SURE-LET for Orthonormal Wavelet-Domain Video Denoising

Florian Luisier, *Member, IEEE,* Thierry Blu, *Senior Member, IEEE,* and Michael Unser, *Fellow, IEEE*

***Abstract*—We propose an efficient *orthonormal* wavelet-domain video denoising algorithm based on an appropriate integration of motion compensation into an adapted version of our recently devised Stein's unbiased risk estimator-linear expansion of thresholds (SURE-LET) approach. To take full advantage of the strong spatio-temporal correlations of neighboring frames, a global motion compensation followed by a *selective* block-matching is first applied to adjacent frames, which increases their temporal correlations without distorting the interframe noise statistics. Then, a multiframe interscale wavelet thresholding is performed to denoise the current central frame. The simulations we made on standard grayscale video sequences for various noise levels demonstrate the efficiency of the proposed solution in reducing additive white Gaussian noise. Obtained at a lighter computational load, our results are even competitive with most state-of-the-art *redundant* wavelet-based techniques. By using a cycle-spinning strategy, our algorithm is in fact able to outperform these methods.**

***Index Terms*—Block-matching, Stein's unbiased risk estimator-linear expansion of thresholds (SURE-LET), video denoising, wavelet.**

I. Introduction

VIDEO PROCESSING has been an active area of research in the past 20 years. Despite the recent advances in image sequence acquisition and transmission, denoising still remains an essential step before performing higher level tasks, such as coding, compression, object tracking or pattern recognition. Since the origins of the degradations are numerous and diverse [imperfection of the charge-coupled device (CCD) detectors, electronic instabilities, thermal fluctuations, etc.], the overall noise contribution is often modeled as an additive (usually Gaussian) white process, independent from the original uncorrupted image sequence [1].

Manuscript received May 12, 2009; revised August 10, 2009 and November 26, 2009. Date of publication March 15, 2010; date of current version June 3, 2010. This work was supported by the Center for Biomedical Imaging of the Geneva–Lausanne Universities, and the Ecole Polytechnique Fédérale de Lausanne, the Leenaards and Louis-Jeantet foundations, the Swiss National Science Foundation, under Grant No. 200 020-109 415, and the Hong Kong Research Grants Council, under Grant No. CUHK 410 209. This paper was recommended by Associate Editor X. Li.

F. Luisier and M. Unser are with the Biomedical Imaging Group, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland (e-mail: florian.luisier@epfl.ch; michael.unser@epfl.ch).

T. Blu is with the Department of Electronic Engineering, Chinese University of Hong Kong, Shatin NT, 8520 Hong Kong, China (e-mail: thierry.blu@m4x.org).

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

Digital Object Identifier 10.1109/TCSVT.2010.2045819

The huge amount of correlations present in every video sequences has quite early led the researchers to develop combined spatio-temporal denoising algorithms, instead of sequentially applying available 2-D tools. The emergence of new multiresolution tools, such as the wavelet transform [2], [3] then gave an alternative to the standard noise reduction filters that were used for video denoising [4]–[7]. Now, the transform-domain techniques in general, and the wavelet-based in particular [8]–[15], have been shown to outperform these spatio-temporal linear and even nonlinear filtering.

In this letter, we stay within the scope of wavelet-domain video denoising techniques. More precisely, and contrary to most of the existing techniques [8]–[11], [14], [15], we consider an *orthonormal* wavelet transform rather than redundant representations, because of its appealing properties (energy and noise statistics preservation) and its lower computational complexity. To take into account the strong temporal correlations between adjacent frames, we work out a *multiframe* wavelet thresholding based on the recently devised Stein's unbiased risk estimator-linear expansion of thresholds (SURE-LET) strategy [16] and on its multichannel extension [17]. The principle is to parametrize our wavelet estimator as a linear expansion of thresholds (LET) and minimize an extended version of Stein's unbiased risk estimator (SURE) [18] to determine the best linear parameters of this expansion. To increase the correlations between adjacent frames, we compensate for interframe motion using a global motion compensation followed by a *selective* block-matching procedure. The selectivity is obtained by first performing a coarse interframe motion detection and then only matching those blocks inside which, a *significant* motion occurred. Thanks to its selectivity, the proposed block-matching has a negligible influence on the interframe noise covariance matrix. This latter point is crucial for the efficiency of our SURE-LET algorithm. Instead, standard block-matching [19] would make it difficult to track the interframe noise statistics.

This letter is organized as follows. In the next section, we recall the SURE-LET principle and show its multiframe expression; in Section III, we first present a selective block-matching algorithm that can be well integrated in the SURE-LET framework, and then expose the proposed multiframe interscale wavelet thresholding; in Section IV, we demonstrate the efficiency of our solution by comparing the results with those obtained by some state-of-the-art *redundant* techniques. To better outline the potential of our approach, we also provide the results of a cycle-spinning version (five shift averages) of

our algorithm; the quality attained is on par with the best video denoising algorithms.

II. SURE-LET Principle

We denote an original (unknown) video sequence of T frames containing N pixels by

$$\mathbf{v} = [\mathbf{v}_1\ \mathbf{v}_2\ \dots\ \mathbf{v}_N] \text{ with } \mathbf{v}_n = [v_{n,1}\ v_{n,2}\ \dots\ v_{n,T}]^{\mathrm{T}}. \quad (1)$$

We also define a unitary $T \times 1$ vector $\mathbf{e}_t$ such that $\mathbf{e}_t^{\mathrm{T}}\mathbf{v}_n = v_{n,t}$, and we assume that the observed noisy video sequence is given by $\mathbf{u} = \mathbf{v} + \mathbf{n}$, where $\mathbf{n}$ is an additive white Gaussian noise independent of $\mathbf{v}$, with known $T \times T$ interframe covariance matrix $\mathbf{R}$.

In an *orthonormal* wavelet representation, the observation model is preserved in the transformed domain, as well as the interframe noise covariance matrix $\mathbf{R}$. Therefore, each noisy wavelet coefficient $\mathbf{y}_n^j \in \mathbb{R}^T$, $j = 1 \dots J$, $n = 1 \dots N^j$ is given by

$$\mathbf{y}_n^j = \mathbf{x}_n^j + \mathbf{b}_n^j \text{ where } \mathbf{b}_n^j \sim \mathcal{N}(\mathbf{0}, \mathbf{R}). \quad (2)$$

Hereafter, we recall the general principle of the SURE-LET [16] denoising strategy, and show how it can be adapted to video denoising.

A. Stein's Unbiased Risk Estimate (SURE)

SURE [18] is an unbiased statistical estimate of the mean-squared error (MSE) between an original unknown signal and a processed version of its noisy observation. This estimate depends only on the observed data and does not require any prior assumption on the noise-free signal. The only statistical assumption is made on the noise: additive and Gaussian.

Denoting by $\hat{\mathbf{v}}$, an estimate of the noise-free video $\mathbf{v}$, we can define the global MSE as

$$\begin{aligned}\mathrm{MSE} &= \frac{1}{NT}\sum_{t=1}^{T}\underbrace{\sum_{n=1}^{N}\mathbf{e}_t^{\mathrm{T}}(\hat{\mathbf{v}}_n - \mathbf{v}_n)(\hat{\mathbf{v}}_n - \mathbf{v}_n)^{\mathrm{T}}\mathbf{e}_t}_{N\times\mathrm{MSE}_t} \\ &= \frac{1}{NT}\sum_{t=1}^{T}\sum_{j=1}^{J}\underbrace{\sum_{n=1}^{N^j}\mathbf{e}_t^{\mathrm{T}}(\hat{\mathbf{x}}_n^j - \mathbf{x}_n^j)(\hat{\mathbf{x}}_n^j - \mathbf{x}_n^j)^{\mathrm{T}}\mathbf{e}_t}_{N^j\times\mathrm{MSE}_t^j} \quad (3)\end{aligned}$$

where $\mathbf{e}_t^{\mathrm{T}}\hat{\mathbf{x}}_n^j = \theta_t^j(\mathbf{y}_n^j, \mathbf{p}_n^j)$ is the nth pixel of the jth wavelet subband of the denoised frame t. It is obtained by thresholding the nth pixel of the jth wavelet subband of the noisy frame t, taking into account (some of) its neighboring frames. From now on, we will drop the subband superscript "j" and the time frame indication "t" for the sake of clarity, when no ambiguities arise.

Considering this multiframe processing $\theta : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$, the MSE of any wavelet subband j of any frame t can be estimated *without bias* by

$$\epsilon = \frac{1}{N}\sum_{n=1}^{N}\Big[\big(\theta(\mathbf{y}_n, \mathbf{p}_n) - \mathbf{e}_t^{\mathrm{T}}\mathbf{y}_n\big)^2 + 2\mathbf{e}_t^{\mathrm{T}}\mathbf{R}\nabla_1\theta(\mathbf{y}_n, \mathbf{p}_n) - N\mathbf{e}_t^T\mathbf{R}\mathbf{e}_t\Big]. \quad (4)$$

Here, $\mathbf{p}_n$ denotes any random variables statistically independent of $\mathbf{y}_n$. ∇_1 stands for the gradient operator relatively to the *first* variable of the function θ, i.e., $\mathbf{y}_n$ (see [17]).

B. Linear Expansion of Thresholds (LET)

The thresholding function is specified by a linear combination of basic thresholding functions, a strategy that we have coined LET, that is

$$\theta(\mathbf{y}_n, \mathbf{p}_n) = \underbrace{[\mathbf{a}_1^{\mathrm{T}}\ \mathbf{a}_2^{\mathrm{T}}\dots\mathbf{a}_K^{\mathrm{T}}]}_{\mathbf{a}^{\mathrm{T}}}\underbrace{\begin{bmatrix}\boldsymbol{\theta}_1(\mathbf{y}_n, \mathbf{p}_n)\\ \boldsymbol{\theta}_2(\mathbf{y}_n, \mathbf{p}_n)\\ \vdots \\ \boldsymbol{\theta}_K(\mathbf{y}_n, \mathbf{p}_n)\end{bmatrix}}_{\boldsymbol{\theta}(\mathbf{y}_n, \mathbf{p}_n)} \quad (5)$$

where $\mathbf{a}$ and $\boldsymbol{\theta}$ are both $KT \times 1$ vectors. Each $\boldsymbol{\theta}_k : \mathbb{R}^T \times \mathbb{R}^T \longrightarrow \mathbb{R}^T$ is an arbitrary vector-valued thresholding that will be specified in Section III-C.

Thanks to this linear parameterization, the optimal—in the minimum ϵ sense—parameters of (5) are the solution of a *linear* system of equations

$$\mathbf{a}_{\mathrm{opt}} = \mathbf{M}^{-1}\mathbf{C} \quad (6)$$

where
$$\begin{cases}\mathbf{M} = \displaystyle\sum_{n=1}^{N}\boldsymbol{\theta}(\mathbf{y}_n, \mathbf{p}_n)\boldsymbol{\theta}(\mathbf{y}_n, \mathbf{p}_n)^{\mathrm{T}} \\ \mathbf{C} = \displaystyle\sum_{n=1}^{N}\Big(\boldsymbol{\theta}(\mathbf{y}_n, \mathbf{p}_n)\mathbf{y}_n^{\mathrm{T}} - \big(\nabla_1\boldsymbol{\theta}(\mathbf{y}_n, \mathbf{p}_n)\big)^{\mathrm{T}}\mathbf{R}\Big)\mathbf{e}_t.\end{cases}$$

In video denoising, SURE is particularly robust (i.e., close to the actual MSE) due to the high number of available samples. Therefore, it can be reliably used to optimize a large number of parameters ($K > 100$ per frame).

III. Algorithm

The video is going to be denoised *frame by frame*, by considering a sliding temporal window of τ (odd) neighboring frames centered around the current frame. For instance, the denoising of the *reference* frame t will involve frames $t - (\tau - 1)/2$ to $t + (\tau - 1)/2$.

The various steps of the proposed algorithm (Fig. 1) are the following. We first align all the neighboring frames (*global* registration) and compensate for their (*local*) motion, with respect to the the frame t. Then, this reference frame is processed in the wavelet domain, using thresholds based on the values of the wavelet coefficients of the aligned neighboring frames, and on their own coarser scale coefficients (multiframe interscale SURE-LET thresholding). Finally, an inverse wavelet transform is performed on the denoised coefficients of this reference frame. These steps are detailed in Sections III-A, III-B, and III-C.

A. Global Motion Compensation

As a global motion model, we can simply consider the translations due to camera motions (pan/tilt). The optimal

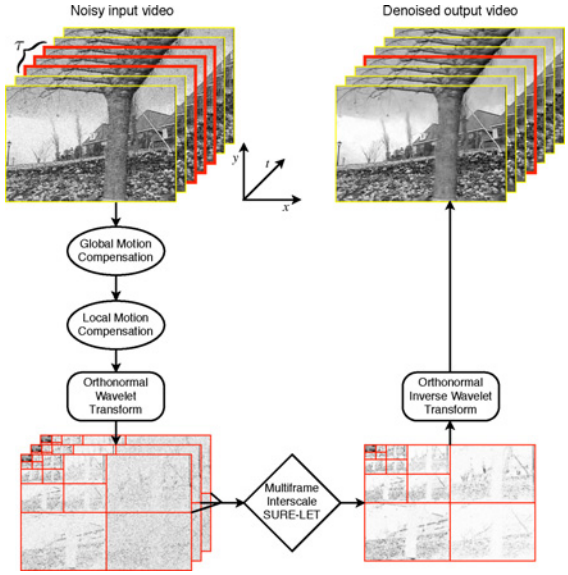

Fig. 1. Overview of the proposed denoising algorithm.

integer shift s_{opt} required to register a given frame $t+\Delta t$ with respect to the reference frame t, is the index of the maximum of the cross-correlation function between the two frames [14], [20], that is

$$s_{\text{opt}} = \text{argmax}_s \, \mathcal{F}^{-1}\left\{U_t(\cdot)U^*_{t+\Delta t}(\cdot)\right\}(s) \tag{7}$$

where $\mathcal{F}^{-1}\{\cdot\}$ denotes the inverse discrete Fourier transform and $U_t(\omega)$, $U_{t+\Delta t}(\omega)$ are respectively the discrete Fourier transforms of the reference frame and of the current frame.

B. Local Motion Compensation by Selective Block-Matching

A global motion model does not reflect the local interframe motions. Block-matching [19] is a standard procedure used in video processing to compensate for these local interframe motions. Here, each of the $\tau - 1$ neighboring frames is replaced by a version that is motion-compensated with respect to the reference frame. Considering one of these neighboring frames, motion compensation is performed as follows: the reference frame is divided into blocks;[1] then, for each block of this frame, a search for similar blocks is performed in the neighboring frame; the compensated frame is then built by pasting the best matching block of the neighboring frame at the location of the reference block. Several parameters are therefore involved.

1) The size of the considered blocks: We found that rectangular blocks of fixed size 8×16 were a good trade-off between accurate motion estimation, robustness toward noise and computational complexity. Note, that a rectangular shape is well-adapted to the standard video format, which are not of squared size.
2) The size of the search region: Here again, the trade-off evoked above led us to consider a square region of 15×15 pixels centered around the position of the reference block. Note that we obtained similar results with a rectangular search region of 11×21 pixels.
3) The criterion used for measuring the similarity between blocks: The two most popular measures of similarities are the mean of the absolute difference and the MSE. We experimentally observed that the MSE gave slightly better results.
4) The way of exploring the search region: We retained the exhaustive search because of its simplicity and accuracy. Note that there is a huge amount of literature (e.g., [21]–[23]) exposing fast algorithms for efficiently exploring the search region.

[1]In this letter, we only consider nonoverlapping blocks. Note that better peak signal-to-noise ratio (PSNR) results (0.2–0.7 dB) can be obtained with overlapping blocks, but the computational burden then becomes heavier.

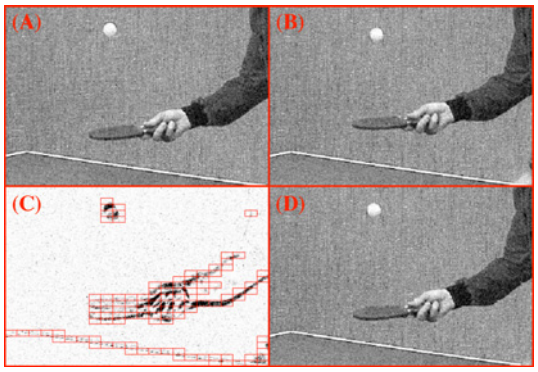

Fig. 2. (a) Frame no. 3 of the *Tennis* sequence: PSNR = 22.11 dB. (b) Frame no. 6 of the *Tennis* sequence (reference frame): PSNR = 22.11 dB. (c) Detected motion with corresponding blocks to be matched. (d) Motion compensated frame no. 3.

Instead of trying to find the best matches for every block of the reference frame, we consider only blocks where a significant motion occurred. Indeed, in noisy video sequences there is a strong risk of matching the noise component in the still regions. In that case, the interframe noise becomes locally highly correlated [see Fig. 3(b)]. To avoid this risk and still be able to consider the interframe noise as stationary (with a good approximation), we propose to perform motion compensation only in the blocks, where a significant motion between frames was detected, as illustrated in Fig. 2.

The proposed motion detection involves the following two steps.

1) In order to be robust with respect to noise, the considered frames are smoothed by the following regularized Wiener filter:

$$H(\omega) = \begin{cases} 1 - \dfrac{|N(\omega)|^2}{|U(\omega)|^2}, & \text{if } |U(\omega)|^2 > \lambda_1 |N(\omega)|^2 \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where $|N(\omega)|^2$ and $|U(\omega)|^2$ are respectively the power spectrum of the noise (constant for white Gaussian noise) and of the noisy frame. $\lambda_1 \geq 1$ is the regularization parameter; its value will be discussed hereafter.
2) The MSEs between the two considered frames are then computed inside each block. The minimum of these MSEs (MSE_{min}) is considered as the "no motion level." Consequently, the block-matching will be only performed for those blocks of the reference frame having

a MSE above a given threshold of motion λ_2 $\text{MSE}_{\min}$, where $\lambda_2 \geq 1$.

In our experiments, we found that any values of λ_1 and λ_2 chosen in the range $[2;3]$ gave similar results (± 0.1 dB). A smaller value of these two parameters will decrease the robustness with respect to noise. A higher value of the regularization parameter λ_1 will oversmooth the frames, decreasing the accuracy of the subsequent block-matching. A higher value of the parameter λ_2 will speed up the algorithm, but the subsequent motion compensation will be less effective. In practice, we have selected $\lambda_1 = \lambda_2 = \sqrt{6}$.

The block-matching itself is performed on the smoothed frames, in order to decrease the sensitivity to noise. For each frame and for each detected block, the minimum MSE (computed between the reference block and its best matching block) is stored; the inverse of the average of these MSEs will then serve as a weight q_t for the considered frame t in the subsequent wavelet-domain thresholding (Section III-C). These weights are especially important when there is no or little correlation between adjacent frames; this situation appears when, for example, a quick change of camera occurs.

The proposed *selective* block-matching procedure has two key advantages.

1) It leads to a fast local motion compensation, despite the fact that an exhaustive search is performed.
2) The interframe noise covariance matrix can be assumed to be unaffected by the local motion compensation [Fig. 3(c)], contrary to standard block-matching [Fig. 3(b)].

C. Multiframe Interscale Wavelet Thresholding

Once the motion between a reference frame and a reasonable number of adjacent frames has been compensated, a 2-D *orthonormal* wavelet transform is applied to each motion-compensated frame. Each highpass subband of the reference frame is then denoised according to the generic procedure described in Section II-B, (5), in which $K = 4$ and

$$\begin{aligned}\theta(\mathbf{y}_n, \mathbf{p}_n) = \mathbf{a}_1^{\mathrm{T}} & \underbrace{\gamma(\mathbf{p}_n^{\mathrm{T}}\mathbf{W}\mathbf{p}_n)\gamma(\mathbf{y}_n^{\mathrm{T}}\mathbf{W}\mathbf{y}_n)\mathbf{y}_n}_{\boldsymbol{\theta}_1(\mathbf{y}_n,\mathbf{p}_n)} \\ + \mathbf{a}_2^{\mathrm{T}} & \underbrace{\overline{\gamma}(\mathbf{p}_n^{\mathrm{T}}\mathbf{W}\mathbf{p}_n)\gamma(\mathbf{y}_n^{\mathrm{T}}\mathbf{W}\mathbf{y}_n)\mathbf{y}_n}_{\boldsymbol{\theta}_2(\mathbf{y}_n,\mathbf{p}_n)} \\ + \mathbf{a}_3^{\mathrm{T}} & \underbrace{\gamma(\mathbf{p}_n^{\mathrm{T}}\mathbf{W}\mathbf{p}_n)\overline{\gamma}(\mathbf{y}_n^{\mathrm{T}}\mathbf{W}\mathbf{y}_n)\mathbf{y}_n}_{\boldsymbol{\theta}_3(\mathbf{y}_n,\mathbf{p}_n)} \\ + \mathbf{a}_4^{\mathrm{T}} & \underbrace{\overline{\gamma}(\mathbf{p}_n^{\mathrm{T}}\mathbf{W}\mathbf{p}_n)\overline{\gamma}(\mathbf{y}_n^{\mathrm{T}}\mathbf{W}\mathbf{y}_n)\mathbf{y}_n}_{\boldsymbol{\theta}_4(\mathbf{y}_n,\mathbf{p}_n)} \end{aligned} \quad (9)$$

where

1) $\gamma(x) = \exp\left(-\frac{|x|}{2\lambda_3^2}\right)$ and $\overline{\gamma}(x) = 1 - \gamma(x)$ are two discriminative functions that classify the wavelet coefficients in four groups, based on their magnitude and the magnitude of their parent $\mathbf{p}_n$.[2] λ_3 is a threshold that rules this categorization of the wavelet coefficients. The numerous[3] linear parameters involved in the multiframe thresholding bring a high level of flexibility to the denoising process. As a consequence, the nonlinear parameter λ_3 does not require a data-dependent optimization; we experimentally found that $\lambda_3 = \lambda_2 = \lambda_1 = \sqrt{6}$ gave the best results. Note that this value is the same that we used for multichannel denoising in [17].
2) Each $\mathbf{a}_k$ is a $\tau \times 1$ vector of linear parameters that is optimized for each subband by the procedure described in Section II-B, (6).
3) $\mathbf{W} = \mathbf{Q}^{\mathrm{T}}\mathbf{R}_\tau^{-1}\mathbf{Q}$ is a $\tau \times \tau$ weighting matrix that takes into account:
   a) the potential interframe signal-to-noise (SNR) disparities, through the inverse of the $\tau \times \tau$ interframe noise covariance matrix $\mathbf{R}_\tau$;
   b) the potential weak interframe correlations, through the weights q_t resulting from the block-matching (Section III-B) and stored in the $\tau \times \tau$ diagonal matrix $\mathbf{Q}$. The weights are normalized to ensure that the Frobenius norm of $\mathbf{Q}$, $\|\mathbf{Q}\|_2 = \sqrt{\text{trace}\{\mathbf{Q}^{\mathrm{T}}\mathbf{Q}\}} = 1$.

[2]These interscale predictors (parents) $\mathbf{p}_n$ are obtained by a rigorous procedure based on group-delay compensation (see [17]).

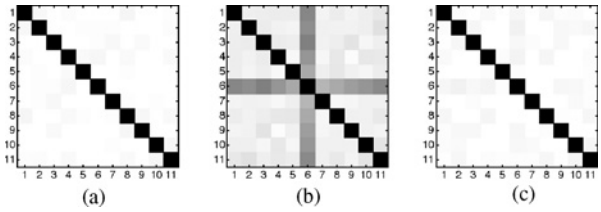

Fig. 3. (a) Interframe noise covariance matrix for the 11 first frames of the noisy *Tennis* sequence before motion compensation (frame no. 6 is the reference frame). (b) Interframe noise covariance matrix after a standard block-matching algorithm. (c) Interframe noise covariance matrix after the proposed selective block-matching algorithm.

D. Computational Complexity

To denoise a given reference frame using its τ neighboring frames, the computational complexity of our algorithm can be evaluated as follows.

1) Global motion compensation: $\mathcal{O}((\tau - 1) \cdot N \cdot \log_2(N))$.
2) Local motion compensation: $\mathcal{O}((\tau - 1) \cdot B_x \cdot B_y \cdot R_x \cdot R_y \cdot \frac{N}{S_x \cdot S_y})$, where $B_x \times B_y = 8 \times 16$ is the block size, $R_x \times R_y = 15 \times 15$ the size of the search region and $(S_x, S_y) = (8, 16)$ the step size between two adjacent reference blocks.
3) Orthonormal discrete wavelet transform: $\mathcal{O}(\tau \cdot N \cdot \log_2(N))$.
4) Construction of the interscale predictor: $\mathcal{O}(\tau \cdot N \cdot \log_2(N))$.
5) Application of the interscale wavelet thresholding: $\mathcal{O}(\tau \cdot K \cdot N)$.

When summing up all these operations, we get approximately 2800 operations per pixel (ops/pix) to denoise one $N = 288 \times 352$ frame using its $\tau = 11$ neighboring frames. However, since the proposed block matching procedure is selective, the actual number of operations is much lower in practice.

[3]Four times the considered number of adjacent frames per subband.

IV. Experiments

We propose now to evaluate the performance of our algorithm in comparison to some other state-of-the-art video denoising methods (all are redundant).

1) Pižurica *et al. sequential wavelet domain and temporal filtering (SEQWT)* [9]: A spatially adaptive Bayesian shrinkage is applied in the undecimated wavelet-domain, followed by a recursive temporal filtering.
2) Zlokolica *et al. wavelet domain recursive spatio-temporal filtering (WRSTF)* [10]: In the undecimated wavelet domain, motion estimation and adaptive temporal filtering are recursively performed, followed by an intraframe spatially adaptive filter.
3) Dabov *et al.* video block-matching and 3-D filtering (*VBM3D*) [24]: The first step of this hybrid two-step algorithm consists of a 3-D spatio-temporal block-matching followed by a 3-D wavelet-domain hard-thresholding. A first denoised estimate is then obtained by an aggregation of the redundant blockwise estimates. The second step is very similar, except that: the block-matching is performed on the first estimate; the 3-D wavelet transform is replaced by a 2-D discrete cosine transform, followed by a 1-D wavelet transform and the hard-thresholding is replaced by a Wiener filter. Up to our knowledge, the PSNRs obtained by the *VBM3D* are among the best published so far for video denoising.
4) Jovanov *et al.* algorithm [15]: This very recent video denoising algorithm extends the *SEQWT* by integrating a filtered version of the motion field estimated by a standard real-time video codec. Contrary to the other video denoising methods evaluated in this section, this algorithm is designed for *real-time* applications, and thus, it uses one previous frame only.

The results of the above first two denoising algorithms can be downloaded at http://telin.ugent.be/ vzlokoli/Results_J/. Some noise-free and noisy video sequences can be downloaded at http://bigwww.epfl.ch/luisier/VideoDenoising/, together with our own denoising results. The results of the *VBM3D* have been obtained by running the corresponding MATLAB code.[4] The authors of [15] have kindly provided us with their denoised sequences. This allows a fair comparison between the various methods. The noisy video sequences have been simulated by adding (without clipping) independent white Gaussian noises of given variance σ^2 on each frame, i.e., $\mathbf{R} = \sigma^2\mathbf{Id}$. For our algorithm, we performed four levels of decomposition of an *orthonormal* wavelet transform using Daubechies *symlet* filters with eight vanishing moments [3]. $\tau = 11$ adjacent frames (five past, five future and the current frame) were considered in our multiframe interscale thresholding (9) to denoise each current frame.

In Fig. 4, we show the peak signal-to-noise ratio (PSNR = $10\log_{10}\frac{255^2}{\mathrm{MSE}}$ dB) in each frame of various video sequences at various input PSNRs. We can observe that our nonredundant solution achieves globally, and for almost every frame, significantly better results than the three purely wavelet-based techniques. Yet, it is usually outperformed by the more sophisticated *VBM3D*. We must point out that these results are very encouraging since we are only considering a *nonredundant* procedure, whereas the other algorithms take advantage of their redundancy. Note that significantly better results can be obtained by increasing the shift-invariance of the proposed algorithm. Indeed, by averaging five cycle-spins (CS) of the whole denoising process, we can reach a PSNR gain of up to 1 dB (see Table I), while maintaining a reasonable computational complexity (at most $\sim 5 \times 2800 = 14\,000$ ops/pix).

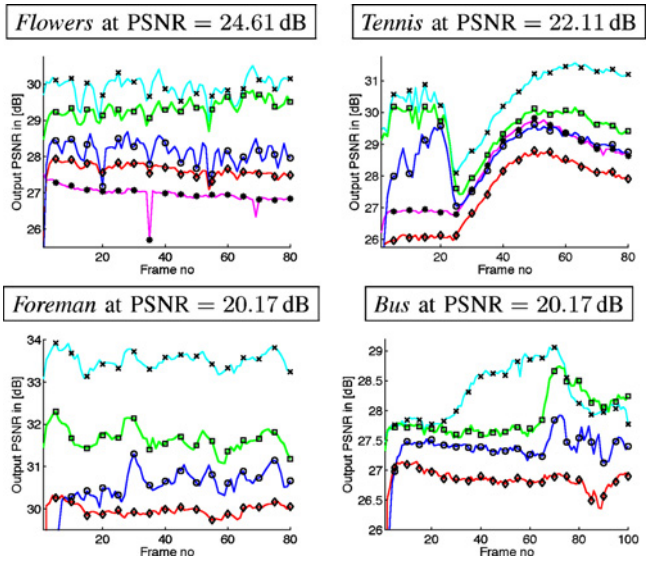

Fig. 4. Comparison of the PSNR evolution for various video sequences and denoising algorithms."□" markers refer to the proposed algorithm, "∗" to [9], "○" to [10], "◇" to [15] (to be added), and "×" to [24].

In Table I, we show a global PSNR comparison of the various algorithms. As observed, the *nonredundant* variant of the proposed algorithm consistently gives higher PSNR (+1 dB) than the other *redundant* wavelet-based approaches, while being usually outperformed by the *VBM3D*. Yet, we notice that the slightly redundant variant of our algorithm is very competitive with the state-of-the-art *VBM3D*.

We have also reported in Table I the results obtained by the recent K-means singular value decomposition (*K-SVD*) video denoising algorithm [25]. Note that, for this algorithm, the noisy videos have been clipped prior to denoising and the denoised videos have been clipped and cropped prior to PSNR computation.[5] This brings a significant gain (up to 1 dB) over the other denoising algorithms compared in this section (especially under heavy noise conditions).

From a computational standpoint, the method described in [25] requires ∼75 000 ops/pix, which corresponds to the cost of averaging 26 CS of the proposed algorithm. In [24], there is no analysis of algorithm's complexity. However, based on the complexity formula given in their paper on image denoising [26], we obtain an overall number of ∼15 500 ops/pix (similar to averaging ∼5 CS of the proposed algorithm). From a visual point of view, our solution provides a good trade-off between noise reduction and preservation of small features (see Fig. 5).

[4]Available at: http://www.cs.tut.fi/~foi/GCF-BM3D /#ref_software.

[5]Private communication with the authors of [25].

TABLE I
COMPARISON OF SOME STATE-OF-THE-ART VIDEO DENOISING ALGORITHMS

σ	5	10	15	20	25	30	50	100
Input PSNR	34.15	28.13	24.61	22.11	20.17	18.59	14.15	8.13
Method	***Tennis*** 240 × 352							
SEQWT [9]	N/A	30.96	28.34	26.88	N/A	N/A	N/A	N/A
WRSTF [10]	N/A	32.57	30.45	28.96	27.83	N/A	N/A	N/A
VBM3D [24]	**37.82**	**34.14**	**32.10**	30.39	28.84	27.41	24.81	22.13
Proposed	37.38	33.52	31.41	29.96	28.85	27.94	25.50	22.60
Proposed (5 CS)	37.81	34.01	31.90	**30.44**	**29.32**	**28.41**	**26.04**	**23.42**
K-SVD [25]	*38.16*	*34.33*	*32.10*	*30.51*	*29.32*	*28.43*	*26.34*	*N/A*
Method	***Flowers*** 240 × 352							
SEQWT [9]	N/A	29.62	27.10	25.32	N/A	N/A	N/A	N/A
WRSTF [10]	N/A	30.77	28.10	26.33	24.92	N/A	N/A	N/A
VBM3D [24]	36.51	32.04	29.66	**28.07**	**26.82**	**25.74**	21.53	17.27
Proposed	36.22	31.63	29.11	27.31	25.89	24.76	21.73	18.54
Proposed (5 CS)	**36.59**	**32.18**	**29.75**	27.99	26.59	25.44	**22.32**	**18.93**
K-SVD [25]	*36.73*	*32.16*	*29.69*	*28.03*	*26.80*	*25.56*	*22.80*	*N/A*
Method	***Foreman*** 288 × 352							
WRSTF [10]	N/A	35.33	33.14	31.55	30.30	N/A	N/A	N/A
VBM3D [24]	40.21	**37.19**	**35.59**	**34.39**	**33.39**	**32.52**	**29.75**	24.02
Proposed	39.60	36.13	34.13	32.73	31.61	30.71	28.15	24.85
Proposed (5 CS)	**40.26**	36.87	34.94	33.54	32.42	31.49	28.91	**25.75**
Method	***Bus*** 288 × 352							
WRSTF [10]	N/A	32.78	30.40	28.76	27.46	N/A	N/A	N/A
VBM3D [24]	37.28	32.88	30.49	28.95	27.81	26.93	24.40	20.74
Proposed	37.45	33.13	30.69	28.99	27.74	26.74	24.04	21.12
Proposed (5 CS)	**37.97**	**33.79**	**31.39**	**29.71**	**28.46**	**27.45**	**24.67**	**21.62**

Note: PSNRs displayed in this table correspond to the averaged values over frames 10–20 of the various video sequences, using frames 5–25 to avoid potential boundary artifacts in the temporal dimension. PSNR results of the *K-SVD* have not been obtained under the same conditions as the other algorithms (see the text).

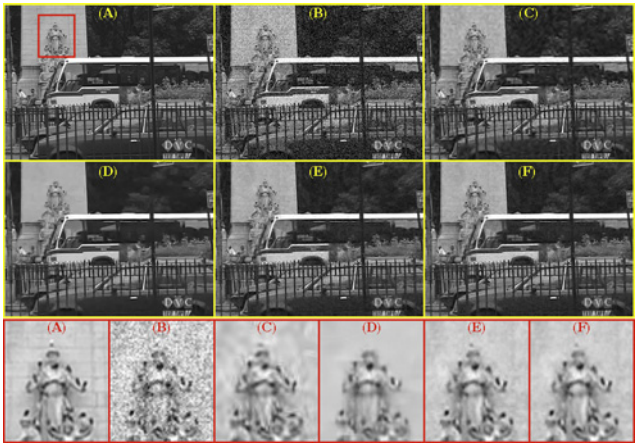

Fig. 5. (a) Part of the frame no. 29 of the *Bus* video sequence. (b) Noisy version of it: PSNR = 20.17 dB. (c) Result of [15]: PSNR = 26.82 dB. (d) Result of [24]: PSNR = 27.92 dB. (e) Result of nonredundant SURE-LET: PSNR = 27.58 dB. (f) Result of SURE-LET (5 CS): PSNR = 28.36 dB.

V. CONCLUSION

In this letter, we have presented a relatively simple and yet very efficient *orthonormal* wavelet-domain video denoising algorithm. Thanks to a proper selective block-matching procedure, the effect of motion compensation on the noise statistics became negligible, and an adapted multiframe interscale SURE-LET thresholding could be applied. The proposed algorithm has been shown to favorably compare with most state-of-the-art *redundant* wavelet-based approaches, while having a lighter computational load. However, it is necessary to increase the shift-invariance of the proposed solution to reach the same level of performance as the very best video denoising algorithms available [24], [25].

REFERENCES

[1] A. C. Bovik, "Multi-frame image restoration," in *Handbook of Image and Video Processing*, A. C. Bovik, Ed. Amsterdam, The Netherlands: Elsevier, Jun. 2005, pp. 219–234.
[2] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.
[3] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pure Appl. Math.*, vol. 41, no. 7, pp. 909–996, 1988.
[4] M. Ozkan, M. Sezan, and A. Tekalp, "Adaptive motion-compensated filtering of noisy image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 4, pp. 277–290, Aug. 1993.
[5] J. C. Brailean, R. P. Kleihorst, S. Efstratiadis, A. K. Katsaggelos, and R. L. Lagendijk, "Noise reduction filters for dynamic image sequences: A review," *Proc. IEEE*, vol. 83, no. 9, pp. 1272–1292, Sep. 1995.
[6] J. Kim and J. Woods, "Spatio-temporal adaptive 3-D Kalman filter for video," *IEEE Trans. Image Process.*, vol. 6, no. 3, pp. 414–424, Mar. 1997.
[7] 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.
[8] I. W. Selesnick and K. Y. Li, "Video denoising using 2-D and 3-D dual-tree complex wavelet transforms," in *Proc. 10th Soc. Photo-Optic. Instrum. Eng. Conf. Wavelets: Applicat. Signal Image Process.*, vol. 5207. Nov. 2003, pp. 607–618.
[9] A. Pižurica, V. Zlokolica, and W. Philips, "Noise reduction in video sequences using wavelet-domain and temporal filtering," in *Proc. Soc. Photo-Optic. Instrum. Eng. Conf. Wavelet Applicat. Ind. Process.*, vol. 5266. Feb. 2004, pp. 48–59.

[10] V. Zlokolica, A. Pižurica, and W. Philips, "Wavelet-domain video denoising based on reliability measures," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 8, pp. 993–1007, Aug. 2006.
[11] 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. Video Technol.*, vol. 16, no. 2, pp. 220–230, Feb. 2006.
[12] N. Gupta, M. Swamy, and E. Plotkin, "Wavelet domain-based video noise reduction using temporal discrete cosine transform and hierarchically adapted thresholding," *IET Image Process.*, vol. 1, no. 1, pp. 2–12, Jan. 2007.
[13] S. M. M. Rahman, F. M. Omair Ahmad, and M. N. S. 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.
[14] G. Varghese and Z. Wang, "Video denoising using a spatiotemporal statistical model of wavelet coefficients," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2008, pp. 1257–1260.
[15] L. Jovanov, A. Pizurica, S. Schulte, P. Schelkens, A. Munteanu, E. Kerre, and W. Philips, "Combined wavelet-domain and motion-compensated video denoising based on video codec motion estimation methods," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 417–421, Mar. 2009.
[16] T. Blu and F. Luisier, "The SURE-LET approach to image denoising," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2778–2786, Nov. 2007.
[17] F. Luisier and T. Blu, "SURE-LET multichannel image denoising: Interscale orthonormal wavelet thresholding," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 482–492, Apr. 2008.
[18] C. Stein, "Estimation of the mean of a multivariate normal distribution," *Ann. Stat.*, vol. 9, no. 6, pp. 1135–1151, 1981.
[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] R. Manduchi and G. Mian, "Accuracy analysis for correlation-based image registration algorithms," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 1993, pp. 834–837.
[21] R. Li, B. Zeng, and M. Liou, "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.
[22] J. Lu and M. Liou, "A simple and efficient search algorithm for block-matching motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 429–433, Apr. 1997.
[23] C.-H. Cheung and L.-M. Po, "A novel cross-diamond search algorithm for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 12, pp. 1168–1177, Dec. 2002.
[24] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3-D transform-domain collaborative filtering," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Poznań, Poland, Sep. 2007, pp. 1257–1260.
[25] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 27–35, Jan. 2009.
[26] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.