NonLocal image and movie denoising

Antoni Buades, Bartomeu Coll, Jean-Michel Morel

Nonlocal image and movie denoising

Antoni Buades Bartomeu Coll Jean-Michel Morel

Abstract

Neighborhood filters are nonlocal image and movie filters which reduce the noise by averaging similar pixels. The first object of the paper is to present a unified theory of these filters and reliable criteria to compare them to other filter classes. A CCD noise model will be presented justifying the involvement of neighborhood filters. A classification of neighborhood filters will be proposed, including classical image and movie denoising methods and discussing further a recently introduced neighborhood filter, NL-means. In order to compare denoising methods three principles will be discussed. The first principle, "method noise", specifies that only noise must be removed from an image. A second principle will be introduced, "noise to noise", according to which a denoising method must transform a white noise into a white noise. Contrarily to "method noise", this principle, which characterizes artifact-free methods, eliminates any subjectivity and can be checked by mathematical arguments and Fourier analysis. "Noise to noise" will be proven to rule out most denoising methods, with the exception of neighborhood filters. This is why a third and new comparison principle, the "statistical optimality", is needed and will be introduced to compare the performance of all neighborhood filters.

The three principles will be applied to compare ten different image and movie denoising methods. It will be first shown that only wavelet thresholding methods and NL-means give an acceptable method noise. Second, that neighborhood filters are the only ones to satisfy the "noise to noise" principle. Third, that among them NL-means is closest to statistical optimality. A particular attention will be paid to the application of the statistical optimality criterion for movie denoising methods. It will be pointed out that current movie denoising methods are motion compensated neighborhood filters. This amounts to say that they are neighborhood filters and that the ideal neighborhood of a pixel is its trajectory. Unfortunately the aperture problem makes it impossible to estimate ground true trajectories. It will be demonstrated that computing trajectories and restricting the neighborhood to them is harmful for denoising purposes and that space-time NL-means preserves more movie details.

1 Introduction

1.1 Neighborhood filters

The main objective of this paper is to set under a common framework and give comparison principles to all neighborhood filters, including movie filters which are usually treated as a different class. We shall call *neighborhood filters* all image and movie filters which reduce the noise by averaging similar pixels. General CCD noise models (briefly presented in Section 2) imply that noise in digital images and movies is signal dependent. Fortunately two pixels which received the same energy from the outdoor scene undergo the same kind of perturbations and therefore have the same noise model. Under the fairly general assumption that at each energy

level the noise model is additive and white, denoising can be achieved by first finding out the pixels which received the same original energy and then averaging their observed grey levels. This observation has led to the wide class of neighborhood filters classified in [48]. Since the original image value is lost these filters proceed by picking for each pixel i the set of pixels $J(i)$ spatially close to i and with a similar grey level value.

Neighborhood filters proceed by replacing the grey level value of i, $u(i)$, by the average $NFu(i) = \frac{1}{|J(i)|}\sum_{j\in J(i)} u(j)$. (Depending on the noise model other statistical estimates are of course possible like the median, etc.) Under the assumption that pixels $j \in J(i)$ indeed received the same original energy as i, $NFu(i)$ is a denoised version of $u(i)$. The more famous neighborhood filters are Lee's σ-filter [28], SUSAN [42] and the bilateral filter [44] where the neighborhoods are gaussian in space and grey level.

1.2 Non local means

In a recent communication [8] (see also [7] for a mathematical analysis) the authors of the present paper extended the above mentioned neighborhood filters to a wide class which they called *non-local means (NL-means)*. This algorithm class defines the neighborhood $J(i)$ of i by the condition: $j \in J(i)$ if the grey level of a whole window around j is close to the grey level of the window around i. The spatial constraint is instead relaxed.

NL-means filters can be given two origins beyond classical neighborhood filters. The same Markovian pixel similarity model was used in the seminal paper [15] for texture synthesis from a texture sample. In that case the neighborhood $J(i)$ is not used for denoising. The aim was to estimate from the texture sample the law of i knowing its neighborhood. This law is used to synthesize similar texture images by an iterative algorithm.

1.3 Movie denoising

Last but not least, most state of the art movie denoising methods are neighborhood filters and some of them, in some sense, NL-means filters. Indeed, motion compensated denoising methods start with the search for a temporal neighborhood $J(i)$, the trajectory, followed by an averaging process. By the Lambertian assumption a pixel belonging to a certain object conserves the same grey level value during its trajectory. Therefore this is computed as a grey level neighborhood of i in the sense of neighborhood filters. The comparison of grey levels is not a sufficient criterion, a difficulty usually called the *aperture problem*. Thus several motion compensated filters involve *block matching*. They construct $J(i)$ by comparing a whole block around j to a whole block around i.

In all of these movie denoising algorithms the neighborhood $J(i)$ picks a single pixel per frame. One of our objectives is to prove that this restriction is actually counterproductive. In fact the performance of movie denoising filters improves significantly by forgetting about trajectories and using all similar pixels in space-time, no matter how many are picked per frame. For this reason the NL-means filter treats movies as a union of images, rather than an image sequence. The time ordering of this union is irrelevant. The aperture problem results in the existence of more samples for each pixel and therefore increases the denoising performance by nonlocal means.

1.4 Three comparison principles

A systematic comparison of the huge variety of denoising methods is requested. On the other hand a comparison between methods which are based on very different principles cannot be performed without formal comparison criteria. Visual comparison of artificially noisy images with their denoised version is subjective. Tables comparing distances of the denoised image to the original are useful. They have two drawbacks, though. The added noise is usually not realistic, generally a white uniform noise with too large variance. Such comparison methods depend strongly on the choice of the image and do not permit to address the main issues: the loss of image structure in noise and the creation of artifacts.

Thus we shall apply three principles aiming at more objective benchmarks. The first principle (already presented in [8]) asks that noise and only noise be removed from an image. It has to be perceptually tested directly on an image with no artificial noise added. The comparison of methods is performed on the difference between the image and its denoised version. We called this difference *method noise*. It is much easier to evaluate whether a method noise contains some structure removed from the image or not. The outcome of such experiments is clear cut on a wide class of denoising filters of all origins including all mentioned neighborhood filters.

We shall introduce here a second principle, *noise to noise*, which requires that a denoising algorithm transforms a white noise into a white noise. This paradoxical requirement seems to be the best way to characterize artifact-free algorithms. It is affordable to mathematical analysis and to Fourier spectrum testing. Mathematical and experimental arguments will show that bilateral filters and NL-means are the only ones satisfying the noise to noise principle.

The third principle, the *statistical optimality* is restricted to neighborhood filters. It questions whether a given neighborhood filter is able or not to retrieve faithfully the neighborhood $J(i)$ of any pixel i. NL-means will be shown to best match this requirement. We shall apply this principle to demonstrate that, contrarily to the current dominant technology, motion estimation or compensation is not needed, and even harmful, to perform movie denoising.

1.5 The extension of patch-based denoising methods

Since a first version [9] of the present paper was disclosed in May 2005, several variants of non-local, or "patch-based" methods and very careful denoising benchmarks have been published by several authors. Mahmoudi and Sapiro [30] reported excellent denoising results and acceleration methods for the non local means algorithms applied to images and movies. Kervrann et al. [23], [24] proposed an adaptive extension of NL-means with variable window size depending on statistical estimates, and performed an impressive comparison benchmark. Awate and Whitaker [4] have simultaneously proposed a method whose principles stand close to the NL-means algorithm, since the method involves comparison between subwindows to estimate a restored value. The objective of the algorithm "*UINTA, for unsupervised information theoretic adaptive filter*", is to denoise the image by decreasing the randomness of the image. Azzabou and Paragios [5] have proposed an acceleration of NL-means by a random walk exploration around each pixel and have again reported impressive results for this accelerated NL-means. Kindermann et al. [25] and Gilboa et al. [18] have given a new, variational framework to non-local denoising leading to iterated algorithms. They have also performed extensive comparisons with total variation denoising.

Probably the most impressive results for a block matching based denoising have been just reported by Dabov et al. [12]. Let us summarize their methods in their own terms: *(We start*

by) "grouping similar 2D image fragments (e.g. blocks) into 3D data arrays called "groups". Collaborative filtering is a special procedure developed to deal with these 3D groups. We realize it using the three successive steps: 3D transformation of 3D group, shrinkage of transform spectrum, and inverse 3D transformation. The result is a 3D estimate that consists of the jointly filtered grouped image blocks. By attenuating the noise, the collaborative filtering reveals even the finest details shared by grouped blocks and at the same time it preserves the essential unique features of each individual block. The filtered blocks are then returned to their original positions. Because these blocks are overlapping, for each pixel we obtain many different estimates which need to be combined. Aggregation is a particular averaging procedure which is exploited to take advantage of this redundancy. (...) The experimental results presented here demonstrate that the developed methods achieve state-of-the-art denoising performance in terms of both peak signal-to-noise ratio and subjective visual quality."

The most striking point of this method is the fusion of NL-means with a transform-domain shrinkage. The main step remains block matching, but then the process is analogous to a motion compensated movie denoising algorithm : all similar blocks are put into a single 3D volume, where time is replaced by a similarity order. We may anticipate that our conclusions are similar to the conclusions of [12]. Applied to a movie, their algorithm will lead to substitute to the movie time a "similarity time", where all blocks are put after each other based on their similarity and not on their time order. This yields much more redundancy, and therefore to a better denoising.

1.6 Plan

We shall proceed as follows. Section 2 presents a realistic CCD noise model which leads to the basic hypothesis justifying neighborhood filters. Neighborhood filters including NL-means and motion compensated movie denoising filters are defined in Section 3. This section describes and discusses some main movie filters. Section 4 proposes the three principles to evaluate the performance of any denoising method. All three principles are designed to serve comparative experiments. Finally, the last section is devoted to a more mathematical comparison of classical neighborhood filters and the NL-means.

2 Noise model

Most digital images and movies are obtained by a CCD device. Following [10, 13, 19], CCD's show three kinds of noise. The first one is the *shot noise* proportional to the square root of the number of incoming photons in the captors during the exposure time, namely

$$n_0 = \sqrt{\frac{\Phi}{h\nu} t \cdot A \cdot \eta},$$

where Φ is the light power (W/m^2), $h\nu$ the photon energy (Ws), t the exposure time in seconds (s), A the pixel area (m^2) and η the quantum efficiency. The other constants being fixed we can simply retain $n_0 = c\sqrt{\Phi}$ where Φ is the "true image" and C a constant (see Figure 1).

Second, a *dark* or *obscurity noise* n_1 is due to spurious photons produced by the captor itself. We can assume the dark noise to be white, additive and with zero mean. The zero mean property is due to the substraction of a *dark frame* from the raw image. The dark frame is obtained by averaging the obscurity noise over a long period of time.

Figure 1: Simulated shot noise. Left: original image u. Right: noise image $\sqrt{u}n$ where n is the realization of a zero mean white noise with standard deviation $\sigma = 1$. The noise in bright parts is larger than in dark parts, an effect which is corrected and sometimes reversed by gamma-correction.

Third, the *read out* noise n_2 is another electronic additive and signal independent noise. This noise can be assumed to have zero mean by the substraction from the raw image of a *bias frame*.

Digital images eventually undergo a "gamma" correction, i.e. a nonlinear increasing contrast change g: "Gamma correction is the name of an internal adjustment made in the rendering of images through photography, television, and computer imaging. The adjustment causes the spacing of steps of shade between the brightest and dimmest part of an image to appear "appropriate" [19]." Summarizing,

$$u(i) = g\left(\Phi(i) + c\sqrt{\Phi(i)}n_0(i) + n_1(i) + n_2(i)\right),$$

where $u(i)$ is the observed intensity at a pixel i, $\Phi(i)$ the "true physical" light intensity average power sent by the scene to pixel i, c a constant, $n_0(i)$, $n_1(i)$ and $n_2(i)$ three independent and signal independent white noises. In practice $g(s) = s^\alpha$ with $0 < \alpha < 1$. When $\Phi(i)$ is large the shot noise $\sqrt{\Phi(i)}$ dominates n_1 and n_2 and is dominated by the signal $\Phi(i)$. Thus we can expand $u(i)$ as

$$u(i) \simeq g(\Phi(i)) + g'(\Phi(i))\left(c\sqrt{\Phi(i)}n_0(i) + n_1(i) + n_2(i)\right) =: g(\Phi(i)) + n(i). \tag{1}$$

If instead $\Phi(i)$ is small with respect to $n_1(i) + n_2(i)$,

$$n(i) \simeq u(i) \simeq g(n_1(i) + n_2(i)). \tag{2}$$

Let us mention a case of particular interest. If $g(s) \simeq s^{\frac{1}{2}}$, the noise $n(i)$ reads

$$n(i) \simeq \begin{cases} n_0(i) & \text{in the bright parts of the image} \\ \sqrt{n_1(i) + n_2(i)} & \text{in the dark parts of the image} \end{cases} \tag{3}$$

In all cases the noise is signal dependent but independent at different pixels. Figure 1 displays a simulated shot noise associated to the Lena image. This noise is signal dependent and much stronger in bright regions than in dark regions. In order to apply many computer vision

algorithms, the noise parameters must be first estimated. For the study of these parameters for the previous real CCD model we refer the reader to [29].

In the following we aim at recovering $g(\Phi(i))$, namely the true image up to the unknown gamma correction. Equations (1), (2) and the white noise and independence assumptions on n_0, n_1 and n_2 legitimate the following hypothesis:

Hypothesis 1 *In a digital image, the noise model at each pixel i only depends on the original pixel value $\Phi(i)$ and is additive. Let $J(i)$ be the set of pixels with the same original value as i. Then $n(j)$, $j \in J(i)$ are independent and identically distributed.*

Hypothesis 1 cannot be used directly because $\Phi(i)$ is unknown. The challenge is to find out $J(i)$ for every i. The simplest idea to do so is to assume that all pixels with the same observed value $u(i)$ have the same noise model. A more sophisticated use of Hypothesis 1 is the following : *for a given pixel in an image, detect all pixels which have the same underlying model.* By Hypothesis 1 each j in $J(i)$ obeys a model $u(j) = v(i) + n(j)$ where $n(j)$ are i.i.d. It is then licit to perform a denoising of $u(i)$ by replacing it by $NFu(i) =: \frac{1}{|J(i)|}\sum_{j\in J(i)} u(j)$. By the variance formula for independent variables one then obtains $NFu(i) = v(i)+\tilde{n}(i)$ where

$$\mathrm{Var}(\tilde{n}(i)) = \frac{1}{|J(i)|}\mathrm{Var}(n(i)). \tag{4}$$

By (4) if nine pixels with the same color plus some uncorrelated noise are averaged the noise is divided by three. Algorithms proceeding in this way will be called neighborhood filters. We shall now examine several classical or new instances.

3 General neighborhood filters

3.1 Local neighborhood filters

The more primitive neighborhood filters replace the color of a pixel with an average of the nearby pixels colors. Thus $J(i)$ is a spatial neighborhood. The filtered value can be written as

$$\mathcal{M}_\rho u(\mathbf{x}) = \frac{1}{\pi\rho^2}\int_{\mathbb{R}^2} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{\rho^2}} u(\mathbf{y})\, d\mathbf{y},$$

where the parameter ρ is roughly the size of the spatial neighborhood involved in the filtering. Now, the closest pixels to i have not necessarily the same color as i. For instance the red pixel placed in the middle of Figure 2 has five red neighbors and three blue ones. If its color is replaced by the average of the colors of its neighbors, it turns blue. The same process would likewise redden the blue pixels of this figure. Thus, the border between red and blue would be blurred.

In order to denoise the central red pixel, it is better to average the color of this pixel with the nearby red pixels and only them, excluding the blue ones. This is exactly the technique of the sigma-filter. This famous algorithm is generally attributed to J.S. Lee [28] in 1983 but can be traced back to L. Yaroslavsky and the Soviet Union image processing school [48]. The idea is to average neighboring pixels which also have a similar color value. The filtered value by this strategy can be written

$$NF_{h,\rho}\, u(\mathbf{x}) = \frac{1}{C(\mathbf{x})}\int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} u(\mathbf{y}) d\mathbf{y}, \tag{5}$$

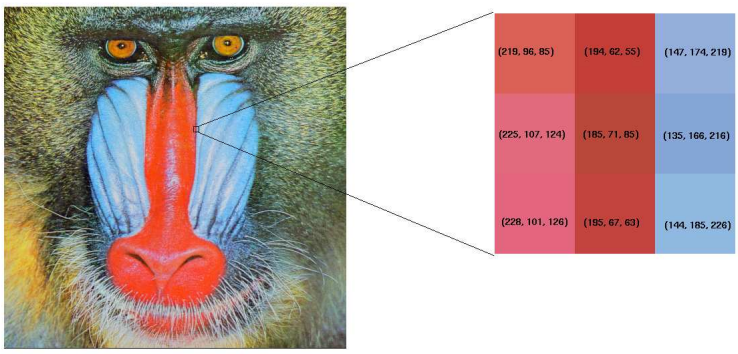

Figure 2: The nine pixels in the baboon image on the left have been enlarged. They present a high red-blue contrast. In the red pixels, the first (red) component is stronger. In the blue pixels, the third component, blue, dominates. Neighborhood filters select pixels with the same color for averaging. In this case the neighborhood of the central pixel should contain the six red pixels or, still better, the pixels of the central column.

where $u(\mathbf{x})$ is the color at $\mathbf{x}$ and $NF_{h,\rho}\, u(\mathbf{x})$ its denoised version. Only pixels inside $B_\rho(\mathbf{x})$ are averaged, h controls the color similarity and $C(\mathbf{x})$ is the normalization factor. SUSAN [42] and the bilateral filter [44] make this process more symmetric by involving a "bilateral" gaussian depending on both space and grey level. This leads to

$$SNF_{h,\rho}\, u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{\rho^2}} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} u(\mathbf{y}) d\mathbf{y}$$

There is another way to avoid the blurring effect of the spatial filtering $\mathcal{M}_\rho$ by a statistical correction which we are going to use in the sequel. When the gaussian mean is performed on an edge, the variance of the performed mean can become larger than the variance of the noise. This is a clue that the average is not licit. A statistically optimal correction was proposed by Lee again [27],

$$L\mathcal{M}_\rho u(\mathbf{x}) = \mathcal{M}_\rho u(\mathbf{x}) + \frac{\sigma_{\mathbf{x}}^2}{\sigma_{\mathbf{x}}^2 + \sigma^2}(u(\mathbf{x}) - \mathcal{M}_\rho u(\mathbf{x})),$$

where

$$\sigma_{\mathbf{x}}^2 = \max(0, \frac{1}{\pi\rho^2} \int_{\mathbb{R}^2} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{\rho^2}} (u(\mathbf{y}) - \mathcal{M}_\rho u(\mathbf{x}))^2 \, d\mathbf{y} - \sigma^2)$$

and σ is the noise standard deviation. The original noisy values are less altered when the variance of the performed mean dominates the variance of the noise. This happens near the edges or in textured regions. In consequence, the noise is mainly reduced in flat zones as displayed in Figure 4.

The bilateral filters perform a better denoising than Lee's correction. They maintain sharp boundaries, since they average pixels belonging to the same region as the reference pixel. Bilateral filters fail when the standard deviation of the noise exceeds the edge contrast. This fact is more extensively exposed in Section 5. In the following experiments we do not distinguish between the classical neighborhood filter and SUSAN or the bilateral filter. The following experiments have been performed using a fixed spatial neighborhood and a Gaussian weighted grey level difference.

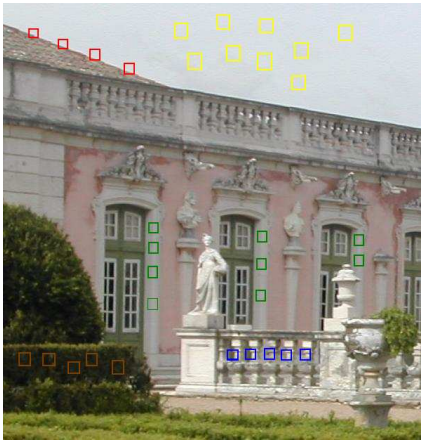

Figure 3: Most image details occur repeatedly. Each color indicates a group of squares in the image which are almost indistinguishable. Image self-similarity can be used to eliminate noise. It suffices to average the squares which resemble each other.

Let us finally mention that the mean operation can be replaced by nonlinear operator like the median filter. The median filter [45] chooses the median value, that is, the value which has exactly the same number of grey level values above and below in a fixed neighborhood. The median filter preserves the main boundaries, but it tends to remove the details. This filter is optimal for the removal of impulse noise on images and does'nt blur edges. It is equivalent to an average of the pixels in a direction orthogonal to the gradient, that is to an *anisotropic diffusion* or *mean curvature motion* [32]. In the following, we shall also show experiments based on the mean curvature motion implemented as originally proposed in [1]. For a more recent review of the subject see [22].

Figure 4 compares the performance of the various considered local neighborhood filters on a noisy image. Even if each method provides a reasonable solution (all except the gaussian filtering maintain sharp edges), none of them is fully acceptable. The anisotropic filter removes small details and fine structures. These features are nearly untouched by Lee's statistical filter and therefore completely noisy. The bilateral filters create irregularities on the edges and leave some residual noise on flat zones.

3.2 Non local averaging

The most similar pixels to a given pixel have no reason to be close to it. Think of periodic patterns, or of the elongated edges which appear in most images. In 1999 Efros and Leung [15] used non local self-similarities as the ones illustrated in Figure 3 to synthesize textures and to fill in holes in images. Their algorithm scans a vast portion of the image in search of all the pixels that resemble the pixel in restoration. The resemblance is evaluated by comparing a whole window around each pixel, not just the color of the pixel itself. Applying this idea to neighborhood filters leads to a generalized neighborhood filter which we called non-local means (or NL-means) [7, 8]. NL-means has a formula quite similar to the sigma-filter,

$$NLu(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{\Omega} e^{-\frac{(G_\rho * |u(\mathbf{x}+.)-u(\mathbf{y}+.)|^2)(0)}{h^2}} u(\mathbf{y})\, d\mathbf{y}, \tag{6}$$

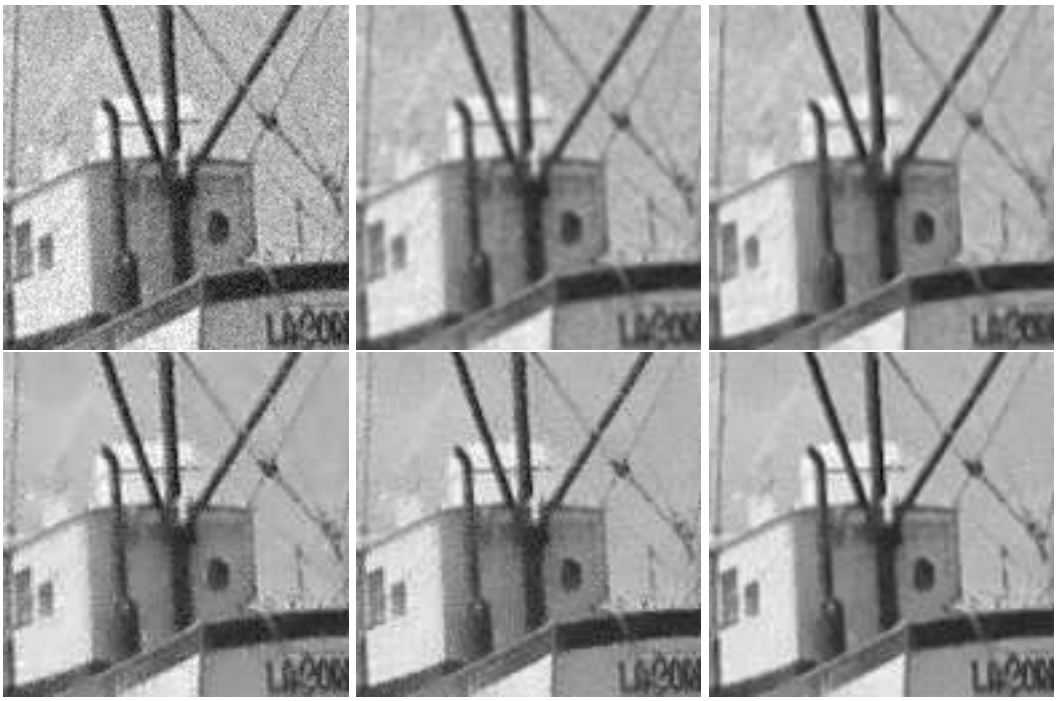

Figure 4: Comparison of neighborhood filters. From top to bottom and left to right: noisy image ($\sigma = 15$), gaussian filtering, anisotropic filtering, Lee's statistical filter, sigma or bilateral filter and the NL-means algorithm. All methods except the gaussian filtering maintain sharp edges. However, the anisotropic filtering removes small details and fine structures. These features are nearly untouched by Lee's statistical filter and therefore completely noisy. The comparison of noisy grey level values by the sigma or bilateral filter is not so robust and irregularities are created on the edges. The NL-means better cleans the edges without losing too many fine structures and details.

where G_ρ is the Gauss kernel with standard deviation ρ, $C(\mathbf{x})$ is the normalizing factor, h acts as a filtering parameter and

$$(G_\rho * |u(\mathbf{x}+.) - u(\mathbf{y}+.)|^2)(0) = \int_{\mathbb{R}^2} G_\rho(\mathbf{t})|u(\mathbf{x}+\mathbf{t}) - u(\mathbf{y}+\mathbf{t})|^2 d\mathbf{t}.$$

The formula (6) means that $u(\mathbf{x})$ is replaced by a weighted average of $u(\mathbf{y})$. The weights are significant only if a gaussian window around $\mathbf{y}$ looks like the corresponding gaussian window around $\mathbf{x}$. Thus the non-local means algorithm uses image self-similarity to reduce the noise as illustrated in Figure 4. As Figure 5 shows, the NL-means seems to be well adapted to text denoising since characters and combinations of characters are easily repeated in a text. One of the limitations of the NL-means algorithm is the removal of highly structured noise as in jpeg compressed images (see Figure 6). The NL-means is able to remove the block artifact due to compression but at the cost of removing some details as the difference between the compressed and restored images shows.

3.3 NL-means implementation details

A NL-means simple version averages pixels which have a grey level window around at a distance less or equal than a certain threshold. The comparison of both windows is made by an Euclidean norm of their difference. Indeed, if the noise samples are locally i.i.d. with zero mean and variance σ^2 ,

$$E||u(\mathcal{N}_i) - u(\mathcal{N}_j)||^2 = ||u_0(\mathcal{N}_i) - u_0(\mathcal{N}_j)||^2 + 2\sigma^2,$$

where u_0 denotes the original (unknown) image and u the noisy one obtained by the addition of a white noise. Thus using a threshold function and setting this hard threshold to $2\sigma^2$ leads to take an average of pixels which originally had an almost identical window around them.

So the point is: why using a soft gaussian threshold instead of a hard one? We may find pixels for which there is no identical or nearly identical window in the image. In that case, the threshold strategy should leave exactly the noise value at such points. The result would visually be identified as an impulse noise and the *noise to noise* principle would be violated. An exponential function is used instead of the threshold and makes a more adaptive weighting distribution.

The Euclidean distance between two windows is also weighted by a gaussian-like kernel decaying from the center of the window to its boundary. Indeed in digital images, closer pixels are more dependent and therefore closer pixels to the reference one should have more importance in the window comparison. In order to involve the current pixel in its own average, the distance between the window centered at the reference pixel and itself is set equal to the minimum of the other distances. Otherwise, the probability distribution should be excessively large at the pixel itself.

For computational purposes of the NL-means algorithm, we can restrict the search of similar windows in a larger "search window" of size $S \times S$ pixels. In all experiments we have fixed a 21×21 pixels search window and a similarity square neighborhood of 7×7 pixels (this can be reduced for color images). If the image has a size $N \times N$, then the final complexity of the algorithm is about $49 \times 441 \times N^2$. The acceleration of the algorithm by multiresolution strategies has been proposed in [7, 30].

The filtering parameter h has been fixed to $k * \sigma$ with $k \in [0.75, 1]$, when a noise of standard deviation σ is added. Due to the fast decay of the exponential kernel, large Euclidean distances lead to nearly zero weights acting as an automatic threshold (see Figures 16 and 18).

3.4 Movie denoising

Nearly all state of the art movie filters are motion compensated. The underlying idea is the existence of a "ground true" physical motion, which motion estimation algorithms should be able to estimate. Legitimate information should exist only along these physical trajectories. The *motion compensated filters* estimate explicitly the motion of the sequence by a motion estimation algorithm. The motion compensated movie yields a new stationary data on which a static filter can be applied.

One of the major difficulties in motion estimation is the ambiguity of trajectories, the so called *aperture problem*. This problem is illustrated in Figure 8. At most pixels, there are several options for the displacement vector. All of these options have a similar grey level value and a similar block around them. Now, motion estimators have to select one by some additional criterion. The most classical approaches to motion estimation are the *optical flow constraint (OFC)* based methods and *the block matching* algorithms. OFC methods assume that the grey

Figure 5: Comparison of different denoising methods on a text image. Top: noisy image. Below and from left to right: crop of the noisy image, the total variation minimization, the stationary wavelet thresholding, the neighborhood filter with spatial neighborhood 11×11 and the NL-means with the whole image as search window. NL-means seems well adapted to text denoising since characters and combinations of characters are easily repeated in a text. Since we have used only three paragraphs the result can be substantially improved by using a complete page or even more pages. In that case the problem would be the huge time of computation.

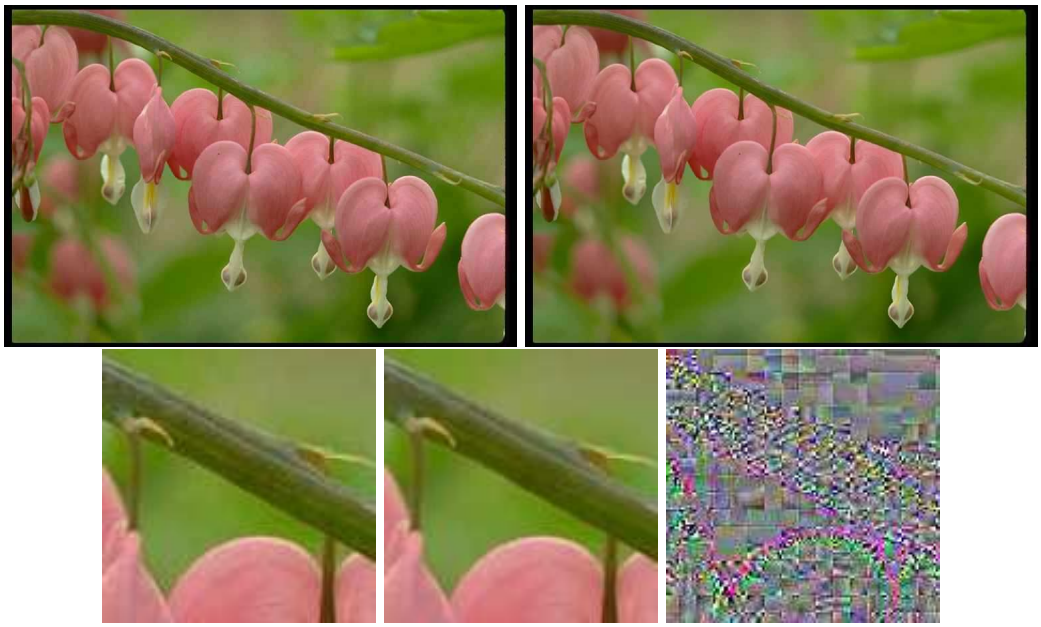

Figure 6: Application of NL-means to restore a highly compressed image. Top: Compressed and restored image by NL-means. Bottom: detail of previous images and its difference. NL-means is able to remove block artifacts due to compression but at the cost of removing some details, as the difference between the compressed and restored image shows.

level value of the objects during their trajectory is nearly constant (Lambertian assumption). In order to obtain a unique flow they impose the flow field to vary smoothly in space. There has been a constant progress in this estimation Horn-Schunck [20], Nagel and Weickert [36, 46], Weickert and Schnor [47]. Other constraints enforcing the constancy of the gradient and the Laplacian can be added as proposed in [34].

The second class of motion estimation algorithms computes the displacement at each pixel by comparing the grey level values in a whole block around it. The similarity is measured by a l^1 or a l^2 distance. As Figure 8 shows, there can be many blocks with similar configurations in the reference frame.

Once the motion compensation has been performed one can classify the movie denoising methods by the kind of 3D neighborhood filter they apply to the compensated movie. We refer to [6] for a comprehensive review. Samy [40] and Sezan et al. [41] proposed the LMMSE filter which is a *motion compensated Lee's filter*. Ozkan et al [37] proposed the AWA filter, a *motion compensation of the neighborhood filters*. Huang [21] and Martinez [31] implemented *motion compensated median filters*. *Motion compensated Wiener filters* were earlier proposed by Kokaram [26].

Figure 9 illustrates the improvement of motion compensated algorithms compared with their static version. The static filters are obtained by extending the 2D image filter support to the 3D time-space support. We call them static because they do not take into account the dynamic character of image sequences. As this figure illustrates, the details are better preserved and the boundaries less blurred with motion compensation. This explains why most recent papers

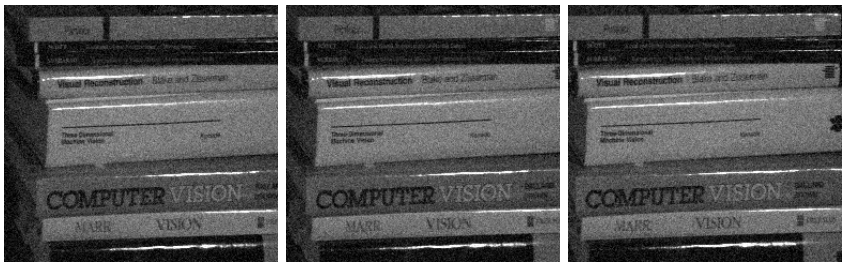

Figure 7: Three consecutive frames of a degraded image sequence. The sparse time sampling in film sequences makes restoration more difficult than in 3D images.

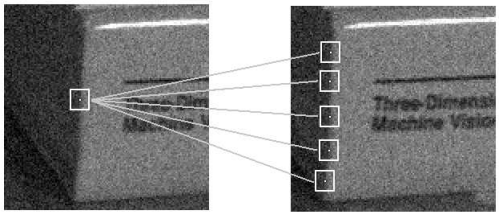

Figure 8: Aperture problem and the ambiguity of trajectories are the most difficult problem in motion estimation: There can be many good matches. The motion estimation algorithms must pick one.

propose motion compensated algorithms.

Non local means for movies

The above description of movie denoising algorithms and its juxtaposition to the NL-means principle shows how the main problem, motion estimation, can be circumvented. In denoising, the more samples we have the happier we are. The *aperture problem* is just a name for the fact that there are many blocks in the next frame similar to a given one in the current frame. Thus, singling out one of them in the next frame to perform the motion compensation is an unnecessary and probably harmful step. A much simpler strategy which takes advantage of the aperture problem is to denoise a movie pixel by involving indiscriminately spatial and temporal similarities: let the best win! In that way, we propose to apply the NL-means to a movie in the following way:

$$NLu(\mathbf{x},t) = \frac{1}{C(\mathbf{x},t)} \int_{\Omega} \int_{\mathbb{R}} e^{-\frac{(G_\rho * |u(\mathbf{x}+.,t) - u(\mathbf{y}+.,s)|^2)(0)}{h^2}} \, u(\mathbf{y},s) \, d\mathbf{y} ds. \tag{7}$$

Notice that the Gauss kernel G_ρ is 2D but the integral involves time and space. In practical terms NL-means takes a spatial block around pixel $\mathbf{x}$ in frame t and looks for all similar blocks in all frames *including the current one*. Then a weighted average of all the similar pixels $(\mathbf{y}, s)$ is performed. Figure 9 compares NL-means with motion compensated neighborhood filters. The NL-means also reduces motion blur since it is not affected by possible errors of motion estimation algorithms.

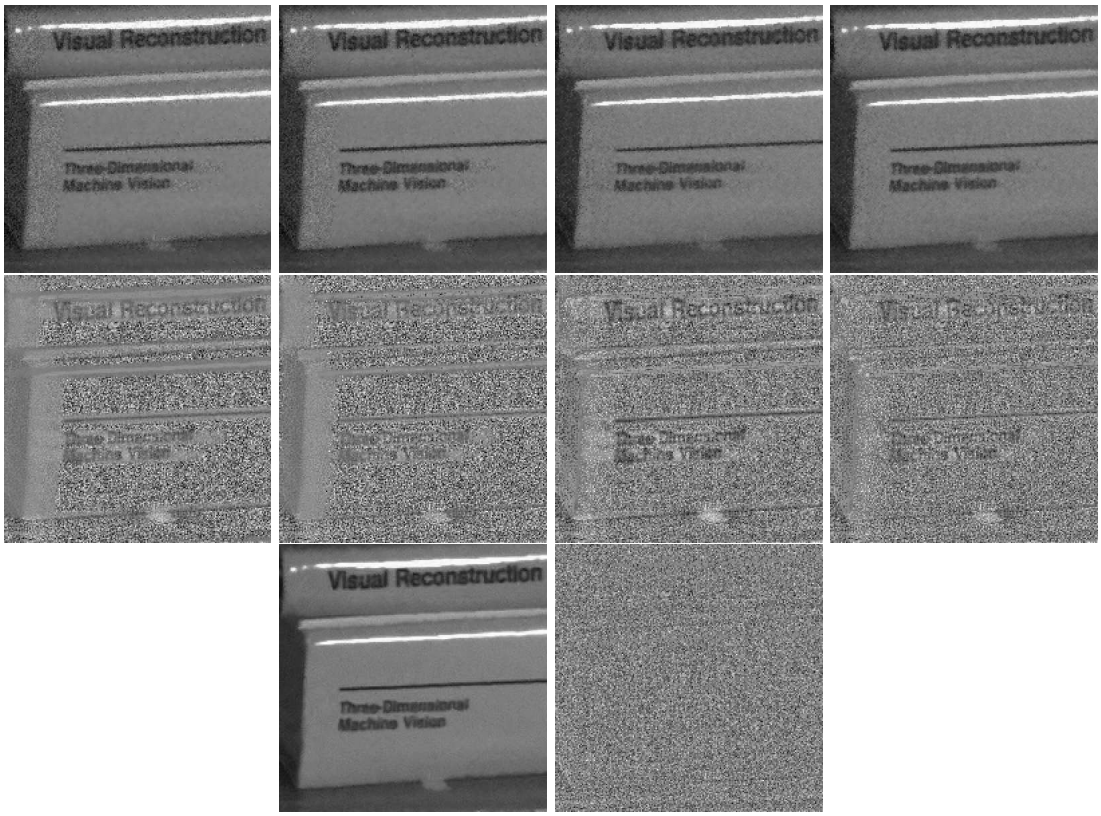

Figure 9: Comparison of static filters, motion compensated filters and NL-means applied to the sequence of Figure 7. Only a piece of the central frame is displayed. From top to bottom and left to right: Lee's correction, Lee's correction with block matching (BM)-(LMMSE), sigma filter and sigma filter with block-matching (AWA). Middle: the noise removed by each method (difference between the noisy and filtered frame). Motion compensation improves the static algorithms by better preserving the details and creating less blur. The noise removed by LMMSE is nearly zero on the strong boundaries. Thus, these boundaries are kept noisy. We can read the titles of the books in the noise removed by AWA. Therefore, that much information has been removed from the original. Finally, the NL-means algorithm (bottom row) has almost no noticeable structure in its removed noise. As a consequence, the filtered sequence has kept more details and is less blurred.

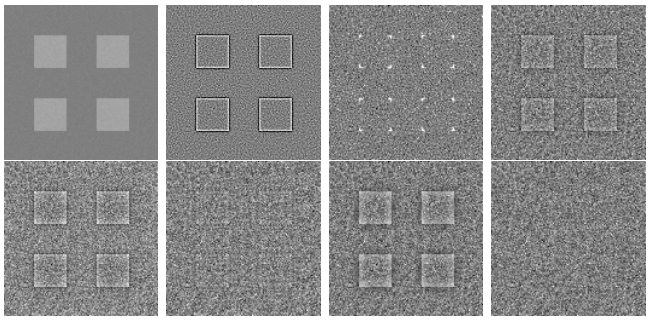

Figure 10: *Method noise* obtained by various denoising methods applied to a slightly noisy image ($\sigma = 2.5$). From top to bottom and left to right: noisy image, gaussian mean, mean curvature motion, total variation minimization, translation invariant soft and hard thresholding, bilateral filter and NL-means. The various method parameters have been adjusted so that the method noise has a per pixel variance equal to σ^2.

4 Three principles for denoising algorithms evaluation

In this section, we shall address the problem of comparing different denoising methods by proposing three formal comparison criteria.

4.1 Method noise

A difference between the original image and its filtered version shows the "noise" removed by the algorithm. This procedure has been introduced recently in [7] and this difference or residue is called *method noise*. In principle the method noise should look like a noise. Otherwise, the method noise can be filtered again and its deterministic part turned back to the image. Recent denoising methods adopted this recursive strategy to recover image information lost in method noise [38, 43]. When the standard deviation of the noise is higher than the feature contrast a visual exploration of the method noise is not reliable. Image features can be masked in method noise. Thus the evaluation of a denoising method should not rely on experiments where a white noise with standard deviation larger than 5 has been added to the original. The best way is actually not to add noise at all.

Definition 1 (method noise) *Let u be a (not necessarily noisy) image and D_h a denoising operator depending on h. The* method noise of u *is the image difference*

$$n(D_h, u) = u - D_h(u). \tag{8}$$

Principle 1 *For every denoising algorithm, the method noise must be zero if the image contains no noise and should be in general an image of independent zero-mean random variables.*

Figure 10 and 11 display the method noise of various denoising methods on a simple geometrical image. The algorithms are applied to a slightly noisy version of the image ($\sigma = 2.5$). Method parameters are fixed so that the method noise has exactly σ^2 variance per pixel. The same parameters have been used in the second experiment on a real image. These method noise images should look like a white noise in Figure 10 and like a constant image in Figure 11. The

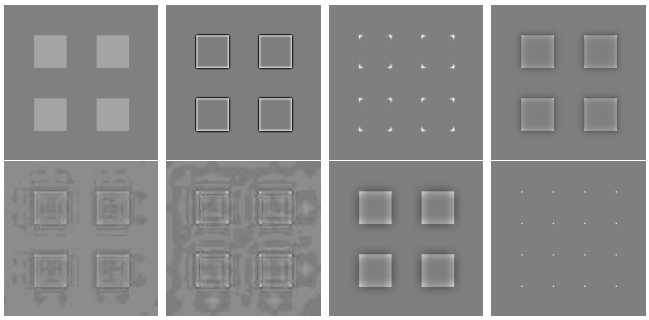

Figure 11: Method noise experiment. Application of various denoising methods to the non noisy image of Figure 10 with the same filtering parameters. From top to bottom and left to right: noisy image, gaussian mean, mean curvature motion, total variation minimization, translation invariant soft and hard thresholding, bilateral filter and the NL-means.

gaussian filter method noise highlights all the boundaries and corners of the image. Averages are performed on a radial neighborhood and therefore do not adapt to the geometrical configuration of the image. The anisotropic (median, mean curvature equation) filter averages pixels in the direction of contours and therefore tends to preserve straight edges. However, the corners are not well preserved since they move at the speed of their high curvature. The iteration of the median or the application of the mean curvature motion for larger times would completely modify the image and even straight edges would not be preserved. The total variation minimization [39] is praised for maintaining sharp boundaries. However, most structures are modified and even straight edges are not well preserved. This fact has received a mathematical proof in [35]. The wavelet thresholding [14] method noise is concentrated on the edges and corners. These structures lead to coefficients of large enough value but lower than the threshold and which are erroneously canceled. The method noise of the soft thresholding is not only based on the small coefficients but also on an attenuation of the large ones, leading to a general alteration of the original image.

The bilateral filter preserves the flat zones, but the edges with a low contrast have been modified. The NL-means method noise is the one which looks the more like a white noise. When applying the algorithms to the non noisy image, the removed features are more noticeable. The corners of the squares can now be seen in the NL-means method noise. These are the only features with a reduced amount of similar samples, since for every corner there are only three similar corners in the image. The experiment of Figure 11 has been designed to illustrate the usage of the method noise on images without noise at all, usually synthetic images. Such experiments characterize immediately the image features sensitive to a given denoising method.

Finally, Figure 12 displays the method noise on a real image. The algorithms were applied to the original Lena image scanned in 1973, in grey level. It contains a little amount of noise. None of the methods can be claimed to deliver a method noise looking like a noise. The hard thresholding and NL-means give the least structured method noise. The method noise of Lena in color is displayed in Figure 13. Color images are obviously easier to denoise by neighborhood filters.

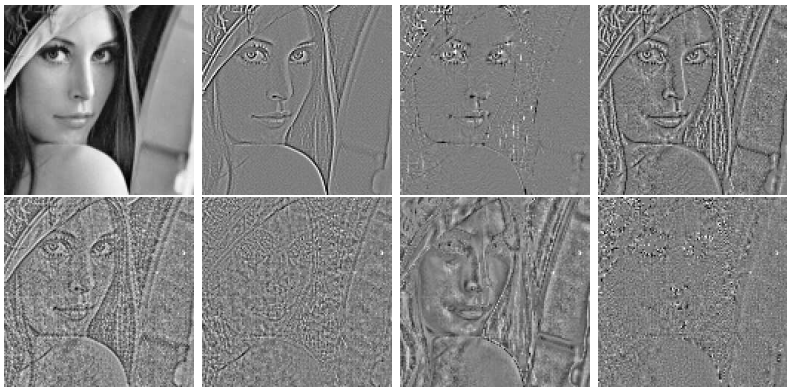

Figure 12: Method noise experiment on Lena (gray levels only). ¿From top to bottom and left to right: original image, gaussian mean, mean curvature motion, total variation minimization, translation invariant soft and hard thresholding, bilateral filter and NL-means.

Figure 13: Lena method noise (in color) by the sigma filter and the NL-means. The results of neighborhood filters improve dramatically on color images because similar pixels are much better identifiable with three components.

4.2 Noise to noise principle

The noise to noise principle asks that a white noise be transformed into a white noise. This requirement may look paradoxical since noise is what we wish to get rid of. Now, it is impossible to totally remove noise. The question is how the remnants of noise look like. The transformation of a white noise into any correlated signal creates structure and artifacts. Only white noise is perceptually devoid of structure, as was pointed out by Attneave [2].

Principle 2 *A denoising algorithm must transform a white noise image into a white noise image (with lower variance).*

There are two ways to check this principle for a given denoising method. One of them is to find a mathematical proof that the pixels remain independent (or at least uncorrelated) and identically distributed random variables. The experimental device simply is an observation the effect of denoising on the simulated realization of a white noise. Since the Fourier transform of a white gaussian noise is a white gaussian noise, the visualization of the Fourier transform also is an adequate tool. Let us review how well classical algorithms match the *noise to noise* principle. Figures 14 and 15 respectively display the filtered noises and their Fourier transforms.

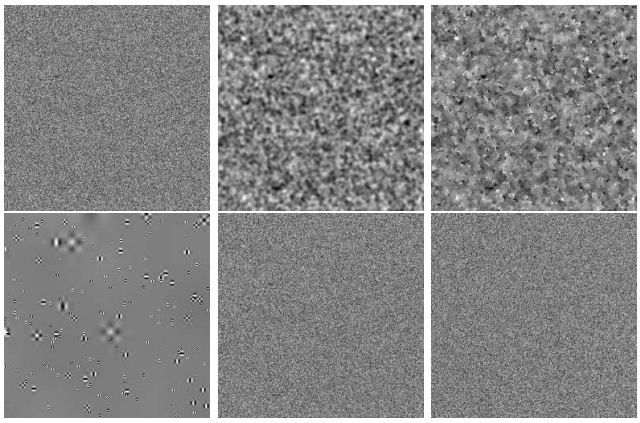

Figure 14: Noise to noise principle: Application of the denoising algorithms to a noise sample. From left to right and top to bottom: noise sample ($\sigma = 15$), filtered noise by the gaussian filtering, total variation minimization, hard wavelet thresholding, bilateral filter and the NL-means algorithm. The parameters of each algorithm have been tuned in order to have a filtered noise of standard deviation 2.5. For the neighborhood or bilateral filter the research zone has been fixed to 21×21 and for NL-means we have used the whole image. Therefore, only the h parameter has been tuned in order to obtain the desired standard deviation.

Gaussian convolution. The convolution with a Gauss kernel G_h is equivalent to the product in the Fourier domain with a Gauss kernel of inverse standard deviation $G_{1/h}$. Therefore, convolving the noise with a kernel reinforces the low frequencies and cancels the high ones. Thus, the filtered noise will no more be a white noise and actually shows big grains due to its prominent low frequencies.

Total variation minimization. The Fourier transforms of the total variation minimization and the gaussian filtering are quite similar even if the total variation preserves some high frequency components.

Wavelet Thresholding. Noise filtered by a wavelet thresholding is no more a white noise. The few coefficients with a magnitude larger than the threshold are spread all over the image. The pixels which do not belong to the support of one of these coefficients are set to zero. The visual result is a constant image with superposed wavelets as displayed in Figure 14. It is easy to prove that the denoised noise is spatially highly correlated.

Bilateral filter. For simplicity consider the case where the grey level neighborhood is an interval. Given a noise realization, the filtered value by the bilateral filter at a pixel i only depends on its value $n(i)$ and the parameters h and ρ. The bilateral filter averages noise values at a distance from $n(i)$ less or equal than h. Thus as the size ρ of the neighborhood increases by the law of large numbers the filtered value tends to the expectation of the Gauss distribution restricted to the interval $(n(i) - h, n(i) + h)$. The filtered value is therefore a deterministic function of $n(i)$ and h. Independent random variables are mapped by a deterministic function on independent variables. Thus the *noise to noise* requirement is asymptotically satisfied by the bilateral filter. A visual check in Figure 15 fully confirms this theoretical result.

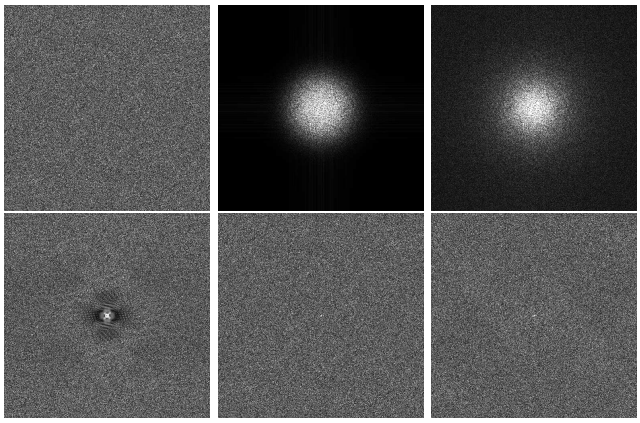

Figure 15: Noise to Noise principle: Fourier transforms of the filtered noises displayed in Figure 14. The Fourier transform of a gaussian white noise is a gaussian white noise.

NL-means algorithm. Figure 15 indicates that NL-means satisfies the *noise to noise* principle in the same extent as a classical bilateral filter. However, a mathematical statement and proof of this property are intricate and we shall skip them.

4.3 Statistical optimality

The statistical optimality means the ability of a generalized neighborhood filter to find the right set of pixels $J(i)$ for performing the average yielding the new estimate for $u(i)$. This principle for the comparison of denoising algorithms applies for all algorithms performing an average over a set of selected pixels, namely, all neighborhood filters including NL-means and all motion compensated movie denoising algorithms (AWA, LMMSE,...).

Principle 3 *A generalized neighborhood filter is optimal if it finds for each pixel i all and only the pixels j having the same model as i.*

Returning to the signal dependent noise model given by Hypothesis 1, we notice that the ideal denoising algorithm from that point of view would give for $J(i)$ the set of all pixels j with the same original, noiseless value $\Phi(j) = \Phi(i)$ as i. This aim is not attainable and can be replaced by a search for pixels j which are *likely* to have the same value as i. In movies, by the Lambertian assumption, it can be assumed that a non occluded pixel keeps the same grey level in several frames. Thus in motion compensated movie filters, $J(i)$ is the trajectory of i. In the case of NL-means, it is assumed that pixels having similar neighborhoods for some distance also have close colors. Principle 3 cannot be checked in theory but can be in practice explored by displaying the probability distribution of $w(j)$, $j \in J(i)$ for various algorithms and images. In that way it can be checked whether $J(i)$ corresponds to the pixels j perceptually equivalent or similar to i. This visualization is actually quite informative as illustrated in Figures 16 and 18.

As displayed in Figure 16, the orientation computed by the anisotropic filter is not exactly the expected one. This fact is due to the noise interference on the gradient computation. The noise also degrades the probability distribution of the bilateral filter. The window comparison of NL-means is more robust to noise and yields a more adapted weight configuration. In the

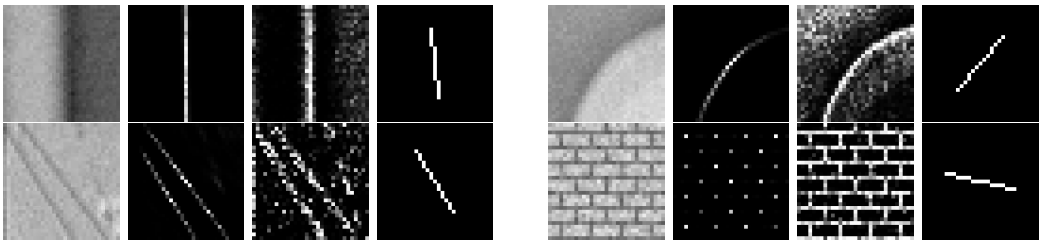

Figure 16: Weight distribution of NL-means, the bilateral filter and the anisotropic filter used to estimate the central pixel in four detail images.

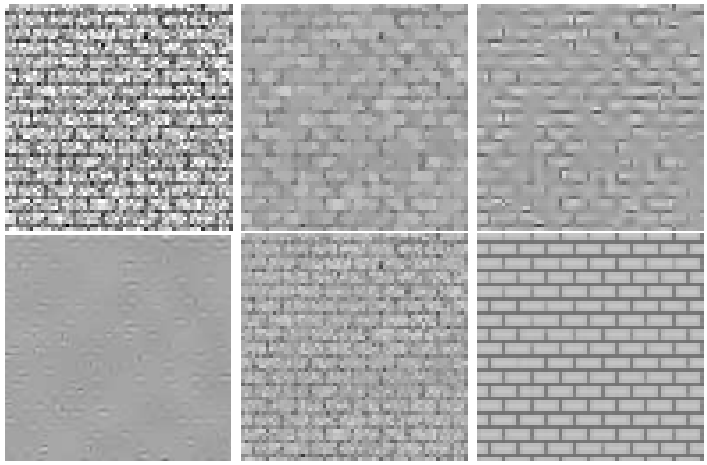

Figure 17: Denoising experiment on a noisy periodic texture. ¿From top to bottom and left to right: noisy image (standard deviation 35), total variation minimization, translation invariant hard thresholding (threshold 3σ), translation invariant hard thresholding (optimal threshold $\sqrt{2\log N}\sigma$), bilateral filter and NL-means.

wall example of Figure 16 NL-means does not find picks the pixels with a similar grey level value while the classical neighborhood does. This avoids mistakes when the standard deviation of the noise is increased as displayed in Figure 17. NL-means shows a good ability for the restoration of binary textures without assuming any prior about the texture statistics. When this prior is available optimal solutions can be obtained by imposing the statistical constraints as recently proposed in [11] via a graph cuts algorithm.

Statistical optimality and the aperture problem.

Let us now address the problem of statistical optimality for movies. We shall sustain the position that, in fact, motion estimation is not only unnecessary, but probably counterproductive. The aperture problem, viewed as a general phenomenon in movies, can be positively interpreted in the following way: There are many pixels in the next or previous frames which can match the current pixel. Thus, it seems sound to use not just one trajectory, but rather *all similar pixels* to the current pixel across time and space.

Motion estimation algorithms try to solve the aperture problem. The block matching algo-

rithm chooses the pixel with the more similar configuration, thus loosing many other interesting possibilities, as displayed in Figure 8. Algorithms based on the optical flow constraint must impose a regularity condition of the flow field in order to choose a single trajectory. Thus, the motion estimation algorithms are forced to choose a candidate among all possible equally good choices. However, when dealing with sequence restoration, the redundancy is not a problem but an advantage. Figure 8 shows all possible and equally good candidates for the averaging. Why not take all of them.

Figure 18 displays the probability distribution of the weights computed by NL-means for three different cases. The algorithm favors pixels with a similar local configuration even if they are far away from the reference pixel. As the similar configurations move, so do the weights. Thus, the algorithm is able to follow the similar configurations when they move without any explicit motion computation. No need to solve any aperture problem. This problem turns out to be a help for denoising purposes.

5 NL-means vs classical neighborhood filters. A simple example

In this section we shall discuss more extensively the application of the neighborhood filters to a simple piece-wise constant image as the one displayed in Figure 19. The analysis of this image can be decomposed in two parts depending if the research window contains the edge or it is totally flat. For this analysis we shall use a simplified version of the NL-means and neighborhood filter, that is,

$$NL_{h,n}u(i) = \frac{1}{|J(i)|}\sum_{j\in J(i)} u(j),$$

where $J(i) = \{j \in R(i) \mid d_n(i,j) \leq h\}$, $R(i)$ denotes a research zone around the interest pixel and $d_n(i,j)$ denotes the squared normalized Euclidean distance between two comparison windows around i and j of size n.

5.1 A research zone totally contained inside the same region.

In that case, the original grey level value of the considered pixels is the same. This case can be viewed as the application of the algorithm to a noise sample and the committed error just as the noise reduction.

As we discussed in section 4.2, the filtered noise value of the neighborhood filter does not depend on the size of the research window but only on the value of h. This means that in order to decrease the error inside flat zones we must increase the value of h as illustrated in Figure 19. As we use the same value for all the image this fact becomes critical when we get near any edge or detail.

The use of a comparison window by NL-means permits a higher noise reduction while keeping a small value of h. We begin by studying the difference between two noise windows of size n. If we denote by d_n the normalized squared Euclidean distance, then

$$\frac{d_n}{2\sigma^2} = \frac{1}{n}\sum_i \left(\frac{m_i}{\sqrt{2}\sigma}\right)^2$$

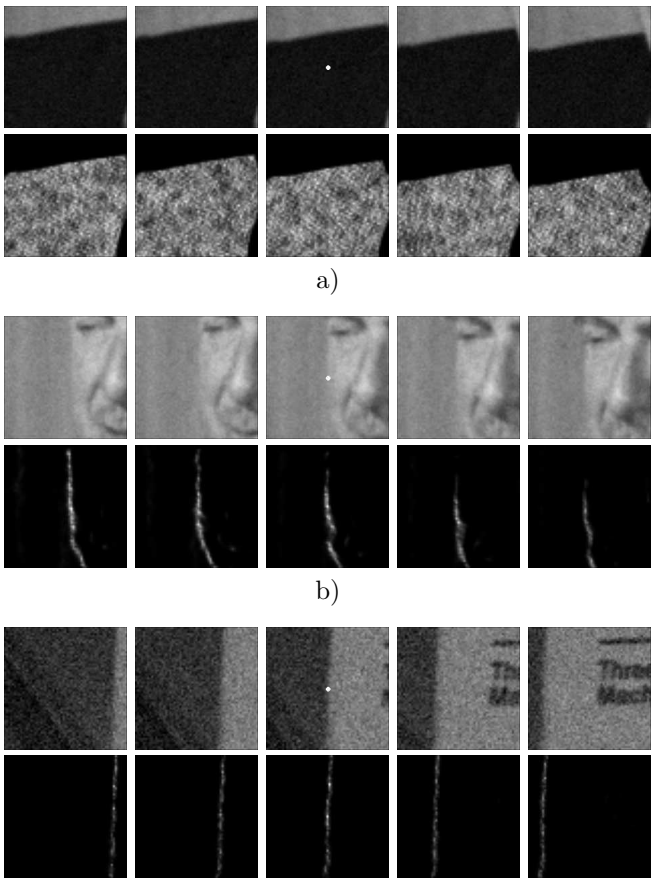

Figure 18: Weight distribution of NL-means applied to a movie. In a), b) and c) the first row shows a five frames image sequence. In the second row, the weight distribution used to estimate the central pixel (in white) of the middle frame is shown. The weights are equally distributed over the successive frames, including the current one. They actually involve all the candidates for the motion estimation instead of picking just one per frame. The aperture problem can be taken advantage of for a better denoising performance by involving more pixels in the average.

where $m_i/\sqrt{2}\sigma$ follows a $N(0,1)$ and then $n\,d_n/2\sigma^2$ follows a χ-squared distribution with n degrees of freedom.

For a sake of simplicity in the mathematical calculations, we assume that the central point of the window is not used in the comparisons. In that case independent values are being averaged and the next theorem estimates the achieved noise reduction.

Theorem 1 *Assume that the $n(i)$ are i.i.d. with zero mean and variance σ^2. Then, the filtered noise by the NL-means algorithm $NL_{h,n}$ satisfies,*

$$Var\ NL_{h,n}\, n(i) = \frac{1}{\beta_n(\frac{h}{2\sigma^2})|R(i)|}\,\sigma^2,$$

where

$$\beta_n(x) = \int_{-\infty}^{n\,x} f_{\chi^2_n}(y)\,dy,$$

and $f_{\chi^2_n}$ denotes the probability distribution function of a χ^2_n.

This result shows that we can increase the noise reduction by increasing the size of the sample and keeping the same value of h as illustrated in Figure 19. We can for example set a conservative threshold $h = 2\sigma^2$ and have a noise reduction tending to $\frac{2}{|R(i)|}$ as n increases.

5.2 A research zone containing the edge

In this case, the risk is to average pixels with an original value different from the pixel of interest. Therefore, we want to separate as much as possible the distances from the correct pixels from the distances of the incorrect ones. Following the same notations of the above section and denoting by c the contrast between the two different regions of the image, we can write the distance between two windows as

$$\frac{d_n}{2\sigma^2} = \frac{1}{n}\left\{\sum_{i=0}^{W}\left(\frac{m_i}{\sqrt{2}\sigma}\right)^2 + \sum_{j=W+1}^{n}\left(\frac{n_i}{\sqrt{2}\sigma}\right)^2\right\},$$

where $m_i/\sqrt{2}\sigma$ and $n_i/\sqrt{2}\sigma$ follow a $N(c/\sqrt{2}\sigma,1)$ and $N(0,1)$ distributions, respectively, and W denotes the number of pixels with a different grey level value inside the comparison window. The distribution of the above distances follows a non central chi squared distribution of parameters n and $\lambda = W * (\sqrt{2}\sigma)^2$.

The above normalized distance has mean $1 + (W/n)\lambda^2$ and variance $\frac{2}{n}(1 + 2\lambda^2(W/n))$. Therefore, the distance between two windows completely contained in different sides of the edge has a mean $1 + \lambda^2$ and a variance $\frac{2}{n}(1 + 2\lambda^2)$. The mean is independent of n while the variance tends to zero as n increases. Thus, windows from the same side of the edge can be better separated from the incorrect ones as n increases. The same argument applies when the ratio W/n is kept constant while n increases. For the neighborhood filter ($n = 1$), the above statistics shows that the comparison of one single pixel is not robust enough. Indeed, the standard deviation of the distance of two pixels on different sides of the edge is larger than the distance between the mean distances of the correct and incorrect choices.

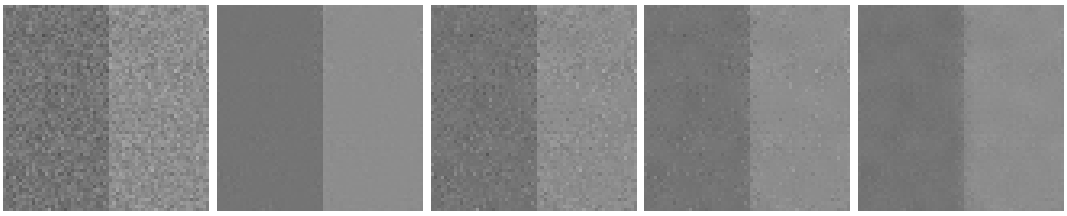

Figure 19: Comparison of neighborhood filters and NL-means on a piecewise constant image. From left to right: input image, NL-means filtered image and the neighborhood filter with an increasing value of h. Using the whole image as research zone, NL-means is able to recover the original image almost perfectly with $h = 2\sigma^2$. For the neighborhood filter we have used a research zone of 7×7 pixels. For h small the neighborhood filter is not able to reduce much noise and by increasing the value of h we begin to filter excessively the edge. In contrast with the neighborhood filter, NL-means is able to reduce the noise with a very conservative value of the filtering parameter.

6 Conclusion

A common framework for the study of the neighborhood filters has been presented. We have proposed three principles for the comparison of denoising methods evaluating the loss of image structure, the creation of artifacts, and the complete usage of image self-similarity. After a structural comparison of non-local denoising methods with other classes, we have shown that movie denoising can avoid the explicit computation of an optical flow estimate. What is left to be done? The very recent extensions of NL-means we mentioned in the introduction point towards the involvement of still more sophisticated statistical instruments. Beyond denoising, we have seen that the association with each pixel of a probability distribution weighting its similarity with the other pixels of the image may become a central tool in image analysis. This rich structure unfolds image information and should be used for algorithm which simultaneously analyze and process images. The recent work of Gilboa and Osher [17] points towards this direction. These authors use NL-means as a segmentation tool.

References

[1] L. Alvarez, F. Guichard, P.L. Lions and J.M. Morel, "Axioms and fundamental equations of image processing". Arch. Ration. Mech. Analysis, vol. 123, 199-257, 1993.

[2] F. Attneave, "Some informational aspects of visual perception", Psych. Rev., Vol. 61, pp. 183-193, 1954.

[3] J.-F. Aujol, G. Aubert, L. Blanc-Feraud and A. Chambolle, "Image decomposition into a bounded variation component and an oscillating component", Journal of Mathematical Imaging and Vision, vol.22(1), pp. 71-88, 2005.

[4] S.P. Awate and Whitaker, R. T., "Higher-Order Image Statistics for Unsupervised, Information-Theoretic, Adaptive, Image Filtering," Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, 2, 2005.

[5] N. Azzabou, N. Paragios and F. Guichard, "Random Walks, Constrained Multiple Hypothesis Testing and Image Enhancement", ECCV (1) : 379-390, 2006.

[6] J. C. Brailean, R. P. Kleihorst, S. Efsratiadis, A. K. Katsaggelos, and R. L. Lagendijk, "Noise reduction filters for dynamic image sequences: a review", Proceedings of the IEEE, vol. 83, pp. 1272-1292, 1995.

[7] A. Buades, B. Coll and J.M. Morel, "A review of image denoising methods, with a new one", Multiscale Modeling and Simulation, vol 4 (2), pp 490-530, 2005.

[8] A. Buades, B. Coll and J.M. Morel, "A non-local algorithm for image denoising", IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2005.

[9] A. Buades, B. Coll and J.M. Morel, "Denoising image sequences does not require motion estimation," preprint, CMLA, http://www.cmla.ens-cachan.fr/Cmla/, N 2005-18, May 2005.

[10] M. Colleen Gino, "Noise, Noise, Noise", *http : //www.astrophys − assist.com/educate/noise/noise.htm.*

[11] D. Cremers and L. Grady, "Statistical priors for efficient combinatorial optimization via graph cuts", European Conference on Computer vision, 2006.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," submitted to IEEE Transactions on Image Processing, December 2006.

[13] S. B. Howell, "Handbook of CCD Astronomy", Cambridge University Press, 2000.

[14] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage", Biometrika, 81 pp. 425-455, 1994.

[15] A. Efros and T. Leung, "Texture synthesis by non parametric sampling," Proc. Int. Conf. Computer Vision (ICCV 99), vol. 2, pp. 1033-1038, 1999.

[16] E. P. Bennett and L. McMillan, "Video Enhancement Using Per-Pixel Virtual Exposures" SIGGRAPH 2005.

[17] G. Gilboa and S. Osher, "Nonlocal Linear Image Regularization and Supervised Segmentation", UCLA CAM Report 06-47, Sept. 2006.

[18] G. Gilboa, J. Darbon, S. Osher and T.F. Chan, "Nonlocal Convex Functionals for Image Regularization", UCLA CAM Report 06-57, Oct. 2006.

[19] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", 2nd edition, Prentice Hall, 2002.

[20] B. Horn and B. Schunck, "Determining optical flow," Artif. Intell., Vol. 17, pp. 185-203, 1981.

[21] T. Huang, "Image Sequence Analysis", Springer-Verlag, 1981.

[22] S.L. Keeling and R. Stollberger, "Nonlinear anisotropic diffusion filtering for multiscale edge enhancement", Inverse Problems, vol 18, pp: 175-190, 2002.

[23] C. Kervrann, J. Boulanger, "Unsupervised patch-based image regularization and representation", Proc. Eur. Conf. Computer Vision (ECCV'06), Graz, Austria, May 2006.

[24] J. Boulanger, C. Kervrann and P. Bouthemy, "Adaptive space-time patch-based method for image sequence restoration." Proc. ECCV'06 Workshop on Statistical Methods in Multi-Image and Video Processing (SMVP'06), Graz, Austria, May 2006.

[25] Kindermann S, Osher S and Jones PW, "Deblurring and denoising of images by nonlocal functionals", Multiscale Modeling and simulation, 4 (4), pp. 1091-1115, 2005.

[26] A. C. Kokaram, "Motion Picture Restoration", PhD thesis, Cambridge University, 1993.

[27] J.S. Lee, "Digital image enhancement and noise filtering by use of local statistics", IEEE Trans. Patt. Anal. Machine Intell., vol. 2, pp. 165-168, 1980.

[28] J.S. Lee, "Digital image smoothing and the sigma filter", Computer Vision, Graphics and Image Processing, vol. 24, pp. 255-269, 1983.

[29] C. Liu, W.T. Freeman, R. Szeliski and S.B. Kang, "Noise estimation from a single image", CVPR 2006.

[30] M. Mahmoudi and G. Sapiro, "Fast Image and Video Denoising via Non-Local Means of Similar Neighborhoods", IEEE Signal Processing Letters, vol. 12 (12), 2005.

[31] D. M. Martinez, "Model-based motion estimation and its application to restoration and interpolation of motion pictures", PhD thesis, Massachusetts Institute of Technology, 1986.

[32] B. Merriman, J. Bence, and S. Osher, "Diffusion generated motion by mean curvature", In Proc. of the Geometry Center Workshop, 1992.

[33] T. Le and L. Vese, "Image decomposition using total variation and div(BMO)", Multiscale Modeling and Simulation, vol. 4, pp. 390-423, 2005.

[34] N. Papenberg, A. Bruhn, T. Brox, S. Didas and J. Weickert, "Highly accurate optic flow computation with theoretically justified warping", International Journal of Computer Vision, Vol. 67, No. 2, pp. 141-158, April 2006.

[35] Y. Meyer, "Oscillating Patterns in Image Processing and Nonlinear Evolution Equations", AMS University Lecture Series, vol. 22, 2002.

[36] H.H. Nagel, "Constraints for the estimation of displacement vector fields from image sequences," Proc. Eighth Int. Joint Conf. on Artificial Intelligence (IJCAI '83), pp. 945-951, 1983.

[37] M.K. Ozkan, M.I. Sezan, and A.M. Tekalp, "Adaptive motion compensated filtering of noisy image sequences", IEEE Trans. Circuits and Systems for Video Technology, vol. 3, pp. 277-290, 1993.

[38] S. Osher, M. Burger, D. Goldfarb, J. Xu and W. Yin, "An iterative regularization method for total variation based image restoration", Multiscale Modelling and Simulation, vol. 4, 460-489, 2005.

[39] L. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms", Physica D, 60, pp. 259-268, 1992.

[40] R. Samy, "An adaptive image sequence filtering scheme based on motion detection", SPIE, Vol. 596, pp. 135-144, 1985.

[41] M.I. Sezan, M.K. Ozkan and S.V. Fogel, "Temporally adaptive filtering of noisy sequences using a robust motion estimation algorithm", Proceedings of the Int. Conf. Acoustics, Speech, Signal Processing 91, pp. 2429-2432, 1991.

[42] S.M. Smith and J.M. Brady, "Susan - a new approach to low level image processing", International Journal of Computer Vision, Volume 23 (1), pp. 45-78, 1997.

[43] E. Tadmor, S. Nezzar, and L. Vese, "A multiscale image representation using hierarchical (BV, L^2) decompositions", Multiscale Modeling and Simulation, vol 2, pp.554–579, 2004.

[44] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," Sixth International Conference on Computer Vision, pp. 839-46. 1998.

[45] J. Tukey, "Exploratory Data Analysis", Addison-Wesley, 1977.

[46] J. Weickert, "On discontinuity-preserving optic flow," Proc. Computer Vision and Mobile Robotics Workshop, pp. 115-122, 1998.

[47] J. Weickert and C. Schnor, "Variational optic flow computation with a spatio-temporal smoothness constraint", Journal of Mathematical Imaging and Vision, vol. 14, pp. 245-255, 2001.

[48] L.P. Yaroslavsky, "Digital Picture Processing - An Introduction", Springer Verlag, 1985.