational filter (Rational) of [3]; and 6) the temporal recursive filter (3RDS) of [16]. For some methods [4], [14], [39], their authors processed our sequences; we implemented some other methods [3], [6], [12], [16] ourselves. The results of the processed sequences, along with the proposed method and with the methods used for comparison, can be viewed at the following link: http://telin.ugent.be/~vzlokoli/Results_J.

A. Denoising Results

We first evaluated the performance of the proposed temporal recursive filter only (WRTF), in terms of temporal blur and noise reduction. Specifically, we have compared the denoising performance of the WRTF method with the 3RDS algorithm, visually (Fig. 4) and in terms of PSNR [Fig. 5(a)]. For implementing the compared temporal recursive 3RDS filter, which uses a block matching based motion estimation and compensation approach

[6]The range of grayscale values is assumed to be [0,255].

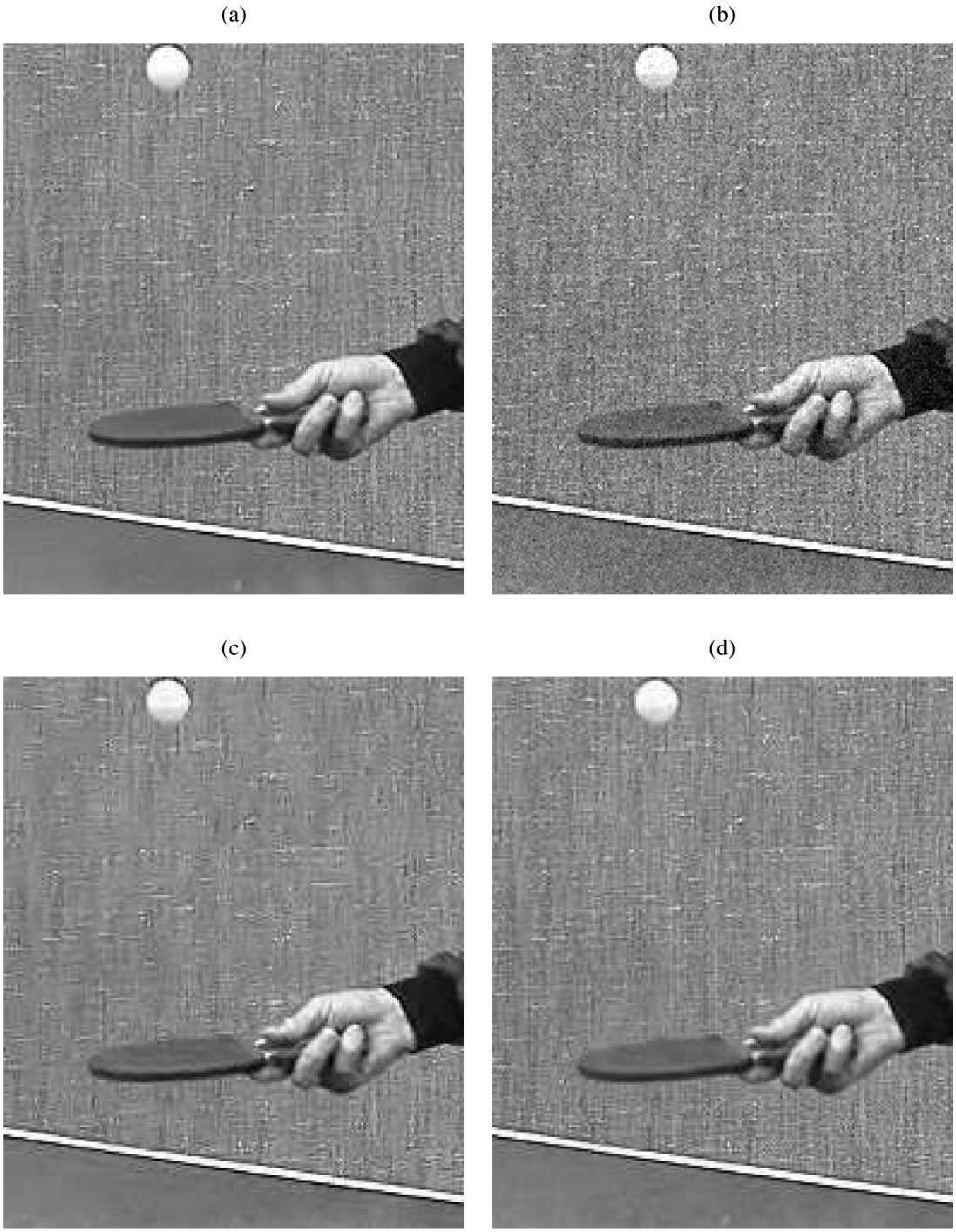

Fig. 9. Results for the 20th frame of the processed "Tennis" sequence with added Gaussian noise, $\sigma_n = 15$, by (c) the SEQWT algorithm and (d) the WRSTF algorithm. (a) Original image frame. (b) Noisy image frame.

in the base domain, we computed the recursion variable k as follows: $k(s) = k_r\sigma_n/\mathrm{MAD}(s)$; $\mathrm{MAD}(s)$ corresponds to the mean absolute difference of displacement of block s for the estimated MV, while k_r is noise reduction parameter (approximately equals 0.3, but the optimal value depends on noise level). For motion estimation, the spatio-temporal neighboring MVs used are as shown in Fig. 2. In Fig. 4, the visual result for a portion of the 29th frame of the "Bicycle" sequence, corrupted with Gaussian noise ($\sigma_n = 15$), is shown. The results show how certain details, such as the spokes of the bicycle wheel, are better restored by the proposed WRTF filter, whereas the 3RDS filter introduces artifacts. In Fig. 5(a) the filters are compared in terms of PSNR for the "Bicycle" sequence with added Gaussian noise of $\sigma_n = 15$. The WRTF filter is approximately 3 dB better than the 3RDS.

Next, the proposed WRSTF filter was compared to the other methods in terms of 1) objective criteria: PSNR and in terms of 2) subjective criteria: visual quality.

TABLE I
SUBJECTIVE EVALUATION OF THE ALGORITHMS' PERFORMANCE; 1=3DWTF; 2=SEQWT; 3=WRSTF. THE ORDER OF THE NUMBERS CORRESPONDS TO THE RANKING OF THE RESULTS ACCORDING TO VISUAL QUALITY

	$\sigma_n = 10$								
image sequence	**Person 1**			**Person 2**			**Person 3**		
	overall quality	noise reduction	least artifacts	overall quality	noise reduction	least artifacts	overall quality	noise reduction	least artifacts
Salesman	3 1 2	3 1 2	3=1 2	3 1 2	3 1 2	1=3 2	3 2 1	3 2 1	3 1 2
FlowerGar.	3 2 1	3 2 1	1 3=2	3 2 1	3 2 1	3 2 1	3 2 1	3=2 1	3 2 1
Tennis	3 1 2	3 2 1	1 3 2	3 2 1	3 2 1	3 1 2	3 1 2	2 3 1	3 1 2
	$\sigma_n = 15$								
Salesman	1=3 2	1=3 2	3 1 2	3 1 2	3 1 2	1 3 2	3 1 2	3 2 1	1 3 2
FlowerGar.	3 2 1	3 2 1	1 3 2	3 2 1	2=3 1	3 1 2	3 1 2	3 2 1	3 1 2
Tennis	3 1 2	3 2 1	1 3 2	3 1 2	3 2 1	1 3 2	3 1 2	3=2 1	3 1 2
	$\sigma_n = 20$								
Salesman	1 3 2	3 1 2	1=3 2	1 3 2	1=3 2	1 3 2	3 1 2	3 2 1	1 3 2
FlowerGar.	3 2 1	3 2 1	1 3 2	3 1 2	3 2 1	3 1 2	3 1 2	3 2 1	3 1 2
Tennis	3 1 2	3 1 2	1 3 2	3 1 2	3 2 1	1 3 2	3 1 2	3 2 1	1 3 2
	$\sigma_n = 10$								
image	**Person 4**			**Person 5**			**Person 6**		
Salesman	3 2 1	3 1 2	3 1 2	3 2 1	3 2 1	1 3 2	3 1 2	3 2 1	1=3=2
FlowerGar.	3 2 1	3 2 1	3 1 2	1 3 2	3 1 2	1 3 2	3 2 1	3 2 1	3 1 2
Tennis	3 2 1	3 2 1	3 2 1	3 1=2	3 2 1	1 3 2	3 2=1	3 2 1	3 1=2
	$\sigma_n = 15$								
Salesman	3 1 2	1 3 2	1 3 2	3 2=1	3 2 1	3 1 2	3 1 2	3 1 2	3 1 2
FlowerGar.	3 1 2	3 1 2	3 1=2	3 1 2	3 1 2	1 3 2	3 2 1	3 2 1	3 1 2
Tennis	3 2 1	3 2 1	3 2 1	3 1 2	3 2 1	3 1 2	3 2=1	3 2 1	3 1=2
	$\sigma_n = 20$								
Salesman	1 3 2	1 3 2	1 3 2	3 2=1	3 2 1	1 3 2	3 1 2	3 1 2	3 1 2
FlowerGar.	3 1=2	3 1 2	3 1=2	3 1 2	3 1 2	1 3 2	3 2 1	3 2 1	3 1 2
Tennis	3 2 1	3 1 2	3 1 2	3 1 2	3 2 1	3=1 2	3 2 1	3 2 1	3 1 2

Figs. 5 and 6 display graphs of the PSNR versus frame index for three image sequences with three different noise levels. In Fig. 5(b)–(d), the PSNR values for the "Salesman" sequence are shown. For all three noise levels ($\sigma_n = 10, 15, 20$), the proposed WRSTF method produces the best PSNR for all time instances. In comparison with the 3DWTF technique, the average improvement is approximately 0.5–1 dB and in comparison with SEQWT the average improvement is 2 dB. The MCWF and the ASTF methods are shown to have very similar PSNR performances, with approximately 3 dB lower PSNR, on average, compared to the proposed WRSTF method. We note that the reduced PSNR of the proposed WRSTF method in first 5–10 frames is due to the convergence time within the proposed recursive scheme.

For the "Tennis" sequence the results are shown in Fig. 6(a)-(c). The proposed WRSTF method yields a slightly better PSNR compared to the 3DWTF and SEQWT methods. Namely, it is on average 0.3, 0.9, 1.5, and 2 dB better than the 3DWTF, SEQWT, MCWF, and the ASTF methods, respectively. The reason for the reduced PSNR in one part of the sequence (frame number 20–35) is mostly due to zooming. For the proposed motion estimation model without a *zooming feature model*, we cannot get reliable MVs in the case of video zooming and hence have to filter less, resulting in a reduced PSNR. This at least does not introduce artifacts (see Fig. 9) and the spatial filter still performs noise suppression to some degree.

In Fig. 6(d)–(f), the PSNR for the "Flower Garden" sequence is shown for three noise levels ($\sigma_n = 10, 15, 20$). The proposed WRSTF method again outperforms the other compared methods, in terms of PSNR, where the average improvement over the 3DWTF method is 0.4 dB. Although this is relatively small improvement, compared to the improvement over the other methods: 1 dB for SEQWT, 1.5 dB MCWF, 3 dB ASTF, and 4 dB for the rational filter, the gain is significant. In certain PSNR graphs the results for some methods were not shown since their performance was significantly lower (less than 4 dB in comparison with the proposed WRSTF method). Because the "Flower Garden" sequence contains a great deal of complicated texture, the subjective improvement is not sufficiently reflected in the PSNR graphs. Hence, to show the real improvement of the proposed WRSTF algorithm we refer the reader to the denoised (processed) video sequence frames (or parts of them, see Fig. 7) on our website.

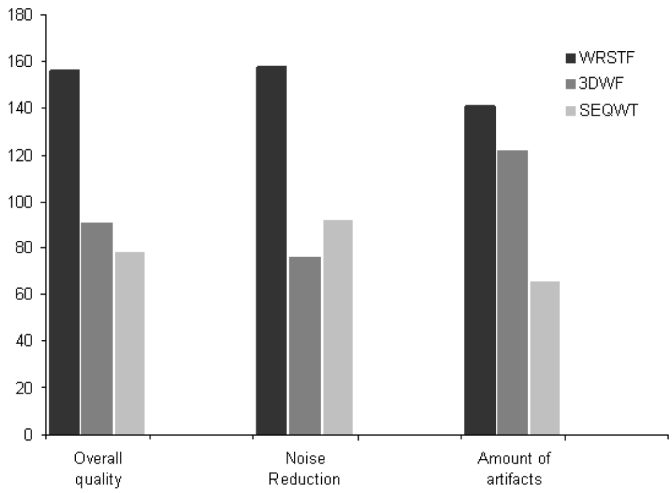

Fig. 10. Results of the subjective evaluation.

(a) Noisy (b) WRSTF

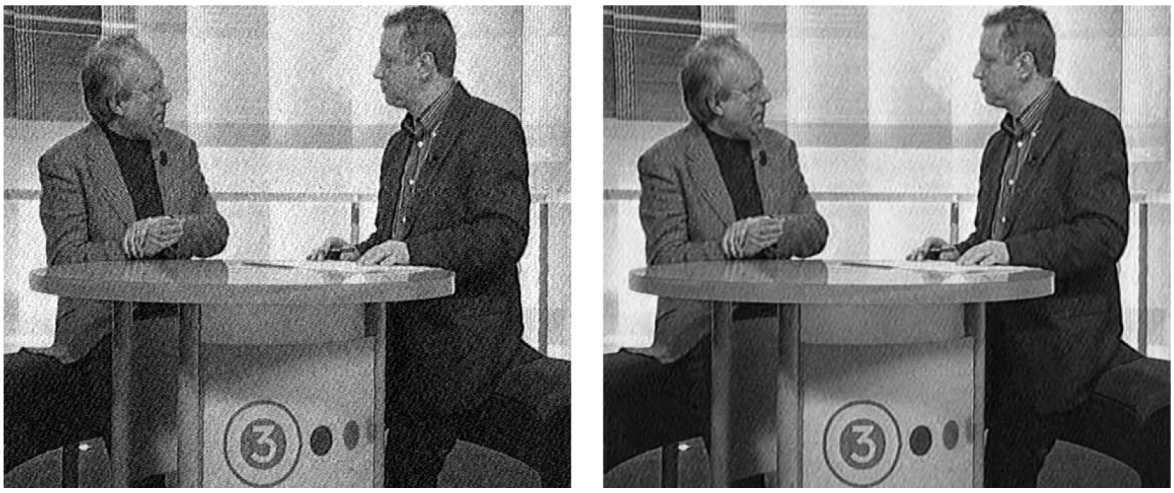

Fig. 11. Result for the 103th frame of the TV sequence (kanaal3). (a) Noisy and (b) processed by the WRSTF filter.

Although from a PSNR point of view our method is not always significantly better than the other methods, it invariably performs best visually, as can be seen in Figs. 7 and 8, in comparison with the 3DWTF filter and in Fig. 9 in comparison to SEQWT filter. Fig. 7 demonstrates that the proposed WRSTF filter better suppresses noise in uniform areas than the 3DWTF, while preserving texture equally well. Fig. 8 illustrates that our method also outperforms the 3DWTF method, in terms of spatio-temporal blur, for a relatively high noise level $(\sigma_n = 20)$ and fast movements (the "Salesman" sequence). Additionally, Fig. 9 shows that the proposed WRSTF method better preserves texture than the SEQWT method. The improvements of the proposed method over the reference ones can be even better seen by viewing the processed image sequences at our website: http://telin.ugent.be/~vzlokoli/Results_J.

The visual (subjective) quality was evaluated by a panel of six people (four experts and two nonexperts in image processing). For the three best methods from the PSNR point of view, i.e., 1) the 3DWTF; 2) the SEQWT; and 3) the WRSTF algorithm, the results were shown simultaneously, along with the input noisy sequence. The panel members were asked to rank the three algorithms in terms of 1) overall quality; 2) noise reduction; and 3) amount of visible artifacts. In the case where the person could not decide which one was better, the corresponding methods were ranked equally. The experiment was carried out on three different sequences, that is "Salesman," "Flower Garden," and "Tennis" and with three different noise levels $(\sigma_n = 10, 15, 20)$. The sequences used for the experiment can be viewed on the website: http://telin.ugent.be/~vzlokoli/Results_J/subj_eval/. Table I shows the results of the experiment, where the numbers denote the filters. 1: 3DWTF; 2: SEQWT; 3: WRSTF, and order of the numbers corresponds to the ranking of the results according to visual quality.

The results in Table I demonstrate that 1) in terms of overall quality the proposed WRSTF algorithm was judged to be best in 90% of the cases; 2) in terms of noise reduction the proposed WRSTF method was found to be best in 87% of the cases; and 3) in terms of the amount of visible artifacts the WRSTF method was preferred in 55% of the cases, in comparison with the other two methods. Fig. 10 summarizes the results of the subjective comparison evaluation in a different way, where each method was given 3 points when it was ranked first, 2 points when it was ranked second, and 1 point when it was ranked third. Specifically, Fig. 10 shows the average score for overall quality, noise and artifacts. The results confirm that the proposed WRSTF method scores best in all three aspects. In terms of visible artifacts, it is very close in performance to the 3DWTF algorithm. However, in terms of noise reduction it scores much better than SEQWT and 3DWTF, which show similar performance here.

TABLE II
AVERAGE COMPUTATION TIME REQUIRED FOR THE PROCESSING OF ONE FRAME OF A CIF SEQUENCE (SIZE: 352 × 288), ON A AMD ATHLON 64 (4000+), 2.4 GHZ PROCESSOR (WITH 2 GB RAM) AND A GNU/LINUX OPERATING SYSTEM (IN C++).

method	**required time per frame** (sec/frame)
Rational	0.092
3RDS	0.146
MCWF	0.756
SEQWT	0.814
WRSTF	0.866
ASTF	0.904

In addition, based on our own observation, we deduced the following. The ASTF of [14], designed for noise reduction in video coding, showed relatively good performance in the case of relatively low noise levels $(\sigma_n \approx 10)$. However, for higher noise levels $(\sigma_n \approx 20)$, it does not sufficiently reduce noise and introduces blocky artifacts that are mostly due to the failure of motion estimation and compensation in a highly noisy environment. The MCWF of [6] also displayed relatively good performance for lower noise levels. For higher noise levels the MCWF filter did not introduce artifacts, but reduced the resolution of the input sequence significantly and in some case introduced spiky impulse-like artifacts. Finally, the rational filter (Rational) of [3] displayed the worst performance of all. Nevertheless, it should be noted that the rational filter is of the lowest complexity and it can still produce relatively good denoising results, for certain images without significant spatial details and slow moving objects.

Finally, the denoising results for the processed video sequences with real noisy scenarios by the proposed WRSTF filter are given on: http://telin.ugent.be/~vzlokoli/Results_J/New_Method/RealSeq/, along with the noisy sequences, for subjective evaluation of the denoising results. Specifically, we processed two sequences corrupted with white Gaussian-like noise (one with relatively lower and the other higher noise level) and two sequences with colored noise. From the results, it can be concluded that the proposed WRSTF algorithm removes noise sufficiently well in case of white Gaussian noise and performs slightly less efficiently in case of colored noise; some noise is still left after denoising, but essentially there is a clear improvement of the noisy sequence. Fig. 11 illustrates the denoising performance of the WRSTF method on one TV sequence corrupted with colored-like noise.

B. Computational Complexity RSTF

We have evaluated the computation complexity of the proposed WRSTF method in terms of the required time for processing. On a AMD Athlon 64 (4000+), 2.4-GHz processor (with 2 GB RAM), and a GNU/Linux operating system (in C++), in the case of CIF sequences (size: 352×288) and 50 frames, we obtained the following results for the tested video denoising algorithms, shown in Table II.[7] The processing time for the WRSTF technique was approximately 35 seconds, which corresponds to approximately 1.5 frames/s.[8]

As can be seen from the Table II, the base-domain techniques, i.e., the Rational and 3RDS, require the least processing time. On the other hand, wavelet-based techniques, the MCWF, SEQWT, WRSTF, and ASTF, are significantly slower and require approximately five times more processing time (the difference in complexity between the proposed WRSTF method and the other compared (MCWF, SEQWT, and ASTF) is small).

For the proposed WRSTF method, about 30% of the total processing time is spent on the wavelet transform and approximately 30% on motion estimation. The rest (40%) is required by the proposed spatio-temporal filter. The proposed WRSTF method could be implemented in real-time video applications with possible optimization concerning the computation of the wavelet transform, with a simplified scheme for motion estimation and compensation (for a specific purpose). We note that the computational complexity of the proposed algorithm was evaluated for an implementation that had not been fully optimized for speed.

V. Conclusion and Future Work

In this paper, we have proposed a new method for motion estimation and image sequence denoising in the wavelet domain. By robustly estimating motion and compensating for it appropriately, we efficiently remove noise without introducing visual artifacts. In future work, we intend to refine our motion estimation framework in order to deal with occlusion and "moving block edges," i.e., to refine the MV estimation process for blocks undergoing two or more different motion.

[7]The 3DWTF method could not be tested because we didn't have access to the corresponding code.

[8]For this experiment, we processed the "Salesman" sequence in progressive format.

Acknowledgment

The authors would like to thank to Prof. Selesnick from the Polytechnic University, New York, for his cooperation and for providing us with the processed video sequences for the 3DWTF algorithm which were used for the comparison. They would also like to give a special thanks to Dr. A. M. Tourapis for providing them with the processed sequences by the ASTF algorithm and his precious suggestions.

References

[1] G. De Haan, *Video Processing for Multimedia Systems*. Eindhoven, The Netherlands: University Press Eindhoven, 2003.

[2] K. Jostschulte, A. Amer, M. Schu, and H. Schroder, "Perception adaptive temporal tv-noise reduction using contour preserving prefilter techniques," *IEEE Trans. Consum. Electron.*, vol. 44, no. 3, pp. 1091–1096, Aug. 1998.

[3] F. Cocchia, S. Carrato, and G. Ramponi, "Design and real-time implementation of a 3-D rational filter for edge preserving smoothing," *IEEE Trans. Consum. Electron.*, vol. 43, no. 4, pp. 1291–1300, Nov. 1997.

[4] W. I. Selesnick and K. Y. Li, "Video denoising using 2d and 3d dual-tree complex wavelet transforms," in *Proc. SPIE Wavelet Applicat. Signal Image Process.*, San Diego, CA, Aug. 2003, pp. 607–618.

[5] N. Rajpoot, Z. Yao, and R. Wilson, "Adaptive wavelet restoration of video sequences," in *Proc. Int. Conf. Image Process.*, Singapore, 2004, pp. 957–960.

[6] V. Zlokolica, A. Pizurica, and W. Philips, "Video denoising using multiple class averaging with multiresolution," in *Proc. Int. Workshop VLBV03*, Madrid, Spain, Sep. 2003, pp. 172–179.

[7] R. Rajagopalan and M. T. Orchard, "Synthesizing processed video by filtering temporal relationships," *IEEE Trans. Image Process.*, vol. 11, no. 1, pp. 26–36, Jan. 2002.

[8] O. A. Ojo and T. G. Kwaaitaal-Spassova, "An algorithm for integradted noise reduction and sharpness enhacement," *IEEE Trans. Consum. Electron.*, vol. 46, no. 3, pp. 474–480, Aug. 2000.

[9] R. Dugad and N. Ahuja, "Video denoising by combining kalman and wiener estimates," in *Proc. IEEE Int. Conf. Image Process.*, Kobe, Japan, Oct. 1999, vol. 4, pp. 156–159.

[10] R. P. Kleihorst, R. L. Legendrijk, and J. Biemond, "Noise reduction of image sequences using motion compensation and signal decomposition," *IEEE Trans. Image Process.*, vol. 4, no. 3, pp. 274–284, Mar. 1995.

[11] P. M. B. Van Roosmalen, S. J. P. Westen, R. L. Lagendrijk, and J. Biemond, "Noise reduction for image sequences using an oriented pyramid thresholding technique," in *Int. Conf. Image Process.*, Lausanne, Switzerland, 1996, pp. 375–378.

[12] A. Pizurica, V. Zlokolica, and W. Philips, "Noise reduction in video sequences using wavelet-domain and temporal filtering," in *Proc. SPIE Conf. Wavelet Applicat. Industrial Process.*, Providence, RI, Oct. 2003, pp. 48–59.

[13] J. C. Brailean, R. P. Kleihorst, S. Efstratidis, K. A. Katsaggeleos, and R. L. Lagendijk, "Noise reduction filters for dynamic image sequences: a review," *Proc. IEEE*, vol. 83, no. 9, pp. 1272–1292, Sep. 1995.

[14] H. Y. Cheong, A. M. Tourapis, J. Llach, and J. Boyce, "Adaptive spatio-temporal filtering for video de-noising," in *Proc. Int. Conf. Image Process.*, Singapore, 2004, pp. 965–968.

[15] K. O. Mehmet, S. Ibrahim, and T. Murat, "Adaptive motion-compensated filtering of noisy image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 4, pp. 277–290, Aug. 1993.

[16] G. De Haan, "IC for motion-compensated de-interlacing, noise reduction and picture rate conversion," *IEEE Trans. Consum. Electron.*, vol. 45, no. 3, pp. 617–623, Aug. 1999.

[17] V. Zlokolica, A. Pizurica, and P. Wilfried, "Recursive temporal denoising and motion estimation of video," in *Proc. Int. Conf. Image Process.*, Singapore, 2004, pp. 1465–1468.

[18] A. M. Tekalp, *Digital Video Processing*. Upper Saddle River, NJ: Prentice Hall, 1995.

[19] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. 29, no. 12, pp. 1799–1808, Dec. 1981.

[20] M. H. Chan, Y. B. Yu, and A. G. Constantinides, "Variable size block matching motion compensation with application to video coding," in *Proc. IEE*, 1990, vol. 137, pp. 205–212.

[21] Y.-Q. Zhang and S. Zafar, "Motion-compensated wavelet transform coding for video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 3, pp. 285–196, Aug. 1992.
[22] Y. Yuan and K. M. Mandal, "Low-band-shifted hierarchical backward motion estimation and compensation for wavelet-based video coding," in *Proc. 3rd Indian Conf. Comput. Vis., Graph. Image Process.*, Ahmedbad, India, 2002, pp. 185–190.
[23] R. Li and M. L. Lio, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 4, pp. 438–442, Aug. 1994.
[24] G. De Haan, W. C. A. Biezen, H. Huijgen, and O. A. Ojo, "True-motion estimation with 3-D recursive search block matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 5, pp. 368–379, Oct. 1993.
[25] H. Blume, J. Von Livonius, and T. G. Noil, "Segmentation in the loop: An iterative, object based algorithm for motion estimation," in *Proc. SPIE Video Commun. Image Process.*, 2004, pp. 464–473.
[26] R. Braspenning and G. De Haan, "True-motion estimation using feature correspodences," in *Proc. SPIE Video Commun. Image Process.*, 2004, pp. 396–407.
[27] J. Zan, M. O. Ahmad, and M. N. S. Swamy, "New techniques for multiresolution motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 9, pp. 793–802, Sep. 2002.
[28] M. K. Mandal, E. Chan, X. Wong, and S. Panchanathalu, "Multiresolution motion estimation techniques for video compression," *Opt. Eng.*, vol. 35, no. 1, pp. 128–136, Jan. 1996.
[29] J. Skowronski, "Pel recursive motion estimation and compensation in subbands," *Signal Process.: Image Commun.*, vol. 14, pp. 389–396, 1999.
[30] H. W. Park and H. S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 577–587, Apr. 2000.
[31] J. Chalidabhongse and C.-C. J. Kuo, "Fast motion vector estimation using multiresolution-spatio-temporal correlations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 3, pp. 477–487, Jun. 1997.
[32] B. C. Song and K.-W. Chun, "Multiresolution block matching algorithm and its vlsi architecture for fast motion estimation in an mpeg-2 video encoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 9, pp. 1119–1137, Sep. 2004.
[33] S. Cui, Y. Wang, and E. J. Fowler, "Multihypothesis motion compensation in redundant wavelet domain," in *Proc. Int. Conf. Image Process.*, 2003, pp. 53–56.
[34] J. C. Ye and M. Van Der Schaar, "Fully scalable 3-d overcomplete wavelet video coding using adaptive motion compensated temporal filtering," in *Proc. SPIE Video Commun. Image Process.*, 2003, pp. 1169–1180.
[35] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. New York: Academic, 1999.
[36] V. Zlokolica, A. Pizurica, and W. Philips, "Wavelet domain noise-robust motion estimation and noise estimation for video denoising," presented at the 1st Int. Workshop Video Process. Quality Metrics Consum. Electron., Scotssdale, AZ, Jan. 2005, Paper no. 200.
[37] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Trans. Image Process.*, vol. 9, no. 9, pp. 1522–1531, Sep. 2000.
[38] M. K. Mihcaak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 300–303, Jun. 1999.
[39] L. Sendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependences," *IEEE Trans. Image Process.*, vol. 50, no. 11, pp. 2744–2756, Nov. 1999.
[40] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Process.*, vol. 11, no. 5, pp. 545–557, May 2002.
[41] J. Portilla, V. Strella, M. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

Vladimir Zlokolica (M'03) was born in Novi Sad, Serbia and Montenegro on June 23, 1975. He received the M.Sc. degree in electrical engineering from the University of Novi Sad, Serbia and Montenegro, in 2001 and the Ph.D. degree in applied sciences from the Department of Telecommunications and Information Processing, Ghent University, Ghent, Belgium.

Currently, he is a Postdoctoral Researcher in the Department of Telecommunications and Information Processing, Ghent University. His research interests include video processing, motion estimation, multiresolution representations, noise estimation, and objective quality assessment of video.

Aleksandra Pižurica (M'98) was born in Novi Sad, Serbia and Montenegro, in 1969. She received the Diploma degree in electrical engineering from the University of Novi Sad, Novi Sad, Serbia and Montenegro, in 1994, the M.Sc. degree in telecommunications from the University of Belgrade, Belgrade, Serbia and Montenegro, in 1997, and the Ph.D. degree in applied sciences from the Ghent University, Ghent, Belgium, in 2002.

Currently, she is a Postdoctoral Researcher in the Department of Telecommunications and Information Processing, Ghent University. Her research interests include statistical signal and image modeling, multiresolution representations, signal detection and estimation, and video processing.

Wilfried Philips (M'92) was born in Aalst, Belgium, on October 19, 1966. he received the Diploma degree in electrical engineering and the Ph.D. degree in applied sciences, both from Ghent University, Ghent, Belgium, in 1989 and 1993, respectively.

From October 1989 until October 1997, he worked at the Department of Electronics and Information Systems, Ghent University, for the Flemish Fund for Scientific Research (FWO-Vlaanderen), first as a Research Assistant and later as a Postdoctoral Research Fellow. Since November 1997, he has been with the Department of Telecommunications and Information Processing, Ghent University, where he is currently a full-time Professor. His main research interests are image and video restoration, image analysis, and lossless and lossy data compression of images and video, and processing of multimedia data.