



ELSEVIER

Signal Processing 76 (1999) 147–165

**SIGNAL  
PROCESSING**

www.elsevier.nl/locate/sigpro

# Multiscale entropy filtering

J.-L. Starck<sup>a,\*</sup>, F. Murtagh<sup>b</sup>

<sup>a</sup>*DAPNIA/SEI-SAP, CEA/Saclay, Orme des Merisiers, F-91191 Gif sur Yvette Cedex, France*

<sup>b</sup>*Faculty of Informatics, University of Ulster, Magee College, Londonderry BT48 7JL, Ireland*

Received 2 April 1998; received in revised form 23 September 1998

---

## Abstract

We present in this paper a new method for filtering an image, based on a new definition of its entropy. A large number of examples illustrate the results. Comparisons are performed with other wavelet-based methods. © 1999 Elsevier Science B.V. All rights reserved.

## Zusammenfassung

Wir stellen in dieser Arbeit eine neue Methode zur Filterung von Bildern vor, die auf einer neuen Definition seiner Entropie beruht. Eine große Anzahl von Beispielen illustriert die Ergebnisse. Vergleiche zu anderen wavelet-basierten Methoden werden angestellt. © 1999 Elsevier Science B.V. All rights reserved.

## Résumé

Nous présentons dans cet article une méthode nouvelle pour le filtrage des images, basée sur une définition nouvelle de son entropie. Un grand nombre d'exemples illustrent les résultats. Des comparaisons sont effectuées avec d'autres méthodes basées sur les ondelettes. © 1999 Elsevier Science B.V. All rights reserved.

*Keywords:* Filtering; Image processing; Entropy

---

## 1. Introduction

The wavelet transform (WT) has been widely used in recent times and furnishes a new approach for describing and modeling the data. Using wavelets, a signal can be decomposed into components of different scales. There are many 2D WT algorithms [35]. The most well-known are perhaps the orthogonal wavelet transform proposed by

Mallat [23], and its bi-orthogonal version [11]. These methods are based on the principle of reducing the redundancy of the information in the transformed data. Other WT algorithms exist, such as the Feauveau algorithm [17] (which is an orthogonal transform, but using an isotropic wavelet), or the à trous algorithm which is non-orthogonal and furnishes a very redundant dataset [20]. All these methods have advantages and drawbacks. Following the content of the data, and the nature of the noise, each of these models can be considered as optimal.

Once the vision model is chosen, the second fundamental point is to estimate the noise behavior

---

\*Corresponding author. Tel.: +33 1 6908 5764; fax: +33 1 6908 6577; e-mail: jstarck@cea.fr

in the transformed data. Linear transforms have in this case the advantage of allowing robust estimation of noise variance. But again, different strategies can be employed, which include soft or hard thresholding [16,14], and in these latter cases threshold level estimation [32].

We review in the second section the algorithms which can be used for a multiresolution decomposition (we call these vision models in the sequel), and which strategies can be used for treating the noise, once the data have been transformed. Then we introduce in Section 3 the Multiscale Entropy Filtering method (MEF), and present a large number of examples. Results of a set of simulations are presented and discussed in Section 4 in order to compare the MEF method to other standard wavelet-based methods.

## 2. Multiresolution and filtering

This section reviews different strategies available for wavelet coefficient filtering. A range of important and widely used transform and filtering approaches are used.

### 2.1. The choice of the multiresolution transform

#### 2.1.1. The (bi-)orthogonal wavelet transform

This wavelet transform [23], often referred to as the Fast Wavelet Transform (FWT), is certainly the most widely used among available discrete wavelet transform algorithms. It is a non-redundant representation of the information. An introduction to this type of transform can be found in [38,13].

A large class of orthogonal wavelet functions are available.

#### 2.1.2. The Feauveau wavelet transform

Feauveau [17] introduced quincunx analysis based on Adelson's work [2]. This analysis is not dyadic and allows an image decomposition with a resolution factor equal to  $\sqrt{2}$ . By this method, we have only one wavelet image at each scale, and not three as in the previous method.

#### 2.1.3. The à trous algorithm [20]

The wavelet transform of an image by this algorithm produces, at each scale  $j$ , a set  $\{w_j\}$ . This has the same number of pixels as the image. Furthermore, using a wavelet defined as the difference between the scaling functions of two successive scales ( $\frac{1}{2}\psi(x/2) = \phi(x) - \phi(x/2)$ ), the original image  $c_0$  can be expressed as the sum of all the wavelet scales and the smoothed array  $c_p$ ,

$$c_0 = c_p + \sum_{j=1}^p w_j, \quad (1)$$

and a pixel at position  $x,y$  can be expressed also as the sum of all the wavelet coefficients at this position, plus the smoothed array:

$$c_0(x,y) = c_p(x,y) + \sum_{j=1}^p w_j(x,y). \quad (2)$$

#### 2.1.4. The multiresolution median transform

The median transform is non-linear, and offers advantages for robust smoothing (i.e. the effects of outlier pixel values are mitigated). The multiresolution median transform [37] (which is not a wavelet transform) consists of a series ( $c_1, \dots, c_p$ ) of smoothings of the input image, with successively broader kernels. Each resolution scale  $w_j$  is constructed from differencing two successive smoothed images ( $w_j = c_{j-1} - c_j$ ). For integer input image values, this transform can be carried out in integer arithmetic only which may lead to computational savings. As in the case of the à trous algorithm, the original image can be expressed by a sum of the scales and the smoothed array.

### 2.2. Non-Gaussian noise

If the noise in the data  $I$  is Poisson, the transformation [4]

$$t(I) = 2\sqrt{I + \frac{3}{8}} \quad (3)$$

acts as if the data arose from a Gaussian white noise model, with  $\sigma = 1$ , under the assumption that the mean value of  $I$  is sufficiently large. The arrival of photons, and their expression by electron counts, on CCD detectors may be modeled by a Poisson

distribution. In addition, there is additive Gaussian read-out noise. The Anscombe transformation has been extended to take this combined noise into account. The generalization of the variance stabilizing Anscombe formula is derived as [25]

$$t(I) = \frac{2}{g} \sqrt{gI + \frac{3}{8}g^2 + \sigma^2 - gm}, \quad (4)$$

where  $g$  is the electronic gain of the detector,  $\sigma$  and  $m$  the standard deviation and the mean of the read-out noise.

This implies that for the filtering of an image with Poisson noise or a mixture of Poisson and Gaussian noise, we will first pre-transform the image  $I$  into another one  $t(I)$  with Gaussian noise. Then  $t(I)$  will be filtered, and the filtered image will be inverse-transformed.

For other kinds of noise, modeling must be performed in order to define the noise probability distribution of the wavelet coefficients [35]. In the following, we will consider only stationary Gaussian noise.

### 2.3. Filtering in the wavelet space

We review in this section some important strategies for treating the noise, once the data have been transformed.

#### 2.3.1. Hard thresholding

This consists of setting to 0 all wavelet coefficients which have an absolute value lower than a threshold  $T_j$  ( $T_j = K\sigma_j$ , where  $j$  is the scale of the wavelet coefficient,  $\sigma_j$  is the noise standard deviation at the scale  $j$ , and  $K$  is a constant generally chosen equal to 3). For an energy-normalized wavelet transform algorithm, we have  $\sigma_j = \sigma$  for all  $j$ .

The appropriate value of  $\sigma_j$  in the succession of wavelet scales is assessed from the standard deviation of the noise  $\sigma$  in the original signal and from study of the noise in the wavelet space. This study consists of simulating a signal containing Gaussian noise with a standard deviation equal to 1, and taking the wavelet transform of this signal. Then we compute the standard deviation  $\sigma_j^e$  at each scale. We get a curve  $\sigma_j^e$  as a function of  $j$ , giving the

behavior of the noise in the wavelet space. Due to the properties of the wavelet transform, we have  $\sigma_j = \sigma\sigma_j^e$  (see [34] for a description of how  $\sigma$  can be automatically calculated directly from the data).

#### 2.3.2. Soft thresholding

Soft thresholding consists of replacing each wavelet coefficient  $w_{j,k}$  ( $j$  being the scale index, and  $k$  the position index) by the value  $\tilde{w}_{j,k}$  where

$$\tilde{w}_{j,k} = \begin{cases} \text{sgn}(w_{j,k})(|w_{j,k}| - T_j) & \text{if } |w_{j,k}| \geq T_j, \\ 0 & \text{if } |w_{j,k}| < T_j. \end{cases} \quad (5)$$

#### 2.3.3. Donoho universal approach

Donoho [16,14] has suggested to take  $T_j = \sqrt{2\log(n)}\sigma_j$  (where  $n$  is the number of pixels) instead of the standard  $K\sigma$  value. This leads to a new soft and hard thresholding approach.

Other threshold-based approaches are available. SURE, Stein unbiased risk estimator [15,7] is adaptive in that it is resolution dependent. The SURE estimator can break down when the wavelet coefficients are mostly around zero. In contrast, the Donoho *universal* hard and soft thresholding approach may overly smooth the data, which is potentially rectified by the minimax criterion proposed in [16]. Note also that Chipman et al. [10] found that SURE create high frequency artifacts.

#### 2.3.4. Multiresolution Wiener filtering

Multiresolution Wiener filtering [32] consists of multiplying all coefficients  $w_{j,k}$  of a given scale  $j$  by

$$\alpha_j = S_j/(S_j + N_j), \quad (7)$$

where  $S_j$  and  $N_j$  are, respectively, the variance of the signal and of the noise at the scale  $j$  ( $N_j = \sigma_j^2$ ). In the absence of any information about the signal, we take  $S_j$  equal to the difference between the variance of the data  $w_j$  and the variance of the noise  $N_j$ .

#### 2.3.5. Hierarchical Wiener filtering

Hierarchical Wiener filtering [32] tries to introduce a prediction  $w_{j,k}^h$  into the estimation of  $\tilde{w}_{j,k}$ .

$$\tilde{w}_{j,k} = \frac{H_j}{N_j + H_j + Q_j} w_{j,k} + \frac{N_j}{N_j + H_j + Q_j} w_{j,k}^h, \quad (8)$$

with

$$Q_j = H_j N_j / S_j, \quad (9)$$

where  $H_j$  is the variance of the image  $D$  obtained by taking the difference of the scale  $j$  and the following one  $j+1$  ( $D = w_j - w_{j+1}$ , and  $H_j = (1/N) \sum_k (D_k - m_D)^2$ , where  $N$  is the number of pixels and  $m_D$  the mean of  $D$ ). If a pyramidal transform is used, the scale  $w_{j+1}$  must be first interpolated to the size of the scale of  $w_j$ .

This prediction  $w_{j,k}^h$  is obtained from the coefficient at the same position but at the following scale. In the case of the à trous algorithm  $w_{j,k}^h = w_{j+1,k}$ , while for a pyramidal transform,  $w_{j,k}^h = w_{j+1,k/2}$ .

### 2.3.6. Hierarchical hard thresholding

The threshold used here,  $T_h$  [32], is equal to  $T_j = K\sigma_j$  if  $|w_{j,k}| \geq T_j$ , and  $T_h = T_j f(|w_{j,k}|/\sigma_{j+1})$  otherwise. The function  $f(a)$  must return a value between 0 and 1. A possible function for  $f$  is

$$f(a) = \begin{cases} 0 & \text{if } a \geq k, \\ 1 - \frac{1}{K}a & \text{if } a < k. \end{cases}$$

If the predicted wavelet coefficient has a high signal-to-noise ratio (SNR) (this means that there is certainly some information at this position), the threshold level becomes null, and the wavelet coefficient will not be thresholded, even if its value is small. The threshold level becomes adaptive.

## 3. Multiscale entropy filtering

### 3.1. Multiscale entropy definition

The term ‘entropy’ is due to Clausius (1865), and the concept of entropy was introduced by Boltzmann into statistical mechanics, in order to measure the number of microscopic ways that a given macroscopic state can be realized. Shannon [30] founded the mathematical theory of communication when he suggested that the information gained in a measurement depends on the number of possible outcomes out of which one is realized. Shannon also suggested that the entropy can be used for maximization of the bits transferred under

a quality constraint. Jaynes [22] proposed to use the entropy measure for radio interferometric image deconvolution, in order to select between a set of possible solutions that which contains the minimum of information, or following his entropy definition, that which has maximum entropy. In principle, the solution satisfying such a condition should be the most reliable. A lot of work has been done in the last 30 years on the use of entropy for the general problem of data filtering and deconvolution [1,6,8,18,19,26,29,31,39]. The main entropy functions used are:

- Burg [8]:

$$H_b(X) = - \sum_{k=1}^N \ln(X_k), \quad (10)$$

- Freiden [18]:

$$H_f(X) = - \sum_{k=1}^N X_k \ln(X_k), \quad (11)$$

- Gull and Skilling [19]:

$$H_g(X) = \sum_{k=1}^N X_k - M_k - X_k \ln\left(\frac{X_k}{M_k}\right). \quad (12)$$

Each of these entropies can be used in practice, and they correspond to different probability distributions that one can associate with an image [26]. (See [18,31] for descriptions.) The last definition of the entropy has the advantage of having zero maximum when  $X$  equals the model  $M$ , usually taken as a flat image. However, as discussed in [26,6,36], all of these definitions present drawbacks.

The different entropy functions (such as those described here) which have been proposed for image restoration have the property of being maximal when the image is flat, and of decreasing when we introduce some information. So minimizing the information is equivalent to maximizing the entropy, and this has led to the well-known Maximum Entropy Method (MEM). For the Shannon entropy (which is obtained from the histogram of the data), this is the opposite. The entropy is null for a flat image, and increases when the data contains some information. So, if the Shannon entropy were used for restoration, this would lead to a Minimum Entropy Method.

A discussion was raised in [36] about what should be a good entropy measurement for signal restoration, and we proposed that the following criteria should be satisfied:

1. The information in a flat signal is zero.
2. The amount of information in a signal is independent of the background.
3. The amount of information is dependent on the noise. A given signal  $Y$  ( $Y = X + \text{Noise}$ ) does not furnish the same information if the noise is high or small.
4. The entropy must work in the same way for a pixel which has a value  $B + \varepsilon$  ( $B$  being the background), and for a pixel which has a value  $B - \varepsilon$ .
5. The amount of information is dependent on the correlation in the signal. If a signal  $S$  presents large features above the noise, it contains a lot of information. By generating a new set of data from  $S$ , by randomly taking the pixel values in  $S$ , the large features will evidently disappear, and this new signal will contain less information. But the pixel values will be the same as in  $S$ .

It is clear that among all entropy functions proposed in the past, it is the Shannon one which best respects these criteria. Indeed, if we assume that the bin of the histogram is defined as a function of the standard deviation of the noise, the first four points are satisfied, while none of these criteria are satisfied with other entropy functions (and only one point is satisfied for Gull and Skilling entropy by taking the model equal to the background).

Following on from these criteria, a possibility is to consider that the entropy of a signal is the sum of the information of each scale of its wavelet transform [36], and the information of a wavelet coefficient is related to the probability of it being due to noise. Noting  $h$  the information relative to a single wavelet coefficient, we have

$$H(X) = \sum_{j=1}^l \sum_{k=1}^{N_j} h(w_{j,k}), \tag{13}$$

with  $h(w_{j,k}) = -\ln p(w_{j,k})$ .  $l$  is the number of scales, and  $N_j$  is the number of samples in the band  $j$  ( $N_j = N$  for the à trous algorithm). For Gaussian noise, we get

$$h(w_{j,k}) = w_{j,k}^2 / 2\sigma_j^2, \tag{14}$$

where  $\sigma_j$  is the noise at scale  $j$ . We see that the information is proportional to the energy of the wavelet coefficients. The higher a wavelet coefficient, the lower will be the probability, and the higher will be the information furnished by this wavelet coefficient. We can see easily that this entropy fulfills all our requirements. As for the Shannon entropy, the information increases with the entropy, and using such an entropy leads to a Minimum Entropy Method.

Since the data is composed of an original signal and noise, our information measure is corrupted by noise, and we decompose our information measure into two components, one ( $H_S$ ) corresponding to the non-corrupted part, and the other ( $H_N$ ) to the corrupted part. We have [36]

$$H(X) = H_S(X) + H_N(X). \tag{15}$$

We will define in the following  $H_S$  as the signal information, and  $H_N$  as the noise information. It must be clear that noise does not contain any information, and what we call ‘noise information’ is a quantity which is measured as information by the multiscale entropy, and which is probably not informative to us.

If a wavelet coefficient is small, its value can be due to noise, and the information  $h$  relative to this single wavelet coefficient should be assigned to  $H_N$ . If the wavelet coefficient is high, compared to the noise standard deviation, its value cannot be due to the noise, and  $h$  should be assigned to  $H_S$ .  $h$  can be distributed as  $H_N$  or  $H_S$  based on the probability  $p_n(w_{j,k})$  that the wavelet coefficient is due to noise, or the probability  $p_s(w_{j,k})$  that it is due to signal. We have  $p_s(w_{j,k}) = 1 - p_n(w_{j,k})$ . For the Gaussian noise case, we estimate  $p_n(w_{j,k})$  that a wavelet coefficient is due to the noise by

$$\begin{aligned} p_n(w_{j,k}) &= \text{Prob}(W > |w_{j,k}|) \\ &= \frac{2}{\sqrt{2\pi}\sigma_j} \int_{|w_{j,k}|}^{+\infty} \exp(-W^2/2\sigma_j^2) dW \\ &= \text{erfc}\left(\frac{|w_{j,k}|}{\sqrt{2}\sigma_j}\right). \end{aligned} \tag{16}$$

For each wavelet coefficient  $w_{j,k}$ , we have to estimate now the fractions  $h_n$  and  $h_s$  of  $h$  which should

be assigned to  $H_n$  and  $H_s$ . Hence signal information and noise information are defined by

$$H_s(X) = \sum_{j=1}^l \sum_{k=1}^{N_j} h_s(w_{j,k}), \quad (17)$$

$$H_n(X) = \sum_{j=1}^l \sum_{k=1}^{N_j} h_n(w_{j,k}).$$

The idea for deriving  $h_s$  and  $h_n$  is the following: we imagine that the information  $h$  relative to a wavelet coefficient is a sum of small information components  $dh$ , each of them having a probability to be noise information, or signal information. Hence,  $h_n$  and  $h_s$  are calculated by

$$h_n(w_{j,k}) = \int_0^{|w_{j,k}|} p_n(|w_{j,k}| - u) \left( \frac{\partial h(x)}{\partial x} \right)_{x=u} du \quad (18)$$

is the noise information relative to a single wavelet coefficient, and

$$h_s(w_{j,k}) = \int_0^{|w_{j,k}|} p_s(|w_{j,k}| - u) \left( \frac{\partial h(x)}{\partial x} \right)_{x=u} du \quad (19)$$

is the signal information relative to a single wavelet coefficient. For Gaussian noise, we have

$$h_n(w_{j,k}) = \frac{1}{\sigma_j^2} \int_0^{|w_{j,k}|} u \operatorname{erfc} \left( \frac{|w_{j,k}| - u}{\sqrt{2}\sigma_j} \right) du, \quad (20)$$

$$h_s(w_{j,k}) = \frac{1}{\sigma_j^2} \int_0^{|w_{j,k}|} u \operatorname{erf} \left( \frac{|w_{j,k}| - u}{\sqrt{2}\sigma_j} \right) du. \quad (21)$$

### 3.2. Filtering

The problem of filtering or restoring data  $D$  can be expressed by the following: We search for a solution  $\tilde{D}$  such that the difference between  $D$  and  $\tilde{D}$  minimizes the information due to the signal, and such that  $\tilde{D}$  minimizes the information due to the noise.

$$J(\tilde{D}) = H_s(D - \tilde{D}) + H_n(\tilde{D}). \quad (22)$$

Furthermore, the smoothness of the solution can be controlled by adding a parameter:

$$J(\tilde{D}) = H_s(D - \tilde{D}) + \alpha H_n(\tilde{D}). \quad (23)$$

In practice [9], we minimize for each wavelet coefficient  $w_{j,k}$ :

$$j(\tilde{w}_{j,k}) = h_s(w_{j,k} - \tilde{w}_{j,k}) + \alpha h_n(\tilde{w}_{j,k}). \quad (24)$$

$j(\tilde{w}_{j,k})$  can be obtained by any minimization routine. In our examples, we have used a simple dichotomy.

Fig. 1 shows the result when minimizing the functional  $j$  with different  $\alpha$  values, and a noise standard deviation equal to 1. The corrected wavelet coefficient is plotted versus the wavelet coefficient. From the top curve to the bottom one,  $\alpha$  is, respectively, equal to 0, 0.1, 0.5, 1, 2, 5, 10. The higher the value of  $\alpha$ , the more the corrected wavelet coefficient is reduced. When  $\alpha$  is equal to 0, there is no regularization and the data are unchanged.

### 3.3. The regularization parameter

The  $\alpha$  parameter can be used in different ways:

- It can be fixed to a given value (user parameter):  $\alpha = \alpha_u$ . This method leads to a very fast filtering using the optimization proposed in the following.
- It can be calculated under the constraint that the residual should have some specific characteristic. For instance, in the case of Gaussian noise, we expect a residual with a standard deviation equal to the noise standard deviation. In this case,  $\alpha = \alpha_c \alpha_u$ . The parameter finally used is taken as the product of a user parameter (defaulted to 1) and the calculated value  $\alpha_c$ . This allows the user to keep open the possibility of introducing an under-smoothing, or an over-smoothing. It is clear that such an algorithm is iterative, and will always take more time than a simple hard thresholding approach.
- We can permit more constraints on  $\alpha$  by using the fact that we expect a residual with a given standard deviation at each scale  $j$  equal to the noise standard deviation  $\sigma_j$  at the same scale. Then rather than a single  $\alpha$  we have an  $\alpha_j$  per scale.

A more sophisticated way to fix the  $\alpha$  value is to introduce a distribution (or a priori knowledge) of how the regularization should work. For instance, in astronomical image restoration, the analyst generally prefers that the flux (total intensity)

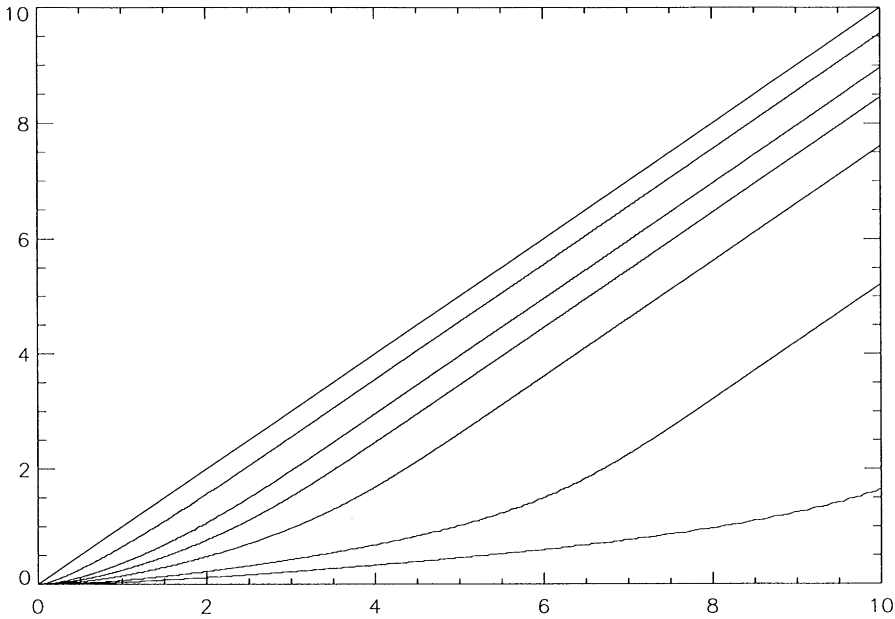


Fig. 1. Corrected wavelet coefficient versus the wavelet coefficient with different  $\alpha$  values (from the top curve to the bottom one,  $\alpha$  is respectively equal to 0, 0.1, 0.5, 1, 2, 5, 10).

contained in a star or in a galaxy is not modified by the restoration process. This means that the residual at positions of astronomical objects will approximately be equal to zero. All zero areas in the residual map obviously do not relate to realistic noise behavior, but from the user's point of view they are equally important. For the user, all visible objects in the filtered map contain the same flux as in the raw data. In order to obtain this kind of regularization, the  $\alpha$  parameter is no longer a constant value, but depends on the raw data. Hence we have one  $\alpha$  per wavelet coefficient, which will be denoted  $\alpha_s(w_{j,k})$ , and it can be derived by

$$\alpha_s(w_{j,k}) = \alpha_j \frac{1 - L(w_{j,k})}{L(w_{j,k})}, \quad (25)$$

with  $L(w_{j,k}) = \text{MIN}(1, |w_{j,k}|/k_s\sigma_j)$ , where  $k_s$  is a user parameter (typically defaulted to 3).

When  $L(w_{j,k})$  is close to 1,  $\alpha_s(w_{j,k})$  becomes equal to zero, and there is no regularization anymore, and the obvious solution is  $\tilde{w}_{j,k} = w_{j,k}$ . Hence, the wavelet coefficient is preserved from any regulariz-

ation. If  $L(w_{j,k})$  is close to 0,  $\alpha_s(w_{j,k})$  tends toward infinity, then the first term in Eq. (24) is negligible, and the solution will be  $\tilde{w}_{j,k} = 0$ . In practice, this means that all coefficients higher than  $k_s\sigma_j$  are untouched as in the hard thresholding approach. We also notice that by considering a distribution  $L(w_{j,k})$  equal to 0 or 1 (1 when  $|w| > K\sigma$  for instance), the solution is then the same as a hard thresholding solution.

### 3.4. The use of a model

Using a model in wavelet space has been successfully applied for denoising (see for example [10,12,21]). If we have a model  $D_m$  for the data, this can also naturally be inserted into the filtering equation

$$J_m(\tilde{D}) = H_s(D - \tilde{D}) + \alpha H_n(\tilde{D} - D_m) \quad (26)$$

or, for each wavelet coefficient  $w_{j,k}$ ,

$$j_m(\tilde{w}_{j,k}) = h_s(w_{j,k} - \tilde{w}_{j,k}) + \alpha h_n(\tilde{w}_{j,k} - w_{j,k}^m), \quad (27)$$

where  $w_{j,k}^m$  is the corresponding wavelet coefficient of  $D_m$ .

The model can be of quite different types. It can be an image, and in this case, the coefficients  $w_{j,k}^m$  are obtained by a simple wavelet transform of the model image. It can also be expressed by a distribution or a given function which furnishes a model wavelet coefficient  $w^m$  from the data. For instance, the case where we want to keep intact high wavelet coefficients (see Eq. (25)) can also be treated by the use of a model, just by calculating  $w_{j,k}^m$  by

$$w_{j,k}^m = p_s(w_{j,k})w_{j,k}, \tag{28}$$

when  $w_{j,k}$  has a high signal-to-noise ratio,  $P_s(w_{j,k})$  is close to 1, and  $w_{j,k}^m$  is equal to  $w_{j,k}$  (Fig. 2). Then  $\alpha h_n(\tilde{w}_{j,k} - w_{j,k}^m)$  is equal to zero and  $\tilde{w}_{j,k} = w_{j,k}$ , i.e. no regularization is done on  $w_{j,k}$ .

Other models may also be considered. When the image contains contours, it may be interesting to derive the model from the detected edge. Zero-crossing wavelet coefficients indicate where the edges are [24]. By averaging three wavelet coefficients in the direction of the detected edge, we get

a value  $w_a$ , from which we derive the SNR  $S_e$  of the edge ( $S_e = 0$  if there is no detected edge). The model value  $w^m$  is set to  $w_a$  if a contour is detected, and 0 otherwise. This approach has the advantage to filter the wavelet coefficient, and even if an edge is clearly detected the smoothing operates in the direction of the edge.

There is naturally no restriction on the model. When we have a priori information of the content of an image, we should use it in order to improve the quality of the filtering. It is clear that the way we use the knowledge of the presence of edges in an image is not a closed question. The model in the entropy function is an interesting direction to investigate in the future.

### 3.5. The multiscale entropy filtering algorithm

The Multiscale Entropy Filtering algorithm (MEF) consists of minimizing for each wavelet coefficient  $w_{j,k}$  at scale  $j$ ,

$$j_m(\tilde{w}_{j,k}) = h_s(w_{j,k} - \tilde{w}_{j,k}) + \alpha_j h_n(\tilde{w}_{j,k} - w_{j,k}^m) \tag{29}$$

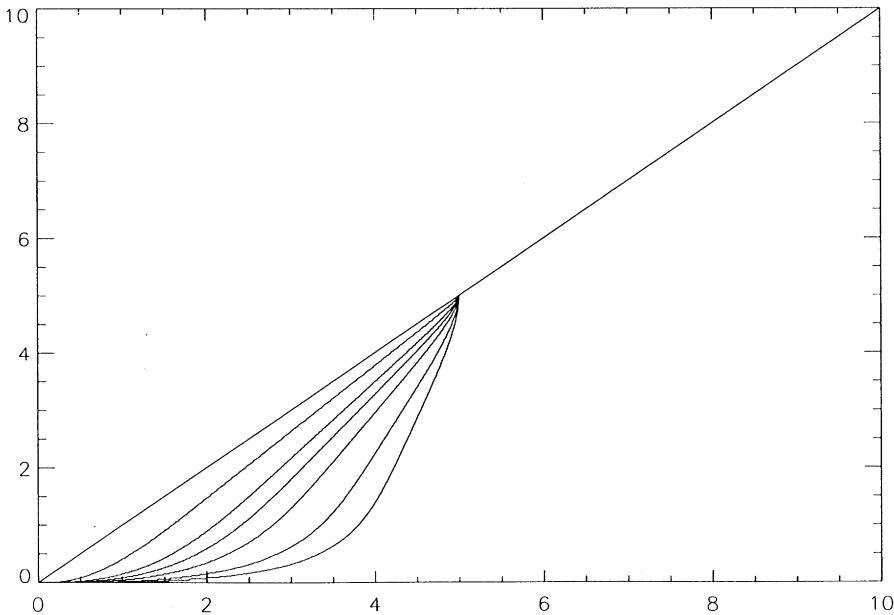


Fig. 2. Corrected wavelet coefficient versus the wavelet coefficient with different  $\alpha$  values.



or

$$j_{ms}(\tilde{w}_{j,k}) = h_s(w_{j,k} - \tilde{w}_{j,k}) + \alpha_j \alpha_s(w_{j,k}) h_n(\tilde{w}_{j,k} - w_{j,k}^m) \quad (30)$$

if the SNR is used. By default the model  $w_{j,k}^m$  is set to 0. There is no user parameter because the  $\alpha_j$  are calculated automatically in order to verify the noise properties. If an over-smoothing (or a under-smoothing) is desired, a user parameter must be introduced. We propose in this case to calculate the  $\alpha_j$  in the standard way, and then to multiply the calculated values by a user value  $\alpha_u$  defaulted to 1. Increasing  $\alpha_u$  will lead to an over-smoothing, while decreasing  $\alpha_u$  implies an under-smoothing.

Using a simple dichotomy, the algorithm becomes

1. Estimate the noise in the data  $\sigma$  (see [28,34]).
2. Wavelet transform of the data.
3. Calculate from  $\sigma$  the noise standard deviation  $\sigma_j$  at each scale  $j$ .
4. Set  $\alpha_j^{\min} = 0$ ,  $\alpha_j^{\max} = 200$ .
5. For each scale  $j$  do
  - 5.1. Set  $\alpha_j = (\alpha_j^{\min} + \alpha_j^{\max})/2$
  - 5.2. For each wavelet coefficient  $w_{j,k}$  of scale  $j$ , find  $\tilde{w}_{j,k}$  by minimizing  $j_m(\tilde{w}_{j,k})$  or  $j_{ms}(\tilde{w}_{j,k})$
  - 5.3. Calculate the standard deviation of the residual:  $\sigma_j^r = \sqrt{(1/N_j) \sum_{k=1}^{N_j} (w_{j,k} - \tilde{w}_{j,k})^2}$
  - 5.4. If  $\sigma_j^r > \sigma_j$  then the regularization is too strong, and we set  $\alpha_j^{\max}$  to  $\alpha_j$ , otherwise we set  $\alpha_j^{\min}$  to  $\alpha_j$  ( $\sigma_j$  is derived from the method described in Section 2.3).
6. If  $\alpha_j^{\max} - \alpha_j^{\min} > \varepsilon$  then go to 5.
7. Multiply all  $\alpha_j$  by the constant  $\alpha_u$ .
8. For each scale  $j$  and for each wavelet coefficient  $w$  find  $\tilde{w}_{j,k}$  by minimizing  $j_m(\tilde{w}_{j,k})$  or  $j_{ms}(\tilde{w}_{j,k})$ .
9. Reconstruct the filtered image from  $\tilde{w}_{j,k}$  by the inverse wavelet transform.

The minimization of  $j_m$  or  $j_{ms}$  (Step 5.2) can be done by any method. For instance, a simple dichotomy can be used in order to find  $\tilde{w}$  such that

$$\frac{\partial h_s(w - \tilde{w})}{\partial \tilde{w}} = -\alpha_j \frac{\partial h_n(\tilde{w})}{\partial \tilde{w}}. \quad (31)$$

The idea to treat the wavelet coefficients such that the residual respects some constraint has also been used in [27,3] using cross-validation.

### 3.6. Optimization

In the case of Gaussian noise, the calculation of erf and erfc functions could lead to a significant time computation, when compared to a simple filtering method. This can be easily avoided by precomputing tables, which is possible due to the specific properties of  $\partial h_s/\partial \tilde{w}$  and  $\partial h_n/\partial \tilde{w}$ .  $h_s$  and  $h_n$  are functions of the standard deviation of the noise, and we denote the reduced functions by  $h_s^r$  and  $h_n^r$ , i.e.  $h_s$  and  $h_n$  for noise standard deviation equal to 1. It is easy to verify that

$$\frac{\partial h_s(w_{j,k})}{\partial \tilde{w}} = \sigma_j \frac{\partial h_s^r\left(\frac{w_{j,k}}{\sigma_j}\right)}{\partial \tilde{w}}, \quad (32)$$

$$\frac{\partial h_n(w_{j,k})}{\partial \tilde{w}} = \sigma_j \frac{\partial h_n^r\left(\frac{w_{j,k}}{\sigma_j}\right)}{\partial \tilde{w}}. \quad (33)$$

Furthermore,  $\partial h_n^r/\partial \tilde{w}$  and  $\partial h_s^r/\partial \tilde{w}$  are symmetric functions,  $\partial h_n^r/\partial \tilde{w}$  converges to a constant value  $C$  ( $C = 0.798$ ), and  $\partial h_s^r/\partial \tilde{w}$  tends to  $C - w$  when  $w$  is large enough ( $> 5$ ). In our implementation, we precomputed the tables using a step-size of 0.01 from 0 to 5. If no model is introduced and if the SNR is not used, the filtered wavelet of coefficients is a function of  $\alpha$  and  $w_j/\sigma_j$ , and a second level of optimization can be performed by precomputed tables of solutions for different values of  $\alpha$ .

### 3.7. Examples

#### 3.7.1. 1D data filtering

Figs. 3–5 show the results of the multiscale entropy method on simulated data (2048 pixels). From top to bottom, each figure shows simulated data, the noisy data, the filtered data, and both noisy and filtered data overplotted. For the two first filterings, all default parameters were taken (noise standard deviation and  $\alpha_j$  automatically calculated,  $\alpha_u = 1$ , and the chosen wavelet transform algorithm is the à trous one). For the block signal (Fig. 5), default parameters were also used, but the multiresolution transform we used is the multiresolution median transform.

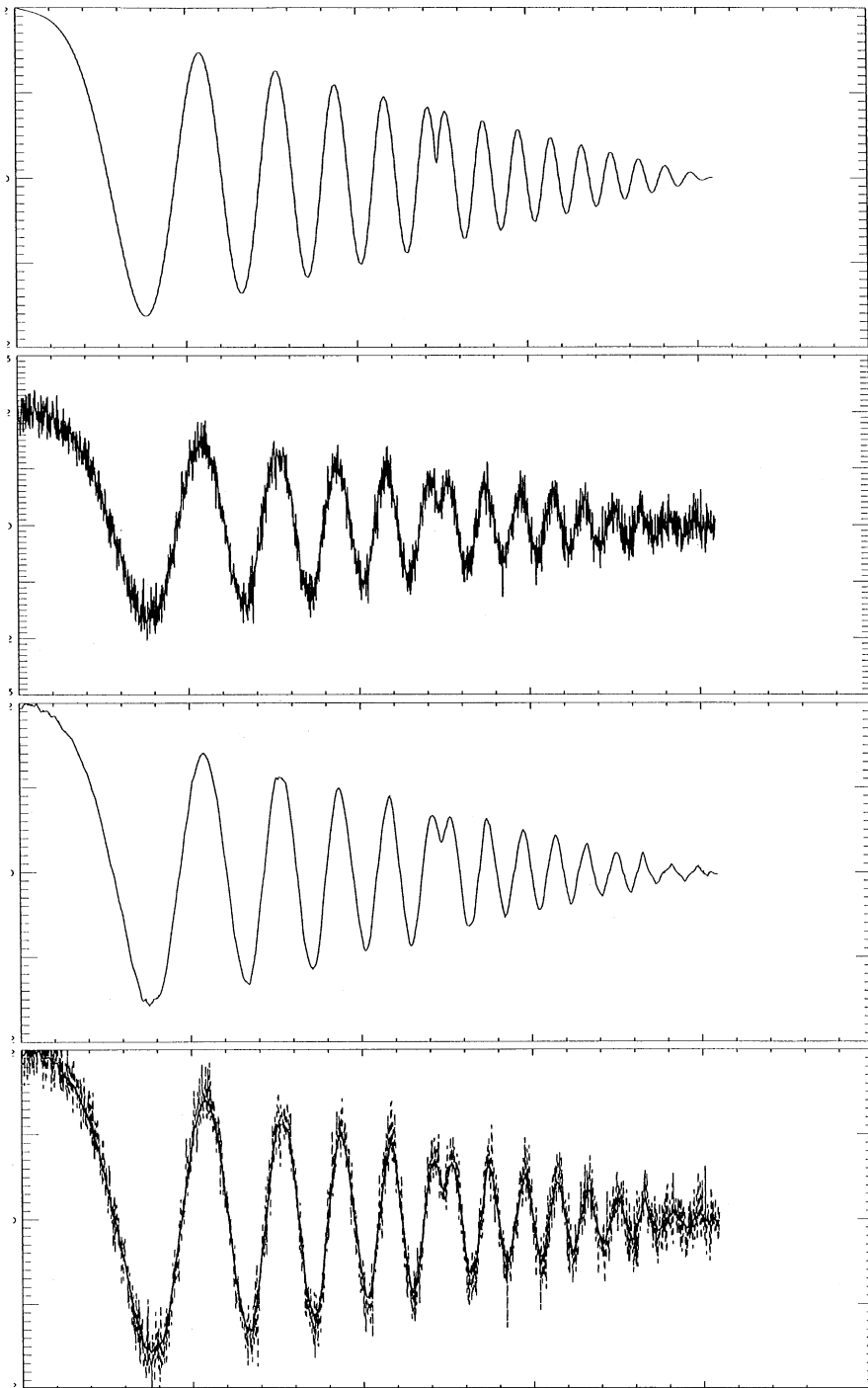


Fig. 3. From top to bottom, simulated data, noisy data, filtered data, and both noisy and filtered data overplotted.

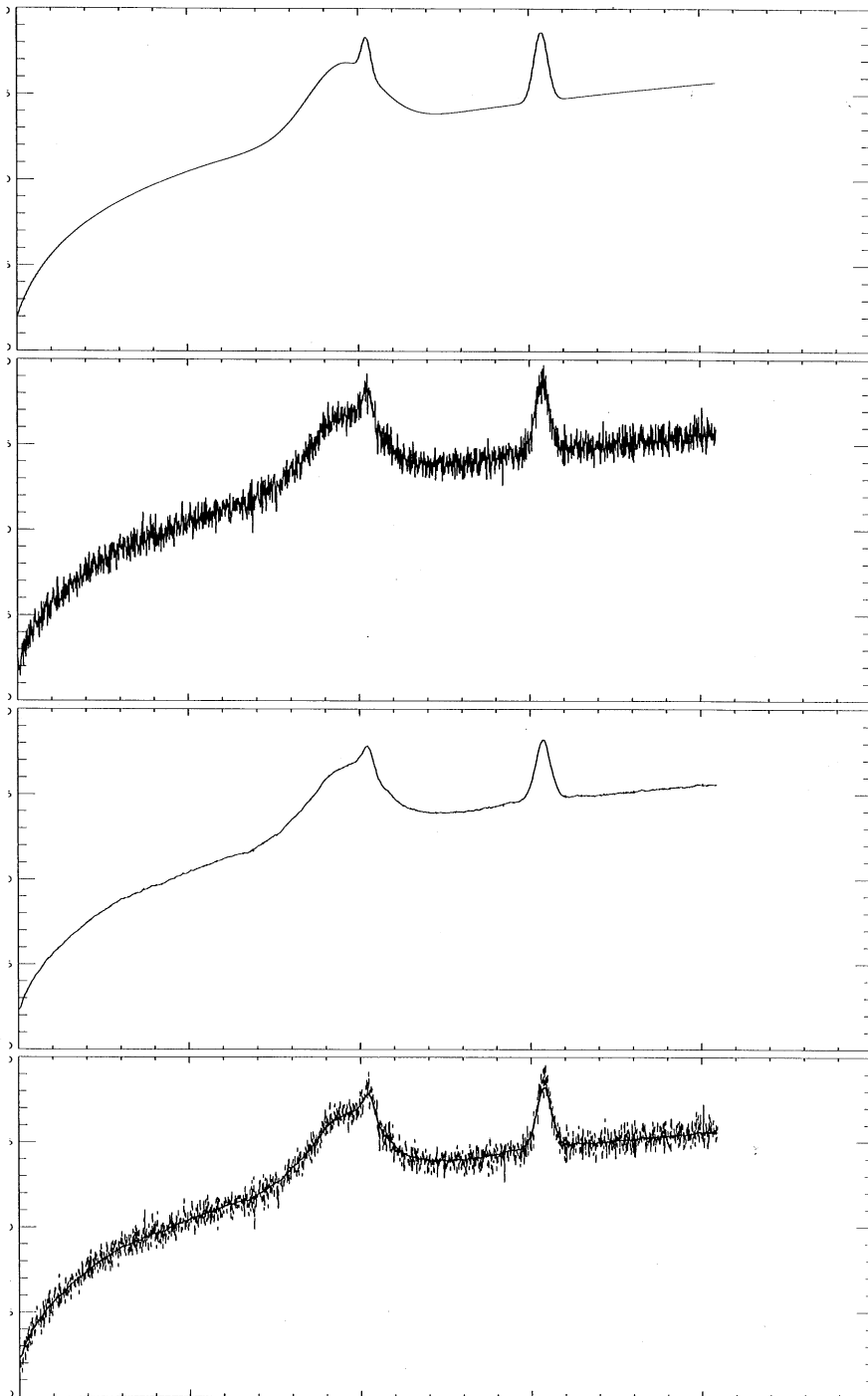


Fig. 4. From top to bottom, simulated data, noisy data, filtered data, and both noisy and filtered data overlotted.

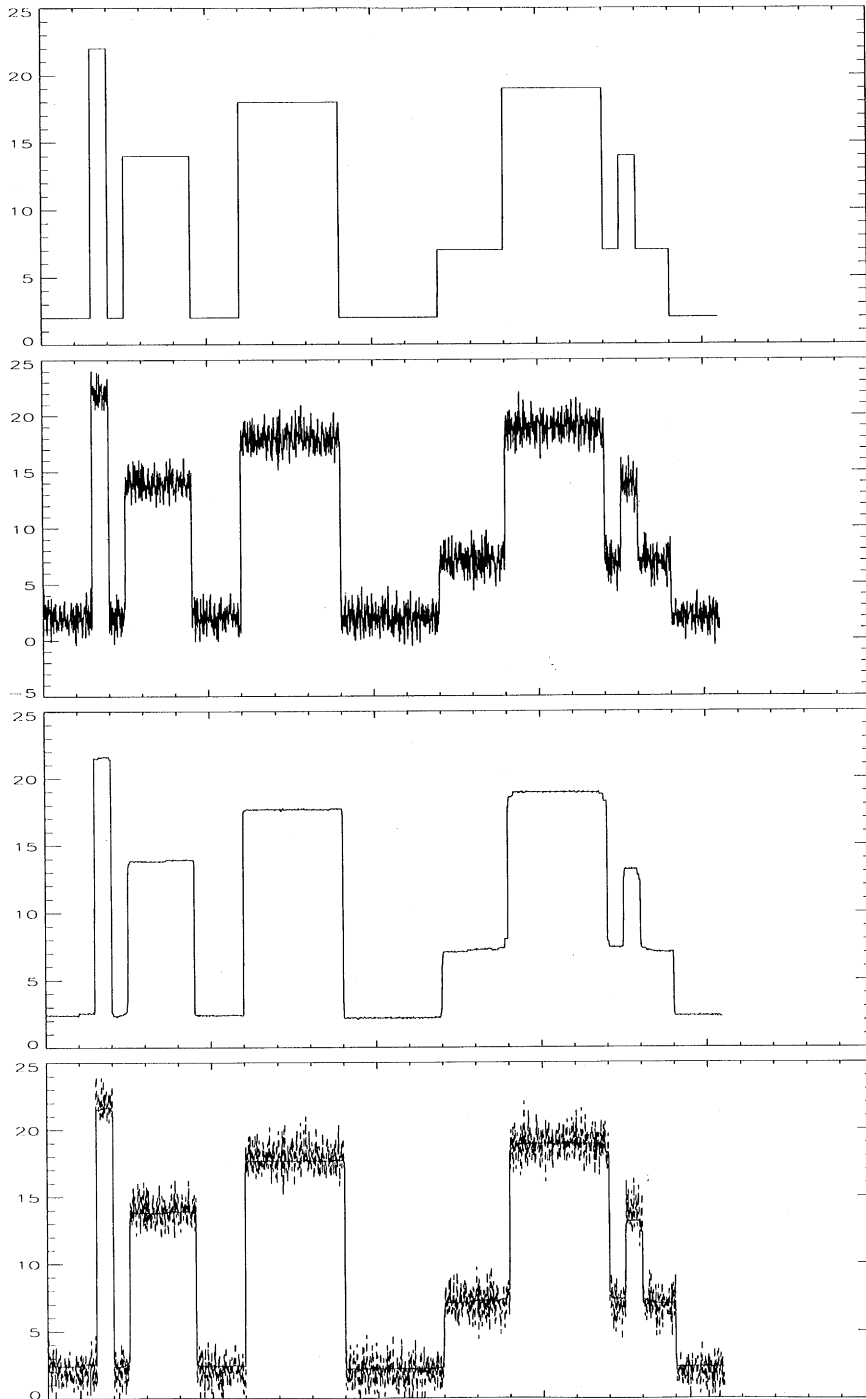


Fig. 5. From top to bottom, simulated block data, noise blocks, filtered blocks, and both noisy and filtered blocks overplotted.

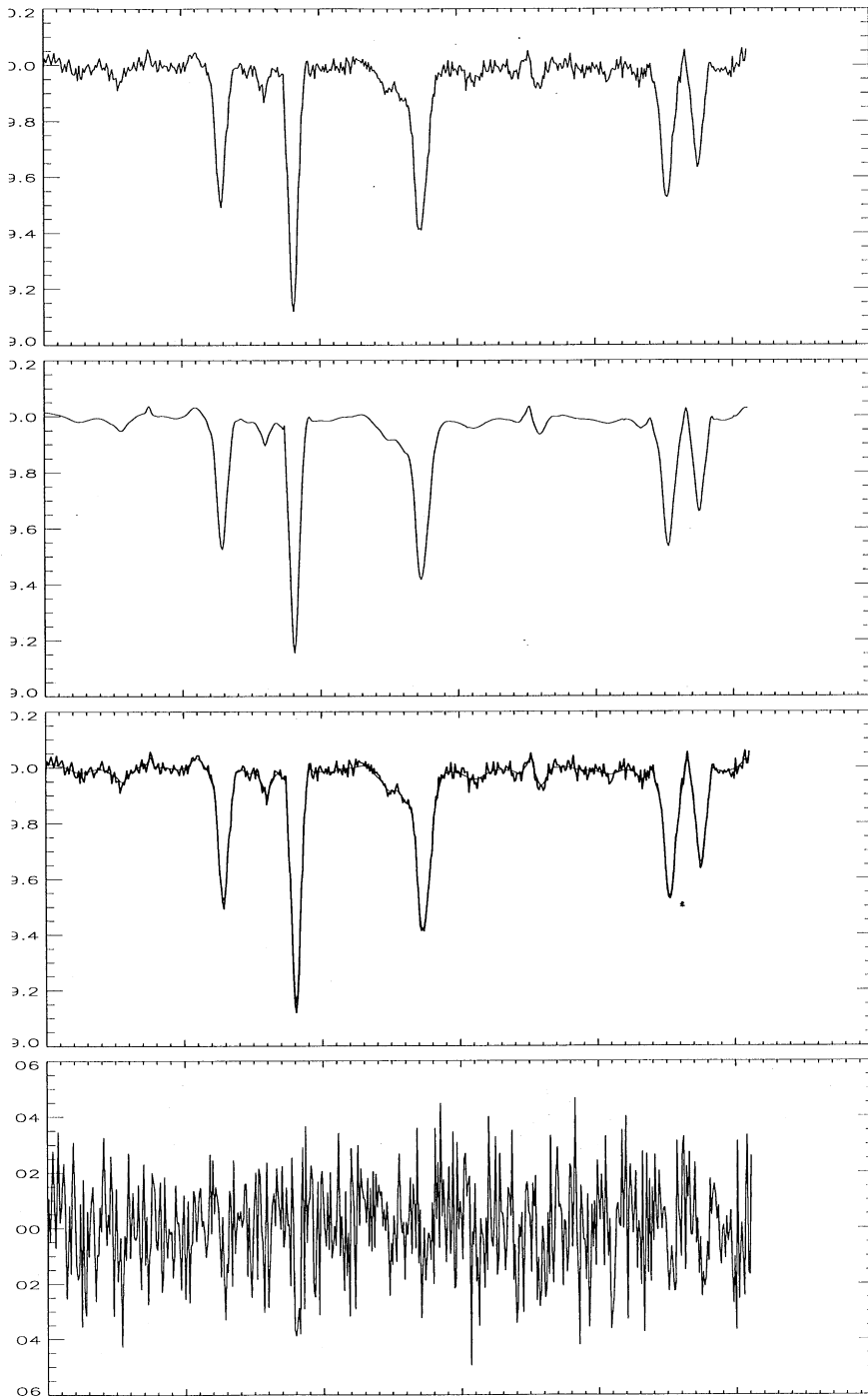


Fig. 6. From top to bottom, real spectrum, filtered spectrum, both noisy and filtered spectrum overplotted, and difference between the spectrum and the filtered data. As we can see, the residual contains only noise.

Fig. 6 shows the result after applying the MEF method to a real spectrum (512 pixels). The last plot shows the difference between the original and the filtered spectrum. As we can see, the residual contains only noise. In this case, we used also default parameters, but we introduce the SNR in the calculation of  $\alpha$ .

### 3.7.2. Image filtering

A simulated  $256 \times 256$  image containing stars and galaxies is shown in Fig. 7 (top left). The

simulated noisy image, the filtered image and the residual image are, respectively, shown in Fig. 7 top right, bottom left, and bottom right. We can see that there is no structure in the residual image.

### 3.8. Comparison with other methods from simulations

#### 3.8.1. Simulation descriptions

A set of simulations have been realized based on two images: the classical Lena  $512 \times 512$  image, and

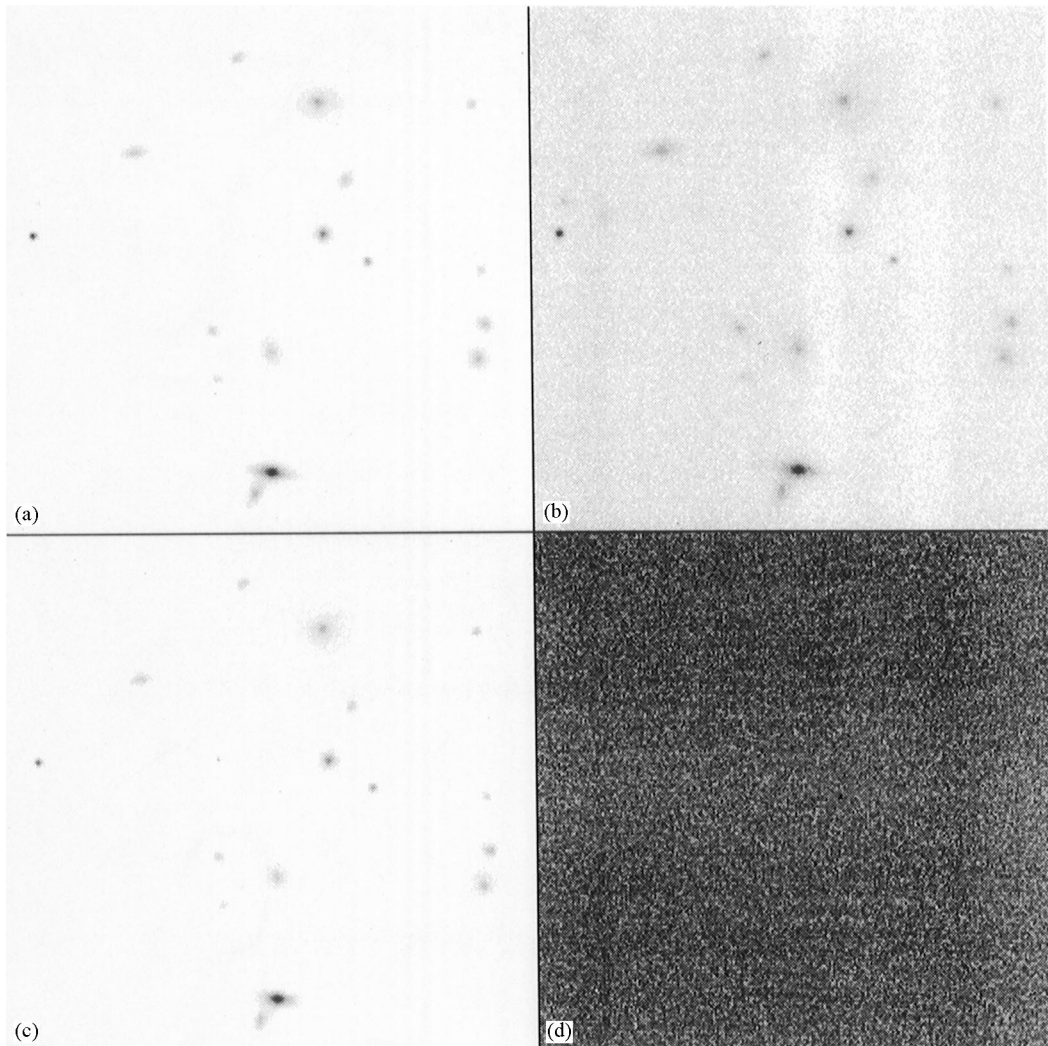


Fig. 7. (a) Simulated image, (b) simulated image and Gaussian noise, (c) filtered image, and (d) residual image.

Table 1  
PSNR after filtering the simulated image (Lena + Gaussian noise ( $\sigma = 5$ ))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	34.63	35.95	33.27	35.20	34.82
Soft thresh.	32.35	33.83	30.67	32.30	32.43
Donoho hard thresh.	33.19	34.62	31.05	33.98	33.68
Donoho soft thresh.	30.69	32.09	28.73	30.76	31.19
Hierarchical thresh.	–	–	–	35.26	34.89
Hierarchical Wiener	–	–	–	33.35	31.91
Multiresol. Wiener	–	–	–	33.42	31.93
Multiscale entropy	35.86	36.76	–	35.82	35.56

Table 2  
PSNR after filtering the simulated image (Lena + Gaussian noise ( $\sigma = 10$ ))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	31.31	32.97	29.87	32.63	31.80
Soft thresh.	29.72	31.29	28.05	30.03	30.15
Donoho hard thresh.	29.94	31.55	27.68	31.33	30.88
Donoho soft thresh.	28.18	29.66	26.77	28.49	29.09
Hierarchical thresh.	–	–	–	32.75	31.93
Hierarchical Wiener	–	–	–	31.71	30.33
Multiresol. Wiener	–	–	–	31.68	30.24
Multiscale entropy	32.12	33.39	–	32.41	31.95

Table 3  
PSNR after filtering the simulated image (Lena + Gaussian noise ( $\sigma = 30$ ))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	26.82	27.97	26.00	28.58	28.19
Soft thresh.	26.27	27.67	25.85	26.85	27.27
Donoho hard thresh.	25.99	27.46	25.80	27.03	27.42
Donoho soft thresh.	25.29	26.78	25.80	25.85	26.58
Hierarchical thresh.	–	–	–	28.97	28.42
Hierarchical Wiener	–	–	–	28.08	27.96
Multiresol. Wiener	–	–	–	27.25	26.81
Multiscale entropy	27.45	28.75	–	28.37	27.96

a  $512 \times 512$  landscape image. From each image, three images were created by adding Gaussian noise with standard deviations of 5, 10, 30. These six images were filtered using different multiresolution methods and different noise treatment methods.

The multiresolution methods were:

1. Haar wavelet transform (FWT-Haar).
2. Mallat–Daubechies bi-orthogonal wavelet transforms using the Daubechies-Antonini 7/9 filters [5] (FWT-7/9).
3. Feauveau wavelet transform.

Table 4  
PSNR after filtering the simulated image (Landscape + Gaussian noise (sigma = 5))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	32.50	33.02	30.48	32.49	31.79
Soft thresh.	30.35	30.97	28.27	29.87	29.59
Donoho hard thresh.	31.04	31.53	28.38	31.23	30.67
Donoho soft thresh.	28.80	29.40	26.70	28.46	28.45
Hierarchical thresh.	–	–	–	32.51	31.82
Hierarchical Wiener	–	–	–	30.59	30.32
Multiresol. Wiener	–	–	–	30.65	30.35
Multiscale entropy	34.63	34.94	–	34.30	33.97

Table 5  
PSNR after filtering the simulated image (Landscape + Gaussian noise (sigma = 10))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	29.32	30.00	27.38	29.88	28.91
Soft thresh.	27.89	28.66	26.18	27.78	27.45
Donoho hard thresh.	28.05	28.74	25.86	28.58	27.98
Donoho soft thresh.	26.53	27.32	25.32	26.50	26.52
Hierarchical thresh.	–	–	–	29.99	28.99
Hierarchical Wiener	–	–	–	29.59	28.04
Multiresol. Wiener	–	–	–	29.64	28.04
Multiscale entropy	30.80	31.35	–	30.70	30.16

Table 6  
PSNR after filtering the simulated image (Landscape + Gaussian noise (sigma = 30))

Method	FWT-Haar	FWT-7/9	Feauveau	à trous	MMT
Hard thresh.	25.44	26.01	24.80	26.55	25.90
Soft thresh.	25.03	25.88	24.75	25.33	25.10
Donoho hard thresh.	24.77	25.60	24.72	25.36	25.19
Donoho soft thresh.	24.26	25.29	24.725	24.61	24.51
Hierarchical thresh.	–	–	–	27.07	26.21
Hierarchical Wiener	–	–	–	26.52	25.83
Multiresol. Wiener	–	–	–	25.89	25.24
Multiscale entropy	26.33	27.11	–	26.88	26.16

4. À trous algorithm using a B-spline scaling function (see [33,35] for more details).

5. Multiresolution median transform (MMT) [37].

The first two belong to the class of fast wavelet transforms. The third is also a non-redundant transform, but compared to the FWT, the wavelet function is isotropic. The à trous algorithm is

redundant and isotropic, and finally the MMT is not a wavelet transform, but does allow a multi-resolution representation.

Using these five transforms, we used eight different strategies for correcting the multiresolution coefficients from the noise:

1. k-sigma hard and soft thresholding,
2. Donoho hard and soft thresholding,



3. Multiscale entropy method,
4. Hierarchical hard thresholding,
5. Multiresolution Wiener filtering,
6. Hierarchical Wiener filtering.

The last three strategies have been used (up to now) only with redundant transforms (à trous algorithm and MMT in our case).

Finally, close to two hundred filtered images were created. Four resolution scales were used for the filtering, and the constant  $k$  for the hard thresholding was always taken as equal to 4 for the first scale, and 3 for the others. For the multiscale entropy method, the parameter  $\alpha$  was determined by the program in order to get a standard deviation of the residual (i.e. image minus filtered image) of the same order as the noise standard deviation.

For each filtered image, the PSNR (peak signal-to-noise) ratio between the original image  $I$  and the filtered image  $F$  was calculated as

$$\text{PSNR}_{\text{dB}} = 10 \log_{10} \frac{255}{\text{NRMSE}^2}, \quad (34)$$

where NRMSE is the normalized root mean square error:

$$\text{NRMSE}^2 = \frac{\sum_{\text{pix}} (I - F)^2}{\sum_{\text{pix}} I^2}. \quad (35)$$

We also calculated the correlation factor, but we found that this does not furnish more information than the PSNR. If the PSNR is an objective measure, it is however not sufficient, because it does not allow us to control whether artifacts are present or not. Images were therefore also visually assessed, in order to decide if artifacts are visible.

Results of the simulations are presented in Tables 1–6.

### 3.9. Simulation analysis

#### 3.9.1. Multiresolution algorithm

Filtering using the Haar transform always produces artifacts, even at low noise levels. When using other filters, artifacts appear only beyond a given noise level. Improving the filter set improves the filtered image quality, which is a well-known result. When the noise increases, artifacts

appear, even with a good filter set such as the Antonini 7/9 one.

- *Feauveau WT*. The standard orthogonal WT is always better than the Feauveau method for filtering.
- *À trous algorithm*. This does not create artifacts when thresholding, and results are significantly better (from the visual point of view) at high noise levels, compared to orthogonal WT approaches. As opposed to the standard WT method, this transform is isotropic and performs better on isotropic structures compared to faint contours. This is the reason for its success on astronomical images where objects are diffuse and more or less isotropic (stars, galaxies, etc.).
- *Multiresolution median transform*. This transform is non-linear, and noise estimation at the different scales cannot be carried out in the same rigorous way as with linear transforms. For pure Gaussian noise, there is clearly no interest in using this transform, even if it respects well the morphology of the objects contained in the image. For some other kinds of noise, the non-linearity can be an advantage, and it can then be considered.

#### 3.9.2. Conclusion

The Feauveau WT and the MMT are not competitive for filtering in the case of Gaussian noise. FWT-7/9 allows better restoration of the edges than the à trous algorithm, but the à trous algorithm is more robust from the visual point of view. The important point to be made is clearly that the way the information is represented is fundamental. At high noise levels, whatever the chosen filter set, we will always have more artifacts using the FWT than with the à trous algorithm.

#### 3.10. Noise treatment strategies

- *The optimal method depends on the noise level*. At low noise levels, simple thresholding using an orthogonal wavelet transform leads to very good results. When the noise increases, artifacts appear. Non-orthogonal transforms produce better results, and soft thresholding strategies lead to more acceptable image quality.
- *Donoho soft and hard thresholding versus the  $k$ -sigma approach*. Whatever the multiresolution

transform and the noise level, the  $k$ -sigma hard (respectively soft) thresholding is always better than the Donoho hard (respectively soft) thresholding. Both PSNR ratio and visual aspect are better using the  $k$ -sigma approach. This outcome is not too surprising. Indeed the threshold, in the Donoho approach, is increasing with the number of pixels (justified in order to have a fixed number of ‘artifacts’). For our  $512 \times 512$  image, this approach is equivalent to thresholding at  $5\sigma$ . But then the thresholding level is too high, because the main coefficients between  $3\sigma$  and  $5\sigma$  are significant. The larger the image size, the stronger will be the over-smoothing.

- *Hierarchical thresholding.* The modification of the thresholding level at a given scale using the information at the following scale improves the result. The PSNR is better, and the visual aspect is similar to the hard thresholding. This procedure could certainly be also introduced into orthogonal transforms.
- *Quality of the multiscale entropy method.* The multiscale entropy method proposes a visually good solution whatever the noise level. It is in fact a method which preserves high wavelet coefficients, and corrects other wavelet coefficients in an adaptive, soft, manner.

#### 4. Conclusion

If a hard or a soft thresholding approach is used, the  $k$ -sigma value should be preferred to the universal  $\sqrt{(2 \log(n))}$  value. Multiresolution Wiener filtering and hierarchical Wiener filtering are not at all competitive.

The multiscale entropy method is an adaptive soft approach which is certainly the best when considering both visual quality and the PSNR criterion. At low noise levels, an FWT can be used, which allows better restoration of edges (assuming the image does contain edges!), and at high noise levels, the à trous algorithm must be chosen since otherwise artifacts related to decimation appear. However, these artifacts are less severe than those produced by poor thresholding.

Fig. 8 shows how a wavelet coefficient is modified using a hard thresholding, a soft thresholding, MEF method, and MEF method with  $\alpha$  as a function of the SNR. As we can see, MEF methods are intermediate between hard and soft thresholding, but do not present any discontinuity as the hard thresholding. This is the reason why good SNR is obtained with the MEF method, while retaining also good visual quality.

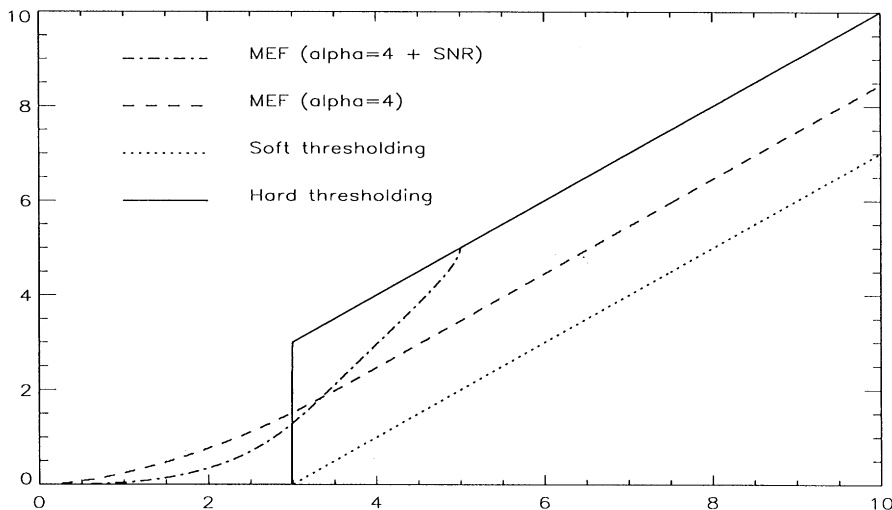


Fig. 8. Filtered wavelet coefficients versus wavelet coefficients (for a noise standard deviation equal to 1) by four methods: hard thresholding, soft thresholding, multiscale entropy filtering, and multiscale entropy filtering with a non-constant  $\alpha$  (SNR depending) value.

## References

- [1] J.G. Ables, *Astron. Astrophys. Suppl. Ser.* 15 (1974) 383–393.
- [2] E.H. Adelson, E. Simoncelli, R. Hingorani, Optimal image addition using the wavelet transform, *SPIE Visual Commun. Image Process.* II 845 (1987) 50–58.
- [3] U. Amato, D.T. Vuzza, Wavelet approximation of a function from samples affected by noise, *Rev. Roumaine Math. Pures Appl.* (1998) in press.
- [4] F.J. Anscombe, The transformation of Poisson, binomial and negative-binomial data, *Biometrika* 15 (1948) 246–254.
- [5] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using wavelet transform, *IEEE Trans. Image Process.* 1 (2) (1992) 205–220.
- [6] T.J.R. Bontekoe, E. Koper, D.J.M. Kester, Pyramid maximum entropy images of IRAS survey data, *Astron. Astrophys.* 294 (1994) 1037–1053.
- [7] A. Bruce, H.-Y. Gao, *S+ Wavelets User's Manual*, StatSci. Division, MathSoft Inc., Seattle, 1994, Version 1.0.
- [8] J.P. Burg, Annual Meeting International Society Exploratory Geophysics, Reprinted in: D.G. Childers (Ed.), *Modern Spectral Analysis*, 1978, IEEE Press, New York, 1978, 1967, pp. 34–41.
- [9] A. Chambolle, R.A. DeVore, N. Lee, B.J. Lucier, Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage, *IEEE Trans. Signal Process.* 7 (3) (1998) 319–335.
- [10] H.A. Chipman, E.D. Kolaczyk, R.E. McCulloch, Adaptive bayesian wavelet shrinkage, *J. Amer. Statist. Assoc.* 92 (440) (1997) 1413–1421.
- [11] A. Cohen, I. Daubechies, J.C. Feauveau, Biorthogonal bases of compactly supported wavelets, *Commun. Pure Appl. Math.* 45 (1992) 485–560.
- [12] M. Crouse, R. Nowak, R. Baraniuk, Wavelet-based statistical signal processing using hidden markov models, *IEEE Trans. Signal Process.* (Special issue on Wavelets and Filterbanks) (1998) in press.
- [13] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Series in Applied Mathematics, Vol. 61, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1992 (NuHAG-lib.).
- [14] D.L. Donoho, Nonlinear wavelet methods for recovery of signals, densities, and spectra from indirect and noisy data, in: *Proceedings of Symposia in Applied Mathematics*, Vol. 47, 1993.
- [15] D.L. Donoho, Translation-invariant de-noising, in: A. Antoniadis, G. Oppenheim (Eds.), *Wavelets and Statistics*, Springer, Berlin, 1995.
- [16] D.L. Donoho, I.M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Technical Report 400, Stanford University, 1993.
- [17] J.C. Feauveau, *Analyse multirésolution par ondelettes non-orthogonales et bancs de filtres numériques*, Ph.D. Thesis, Université Paris Sud, 1990.
- [18] B.R. Frieden, *Image Enhancement and Restoration*, Wiley, Berlin, Springer edition, 1978.
- [19] S.F. Gull, *J. Skilling, MEMSYS5 User's Manual*, 1991.
- [20] M. Holschneider, R. Kronland-Martinet, J. Morlet, P. Tchamitchian, *Wavelets: Time-Frequency Methods and Phase-Space*, Chapter A Real-Time Algorithm for Signal Analysis with the Help of the Wavelet Transform, Springer, Berlin, 1989, pp. 286–297.
- [21] M. Jansen, D. Roose, Bayesian correction of wavelet threshold procedures for image de-noising, in: *Processing of the Joint Statistical Meeting, Bayesian Statistical Science*, 1998, in press.
- [22] E.T. Jaynes, *Phys. Rev.* 106 (1957) 620–630.
- [23] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (7) (1989) 674–693.
- [24] S.G. Mallat, Zero crossings of a wavelet transform, *IEEE Trans. Inform. Theory* 37 (4) (1991) 1019–1033.
- [25] F. Murtagh, J.L. Starck, A. Bijaoui, Image restoration with noise suppression using a multiresolution support, *Astron. Astrophys. Suppl. Ser.* 112 (1995) 179–189.
- [26] R. Narayan, R. Nityananda, Maximum entropy image restoration in astronomy, *Ann. Rev. Astron. Astrophys.* 24 (1986) 127–170.
- [27] G.P. Nason, Wavelet shrinkage by cross-validation, *J. Roy. Statist. Soc. B* 58 (1996) 463–479.
- [28] S.I. Olsen, Estimation of noise in images: An evaluation, *CVGIP* 55 (1993) 319–323.
- [29] E. Pantin, J.L. Starck, Deconvolution of astronomical images using the multiscale maximum entropy method, *Astron. Astrophys. Suppl. Ser.* 315 (1996) 575–585.
- [30] C.E. Shannon, A mathematical theory for communication, *Bell System Tech. J.* 27 (1948) 379–423.
- [31] J. Skilling, *Classic maximum entropy*, in: *Maximum Entropy and Bayesian Methods*, Kluwer, Dordrecht, 1989, pp. 45–52.
- [32] J.L. Starck, A. Bijaoui, Filtering and deconvolution by the wavelet transform, *Signal Processing* 35 (1994) 195–211.
- [33] J.L. Starck, A. Bijaoui, F. Murtagh, Multiresolution support applied to image filtering and deconvolution, *CVGIP: Graphical Models Image Process.* 57 (1995) 420–431.
- [34] J.L. Starck, F. Murtagh, Automatic noise estimation from the multiresolution support, *Publications Astron. Soc. Pacific* 110 (744) (1998) 193–199.
- [35] J.L. Starck, F. Murtagh, A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*, Cambridge University Press, Cambridge, 1998.
- [36] J.L. Starck, F. Murtagh, R. Gstaad, A new entropy measure based on the wavelet transform and noise modeling, *Special Issue on Multirate Systems, Filter Banks, Wavelets, and Applications of IEEE Trans. on CAS II* 45 (8) (1998).
- [37] J.L. Starck, F. Murtagh, B. Pirenne, M. Albrecht, Astronomical image compression based on noise suppression, *Publications Astron. Soc. Pacific* 108 (1996) 446–455.
- [38] G. Strang, T. Nguyen, *Wavelet and Filter Banks*, Wellesley-Cambridge Press, Box 812060, Wellesley MA 02181, 1996.
- [39] N. Weir, A multi-channel method of maximum entropy image restoration, in: D.M. Worral, C. Biemesderfer, J. Barnes (Eds.), *Astronomical Data Analysis Software and System 1*, Astronomical Society of the Pacific, 1992, pp. 186–190.