HOS-based image sequence noise removal

M. El Hassouni, Hocine Cherifi, Driss Aboutajdine

HOS-Based Image Sequence Noise Removal

Mohammed El Hassouni, Hocine Cherifi, and Driss Aboutajdine

***Abstract*—In this paper, a new spatiotemporal filtering scheme is described for noise reduction in video sequences. For this purpose, the scheme processes each group of three consecutive sequence frames in two steps: 1) estimate motion between frames and 2) use motion vectors to get the final denoised current frame. A family of adaptive spatiotemporal L-filters is applied. A recursive implementation of these filters is used and compared with its nonrecursive counterpart. The motion trajectories are obtained recursively by a region-recursive estimation method. Both motion parameters and filter weights are computed by minimizing the kurtosis of error instead of mean squared error. Using the kurtosis in the algorithms adaptation is appropriate in the presence of mixed and impulsive noises. The filter performance is evaluated by considering different types of video sequences. Simulations show marked improvement in visual quality and SNRI measures cost as well as compared to those reported in literature.**

***Index Terms*—Higher order statistics, L-filters, mixed noise, motion compensation, noisy video sequences, recursive implementation, spatiotemporal filters, step-size, video restoration.**

I. Introduction

NOWADAYS, the demand for digital video has increased. Video sequences occur naturally in multimedia services, such as autonomous navigation, telesurveillance, archiving and restoring of old films, defense and astronomical imaging [9], [13]. In many practical circumstances, these video sequences may be corrupted by a significant amount of noise. This noise degrades both visual quality and the effectiveness for subsequent performance such as coding, analysis or interpretation [11], [16], [25].

The classification of the existing image sequence filtering algorithms is based on the dimension of the region of support of the filter [4], [5]. More specifically, we consider one-dimensional (1-D) temporal filters, and three-dimensional (3-D) spatiotemporal filters. Three-dimensional temporal filters avoid the artifacts introduced by spatial filters by modeling the image sequence as a series of 1-D pixel trajectories. That is the intensity at each spatial position is considered as a 1-D signal that transverses along the temporal axis [30]. Spatiotemporal filters take advantage of the correlations that exist in both the temporal and spatial directions. Clearly, two-dimensional (2-D) spatial and 1-D temporal filters may be thought of in most cases as special cases of 3-D spatiotemporal filters.

The existing filters are further classified into non motion-compensated and motion-compensated [3]. Motion compensation is one of the ways for removing the temporal nonstationarity of the sequence which is due to its dynamic nature (see Fig. 1). A motion-compensated sequence is more stationary along the temporal direction than along the spatial ones, and this provides the support for design of temporal filters [10].

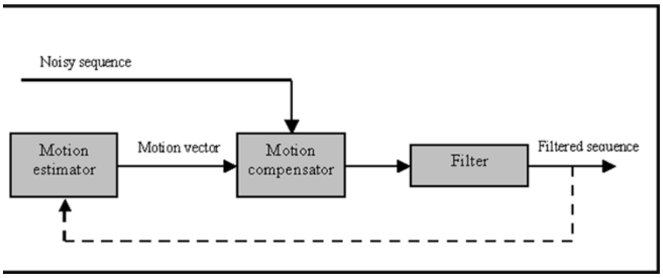


Fig. 1. Motion-compensated filtering scheme consists of three parts: first, the motion is estimated; second, the observed temporal signal is compensated for motion; third, it is filtered.

A Majority of linear filters and nonlinear weighted filters uses least mean square (LMS) algorithm adaptation [7]. Recently, higher order statistics or cumulants have been successfully employed for the detection and classification of non-Gaussian signals in Gaussian noise [22]. Various fourth-order cumulant-based filtering techniques (fixed or adaptive) have been described for the enhancement of signals in either Gaussian or non-Gaussian [14]. The least mean fourth (LMF) and other higher order criteria exhibit improved performance compared to the LMS [28]. The performance of the LMS and the LMF algorithms have been investigated in the literature in the context of adaptive filtering [18]. Also, the combination of the LMS and LMF approaches was applied to image restoration problem [17]. In [21], a nonlinear cumulant-based adaptive filter for image denoising has been proposed. It combines the linear (averaging) characteristics of the linear filter and nonlinear characteristics of the median filter. It uses an impulsive detector to exclude the impulsive points in the input from updating the filter coefficients. Therefore, the deterioration due to the presence of heavily mixed noise corruption could be avoided.

In video sequence applications, cumulant-based methods have been proposed for motion estimation. In [6], the displacement vector is obtained by maximizing a third- and fourth-order statistics criteria. In [15], several algorithms are developed based on a parametric cumulant method, a cumulant matching method and a mean kurtosis error estimation.

In this paper, a spatiotemporal filter is proposed for video sequences corrupted with a variety of noise by considering both spatial intra-frame and temporal inter-frame correlations. Motion compensation along the temporal direction in the sequence is performed by a region recursive-based method. This method uses an affine linear motion model. For each region, motion vectors are computed by minimizing the fourth-order cumulant (Kurtosis)

Manuscript received December 19, 2003; revised February 14, 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mario A. T. (G. E.) Figueiredo.

M. El Hassouni and H. Cherifi are with the LIRSA Laboratory, University of Bourgogne, Dijon, France (e-mail: mohamed.elhassouni@u-bourgogne.fr; hocine.cherifi@u-bourgogne.fr).

D. Aboutajdine is with the GSCM Laboratory, University of Mohammed V, Rabat, Morocco (e-mail: aboutaj@fsr.ac.ma).

Digital Object Identifier 10.1109/TIP.2005.863039

of the displaced frame difference (DFD). The cumulant-based algorithm presented here is found to be noise robust for a wide range of noise signals such as impulsive and mixed distributed. The robustness of the motion estimator to noise effect, as compared to second-order statistics, has been illustrated in [2]. After motion compensation, the spatiotemporal cumulant-based L-filter is applied to motion compensated frames. The filter weights are computed iteratively by minimizing the Kurtosis of the estimation error instead of the mean squared error. In [23], we proposed a family of adaptive spatiotemporal LMS L-filter which is an extension of the spatial filter described in [1]. This filter has proved to be efficient for reducing noise and preserving image details in corrupted image sequences. The main contribution of the paper is in the design and the comparison of several cumulant-based L-filters to second-order statistics-based (LMS)L-filters for noise reduction in video sequences. The weighting coefficients are computed by the least mean kurtosis (LMK) adaptation algorithm. These coefficients are derived for both recursive and nonrecursive implementations. In many cases, improved performance in noise suppression can be obtained if the filter is implemented recursively. However, the weighting coefficients derived for nonrecursive filtering are not optimal for recursive implementation, where the estimate of current pixel is dependent on the past filter outputs [8].

This paper is organized as follows. Section II describes the problem formulation and presents the motion estimation and compensation steps. Section III describes the proposed spatiotemporal filtering scheme. Experimental results are described in Section IV. Section V contains some conclusions.

II. Motion Estimation

A. Problem Formulation

The problem of motion estimation can be stated as follows: Given an image sequence, compute a representation of the motion field that best aligns pixels in one frame of the sequence with those in the next. This is formulated as

$$\begin{aligned} g_{n-1}(s) &= f_{n-1}(s) + \eta_{n-1}(s) \\ g_n(s) &= f_n(s) + \eta_n(s) \\ &= f_{n-1}\left(s - d_n(s)\right) + \eta_n(s) \end{aligned} \tag{1}$$

where $s = (i, j)$ denotes spatial image position of a point; $g_n(s)$ and $g_{n-1}(s)$ are observed image intensities at instant n and $n-1$, respectively; $f_n(s)$ and $f_{n-1}(s)$ are noise-free frames; $\eta_n(s)$ and $\eta_{n-1}(s)$ are assumed to be zero-mean image noise sequences which can be spatially and temporally correlated and $d_n(s)$ is the displacement vector of the object during the time interval $[n, n-1]$.

B. Region-Recursive Approach Based on Motion Modeling

The problem is to estimate $d_n(s)$ from the observation of $g_n(s)$ and $g_{n-1}(s)$. Let us define the $\mathrm{DFD}_n(d)$ as follows:

$$\mathrm{DFD}_n(d) = g_n(s) - g_{n-1}(s - d). \tag{2}$$

To compress an image sequence, the reduction or elimination of image-to-image redundancy is primordial. The presence of motionless objects in the scene creates unvarying image zones in time. Object motion in the scene involves displacement of corresponding projective pattern in the image. Motion estimation methods can be classified into three main families: differential, matching, and stochastic [12].

The region-recursive algorithm is based on a motion model. This algorithm is similar to pel-recursive one [20] extended to parametric motion model applied to region of pixels. This algorithm proceeds in three steps: motion modeling, initialization, estimation and splitting. These steps are iteratively executed until convergence to a given region decomposition is reached.

C. Motion Model

To represent the motion of each region, there are many 2-D motion models (translation, affine, projective linear, quadratic, sampled, and polynomial) [12]. The simplest (translational) model for 2-D motion is used in the existing coding standards. To estimate motions more complex than translations, Labit and Nicolas [19], introduced affine linear motion model of the image with the goal to take into account rotation and divergence. This model has been extensively used in the literature for 2-D motion representation, also in video processing and compression [26].

The region-recursive method described here is a simplified affine linear model (SLM) represented by four parameters $\Theta = (\theta_1, \theta_2, \theta_3, \theta_4)^T$. The vector of displacement is defined as follows:

$$\begin{cases} di = \theta_1 + \theta_3(i - i_g) - \theta_4(j - j_g) \\ dj = \theta_2 + \theta_3(j - j_g) + \theta_4(i - i_g) \end{cases} \tag{3}$$

with (i_g, j_g) are the coordinates of the region center, θ_1 and θ_2 are, respectively, the translation along the axis i and j, θ_3 represents the divergence ratio, and θ_4 is the rotation angle.

D. Initialization

For parameter initialization, there are two solutions (Random and approximate). In general, this initialization depends on the information level that we have *a priori* and which must be exploited rigorously. In our case, we consider that the present information acts on the dense motion field which can be calculated by the pel-recursive algorithm [19].

To calculate d_i et d_j, it is necessary to compute motion vector descriptors from an initial dense motion field. The global motion attributes are computed on image regions issued from a segmentation step. A segmentation procedure is defined by regions of square blocks. For every region R, we define

$$\begin{aligned} \theta_1^0 &= \frac{1}{N}\sum_{s\in R} di \\ \theta_2^0 &= \frac{1}{N}\sum_{s\in R} dj \\ \theta_3^0 &= \frac{1}{N}\frac{\sum_{s\in R}\left[(i-i_g)(di-\theta_1)+(j-j_g)(dj-\theta_2)\right]}{\sum_{s\in R}\left[(i-i_g)^2+(j-j_g)^2\right]} \\ \theta_4^0 &= \frac{1}{N}\frac{\sum_{s\in R}\left[-(j-j_g)(di-\theta_1)+(i-i_g)(dj-\theta_2)\right]}{\sum_{s\in R}\left[(i-i_g)^2+(j-j_g)^2\right].} \end{aligned} \tag{4}$$

Then, the four-dimensional (4-D) parameter vector gives a compact representation of any dense velocity field. The new motion field provides a suitable initialization to optimize a cost function for image reconstruction using motion compensation.

E. Estimation

For every moving region or block, we estimate motion using HOS-based cost function that is maximized or minimized for the desired displacement.

Next, we introduce different criteria to obtain the displacement vector based on second- and fourth-order statistics of the DFD.

SOS-Based Cost Function: The classical solution to obtain the displacement vector from the $\mathrm{DFD}_n(d)$ is the minimum square error

$$J_{2n} = E\left\{\mathrm{DFD}_n^2(d)\right\}. \tag{5}$$

An estimation of this cost function is given by the sample averaging

$$\hat{J}_{2n}(d) = \frac{1}{N}\sum_{s\in R}\mathrm{DFD}_n^2(d) \tag{6}$$

where R denotes the spatial domain that contains the pixels from the region, and N the number of such pixels. Unfortunately, in the presence of colored Gaussian noise or non-Gaussian noise, the above solution may lead to inaccurate results [15].

HOS-based cost function: HOS-based cost functions can be built from different criteria. The approach in [24] is based on a fourth-order statistics cost function that uses the kurtosis of the $\mathrm{DFD}_n(d)$, which is asymptotically unaffected by zero-mean additive noise assumed to have an unknown distribution. With this assumption, we thought to use a HOS-based criterion cost function instead of M-estimators which are robust to outliers, and behave optimally while the statistical distribution follows a known parametric model. The HOS-based cost function is defined as

$$J_{4n}(d) = K\left(\mathrm{DFD}_n(d)\right) \tag{7}$$

where the kurtosis is by definition, as follows:

$$K\left(\mathrm{DFD}_n(d)\right) = E\left\{\mathrm{DFD}_n^4(d)\right\} - 3\left[E\left\{\mathrm{DFD}_n^2(d)\right\}\right]^2. \tag{8}$$

The correct displacement is found by minimizing $J_{4n}(d)$. Tugnait [24] was the first to propose this criterion to estimate the time delay between two signals as an extension to the performance index $J_{2n}(d)$. Later, Anderson and Giannakis [6] used the above cost function to recursively estimate the displacement of pixels between two images. We propose to use the corresponding estimation of $J_{4n}(d)$ given by

$$\hat{J}_{4n}(d) = \frac{1}{N}\sum_{s\in R}\mathrm{DFD}_n^4(d) - 3\left[\frac{1}{N}\sum_{s\in R}\mathrm{DFD}_n^2(d)\right]^2. \tag{9}$$

The presence of colored Gaussian or non-Gaussian noise do not degrade the estimation process and the cost function will depend only on the signal second- and fourth-order moments.

Update equation: The optimization algorithm uses the well known steepest descent gradient method and yields an algorithmic extension of the Anderson–Giannakis [6] cumulant method

$$\hat{\Theta}^{k+1} = \hat{\Theta}^k - \epsilon\Delta_{\hat{d}^k}\hat{J}_{4n}(\hat{d}^k) \tag{10}$$

where ϵ is the 4 × 4 diagonal matrix of gain, k is the region index, and the 4-D gradient vector is defined by

$$\Delta_{\hat{d}^k}\hat{J}_{4n}(\hat{d}^k) = \left[\frac{\partial\hat{J}_{4n}(\hat{d}^k)}{\partial\theta_1}\frac{\partial\hat{J}_{4n}(\hat{d}^k)}{\partial\theta_2}\frac{\partial\hat{J}_{4n}(\hat{d}^k)}{\partial\theta_3}\frac{\partial\hat{J}_{4n}(\hat{d}^k)}{\partial\theta_4}\right] \tag{11}$$

which can be expressed in terms of the DFD and the gradient of J_{4n}, after some analytical development and using the affine linear motion model. The motion descriptor $\Theta = (\theta_1, \theta_2, \theta_3, \theta_4)^T$ is identified for each region.

F. Splitting

The estimation of the motion models needs the choice of a segmentation procedure, either prior, or simultaneous with, the motion estimation step itself, since this operates on a region R of matched pixels. The segmentation rule influences greatly the overall performances of the algorithm. In this study, images are segmented into motion homogeneous regions using an adaptive quad-tree splitting [31]. This segmentation allows the progressive decomposition of the image into smaller and smaller regions making it possible to identify the more global attributes and leading to the identification of local motions at the end of the estimation process. The splitting criterion adopted consists in comparing the entropy of the motion compensated differences as follows:

$$N(R)H(R) - \sum_{i}^{n_d} N(R_i)H(R_i) > S_H \tag{12}$$

where $N(R)$ [resp. $N(R_i)$]: the number of pixels in region R (resp. R_i). $H(R)$: the entropy of the region R in the error image before splitting. $H(R_i)$: the entropy of the sub-region R_i after splitting. n_d: the number of sub-regions ($n_d = 4$) and S_H is the threshold value. This entropy criterion controls the accuracy of the motion estimation procedure in order to optimally balance the amount of information corresponding to the prediction error and to the motion parameters.

An example obtained with a real image sequence is presented in Fig. 2. "campagne" is a complex scene with a global motion of divergence and others local motions (car, a post on the right of the picture, leaf of trees). For the experiments, the decomposition starts from 32 × 32 initial regions to 8 × 8 final regions and the splitting threshold is set to 15.

III. Adaptive Spatiotemporal L-Filter

The currently used spatiotemporal approaches were developed by generalizing well known 2-D filtering techniques. For example, a simple approach to spatiotemporal filtering is to extend the support of a 2-D finite impulse response (FIR) filter in the temporal direction (see Fig. 3). The motion-compensated

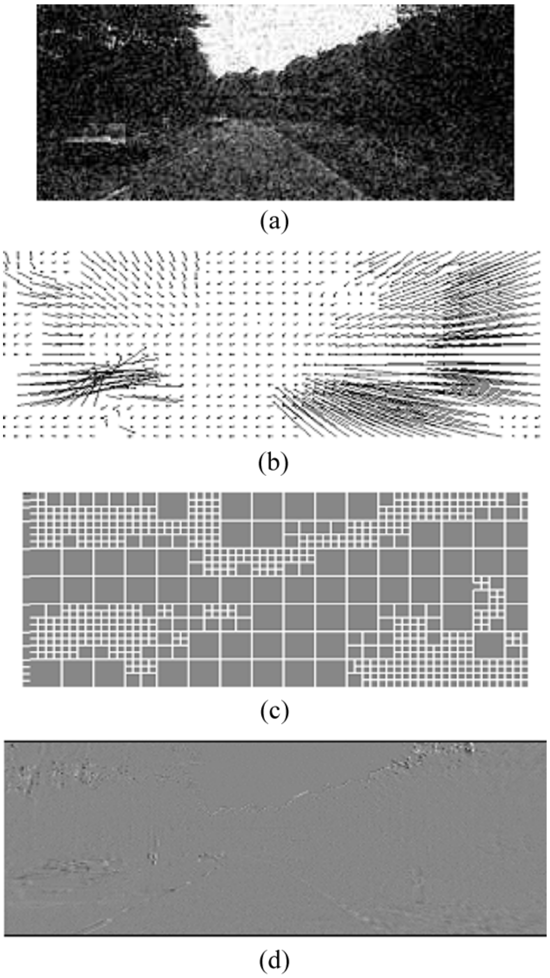


Fig. 2. Motion compensation of the "Campagne" sequence using the HOS-based method. (a) One frame of the noisy images sequence "Campagne" with $\mathrm{SNR} = 10$ dB. (b) Motion vector field. (c) Quadtree segmentation. (d) Motion compensated differences $(\mathrm{MSE} = 26.5)$.

3-D filter is very similar to its no-motion counterpart. Specifically, the generalized spatiotemporal filter is given by

$$\hat{f}_n(i,j) = \sum_{(p,q,l)\in S} \omega(p,q,l) g_{n-l} \times \left(i - p - \hat{d}_x^{n,n-1}(i-p, j-q), j - q - \hat{d}_y^{n,n-1}(i-p, j-q)\right) \quad (13)$$

where $\hat{d}_x^{n,n-1}(i,j)$ and $\hat{d}_y^{n,n-1}(i,j)$ are the horizontal and vertical components of an estimate of the motion between frames n and $n-l$. By motion compensating the sequence prior to averaging, the temporal stationary assumption holds over a much larger region. This allows for the support of the filter to be increased in the temporal direction, improving the filter's ability to suppress noise without incurring additional artifacts due to motion.

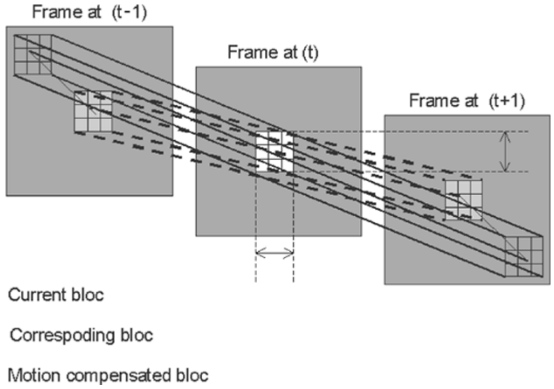


Fig. 3. Spatiotemporal motion-compensated filter support.

A. Three-Dimensional L-Filter Presentation

In video processing, a spatiotemporal neighborhood is the cube around each pixel. Our purpose is to design a filter defined on this neighborhood that aims at estimating the noise-free central image pixel value $f_n(i,j)$ by minimizing a certain criterion. We apply a filter based on sample ordering of the filter cube to a 1-D vector. Let $g^r(k)$ be the ordered input vector at pixel k given by

$$g^r(k) = \mathrm{order}\left(g_{n-l}(k), \ldots, g_{n+l}(k)\right)^T \quad (14)$$

where $g_n(s)$ is a filter window rearranged in a lexicographic order (i.e., row by row) to a 1-D vector

$$g_n(k) = (g_n(i-p, j-q), g_n(i-p, j-q+1), \ldots, g_n(i-p, j+q), \ldots, g_n(i+p, j+q))^T. \quad (15)$$

Let us define s as a scalar running index which can be used instead of the pixel coordinates k. Henceforth, a 1-D notation will be adopted for simplicity. The intensity estimate at image location s is given by:

$$\hat{f}(s) = a^T(s) g^r(s) \quad (16)$$

where $g^r(s)$ is the ordered observations and $a(s)$ is the L-filter coefficient vector that minimizes a criterion between the estimated and the desired images.

Let $\varepsilon(s)$ be the estimation error at pixel s, i.e., $\varepsilon(s) = \hat{f}(s) - f(s)$. To obtain the optimal filter weights from the $\varepsilon(s)$, one can use two criteria:

- a solution based on mean square error (MSE) minimization;
- a solution based on kurtosis minimization of the estimated error.

B. Three-Dimensional LMS L-Filter

In this section, we briefly review the derivation of the nonlinear LMS adaptive algorithms that have been used in the literature for the optimization of several types of nonlinear filters and specially the L-filters. The *optimal* filter weights minimize the MSE cost function

$$J_2(s) = E\left\{\varepsilon^2(s)\right\} = E\left\{\left(\hat{f}(s) - f(s)\right)^2\right\}. \quad (17)$$

The necessary conditions for filter optimality are obtained by setting the gradient of the cost function equal to zero

$$\frac{\partial J_2(s)}{\partial a} = 2E\left\{\varepsilon(s)\frac{\partial \hat{f}(s)}{\partial a}\right\} = 0. \quad (18)$$

The method of steepest descent is a popular technique which addresses this problem by updating the filter weights using the following equation, in an attempt to continually reduce the MSE cost function

$$\hat{a}(s+1) = \hat{a}(s) + \frac{1}{2}\mu \frac{\partial J_2(s)}{\partial a} \tag{19}$$

where $\mu > 0$ is the step-size of the update. Performing the derivation using (17) and (19), we obtain the following equation of the filter weights:

$$\hat{a}(s+1) = \hat{a}(s) + \mu\varepsilon(s)g^r(s) \tag{20}$$

where $\partial\varepsilon(s)/\partial a = \partial\hat{f}(s)/\partial a = g^r(s)$.

C. Three-Dimensional LMK L-Filter

The Kurtosis of the difference between the estimated and the original images is given by

$$J_4(s) = E\left\{\varepsilon^4(s)\right\} - 3E\left\{\varepsilon^2(s)\right\}^2. \tag{21}$$

The algorithm for adjusting the filter coefficient is derived such as

$$\hat{a}(s+1) = \hat{a}(s) + \mu \frac{\partial J_4(s)}{\partial a} \tag{22}$$

the derived value of $J_4(s)$ is denoted by

$$\frac{\partial J_4(s)}{\partial a} = E\left\{4\varepsilon^3(s)\frac{\partial\varepsilon(s)}{\partial a}\right\} - 6E\left\{\varepsilon^2(s)\right\}E\left\{2\varepsilon(s)\frac{\partial\varepsilon(s)}{\partial a}\right\} \tag{23}$$

with $E\{\varepsilon^2(s)\} = \sigma_\varepsilon^2$, (23) becomes

$$\frac{\partial J_4(s)}{\partial a} = E\left\{4\varepsilon^3(s)\frac{\partial\varepsilon(s)}{\partial a}\right\} - 6\sigma_\varepsilon^2 E\left\{2\varepsilon(s)\frac{\partial\varepsilon(s)}{\partial a}\right\} \tag{24}$$

with an instantaneous estimation of $E\{.\}$, the alternative update equation becomes

$$\hat{a}(s+1) = \hat{a}(s) + 4\mu\left[\varepsilon^2(s) - 3\sigma_\varepsilon^2(s)\right]\varepsilon(s)g^r(s). \tag{25}$$

Estimation of $\sigma_\varepsilon^2(s)$ can be done using the recursion [18]

$$\sigma_\varepsilon^2(s) = \beta\sigma_\varepsilon^2(s-1) + \varepsilon^2(s) \qquad 0 < \beta < 1. \tag{26}$$

For a very small value of β, we have $\sigma_\varepsilon^2(s) \approx \varepsilon^2(s)$, and (25) will be approximated by

$$\hat{a}(s+1) \approx \hat{a}(s) + \mu\varepsilon^3(s)g^r(s) \tag{27}$$

which is similar to least mean fourth (LMF) algorithm defined in [28]. The scalar μ is commonly referred to as the step size or the learning rate of the recursive equation, and it controls the speed of the algorithm. The convergence behavior of the LMK algorithm is mainly influenced by cubic error power, $\varepsilon^3(s)$. Basically, the $\varepsilon^3(s)$ will provide fast convergence as the error is larger (i.e., when filter weights are far from optimum) and will provide too smaller residual steady state error when the error is smaller (i.e., the filter weight is close to the vicinity of optimum weight).

D. Convergence Analysis

In theory, there exist sufficient conditions on μ that guarantee the convergence of the LMS algorithm. These conditions depend on the knowledge of the eigenvalues of the correlation matrix R [7], [32], where R is the correlation matrix of the ordered input vector. Under the white Gaussian distributed noise assumption, the mean convergence is studied with expectation of the weight error deviation. The necessary condition for the average LMS L-filter coefficient vector to be convergent is

$$0 < \mu < \frac{2}{\lambda_{\max}} \tag{28}$$

where $\lambda_{\max}$ denotes the maximal eigenvalue of matrix R. Since the eigenvalues are not easily available, a practical definition for $\lambda_{\max}$ is $\lambda_{\max} < tr[R]$. Therefore, the sufficient condition to ensure stability is

$$0 < \mu < \frac{2}{tr[R]} \tag{29}$$

where $tr[.]$ stands for a trace of the matrix inside brackets.

Following the least mean fourth LMF convergence analysis [27], [28], the condition for the average LMK L-filter coefficient to be convergent is

$$0 < \mu < \frac{1}{3\sigma_\eta^2\lambda_{\max}} \tag{30}$$

where $E\{\eta(s)\} = \sigma_\eta^2$ is the noise covariance. A smaller step size will slow down the convergence speed. On the other hand, by properly controlling the step sizes can yield faster convergence speed without instability [29].

E. Three-Dimensional Normalized LMS-LMK L-Filters

When the adaptive filter is going to operate in nonstationnarity environment (as in image processing), it is reasonable to employ a time/space varying step-size parameter $\mu(s)$. After evaluating the *a posteriori* error at location s defined as follows:

$$\acute{\varepsilon}(s) = f(s) - \hat{a}^T(s+1)g^r(s). \tag{31}$$

It can be easily shown that

$$\acute{\varepsilon}(s) = \varepsilon(s)\left(1 - \mu(s)g^{r^T}(s)g^r(s)\right). \tag{32}$$

If $\mu(s)$ is chosen to be

$$\mu(s) = \frac{1}{\lambda + \|g^r(s)\|^2} \tag{33}$$

then $\acute{\varepsilon}(s)$ becomes zero; the coefficients vector of the NLMS L-filter will be modified as follows:

$$\hat{a}(s+1) = \hat{a}(s) + \frac{\mu_0}{\lambda + \|g^r(s)\|^2}\varepsilon(s)g^r(s) \tag{34}$$

μ_0 should be chosen to satisfy the inequality $0 < \mu_0 \leq (2/3)$. The NLMK L-filter becomes [27]

$$\hat{a}(s+1) = \hat{a}(s) + \frac{\mu_0}{\lambda + \|g^r(s)\|^2}\left[\varepsilon^2(s) - 3\sigma_\varepsilon^2(s)\right]\varepsilon(s)g^r(s). \tag{35}$$

These equations describe the adaptation of the NLMS-NLMK L-filters. It is equivalent to linear NLMS-NLMK algorithms. The only difference is that these equations employ the vector of the ordered observations $g^r(s)$ to update the adaptive L-filters coefficients, whereas the linear NLMS-NLMK algorithms employ the input vector $g(s)$.

F. Recursive Implementation

In this section, we address a recursive implementation of the L-filters weights optimized by the second and higher order statistics criteria. In the previous sections, it was noticed that only the nonrecursive implementation has been addressed. Improved performance in terms of noise attenuation can be expected if the filter is applied recursively.

In recursive filtering, the estimate of current pixel is dependent on the new values, instead of the old ones, of the previously processed pixels. The observation sample vector obtained at location s is given by

$$\mathrm{g}^r(s) = \Bigg(\hat{f}^{(1)}(s), \hat{f}^{(2)}(s), \ldots, \hat{f}^{\left(\frac{(N-1)}{2}\right)}(s), \mathrm{g}^{\left(\frac{(N-1)}{2}+1\right)}(s), g^{\left(\frac{(N-1)}{2}+2\right)}(s), \ldots, g^{(N)}(s)\Bigg)^T. \quad (36)$$

In general, the coefficients of recursive L-filters, like those of nonrecursive filters, have to be designed in some optimal fashion. The major objective of the optimization is to derive the best weighting coefficients such that a performance cost criterion is minimized. Based on the LMS and kurtosis optimization algorithms as described in the preceding sections, the weighting coefficients are updated according to (20) and (25). In the case of using $g^r(s)$ as the input observation vector at each location, the estimates of previously processed pixels have to be calculated in order to update the filter weights. This draws an analogy between the recursive L-filters and linear IIR filters, while the nonrecursive L-filters can be viewed as a modification to linear FIR filters.

IV. Experimental Results

We present a set of experiments in order to assess the performance of the adaptive L-filters we have discussed so far. All the sets of experiments have been conducted on real sequences. A number of these sets of experiments suppose that a reference image (e.g., the original image) is available. In practice, reference images are usually transmitted trough TV telecommunications channels to measure the performance of the channel. In such cases, the proposed adaptive L-filters can be proved very useful, if the design of an optimal filter for the specific channel characteristics is required. However, in certain cases (in image sequences), it is reasonable to assume that one noise-free frame can act as a reference image for a number of image frames.

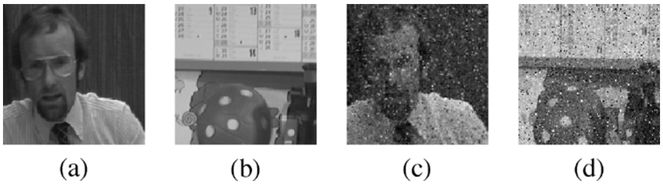


Fig. 4. Some frames of image sequences used in simulations. (a) The third frame of the "Trevor White" original sequence. (b) The third frame of the "Caltrain" image sequence. (c) Frame (a) corrupted by mixed noise. (d) Frame (b) corrupted by mixed noise.

In computer simulations, two 256 × 256 monochrome video sequences, which are parts of the "Trevor White" sequence and the "Caltrain" sequence, were used as original video sequences. Although the "Trevor White" sequence is characterized by temporally slowly varying objects along the temporal direction in a stationary background, it has many spatial nonstationnarities due to vertical strip lines in the background and the shirt worn [one of these frames is shown in Fig. 4(a)]. In contrast, the "Caltrain" sequence consists of a moving objects in a moving background, which give rise to nonstationnarities in large areas along the temporal direction with greater spatial stationary than the 'Trevor White' sequence [see Fig. 4(b)]. The filtered images quality is evaluated by measuring the signal-to-noise ratio (SNR) improvement defined as follows:

$$\mathrm{SNR}i = 10\log_{10}\frac{\sum_{i,j}\left(\hat{f}(i,j) - f(i,j)\right)^2}{\sum_{i,j}\left(g(i,j) - f(i,j)\right)^2}(\text{in dB}) \quad (37)$$

where $f(i,j)$ is the original image pixel, $g(i,j)$ denotes the same pixel corrupted by noise, and (i,j) is the filter output at the same image pixel.

In this section, we compare the performance of the cited filters when the original sequence is corrupted by the following:

1) impulsive noise with probabilities $p = 10\%$ (both positive and negative impulses with equal probability);
2) mixed impulsive $p = 5\%$ and zero-mean additive white Gaussian noise ($\sigma_\eta = 20$).

Then, we shall compare all combinations of the SOS and HOS used methods both in the motion estimation and filtering steps. In a second part, we tend to use a recursive implementation (RLMS, RLMK L-filters) of the chosen adaptive filters, and comparing the performance with the nonrecursive methods (LMS, LMK L-filters).

The notation used in our figure legends and also later in summarizing our results is explained as follows.

- **LMS**, **RLMS** ***L*-filters** refer, respectively, to the SOS-based filter and its recursive implementation.
- **LMK**, **RLMK** ***L*-filters** refer, respectively, to the HOS-based filter and its recursive implementation.
- **SOS–SOS** means that we have used motion estimation method based on SOS and spatiotemporal LMS L-filter.
- **HOS–SOS** means that we have used motion estimation method based on HOS and spatiotemporal LMS L-filter.
- **SOS–HOS** means that we have used motion estimation method based on SOS and spatiotemporal LMK L-filter.
- **HOS–HOS** means that we have used motion estimation method based on HOS and spatiotemporal LMK L-filter.

A. Combining SOS- and HOS-Based Methods

A summary of the SNRI performance for all combinations of HOS and SOS methods applied to the above mentioned se-

TABLE I
SNRI (IN DECIBELS) ACHIEVED BY COMBINING VARIOUS HOS- AND SOS-BASED MOTION ESTIMATORS AND L-FILTERS IN SMOOTHING SOME FRAMES OF TREVOR WHITE SEQUENCE THAT HAS BEEN CORRUPTED BY MIXED GAUSSIAN $(\sigma = 20)$ AND IMPULSE $(p = 5\%)$

Method	3th fr	4th fr	5th fr	6th fr	7th fr	8th fr
SOS-SOS	-5.04	-5.24	-5.18	-5.69	-5.16	-5.06
SOS-HOS	-5.62	-5.39	-5.75	-5.32	-5.36	-5.42
HOS-SOS	-7.37	-7.41	-7.31	-7.57	-7.51	-7.61
HOS-HOS	-9.73	-9.61	-9.38	-9.54	-9.72	-9.63

TABLE II
SNRI (IN DECIBELS) ACHIEVED BY COMBINING VARIOUS HOS- AND SOS-BASED MOTION ESTIMATORS AND L-FILTERS IN SMOOTHING SOME FRAMES OF TREVOR WHITE SEQUENCE THAT HAS BEEN CORRUPTED BY IMPULSIVE NOISE WITH A PERCENTAGE (10%)

Method	3th fr	4th fr	5th fr	6th fr	7th fr	8th fr
SOS-SOS	-5.48	-5.33	-5.40	-5.22	-5.01	-5.12
SOS-HOS	-5.54	-5.32	-6.09	-5.43	-5.21	-5.18
HOS-SOS	-10.82	-10.94	-11.17	-10.58	-11.08	-11.12
HOS-HOS	-11.47	-11.57	-11.86	-11.88	-11.68	-11.41

TABLE III
SNRI (IN DECIBELS) ACHIEVED BY COMBINING VARIOUS HOS- AND SOS-BASED MOTION ESTIMATORS AND L-FILTERS IN SMOOTHING SOME FRAMES OF CALTRAIN SEQUENCE THAT HAS BEEN CORRUPTED BY MIXED GAUSSIAN $(\sigma = 20)$ AND IMPULSE $(p = 5\%)$

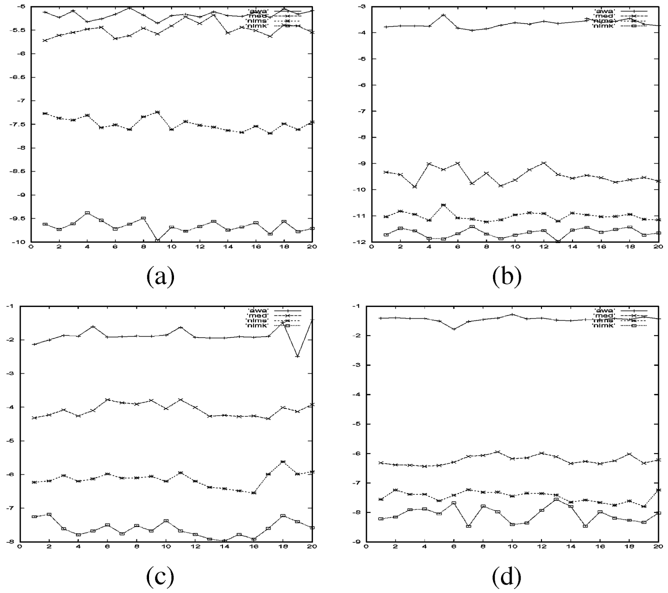
Method	3th fr	4th fr	5th fr	6th fr	7th fr	8th fr
SOS-SOS	-3.64	-3.74	-3.83	-3.56	-3.98	-4.23
SOS-HOS	-4.89	-5.47	-5.53	-5.58	-5.86	-6.04
HOS-SOS	-6.19	-6.03	-6.20	-6.13	-5.98	-6.11
HOS-HOS	-7.18	-7.61	-7.79	-7.68	-7.50	-7.76

TABLE IV
SNRI (IN DECIBELS) ACHIEVED BY COMBINING VARIOUS HOS- AND SOS-BASED MOTION ESTIMATORS AND L-FILTERS IN SMOOTHING SOME FRAMES OF CALTRAIN SEQUENCE THAT HAS BEEN CORRUPTED BY IMPULSIVE NOISE WITH A PERCENTAGE (10%)

Method	3th fr	4th fr	5th fr	6th fr	7th fr	8th fr
SOS-SOS	-4.03	-4.19	-4.08	-4.017	-4.23	-4.31
SOS-HOS	-5.61	-6.14	-6.03	-5.94	-6.31	-6.21
HOS-SOS	-7.24	-7.39	-7.31	-7.61	-7.42	-7.23
HOS-HOS	-8.16	-7.91	-7.88	-8.05	-7.68	-8.48

quences are described in Tables I–IV. The following observations can be made.

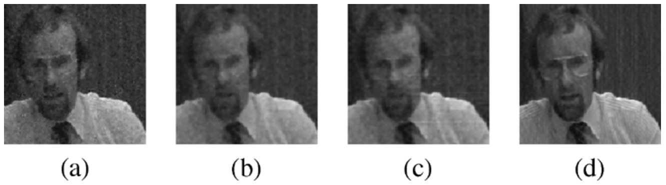
- The joint estimate HOS–HOS-based technique provides an improvement of 4 to 6 dB of SNRI over other techniques, for both Caltrain and Trevor White sequences, and both type of noise.
- For Caltrain sequence which have high amount of motion, the HOS–HOS does better in terms of SNRI than the other techniques. As explained before, this is because more motion activity usually undermines the accuracy of the motion trajectories, there by limiting the LMK L-filter's ability to exploit the temporal redundancy in the sequence.
- For the "Trevor White" sequence, which has textured frames and a reasonable amount of motion, HOS- and SOS-based motion estimators perform about the same. In contrast, the filters perform much better over the results obtained by "Caltrain" sequence.



(a) (b)

(c) (d)

Fig. 5. SNRI improvements versus frame numbers. (a) Trevor sequence with mixed noise $(\sigma = 20, p = 5\%)$. (b) Trevor sequence with salt and pepper noise $(p = 10\%)$. (c) Caltrain sequence with mixed noise $(\sigma = 20, p = 5\%)$. (d) Caltrain sequence with salt and pepper noise $(p = 10\%)$.



(a) (b) (c) (d)

Fig. 6. Output of the filtered Trevor White third frame 256 × 256 corrupted by mixed impulsive $(p = 5\%)$ and Gaussian additive noise $(\sigma = 20)$, (a) output of the $3 \times 3 \times 3$ AWA filter, (b) output of the $3 \times 3 \times 3$ median filter, (c) output of the $3 \times 3 \times 3$ NLMS L-filter with $(\mu_0 = 0.8)$, and (d) output of the $3 \times 3 \times 3$ NLMK L-filter with $(\mu_0 = 10^{-4})$.

After evaluating the four combinations of different motion compensated techniques, we compare the best (HOS–HOS) and (HOS–SOS) techniques with other filters. Fig. 5 shows the SNRI of the HOS–HOS and HOS–SOS technique applied to a group of frames of the chosen sequences and comparing with a classical existing filter like 3-D median filter and 3-D AWA filter. In each case, a temporal windows composed of three-frame are considered. The size of each spatial window is 3 × 3 pixels. For each sequence we use the two types of noise cited above. The HOS–HOS-based filter produces much better results than the other filters for both sequences.

For mixed impulsive and additive Gaussian noise, we observe that adaptive weighted average (AWA) is far from the proposed filters with a difference of 5 dB, both median and LMS L-filters have a same improvement and the LMK L-filter gives a good performance.

For impulsive noise, we observe that the best results are obtained by the LMK L-filter when comparing with the other filters.

Visually, the LMK L-filter denoised frames look better than the other filters denoised frames. We show in Fig. 6 the visual

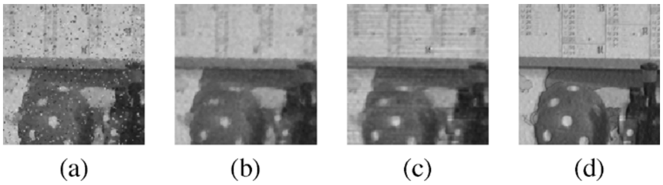


Fig. 7. Output of the filtered Caltrain third frame 256 × 256 corrupted by mixed impulsive ($p = 5\%$) and Gaussian additive noise ($\sigma = 20$), (a) output of the $3 \times 3 \times 3$ AWA filter, (b) output of the $3 \times 3 \times 3$ median filter, (c) output of the $3 \times 3 \times 3$ NLMS L-filter with ($\mu_0 = 0.8$), and (d) output of the $3 \times 3 \times 3$ NLMK L-filter with ($\mu_0 = 10_{-4}$).

TABLE V
SNRI (IN DECIBELS) ACHIEVED BY THE 3-D LMS AND 3-D LMK L-FILTERS WHEN A REFERENCE IMAGE IS NOT AVAILABLE

Filtered Image	SNRI
Trevor White using the LMS L-filter coefficients of Caltrain	-7.02 (-7.37)
Caltrain using the LMS L-filter coefficients of Trevor White	-5.92 (-6.19)
Trevor White using the LMK L-filter coefficients of Caltrain	-8.94 (-9.73)
Caltrain using the LMK L-filter coefficients of Trevor White	-6.34 (-7.18)

comparison between the 3-D filters in suppressing mixed impulsive and additive white Gaussian noise in the third frame of Trevor White sequence. We can notice that the LMK L-filter preserves well the texture in the foreground.

Fig. 7 shows the comparison between the 3-D filters in suppressing mixed impulsive and additive white Gaussian noise in the third frame of Caltrain sequence. From this set of experiments, we have finally adopted the 3-D LMK L-filter which supplies the best results of restoration quality. The results on the Caltrain sequence show that the L-filters are less sensitive to motion estimation errors than the other filters. We can conclude globally that the 3-D LMK filter outperforms the other filters.

When a reference image is not available, we have tested the robustness of the L-filter coefficients that are determined at the end of a training session and are applied to filter a noisy image that has been produced by corrupting a different reference image than the one used in the training session. More specifically, we have run the L-filter algorithm on Trevor White corrupted by mixed noise using as a reference image the original Trevor White. Subsequently, we have averaged the L-filter coefficients that were derived during the operation of the algorithm on the last image row. The resulting L-filter coefficients have been applied to filter the Caltrain image corrupted by the same mixed noise. In Table V, we present the SNRI achieved by filtering the noisy Caltrain by using the L-filter coefficients determined at the end of a training session on Trevor White and vice versa. In parentheses, we have included the same filters performances using the noise-free image as reference. It is seen that the attained noise reduction is close to the one achieved when the reference image is available.

B. Recursive Least Kurtosis and Mean Square Implementation

In our simulations, original images are corrupted by adding zero-mean additive white Gaussian noise with standard deviation σ_η and impulse noise with probability p together.

TABLE VI
SNRI (IN DECIBELS) IN SUPPRESSING MIXED GAUSSIAN ($\sigma = 20$) AND IMPULSE ($p = 5\%$) WITH LMS, RLMS, LMK, RLMK L-FILTERS, MEDIAN, AND RECURSIVE MEDIAN FILTERS

Filters	Sequence frames	
	3^{rd} Caltrain frame	3^{rd} Trevor White frame
Median	-4.08	-5.55
Recursive median	-5.85	-6.11
LMS L-filter	-6.19	-7.37
RLMS L-filter	-7.67	-8.98
LMK L-filter	-7.18	-9.73
RLMK L-filter	-8.79	-10.93

TABLE VII
SNRI (IN DECIBELS) IN SUPPRESSING IMPULSIVE NOISE ($p = 10\%$) WITH LMS, RLMS, LMK, RLMK L-FILTERS, MEDIAN, AND RECURSIVE MEDIAN FILTERS

Filters	Sequence frames	
	3^{rd} Caltrain frame	3^{rd} Trevor White frame
Median	-6.40	-9.89
Recursive median	-8.16	-10.96
LMS L-filter	-7.24	-10.82
RLMS L-filter	-8.96	-11.56
LMK L-filter	-8.16	-11.47
RLMK L-filter	-9.98	-12.34

Tables VI and VII present the comparative results of filtering the two video sequence corrupted by noise with different characteristics. The restoration performance of the recursive LMK L-filter is compared with that of its nonrecursive version and the (least mean squared) LMS L-filter. The step-size parameter μ has been experimentally tuned in order to get the optimal values. For each algorithm, the values μ are chosen as $\mu_{\text{LMS}} = 0.8$, $\mu_{\text{LMK}} = 10^{-4}$. In all experiments, the size of the filter support is chosen to be a $3 \times 3 \times 3$ cube.

The recursive filtering design consistly provides better performance than the nonrecursive implementation. In particular, the SNRI gain obtained by using recursive filtering over its nonrecursive design is 0.87–1.92 dB, which is sequence dependent. It is observed that improvement in terms of SNRI is achieved when the image is heavily corrupted. Similar results are also obtained for the other test sequences. In addition to the achieved gains in SNRI, the recursive implementation yields visually better quality in respect of noise suppression than its nonrecursive counterpart. Results for the two sequences are reported in Figs. 8 and 9. The resultant frames obtained by nonrecursive filtering manifest insufficient reduction of noise, while the recursive filter is more efficient in noise removal without excessive blurring of sequence details.

C. Computational Cost

In this section, we evaluate the efficiency of the proposed algorithms according to the computational cost. Table VIII shows the computational time carried out by using the different filters discussed above applied to the third frame of Trevor White sequence. These computing times are based on a C++ implementation of the algorithms running on a 1-GHz Pentium III-based machine under the Linux operating system. There is no optimization that has been handled. When comparing the computational time of the proposed algorithms with the existing ones, we

TABLE VIII
COMPUTATIONAL TIME PER SECOND FOR THE THIRD TREVOR WHITE FRAME CORRUPTED BY A MIXED IMPULSIVE AND GAUSSIAN NOISE

3-D order statistic filters	LMK L-filter	NLMK L-filter	LMS L-filter	NLMS L-filter	Median
Without motion estimation	0.91	1.24	0.88	0.99	0.75
With SOS-based motion estimator	2.63	2.96	2.60	2.72	2.47
With HOS-based motion estimator	2.99	3.32	2.96	3.07	2.83
3-D linear filters	LMK filter	NLMK filter	LMS filter	NLMS filter	AWA
Without motion estimation	0.26	0.61	0.24	0.39	0.18
With SOS-based motion estimator	1.98	2.33	1.96	2.11	1.90
With HOS-based motion estimator	2.34	2.69	2.32	2.47	2.26

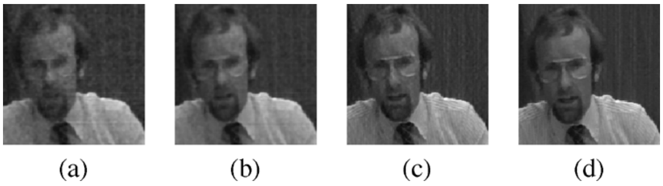


Fig. 8. Output of filtered Trevor White third frame 256 × 256 corrupted by mixed impulsive ($p = 5\%$) and Gaussian additive noise ($\sigma = 20$): (a) 3-D NLMS L-filter with ($\mu_0 = 0.8$), (b) 3-D recursive NLMS L-filter with ($\mu_0 = 0.8$), (c) 3-D NLMK L-filter with ($\mu_0 = 10^{-4}$), and (d) 3-D recursive NLMK L-filter with ($\mu_0 = 10^{-4}$).

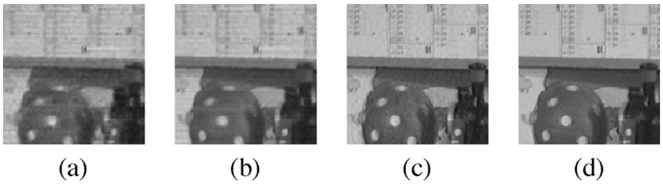


Fig. 9. Output of filtered Caltrain third frame 256 × 256 corrupted by mixed impulsive ($p = 5\%$) and Gaussian additive noise ($\sigma = 20$): (a) 3-D NLMS L-filter with ($\mu_0 = 0.8$), (b) 3-D recursive NLMS L-filter with ($\mu_0 = 0.8$), (c) 3-D NLMK L-filter with ($\mu_0 = 10^{-4}$), and (d) 3-D recursive NLMK L-filter with ($\mu_0 = 10^{-4}$).

can see that there is a trade-off between accuracy and the computational efficiency (the SNRI values have been given above). So, the NLMK L-filter has a larger computational time and the more significant SNRI.

The computational complexity of both algorithms are very similar for HOS-based filters [18]. The LMS requires $\mathcal{O}(2N + 1\text{M}, N + 1\ A/S)$ and LMK requires $\mathcal{O}(2N + 5\text{M}, N + 3\text{A/S})$ where N is the number of adaptive coefficients. Therefore, only four extra multiplications (M) and two extra additions (A/S) are necessary for LMK independent of N. There are more operations required when using the LMS and LMF L-filters [for sorting N pixels, the temporal complexity is $\mathcal{O}(N \log N)$].

V. CONCLUSION

In this paper, a 3-D adaptive motion-compensated LMK L-filter for non-Gaussian noise suppression in video sequences was investigated. Two important findings have been made. First, the motion estimates are accurate enough to improve future time estimates for noisy video sequences. In particular, HOS-based region-recursive method yielded higher SNRI values and good image quality when using a simple parametric motion model. The choice of the motion estimator becomes an important factor. Visibly, the HOS-based region-recursive method produced better image estimates than the SOS-based method. Second, we used spatiotemporal LMK/NLMK L-filters to the motion compensated frames. This is comparable to results reported in literature. Perceptual improvement with our scheme is demonstrated on various sequences. Note that the inclusion of the recursive implementation of the above filters also improved the estimated video sequences, when compared to the non recursive ones.

The cost of employing HOS-based estimators, though, is an issue that cannot be ignored. Using HOS-based motion estimator and filter described earlier increased the processing time. Depending on the application and the resources available, the proposed schemes should be used only if the best results are desired and the computational cost is not a crucial issue. Otherwise, SOS-based algorithms could be employed.

REFERENCES

[1] C. Kotropoulos and I. Pitas, "Adaptive LMS L-filters for noise suppression in images," *IEEE Trans. Image Process.*, vol. 5, no. 12, pp. 1596–1609, Dec. 1996.

[2] E. Ibn-elhaj, D. Aboutajdine, S. Pateux, and L. Morin, "HOS-based method of global motion estimation for noisy image sequences," *Electron. Lett.*, vol. 35, no. 16, pp. 1320–1322, Aug. 1999.

[3] J. C. Brailean, R. P. Kleihorst, S. N. Efstratiadis, and A. K. Katsaggelos, "Noise reduction filters for dynamic image sequences: A review," *Proc. IEEE*, vol. 83, no. 9, pp. 1236–1251, Sep. 1995.

[4] C. Fan and N. M. Namazi, "Simultaneous motion estimation and filtering of image sequences," *IEEE Trans. Image Process.*, vol. 8, no. 12, pp. 1788–1795, Dec. 1999.

[5] R. P. Kleirhost, R. L. Lagendijk, and J. Biemond, "Noise reduction of severly corrupted image sequences," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, Minneapolis, MN, Apr. 1993, pp. 293–296.

[6] J. M. Anderson and G. B. Giannakis, "Image motion estimation algorithms using cumulants," *IEEE Trans. Image Process.*, vol. 4, no. 3, pp. 346–357, Mar. 1995.

[7] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[8] T. Chen and H. R. Wu, "Recursive LMS L-filters for noise removal in images," *IEEE Signal Processing Lett.*, vol. 8, no. 2, pp. 36–38, Feb. 2001.

[9] S. A. C. Kokaram, *Motion Picture Restoration: Digital Algorithms for Artefacts Suppression in Degraded Motion Picture Film and Video*. New York: Springer-Verlag, 1998.

[10] B. Elliger, "Analysis of motion compensated filters concerning motion correctness and accuracy," *Signal Process.: Image Commun.*, vol. 14, pp. 697–720, 1999.

[11] L. Atzori, F. G. B. De Natale, and F. Granelli, "A real-time visual postprocessor for MPEG-coded video sequences," *Signal Process.: Image Commun.*, vol. 16, pp. 809–816, 2001.

[12] C. Stiller and J. Konrad, "Estimating motion in image sequences: a tutorial on modeling and computation of 2-D motion," *IEEE Signal Process. Mag.*, vol. 16, no. 7, pp. 70–91, Jul. 1999.

[13] L. Joyeux, S. Boukir, B. Besserer, and O. Buisson, "Reconstruction of degraded image sequences: application to film restoration," *Image Vis. Comput.*, vol. 19, pp. 503–516, 2001.

[14] C. C. Feng and C. Y. Chi, "Design of wiener filters using a cumulant based MSE criterion," *Signal Process.*, vol. 54, pp. 23–48, 1996.

[15] E. Sayrol, A. Gasull, and J. R. Fonollosa, "Motion estimation using higher order statistics," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 1077–1084, Jun. 1996.

[16] P. V. M. Roosmalen, A. C. Kokaram, and J. Biemond, "Noise reduction of image sequences as preprocessing for mpeg2 encoding," in *Proc. Eur. Conf. Signal Processing*, vol. 4, Sep. 1998, pp. 2253–2256.

[17] M.-C. Hong, T. Stathaki, and A. K. Katsaggelos, "Iterative regularized least-mean mixed-norm image restoration," *Opt. Eng.*, vol. 41, pp. 2515–2524, Oct. 2002.

[18] O. Tanrikulu and A. G. Constantinides, "Least-mean kurtosis: a novel higher-order statistics based adaptive filtering algorithm," *Electron. Lett.*, vol. 30, no. 3, pp. 189–190, Feb. 1994.

[19] H. Nicolas and C. Labit, "Region-based motion estimation using deterministic relaxation schemes for image sequence coding," in *Proc. ICASSP*, vol. 3, San Francisco, CA, Mar. 1992, pp. 265–268.

[20] A. N. Netravali and J. D. Robbins, "Motion-compensated television coding: part I," *Bell Syst. Tech. J.*, vol. 58, pp. 631–670, Mar. 1979.

[21] M. Ibrahim, R. Garieb, and H. Ibrahim, "Nonlinear cumulants based adaptive filter for simultaneous removal of Gaussian and impulsive noises in images," presented at the *ICASSP*, Salt Lake City, UT, May 7–11, 2001.

[22] C. L. Nikias and J. M. Mendel, "Signal processing with higher-order spectra," *IEEE Signal Process. Mag.*, no. 7, pp. 10–37, Jul. 1993.

[23] M. El Hassouni, H. Chérifi, and D. Aboutajdine, "An efficient spatiotemporal filter for noise reduction in video sequences," in *Proc. 3rd Int. Symp. Video Processing and Multimedia Communications*, Zadar, Croatia, Jun., 13–15 2001, pp. 61–64.

[24] J. K. Tugnait, "Time delay estimation in unknown spatially correlated Gaussian noise using higher order statistics," in *Proc. 23rd Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, 1989, pp. 211–215.

[25] S. D. Kim and J. B. Ra, "Efficient block-based video encoder embedding a wiener filter for noisy video sequences," *J. Vis. Commun. Image Represent.*, vol. 14, pp. 22–40, 2003.

[26] F. Charot, C. Labit, and P. Lemonnier, "Architectural study of a block-recursive motion estimation algorithm," *Real-Time Imag.*, vol. 3, pp. 111–128, 1997.

[27] M. K. Chan, A. Zerguine, and C. F. N. Cowan, "An optimized normalized LMF algorithm for sub-Gaussian noise," in *Proc. ICASSP*, Hong Kong, China, Apr. 6–10, 2003, pp. 377–380.

[28] E. Walach and B. Windrow, "The least mean forth (LMF) adaptive algorithm and its family," *IEEE Trans. Inf. Theory*, vol. 30, no. 1, pp. 275–283, Mar. 1984.

[29] S. Koike, "Stability conditions for adaptive algorithms with nonquadratic error criteria," in *Proc. X Eur. Signal Processing Conf.*, vol. 2, Sep. 2000, pp. 131–134.

[30] A. K. Katsaggelos, R. P. Kleihorst, S. N. Efstratiadis, and R. L. Lagendijk, "Temporal motion compensated noise filtering of image sequences," in *Proc. Vis. Communication and Image Processing*, Boston, MA, Nov. 1989, pp. 61–70.

[31] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," in *Proc. ICASSP*, Toronto, ON, Canada, May 1991, pp. 2661–2664.

[32] A. Feure and E. Weinstein, "Convergence analysis of LMS filers with uncorrelated Gaussian data," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 1, pp. 222–230, Feb. 1985.

Mohammed El Hassouni, photograph and biography not available at the time of publication.

Hocine Cherifi, photograph and biography not available at the time of publication.

Driss Aboutajdine, photograph and biography not available at the time of publication.