Adaptive Space-time Examplar-Based Method for Image Sequence Restoration

No Author Given

No Institute Given

Abstract. We present a novel space-time exemplar-based method for image sequence restoration. We have designed a locally adaptive estimation framework which relies on the analysis of the bias-variance trade-off. It allows us to select on-line the best adapted filtering space-time neighborhood at each pixel. Furthermore, it involves an original space-time exemplar-based restoration which improves the overall performance.
This method is unsupervised and requires no motion estimation. It nevertheless can be combined with motion estimation to cope with very large displacements due to camera motion. Experiments show that this method is able to drastically improve the quality of highly corrupted image sequences. Quantitative results on standard image sequences demonstrate that our approach outperforms other recent methods.

1 Introduction

Image sequence restoration takes a crucial place in several important domains. In biology, confocal microscopes provide noisy image sequences due to the limited number of photons. In astronomy, the same phenomenon occurs in telescopes and image sequences are also corrupted with an important noise level. In medicine, image sequence denoising is critical especially in ultra-sound and X-ray imaging. Video surveillance also provides some noisy image sequences. Finally, we can notice that the noise increases the required bandwidth for video sequence broadcasting and that a denoising algorithm can then produce a more compact image sequence digital representation [1].

The main difficulties in image sequence restoration arises from the non-stationarity of the spatio-temporal signal. Denoising methods must preserve these space-time discontinuities while minimizing the error between the unknown noise-free image sequence and the denoised sequence. The later is usually stated as the minimization of a local quadratic objective function.

In order to recover the correct intensity at every pixel of the image sequence, we can exploit the local correlation of the signal and use the neighboring pixels. A method based on a pre-determined shape and size (scale) will not be able to take into account the spatio-temporal discontinuities of the intensity function. This would result in a blurred image sequence. As for video sequences, temporal discontinuities are mainly due to motion, image sequence denoising methods have to be able to filter along motion trajectories. Thus, the local neighborhood can be oriented according to the motion direction; or conversely, the motion can be compensated for. This implies a motion estimation stage. However, motion estimation is usually not that easy specially in noisy sequences.

Another way to remove noise is to find similar samples in the image sequence. Exemplar-based approach has proved its efficiency in texture synthesis [2], image in-painting [3] and also video completion [4]. In [5], an exemplar-based image sequence filtering method is also proposed: the *non-local mean* algorithm. In these methods the similarity between two points is measured by using patches that describe their local context. Examples are usually searched in the whole image or in an area around the considered pixel.

The original method we propose is an exemplar-based adaptive statistical method for image sequence restoration. It is related to the framework already described for still images [6–8] and image sequences [9]. Methods designed for still image like wavelet shrinkage [13], Wiener filtering [14] or PDE based filters [12] have been extended to image sequences. Unlike robust anisotropic diffusion [10] and non-linear Gaussian filtering [11], our local adaptive estimation approach supplies scale selection for each pixel by locally estimating the appropriate space-time filtering window. Moreover, if required, the proposed method is able to process in a decoupled way the space and the time dimensions which is not the case for methods that simply consider the image sequence as a $3D$ volume [12, 13]. Unlike [7, 9] our approach is not based on a geometrical partition of the neighborhood in sectors but use an appropriate and more flexible weighted sum of data points in a neighborhood. The weights are defined by computing the distance between a patch centered on a given pixel and examples patches in the estimated space-time neighborhood. Additionally, a confidence level (i.e. variance) attached to each restored pixel is provided. Lets us finally point out that the Total Variation minimization method [15] cannot be easily extended to space-time domain.

The remainder of the paper is organized as follows. Section 2 describes the general framework of our method and the intensity estimation is defined. In Section 3, the adaptive choice of the local space-time neighborhood is introduced. Section 4 deals with the similarity measure involved in the selection of the patches in the space-time neighborhood of the pixel to be restored and Section 5 gives details of the algorithm implementation. In Section 6, we report an important set of experimental results. Intensive comparisons with other recent methods has been carried out, demonstrating that our method outperforms them. We also present how our denoising method can be combined with a motion estimation method and we comment the usefulness of a motion compensation stage. Finally, Section 7 contains concluding remarks.

2 Proposed exemplar-based space-time approach

We consider the following statistical image model:

$$Y_i = u(\mathbf{x}_i) + \xi_i, \tag{1}$$

where $\mathbf{x}_i \in \Omega$ denotes the pixel location in the space-time volume $\Omega \subset \mathrm{I\!R}^3$. The image function $u_i = u(\mathbf{x}_i)$ has to be recovered from observations Y_i. The errors ξ_i are assumed to be independent zero-mean Gaussian variables with unknown variance τ^2.

We need minimal assumptions on the structure of the image for recovering u. We assume that $u(\mathbf{x}_i)$ is a locally piecewise constant function in a space-time neighborhood of pixel $\mathbf{x}_i$. However, the size and shape of the neighborhood should varie in the image sequence because of non-stationarities and presence of discontinuities. If we can determine the adequate neighborhood for each pixel, then the regression function u can be estimated by optimizing a local maximum likelihood (ML) criterion. The proposed method addresses these two issues as described below.

We construct a sequence of increasing nested space-time neighborhoods $(W_{i,n})_{n\in[0:N]}$ centered at each point $\mathbf{x}_i$, *i.e.* $W_{i,n} \subset W_{i,n+1}$ with N indicating the largest window. Further details about the neighborhood design are described in Section 3. At the initialization, we choose a small neighborhood (the 8 nearest neighbors in the $2D$ space domain) as the *pilot* (starting) window $W_{i,0}$ at point $\mathbf{x}_i, \forall \mathbf{x}_i \in \Omega$. Then, we compute an initial estimate $\hat{u}_{i,0}$ of $u(\mathbf{x}_i)$ and its associated variance $\hat{\sigma}^2_{i,0}$ as:

$$\hat{u}_{i,0} = \sum_{\mathbf{x}_j \in W_{i,0}} \omega_{ij} Y_j \qquad \text{and} \qquad \hat{\sigma}^2_{i,0} = \hat{\tau}^2 \sum_{\mathbf{x}_j \in W_{i,0}} \omega_{ij} \tag{2}$$

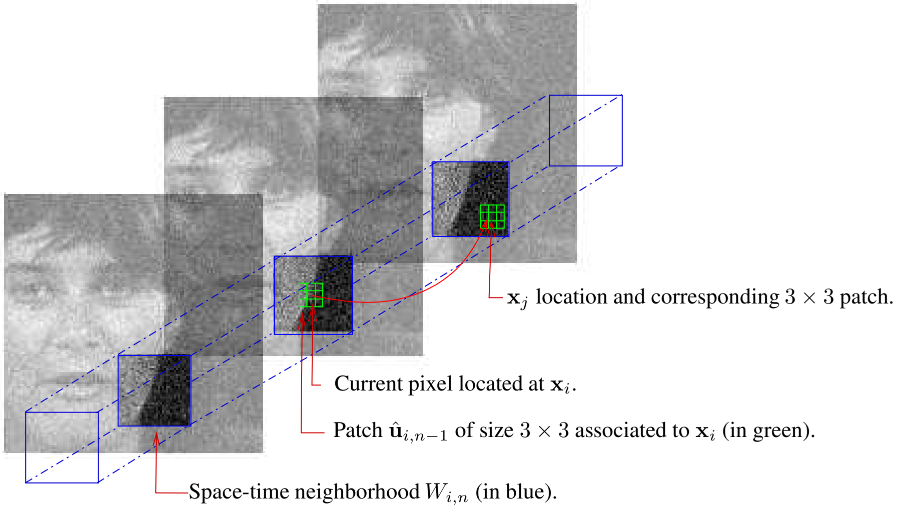
$\mathbf{x}_j$ location and corresponding 3×3 patch.
Current pixel located at $\mathbf{x}_i$.
Patch $\hat{\mathbf{u}}_{i,n-1}$ of size 3×3 associated to $\mathbf{x}_i$ (in green).
Space-time neighborhood $W_{i,n}$ (in blue).

Fig. 1. Exemplar-based space-time approach. To each point of the image sequence is associated an estimated space-time neighborhood $W_{i,n}$. To each point $\mathbf{x}_j$ of this neighborhood is associated a 3×3 patch. The weights ω_{ij} are defined as a function of the distance between the patch centered in point $\mathbf{x}_i$ and patches centered in point $\mathbf{x}_j \in W_{i,n}$.

where $\hat{\tau}^2$ is an empirical estimate of the noise variance τ^2 as described in Section 5. At the initialization step, the weights ω_{ij} are defined as a function of the distance between two $2D$ image patches of p^2 pixels centered in $\mathbf{x}_i$ and $\mathbf{x}_j$.

At the next iteration, we consider a larger space-time neighborhood $W_{i,1}$ such that $W_{i,0} \subset W_{i,1}$ and calculate new estimates $\hat{u}_{i,1}$ and $\hat{\sigma}^2_{i,1}$ over $W_{i,1}$. We continue this way, and at iteration n, we define the estimator as:

$$\hat{u}_{i,n} = \sum_{\mathbf{x}_j \in W_{i,n}} \omega_{ij} Y_j \quad \text{and} \quad \hat{\sigma}^2_{i,n} = \hat{\tau}^2 \sum_{\mathbf{x}_j \in W_{i,n}} \omega_{ij}^2, \tag{3}$$

The estimate $\hat{u}_{i,n}$ corresponds to a weighted average of the data points located in the space-time neighborhood. We propose to define weights ω_{ij} as a function of the distance between two $2D$ regularized image patches $\hat{\mathbf{u}}_{i,n-1}$ and $\hat{\mathbf{u}}_{j,n-1}$ defined at iteration $n-1$ and centered in $\mathbf{x}_i$ and $\mathbf{x}_j$ as shown in Figure 1. In the two next sections, we address the problems of the definition of the neighborhood $W_{i,n}$ and propose the design of a distance to compare image patches.

3 Space-time neighborhood adaptation

3.1 Space-time neighborhood geometry

One important contribution of this work is the on-line adaptation of the space-time neighborhood sequence $(W_{i,n})_{n \in [0:N]}$. We consider a simple hyper-cubic

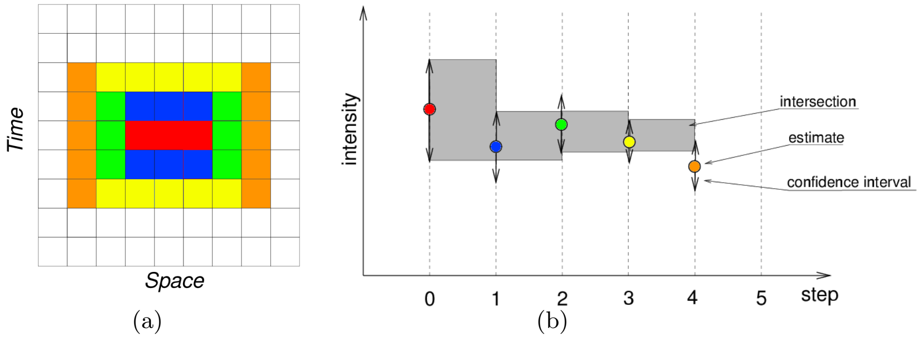

Fig. 2. (a) Spatio-temporal neighborhood: colors correspond to iterations plotted in (b); (b) confidence intervals: circles represent estimates $\hat{u}_{i,n}$ obtained at each iteration n. The gray rectangles represent the intersection between the current confidence interval and the previous one. As long as the estimate belongs to this intersection, the estimation process is updated.

space-time volume as neighborhood shape. Nevertheless, we separate the space dimensions from the time dimension. Then, the space-time neighborhood is characterized by its extent in the space domain and its extent in the time domain. The use of two distinct extents in space and in time allows us to respect space-time discontinuities and the space-time domain is not considered as an isotropic $3D$ volume. In Figure 2(a) the increasing neighborhood sequence is illustrated. At each iteration, the spatial extent and the temporal extent are alternatively increased until a stopping rule is satisfied.

3.2 Space-time neighborhood selection

A point-wise rules is used to guide the space-time neighborhood estimation process. This rule aims at estimating the optimal neighborhood at $\mathbf{x}_i$ and is based on the measure of the closeness of the estimator $\hat{u_i}$ to the unknown function u_i given by the local L_2 risk. This measure of performance can be decomposed in two terms, that is the squared bias and the variance as

$$\mathbb{E}\left[\hat{u}_{i,n} - u_i\right]^2 \;=\; \left[\mathrm{bias}\left(\hat{u}_{i,n}\right)\right]^2 + \hat{\sigma}^2_{i,n}, \qquad (4)$$

where $\mathbb{E}(.)$ denotes the mathematical expectation. In the sequel, we can reasonably assume that the squared bias is an increasing function of the neighborhood size and the variance is a decreasing function of the neighborhood size [6–8, 16]. Then, the optimal neighborhood will be the one that achieves an optimal compromise between these two terms. A closed-form optimal solution for the ideal window is not available for such a non-linear estimator but we can assume that the optimal neighborhood is the one for which the squared bias and the variance are nearly the same: $\mathbb{E}\left[u^*_{i,n} - u_i\right]^2 \;\approx\; 2\sigma^{2*}_{i,n}$ [16].

A practical rule based on pairwise comparison of successive estimates can be derived to select the optimal neighborhood. It amounts to define the largest neighborhood satisfying the following point-wise statistical rule [6, 8, 16]

$$|\hat{u}_{i,n} - \hat{u}_{i,n'}| \ < \ \eta \ \hat{\sigma}_{i,n'}, \ \forall n' < n, \tag{5}$$

as the optimal neighborhood. This rule can be interpreted as follows. While the succesive estimates $\hat{u}_{i,n}$ are sufficiently close to each other, we continue the estimation process. As shown in Figure 2, the estimation process is continued, while new estimates belong to the intersection of estimated confidence intervals $[\hat{u}_{i,n} - \eta\hat{\sigma}_{i,n}, \hat{u}_{i,n} + \eta\hat{\sigma}_{i,n}]$. Besides, it shows that we do not need to store all the previous estimates $(\hat{u}_{i,n'})_{n' \leq n}$ but only the intersection of confidence intervals, the last previous estimate and its variance for each pixel. Finally, the factor η can be easily chosen in the range $[2, 4]$ as justified in [6–8].

4 Local similarity measure for patch selection

As an alternative to a geometry-based approach [7, 9], we prefer to use weights that allow us to select the data points in the neighborhood for filtering. This selection is based on the similarity between a given small image patch at point $\mathbf{x}_i$ and patches at points $\mathbf{x}_j$ belonging to the space-time neighborhood $W_{i,n}$. Considering small patches enables to capture the local intensity structure. The sum of square image differences is widely used for similarity measure between image patches. However, in order to take into account the local variance of the estimator we define the following symmetric distance between image patches $\hat{\mathbf{u}}_{i,n-1}$ and $\hat{\mathbf{u}}_{j,n-1}$:

$$\begin{aligned}\delta_{ij}^2 = \tfrac{1}{2} \ \big[&(\hat{\mathbf{u}}_{i,n-1} - \hat{\mathbf{u}}_{j,n-1})V_{i,n-1}^{-1}(\hat{\mathbf{u}}_{i,n-1} - \hat{\mathbf{u}}_{j,n-1}) + \\ &(\hat{\mathbf{u}}_{i,n-1} - \hat{\mathbf{u}}_{j,n-1})V_{j,n-1}^{-1}(\hat{\mathbf{u}}_{i,n-1} - \hat{\mathbf{u}}_{j,n-1})\big]\end{aligned} \tag{6}$$

where $V_{i,n-1}$ denotes the estimator covariance matrix. The two vectors $\hat{\mathbf{u}}_{i,n-1}$ and $\hat{\mathbf{u}}_{j,n-1}$ denote $p \times p$ patches respectively centered on point $\mathbf{x}_i$ and point $\mathbf{x}_j$. The two matrices $V_{i,n-1}$ and $V_{j,n-1}$ are diagonal with the diagonal elements equal to $\hat{\sigma}_{i,n}^2$ and $\hat{\sigma}_{j,n}^2$ respectively. We decide that the two vectors $\hat{\mathbf{u}}_{i,n-1}$ and $\hat{\mathbf{u}}_{j,n-1}$ are similar with a probability of false alarm $1 - \alpha$, under the hypothesis that they are Gaussian distributed, using a classical χ^2 test with p^2 degrees of freedom. In other words, when $\delta_{ij}^2/\lambda < 1$, with λ choosen as a quantile of a $\chi^2_{p^2,1-\alpha}$ distribution we can decide that the two patches are similar. In our experiments, we use a confidence level of 99% for setting α to 0.01 .

The distance δ_{ij}^2 is transformed into a similarity measure using the exponential kernel. We compute the similarity measure for all the points of the neighborhood and normalize it to obtain the following expression for the weights:

$$\omega_{ij} = \frac{e^{-\delta_{ij}^2/2\lambda}}{\displaystyle\sum_{\mathbf{x}_j \in W_{i,n}} e^{-\delta_{ij}^2/2\lambda}}. \tag{7}$$

Then, if the distance δ_{ij} between two patches is important then the weight ω_{ij} associated to pixel $\mathbf{x}_j$ is small and the pixel will not participate in the intensity estimation at point $\mathbf{x}_i$.

Consequently, the weights provide an efficient and flexible way to implicitly choose the appropriate pixels contributing to the intensity estimation in the adaptive space-time neighborhood while effectively preserving space-time discontinuities. Note that the process is entirely data-driven and does not require particular geometry adapted to image content.

5 Algorithm implementation

At the initialization of the algorithm, we have to estimate the noise variance τ^2. It can be robustly estimated by calculating pseudo-residuals ε_i as described in [17]. If we choose a space-time 6-neighborhood (4 spatial neighbors and 2 temporal ones), pseudo-residuals are compactly represented by $\varepsilon_i = (8Y_i - \mathit{\Delta} Y_i)/\sqrt{42}$ where $\mathit{\Delta} Y_i$ is the discrete Laplacian at $\mathbf{x}_i$ and the constant $\sqrt{42}$ is introduced to ensure that $\mathbb{E}[\varepsilon_i^2] = \tau^2$. Given the residuals ε_i, we can then robustly estimate the noise variance τ^2 by: $\tau = 1.4826 \ \mathrm{med}_i \left(\mid \varepsilon_i - \mathrm{med}_j \left| \varepsilon_j \right| \mid \right)$. In our experiments, λ is set to the 0.99 quantile of the $\chi^2_{k,0.99}$ distribution with $k \in \{(2p+1)^2, p \in \{1,2,3,4\}\}$. The last parameter η is set to $2\sqrt{2}$ to ensure a good accuracy of the estimation. During the estimation, we alternate the increasing of the spatial and temporal extents of the space-time neighborhoods. Furthermore, the algorithm can be easily parallelized. Actually, estimation steps use only local information and have been distributed over 8 CPUs dividing the computation time by 8. Finally, another possibility to speed up the algorithm is to use a dyadic scheme when increasing the extent of the neighborhood. To conclude, the proposed method is very simple to implement and do not require the fine adjustment of parameters λ and η which control the estimatation process.

6 Experimental results

In this section, we report a great number of experiments to show how our exemplar-based space-time adaptive filtering method performs on real image sequences with artificially added noise. The described method is compared to others commonly-used methods for denoising image sequences. The performance of the algorithm combined with a global affine motion compensation stage is also analyzed. For an objective performance evaluation, we use the Peak-Signal-to-Noise-Ratio global measure defined as $PSNR = 20 \log_{10}(255/mse)$ where mse denotes the mean squared error between the original noise-free image sequence and the denoised image sequence.

6.1 Performance assessment and comparison with other recent methods

We first report experiments to evaluate the influence of the noise level and the patch size on the overall method performance. Figure 3 plots $PSNR$ values ob-

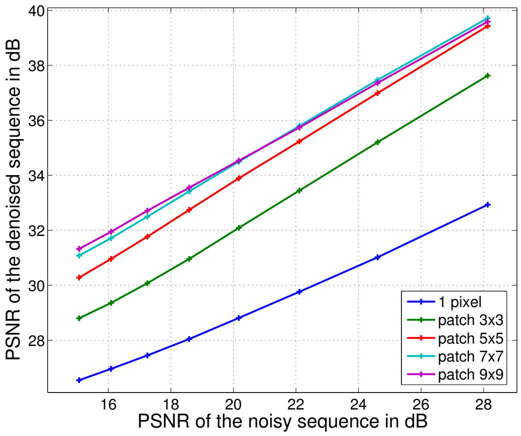

Fig. 3. Denoising performance for several noise levels and for several patch sizes. The $PSNR$ is used to measure the overall performance of the filtering and the test sequence is Akyio ($176 \times 144 \times 300$). This sequence mainly exhibits low motion. Experiments show that the introduction of patches improve the $PSNR$ of at least $2dB$. We can also remark that the results for patch 7×7 and 9×9 are similar.

tained for 8 noise levels and 5 patch sizes. First, we can note that the exemplar-based method is stable while the noise level increases. The improvement gained by introducing patches (to be compared to pixel-wise version) is clearly demonstrated. As it could be expected, it is useless to consider too larges patches. When the size of the patch increases, the $PSNR$ increases too, however, the result for size 7×7 and 9×9 are quite similar while the computation time is proportional to the number of points in the patch.

In Figure 4, we can observe the improvement brought by the progressive increasing neighborhood scheme in term of $PSNR$ as well as in term of visual quality. Figure 4(c) contains denoised image sequence with the same weighted intensity estimation as in our method but with a fixed $11 \times 11 \times 11$ neighborhood size which corresponds to the biggest one used to filter the image sequence with our method whose result is shown in Figure 4(d). The image sequence denoised using our method is smoother and its edges are better preserved compared to the sequence denoised with a fixed neighborhood which appears as blurred. This is due to the fact that the weights are calculated from a regularized version of the image sequence and not the original noisy sequence.

We have compared our method with 4 other recent methods: a combination of a spatial Wiener filter with a motion-compensated temporal Kalman filter [1], a space-time non linear adaptive K-NN filter [18], a $3D$ wavelet-based method [13] and a $3D$ point-wise adaptive estimate using a geometrical design for the neighborhood [9].

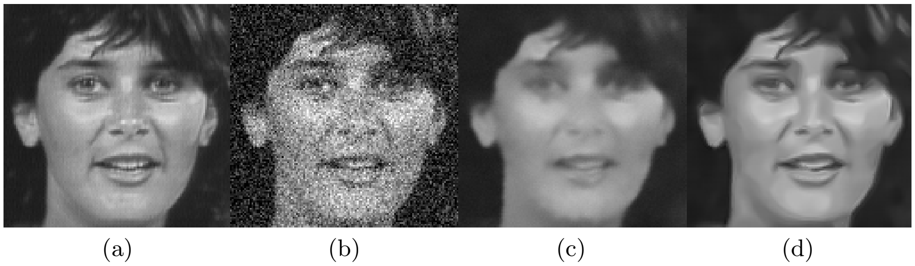

Fig. 4. (a) Original sequence (b) noisy sequence with a additive Gaussian white noise of standard deviation $\tau = 30$ ($PSNR = 18.58$) (c) denoised sequence using a fixed $11 \times 11 \times 11$ neighborhood and a 5×5 patch. The $PSNR$ is $28.72dB$ and we can see that the sequence is quite blurred (d) denoised sequence with the proposed method. The $PSNR$ is $32.14dB$.

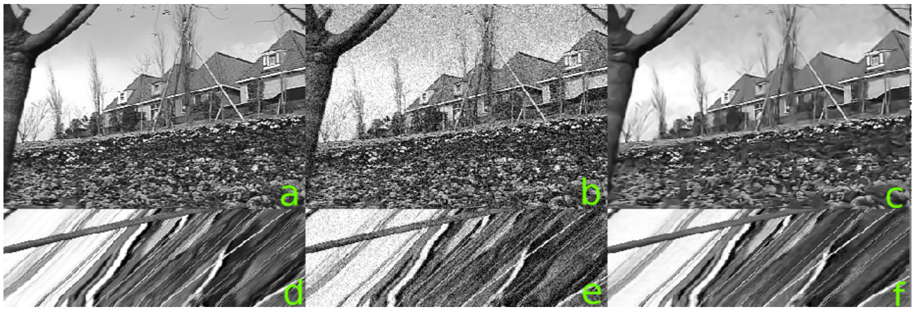

Fig. 5. Sequence "Garden". (a) original sequence, (b) noisy sequence with a additive Gaussian white noise of standard deviation $\tau = 30$ ($PSNR = 18.58dB$) and (c) denoised sequence, $PSNR = 23.59dB$. (d), (e) and (f) represent corresponding XT slices in space-time domain. The camera motion appear as lines in the XT slices.

8 test sequences corrupted with noise of different levels are used. For a fair evaluation, we have considered the results supplied by the authors themselves in the referenced papers. Therefore, we cannot provide the $PSNR$ measure for all the test sequences. Table 1 contains all the available results. Our method clearly outperform all the other methods since it supplies the best $PSNR$ results for all the test sequences sometimes with a quite significative improvement (up to $4dB$). We have also to stress that the implementation of our method is easy and it involves no parameters tuning.

Figure 5 reports another experiments on the "Flower Garden" sequence. It involves an important apparent (varying) motion due to the camera motion. In order to give insights into the spatio-temporal behavior of the denoising method, we have displayed XT slices of the image sequence. Reported results demonstrate that our method can cope with the presence of motion while preserving as well as temporal discontinuities.

Sequence name	$PSNR$	(a)	(b)	(c)	(d)	(e)
Akiyo	22	–	–	–	33.86	**34.31**
Salesman	28	34.4	32.5	–	–	**35.13**
	24	31.1		–	–	**32.60**
Garden	28	–	28.2	–	–	**31.33**
Miss America 1	28	–	35.3	–	–	**39.39**
Miss America 2 (128^3)	7	–	–	26.36	–	**26.69**
	12	–	–	28.16	–	**29.63**
	17	–	–	30.46	–	**32.05**
	22	–	–	32.66	–	**34.20**
Suzie	28	34.8	–	–	–	**37.07**
	24	32.0	–	–	–	**35.11**
Trevor	28	33.9	34.1	–	–	**36.68**
	24	31.3	–	–	–	**34.79**
Foreman	28	33.9	–	–	–	**34.94**
	24	31.1	–	–	–	**32.90**

Table 1. PSNR results for 8 test sequences and 5 denoising methods. (a) Join Kalman and Wiener denoising with motion compensation using dense motion field [1], (b) Adaptive K-NN space-time filter [18],(c)Wavelet based method for image sequence denoising: TIWP3D [13], (d) 3D non-parametric regression approach [9], (e) the proposed adaptive method with 7×7 patches and 6 iterations. Numerical results for the other method are taken from the related publications.

6.2 Filtering with motion compensation

In this section, we analyze the case of image sequences with a global camera motion and we investigate the addition of a motion compensation stage. Global parametric motion can be accurately computed using a robust estimation method even if independent moving objects are present in the scene as described in [19].

In Figure 6, we give the overall block-diagram of the resulting method forming by adding a motion compensated stage to our denoising method. Motion estimation and denoising are not performed alternatively. Actually, we consider that the global affine motion is correctly estimated with the method described in [19] (We used the software made available on the web). A similar exploitation of a parametric motion compensation was proposed in [14] and associated with a $3D$ Wiener filtering technique. After the motion estimation step, the whole sequence is back-warped in the frame (referential) of the first image. Then, we apply our denoising method to this motion-compensated sequence. The result of the filtering has to be warped again to recover the images into their original frames. This is accomplished by using a two-step procedure. First, the denoised images are warped by using the initially estimated motion. Then, resulting images undergo a second transformation inferred from motion estimated between each resulting denoised image and its corresponding image of the original sequence. The

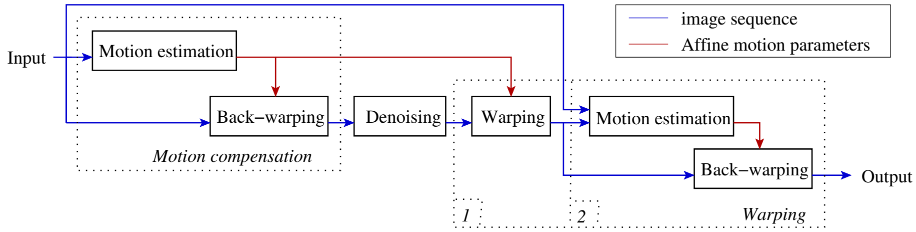

Fig. 6. Block diagram of the denoising method with motion compensation. A motion compensation is applied to the image sequence before the denoising stage. It corresponds to the dotted box on the left. This module is composed of the estimation of affine motion models and the projection of the image sequence in the coordinates system of the first image. After the denoising stage, a two-step warping procedure is applied to get the images in their initial frames. The second step estimates the motion between each image of the input sequence and its corresponding image in the denoised warped image sequence.

later step is made necessary by the error accumulation in combining the motion models estimated from frame to frame in the warping procedure.

Figure 7(c) shows one image the sequence denoised without motion compensation and Figure 7(d) shows the denoising result with the proposed motion compensation stage. The processed image sequence contains two cars tracked by the camera from an helicopter. The warping stage involved in the motion-compensated version of the algorithm require bilinear interpolations that somewhat blur the image sequence. In this expermient, the $PSNR$ value of the proposed motion-compensated method is thus slightly lower than the $PSNR$ value of the output of the denoising method without motion compensation.

We have shown that interpolations involved in the warping stage can induce blur effects that reduce the $PSNR$ value of the filtered image sequence. Then, if interpolations could be avoided, the results of the motion-compensated denoising method would be improved. This could be achieved by transforming the neighborhood instead of warping the data. Such an investigation is in progress. In Figure 8, we have simulated an image sequence by sliding a window over an aerial photography with translation of 10 pixels between successive frame. Since all the motions are translations of entire pixels, no interpolation is required in the warping steps. This experiment shows that the motion-compensated denoising method performs better in the case of large displacements. Consequently, the denoising method without motion-compensation is more appropriate for denoising image sequences with not too large motion magnitude.

7 Conclusion

We have described a novel and efficient unsupervised method for denoising image sequences. The proposed method is based upon an adaptive estimation statistical

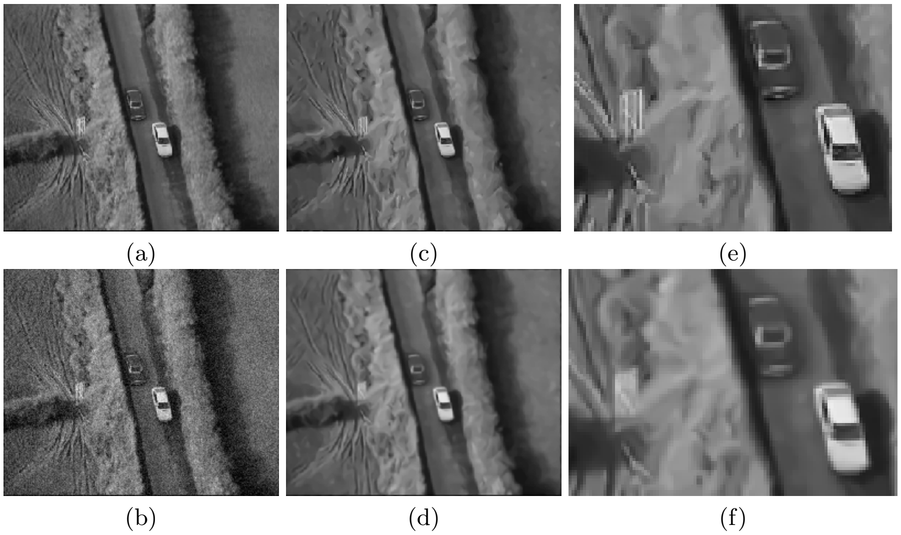

Fig. 7. Sequence "Avenger". (a) original sequence (b) noisy sequence with a additive Gaussian white noise of variance 100 (c) denoised sequence with the proposed method $PSNR = 31.36$ (d) denoised with the proposed motion-compensated method $PSNR = 31.04$. (e) and (f) are details of respectively (c) and (d). Some spatial blurring effects due to interpolation are visible in (f)

framework. It can specify, in a simple data-driven way, the most appropriate space-time neighborhood and associate weights to select the data points involved in the intensity estimation at each pixel. Moreover, it involves an exemplar-based approach extended to the space-time domain. All the parameters of the algorithm are well calibrated and our method does not require any fine tuning. Quite satisfactory results have been obtained on several image sequences. Furthermore, it was experimentally demonstrated that our method outperforms other recent methods. The visual quality of the denoised image sequences is noticeable since noise is well smoothed out while spatial and temporal discontinuities are well preserved. Finally, some improvements are proposed to incorporate a motion-compensated stage It was shown that motion-compensation was really needed for large displacement only provided interpolations involved in the warping steps could be avoided which is under progress.

References

1. Dugad, R., Ahuja, N.: Noise reduction in video by joint spatial and temporal processing. In: submitted to IEEE Trans. on Circuits and Systems for Video Technology. (2001)

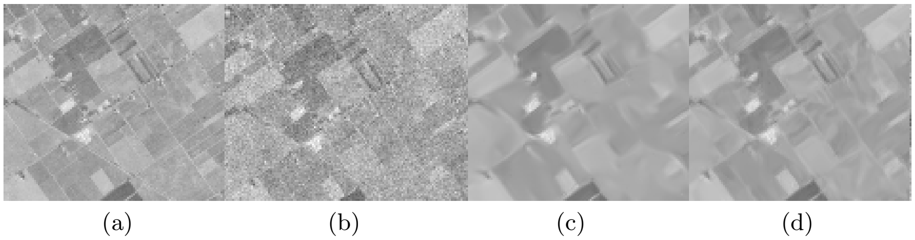

Fig. 8. Image sequence simulated by sliding a window in an aerial photography. (a) one image of original sequence, (b) the same image of the noisy sequence with a additive Gaussian white noise of variance 100, the $PSNR$ is $28.11dB$, (c) the same image of the sequence denoised with the proposed method without motion compensation, the $PSNR$ is $31.59dB$, (d) denoised sequence using the described motion-compensated denoising method with a $PSNR = 32.97dB$.

2. Efros, A., Leung, T.: Texture synthesis by non-parametric sampling. In: Proc. of 7th IEEE Int. Conf. on Computer Vision, ICCV'99, Kerkyra, Greece (1999) 1033–1038
3. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based inpainting. IEEE Trans. Image Processing **13** (2004) 1200–1212
4. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04). Volume 1., Washington (2004)
5. Buades, A., Coll, B., Morel, J.: Denoising image sequences does not require motion estimation. Pre-Print CMLA (2005)
6. Polzehl, J., Spokoiny, V.: Adaptive weights smoothing with applications to image restoration. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **62** (2000) 335–354
7. Katkovnik, V., Egiazarian, K., , Astola, J.: Adaptive window size image denoising based on intersection of confidence intervals (ICI) rule. Journal of Mathematical Imaging and Vision **16** (2002) 223–235
8. Kervrann, C.: An adaptive window approach for image smoothing and structures preserving. In: Proc. of 8th Eur. Conf. on Computer Vision, ECCV'2004, Prague (2004) 132–144
9. Ercole, C., Foi, A., Katkovnik, V., Egiazarian, K.: Spatio-temporal pointwise adaptive denoising in video: 3d non parametric approach. In: Proc. of the 1st International Workshop on Video Processing and Quality Metrics for Consumer Electronics, VPQM'2005, Scottsdale (2005)
10. Black, M., Sapiro, G., Marimont, D., Heeger, D.: Robust anisotropic diffusion. IEEE Trans. on Image Processing **7** (1998) 421–432
11. Barash, D.: A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. IEEE Trans. on Pattern Analysis and Machine Intelligence **24** (2002) 844–847
12. Lee, S.H., Kang, M.G.: Spatio-temporal video filtering algorithm based on 3-D anisotropic diffusion equation. In: Proc. of IEEE Int. Conf. on Image Processing, ICIP'1998. Volume 3(2)., Chicago (1998) 447–450

13. Rajpoot, N., Yao, Z., Wilson, R.: Adaptive wavelet restoration of noisy video sequences. In: Proc. of IEEE Int. Conf. on Image Processing, ICIP'2004, Singapore (2004)
14. Dekeyser, F., Bouthemy, P., Pérez, P.: Spatio-temporal Wiener filtering of image sequences using a parametric motion model. In: IEEE Int. Conf. on Image Processing, ICIP'2000, Vancouver (2000)
15. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60** (1992) 259–268
16. Lepski, O.: Asymptotically minimax adaptive estimation 1: upper bounds. SIAM Journal Theory of Probability and Application **36** (1991) 654–659
17. Gasser, T., Sroka, L., Jennen Steinmetz, C.: Residual variance and residual pattern in nonlinear regression. Biometrika (1986) 625–633
18. Zlokolica, V., Philips, W.: Motion and detail adaptive denoising in video. Proc. of SPIE – Image Processing: Algorithms and Systems III **5298** (2004) 403–412
19. Odobez, J., Bouthemy, P.: Robust multiresolution estimation of parametric motion models. Journal of Visual Communication and Image Representation **6** (1995) 348–365