GRAPHICAL MODELS AND IMAGE PROCESSING
Vol. 57, No. 5, September, pp. 420–431, 1995

Multiresolution Support Applied to Image Filtering and Restoration

JEAN-LUC STARCK

CEA, DSM/DAPNIA, CE-SACLAY, F-91191 Gif-sur-Yvette Cedex, France

FIONN MURTAGH

Space Telescope—European Coordinating Facility, European Southern Observatory, Karl-Schwarzschild-Strasse 2, D-85748 Garching, Germany*

AND

ALBERT BIJAOUI

Observatoire de la Côte d'Azur, B.P. 229, F-06304 Nice Cedex 4, France

Received July 5, 1994; revised April 24, 1995; accepted April 28, 1995

The notion of a multiresolution support is introduced. This is a sequence of Boolean images related to significant pixels at each of a number of resolution levels. The multiresolution support is then used for noise suppression, in the context of image filtering, or iterative image restoration. Algorithmic details, and a range of practical examples, illustrate this approach. © 1995 Academic Press, Inc.

1. INTRODUCTION

1.1. *General Ideas and Plan of Paper*

The human visual system picks out objects of interest at different *scales*. In recent years, therefore, image processing has sought to make use of multiscale or multiresolution representations. A range of theories are available such as quadtrees and pyramid representations, scale-space filtering, and the wavelet transform. For the first two of these, reference may be made to Lindeberg [20].

In this paper, a computationally efficient wavelet transform algorithm is used to generate a sequence of multiresolution views of the image. Following this, in each of the wavelet planes, a *support* is defined, i.e., a Boolean image where significant pixels have a 1 or *true* value, and all other pixels a 0 or *false* value. Contiguous areas of 1-valued pixels are associated with objects in the image being analyzed, at the given resolution or scale. The set of support images, at each resolution level, is called the multiresolution support.

The multiresolution support is an important data structure, which provides a powerful framework for noise filtering and for restoration with noise suppression. The procedure used is to determine statistically significant wavelet coefficients, and from this to specify the support. Thus a statistical image model is used as an integral part of the image processing. The support is used subsequently to handcraft the filtering or restoration (or, although not treated in this paper, object detection). Statistical image models are available in astronomical image processing, and our examples are based on images from this field. We will discuss implementation strategies and experimental results.

This paper is structured as follows. Section 2 introduces the multiresolution support and discusses how it can be used to determine noise in the image. Section 3 deals with noise filtering, and we find that use of the multiresolution support offers a powerful and versatile way to handle noise of different distributions. Section 4 covers image restoration methods. Two appendices provide further detail on some central aspects of the paper. A list of the principal notation used precedes the references.

1.2. *Related Work*

Astronomical images—especially when relating to scenes and objects outside our solar system—have properties which make them quite different from images in industrial vision or remote sensing. Astronomical images for the most part contain point sources (stars and other approximately point symmetric objects) and extended objects (galaxies, nebulae, etc. which are often faint). These objects

* Affiliated with Astrophysics Division, Space Science Department, European Space Agency.

may be superimposed. Edges and alignments rarely play a role. For astronomical image restoration issues, the reader may consult the articles in [14, 44].

An established and successful approach to image restoration and filtering on nonastronomical imagery is to use regularization with a smoothness constraint [17, 21]. This leads to definition of a functional to be optimized, with consideration given to important image properties such as edges. Adaptive choice of the regularization has been used in [13, 16]. As mentioned, astronomical images contain "edges with no extension" (point sources) and diffuse objects. A Tikhonov optimization criterion does not do justice to such objects. Instead we propose an effective heuristic restoration and filtering approach in this field.

"Regularization" as used in this paper involves use of a multiresolution support. A support constraint in the space of wavelet coefficients is in keeping with our vision of the image: superimposed and variably sized point sources and extended objects. The optimization problem is formulated in algorithmic terms, and the greedy solution method is reminiscent of another widely used astronomical restoration method, termed CLEAN (predominent in radio astronomy; it consists of iteratively fitting a point spread function and moving flux from the given "dirty" image to the output "cleaned" image). Similar to CLEAN and to [45], we can argue that our adaptive approach is straightforward, easy to implement, and robust.

Smoothing without reference to astronomical content is used in [26, 35]. Filtering as described in this paper aims at protection of the objects in the image, so that photometric (intensity-related), astrometric (position-related), and morphological information remains faithful (by design) to the input image data.

Previous work of ours has dealt with the choice of effective wavelet transform (see [3, 4]) and a discussion of common noise models [32, 41]. This paper will deal with the adaptive, local regularization implied by constraining the operations of restoration and filtering to respect the multiresolution constraint data structure.

2. MULTIRESOLUTION SUPPORT

2.1. *Definition*

We will say that a multiresolution support of an image describes in a logical or Boolean way if an image I contains information at a given scale j and at a given position (x, y). If $M^{(I)}(j, x, y) = 1$ (or $=$ *true*), then I contains information at scale j and at the position (x, y).

Such a support results from the data, the treatment (noise estimation, etc.), and knowledge on our part of the objects contained in the data (size of objects, linearity, etc.).

The multiresolution support of an image is computed in several steps: compute the wavelet transform of the image, booleanize each scale which yields the multiresolution support, and introduce a priori knowledge by modifying the support.

The last step depends on the knowledge we have of our images. For instance, if we know there is no interesting object smaller or larger than a given size in our image, we can suppress, in the support, anything which is due to that kind of object. This can often be done conveniently by the use of mathematical morphology. In the most general setting, we naturally have no information to add to the multiresolution support.

2.2. *Multiresolution Support from the Wavelet Transform*

There are more than 10 widely used wavelet transform algorithms [3, 7, 11, 25, 29, 31, 40]. We chose the *à trous* algorithm [15, 37, 41] for the following reasons:

1. The transform is carried out in direct space. No artifacts arise due to periodization.
2. The computational requirement is reasonable, as are memory and storage requirements. One property influencing the computational requirement is that the scaling functions are compact.
3. In two dimensions, the transform is practically isotropic (point symmetric).
4. The transform is known at each pixel, allowing reconstruction without any error and without interpolation. We can follow the evolution of the transform from one scale to the next.
5. Invariance under translation is completely verified.

Details of the algorithm are given in Appendix 1. The wavelet transform of an image by this algorithm produces at each scale j, a set $\{w_j\}$ which we will call a wavelet plane throughout the following discussion. This has the same number of pixels as the image. The original image c_0 can be expressed as the sum of all the wavelet planes and the smoothed array c_p

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

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

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

The multiresolution support will be obtained by detecting at each scale the significant coefficients. We will see in the next section how to find these coefficients. The multiresolution support is defined by

$$M(j,x,y) = \begin{cases} 1 & \text{if } w_j(x,y) \text{ is significant} \\ 0 & \text{if } w_j(x,y) \text{ is not significant} \end{cases} \qquad (3)$$

2.3. *Significant Coefficients*

2.3.1. *Statistically Significant Level.* Images generally contain noise. Hence the wavelet coefficients are noisy too. In most applications, it is necessary to know if a coefficient is due to signal or to noise. Generally noise in astronomical images follows a Gaussian or a Poisson distribution.

The wavelet transform yields a set of resolution-related views of the input image. A wavelet image plane at level j has coefficients given by $w_j(x, y)$. If we obtain the distribution of the coefficient $w_j(x, y)$ for each plane, based on the noise, we can introduce a statistical significance test for this coefficient. The procedure is the classical significance-testing one. Let $\mathcal{H}_0$ be the hypothesis that the image is locally constant at scale j. Consider first the case of Gaussian noise. The distribution of $w_j(x, y)$ is Gaussian, with zero mean and standard deviation σ_j. We have the probability density

$$p(w_j(x,y)) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp(-w_j(x,y)^2/2\sigma_j^2). \qquad (4)$$

Rejection of hypothesis $\mathcal{H}_0$ depends (for a positive coefficient value) on

$$P = Prob(w_j(x,y) < W) = \frac{1}{\sqrt{2\pi}\sigma_j} \int_{w_j(x,y)}^{+\infty} \exp(-W^2/2\sigma_j^2)\, dW, \qquad (5)$$

and if the coefficient value is negative, we reject if

$$P = Prob(w_j(x,y) > W) = \frac{1}{\sqrt{2\pi}\sigma_j} \int_{-\infty}^{w_j(x,y)} \exp(-W^2/2\sigma_j^2)\, dW. \qquad (6)$$

Given a threshold, ε, if $P > \varepsilon$ the null hypothesis is not excluded. Although nonnull, the value of the coefficient could be due to noise. On the other hand, if $P > \varepsilon$, the coefficient value cannot be due only to the noise alone, and so the null hypothesis is rejected. In this case, a significant coefficient has been detected.

Given stationary, Gaussian noise, it suffices to compare $w_j(x, y)$ to $k\sigma_j$. Often k is chosen as 3. If $w_j(x, y)$ is small, it is not significant and could be due to noise. If $w_j(x, y)$ is large, it is significant:

$$\begin{aligned} &\text{if } |w_j| \geq k\sigma_j \quad \text{then } w_j \text{ is significant} \\ &\text{if } |w_j| < k\sigma_j \quad \text{then } w_j \text{ is not significant.} \end{aligned} \qquad (7)$$

If the noise in the data I is Poisson, the transform [2]

$$T(I(x,y)) = 2\sqrt{I(x,y) + \frac{3}{8}} \qquad (8)$$

acts as if the data arose from the Gaussian white noise model, with unit standard deviation. In this case, we will take the wavelet transform of $T(I)$, and $w_j^{(I)}(x,y)$ will be significant if $w_j^{(T(I))}(x,y)$ is above a given threshold. (Here the superscript on the wavelet coefficients indicates the image on which the wavelet transform was carried out.) Generalization of transform (8) for combined Poisson and Gaussian noise is discussed in [32], leading to the transform

$$T(I(x,y)) = \frac{2}{\alpha}\sqrt{\alpha I(x,y) + \frac{3}{8}\alpha^2 + \sigma^2 - \alpha g}, \qquad (9)$$

where the Gaussian noise has mean g and standard deviation σ, and where α is the gain (in the case of widely used CCD detectors).

Thus we need to estimate, in the case of Gaussian, Poisson, or additive Poisson and Gaussian noise models, the noise standard deviation at each scale.

2.3.2. *Noise Standard Deviation Estimation at Each Scale.* The appropriate value of σ_j in the succession of wavelet planes is assessed from the standard deviation of the noise σ_I in the original image and from study of the noise in the wavelet space. This study consists of simulating an image containing Gaussian noise with a standard deviation equal to 1 and taking the wavelet transform of this image. Then we compute the standard deviation σ_j^e at each scale. We get a curve σ_j^e as a function of j, giving the behavior of the noise in the wavelet space. (Note that if we had used an orthogonal wavelet transform, this curve would be linear.) Due to the properties of the wavelet transform, we have $\sigma_j = \sigma_I\sigma_j^e$. The standard deviation of the noise at a scale j of the image is equal to the standard deviation of the noise of the image multiplied by the standard deviation of the noise of the scale j of the wavelet transform.

An alternative, here, would be to estimate the standard deviation of the noise σ_1 of the first plane from the histogram of w_1. The values of the wavelet image w_1 are due mainly to the noise. A histogram shows a Gaussian peak around 0. A 3σ clipping is then used to reject pixels where the signal is significantly large. The standard deviation of the noise σ_j is estimated from σ_1. This is done from the study of noise variation between two scales, as described above.

A final alternative to be mentioned here relates to multiple images of the same scene. In this case, a pixel-dependent specification of the noise threshold is used, rather than just level-dependent. A wavelet transform of each of the N images is determined. Thus, we have N wavelet

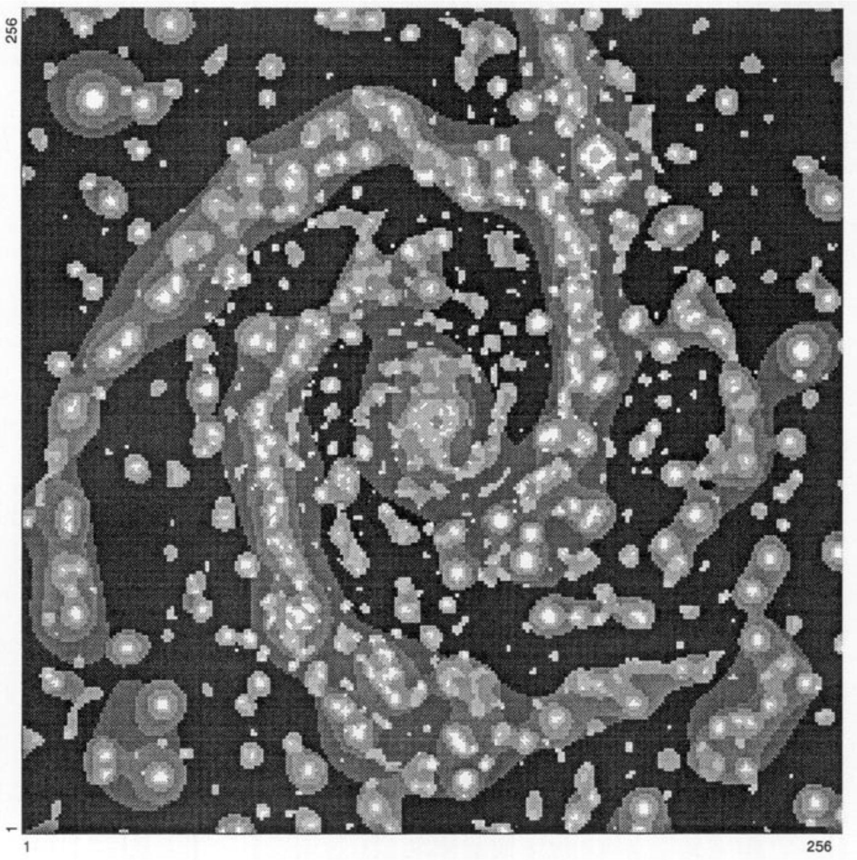

FIG. 1. Multiresolution support representation of a spiral galaxy.

coefficients at each position, (x, y), and at each scale, j. From the N values, $w_j(x, y)$, the standard deviation, $\sigma_j(x, y)$, is calculated. The significance threshold is then defined by

$$k\sigma_j(x, y)/\sqrt{N} \qquad (10)$$

(the denominator is explained by the error of the mean of N Gaussian values varying as $1/\sqrt{N}$).

2.4. *Conclusion*

In order to visualize the support, we can create an image S defined by

$$S(x, y) = \sum_{j=1}^{p} 2^j M(j, x, y). \qquad (11)$$

Figure 1 shows such a multiresolution support visualization of an image of galaxy NGC 2997.

The multiresolution support allows us to integrate, in a visualizable manner, and in a way which is very suitable for ancillary image alteration, information coming from data, knowledge, and processing. We will see below how we can use it in image filtering and in image restoration.

3. FILTERING

3.1. *Filtering from Significant Coefficients*

It has been seen in Section 2.3.1 how significant wavelet coefficients are detected in an image. Reconstruction, after setting nonsignificant coefficients to zero, at full resolution leads to adaptive filtering [40]. The restored image is

$$\tilde{I}(x, y) = c_p(x, y) + \sum_{j=1}^{p} g(\sigma_j, w_j(x, y)) w_j(x, y) \qquad (12)$$

with g defined by

$$g(\sigma_j, w_j) = \begin{cases} 1 & \text{if } |w_j| \geq k\sigma_j \text{ (significant)} \\ 0 & \text{if } |w_j| < k\sigma_j \text{ (nonsignificant).} \end{cases} \qquad (13)$$

3.2. *Iterative Filtering from Significant Coefficients*

In the method just described, we obtain an image $\tilde{I}$ by reconstructing the thresholded coefficients. A satisfactory filtering implies that the error image $E = I - \tilde{I}$, obtained as the difference between the original image and the filtered image, contains only noise and no "structure." Such is not the case in practice with the approach described. However, we can easily arrive at this objective by iterating a few times:

1. $n \leftarrow 0$.
2. Initialize the solution, $I^{(0)}$, to zero.
3. Estimate the significance level (e.g., 3σ) at each scale.
4. Determine the error, $E^{(n)} = I - I^{(n)}$ (where I is the input image, to be filtered).
5. Determine the wavelet transform of $E^{(n)}$.
6. Threshold: only retain the significant coefficients.
7. Reconstruct the thresholded error image. This yields the image $\tilde{E}^{(n)}$ containing the significant residuals of the error image.
8. Add this residual to the solution: $I^{(n)} \leftarrow I^{(n)} + \tilde{E}^{(n)}$.
9. If $|(\sigma_{E^{(n-1)}} - \sigma_{E^{(n)}}/\sigma_{E^{(n)}}| > \varepsilon$ then $n \leftarrow n + 1$ and go to 4.
10. $I^{(n)}$ contains the filtered image, and $I - I^{(n)}$ is our estimation of the noise.

At each iteration, we extract the residual image of significant structures and we introduce them into the solution. We generally used between 6 and 10 iterations. On termination, we are certain that there are no further significant structures in the residual images.

If the noise associated with image I is Poisson, or Poisson and Gaussian, the significant structures are extracted as described in Appendix 2.

3.3. *Iterative Filtering from a Multiresolution Support*

From the iterative algorithm described in the preceding section, we reconstruct a filtered image $\tilde{I}$ such that, for all pixels, we have

$$|I(x, y) - \tilde{I}(x, y)| < k\sigma_I, \tag{14}$$

where σ_I is the standard deviation of the noise contained in the image. This filtering is effective, but does not always correspond to what is wanted. In astronomy, for example, we would prefer not to touch a pixel if it generates a significant coefficient at all scales. In general, we say that if a multiresolution coefficient of the original image is significant (i.e., $|w_j^{(I)}(x, y)| > K$, where K is the significance threshold), then the multiresolution coefficient of the error image (i.e., $w_j^{(E^{(n)})}$) must satisfy the following exactly:

$$w_j^{(E^{(n)})}(x, y) = 0 \quad \text{if } |w_j^{(I)}(x, y)| > K. \tag{15}$$

To arrive at this objective, we use the multiresolution support of the image, and the algorithm becomes:

1. $n \leftarrow 0$.
2. Initialize the solution, $I^{(0)}$, to zero.
3. Determine the multiresolution support of the image.
4. Estimate the significance level (e.g., 3σ) at each scale.
5. Determine the error, $E^{(n)} = I - I^{(n)}$ (where I is the input image, to be filtered).
6. Determine the multiresolution transform of $E^{(n)}$.
7. Threshold: only retain the coefficients which belong to the support.
8. Reconstruct the thresholded error image. This yields the image $\tilde{E}^{(n)}$ containing the significant residuals of the error image.
9. Add this residual to the solution: $I^{(n)} \leftarrow I^{(n)} + \tilde{E}^{(n)}$.
10. If $|(\sigma_{E^{(n-1)}} - \sigma_{E^{(n)}})/\sigma_{E^{(n)}}| > \varepsilon$ then $n \leftarrow n + 1$ and go to 4.

Thus the regions of the image which contain significant structures at all levels are not modified by the filtering. The residual will contain the value zero over all of these regions. The support can also be enriched by any available a priori knowledge. For example, if artifacts exist around objects, a simple morphological dilation of the support can be used to eliminate them.

3.4. *Example*

Figure 2 shows a noisy spectrum (upper left, repeated lower right). For the astronomer, the spectral lines—here mainly absorption lines extending downward—are of interest. The continuum may also be of interest, i.e., the overall spectral tendency. The spectral lines are unchanged in the filtered version (upper center and upper right). To illustrate the damage that can result from another wavelet transform, and another noise suppression policy, the lower center (and lower right) version shows the result of applying Daubechies' [9] coefficient 8, a compactly supported orthonormal wavelet. This was followed by thresholding based on estimated variance of the coefficients [10], but not taking into account the image's noise properties as we have done (see [33]). One sees immediately that a problem- (or image-)driven choice of wavelet and filtering strategy is indispensible.

4. DECONVOLUTION

4.1. *Iterative Restoration Algorithms*

Consider an image characterized by its intensity distribution (the "data") $I(x, y)$, corresponding to the observation of a "real image" $O(x, y)$ through an optical system. If the imaging system is linear and shift-invariant, the relation between the object and the image in the same coordinate frame is a convolution:

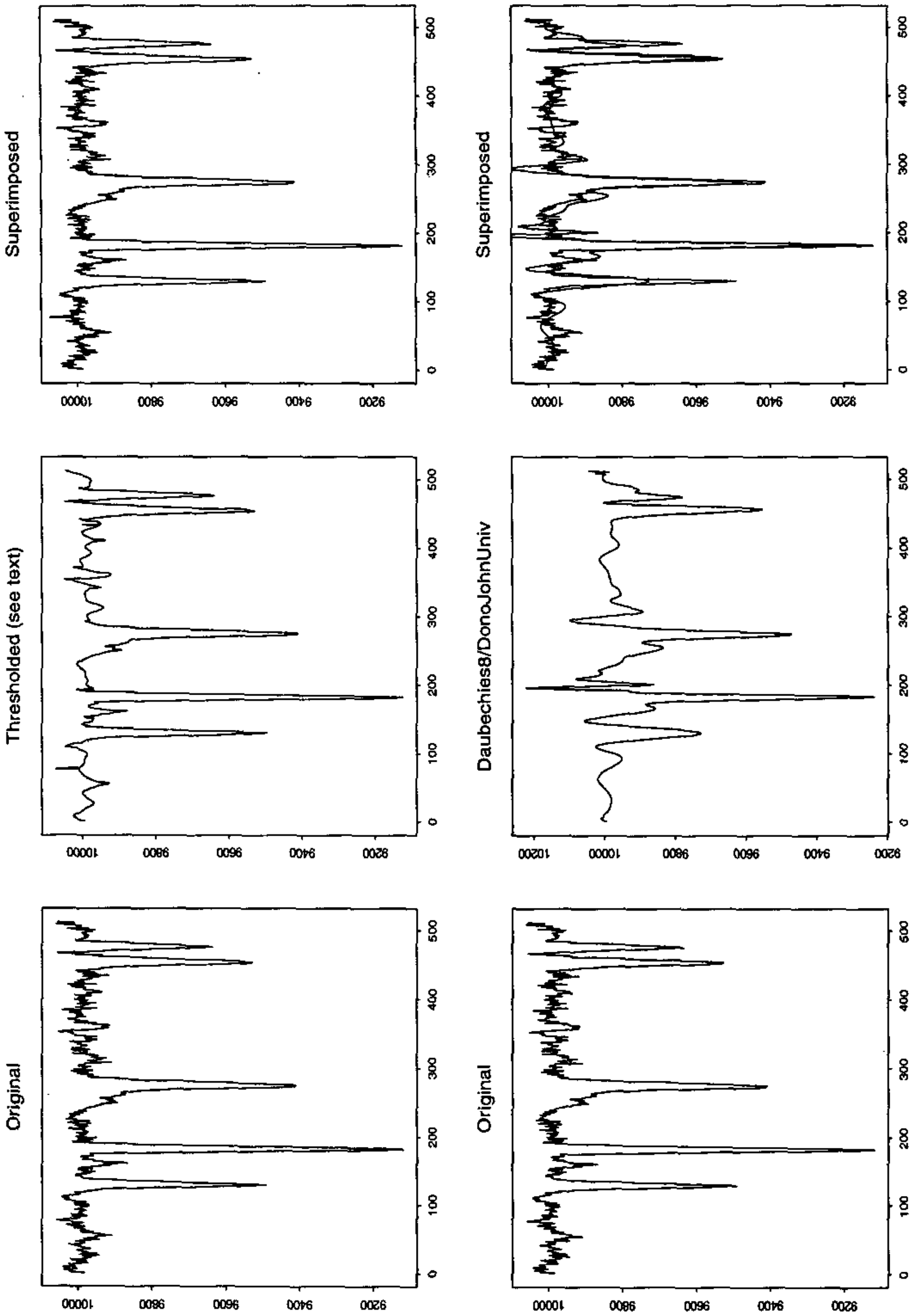

FIG. 2. (Top row) Original noisy spectrum; filtered spectrum; both superimposed. (Bottom row) Original; filtered (using Daubechies coefficient 8, and Donoho and Johnstone "universal" thresholding); both superimposed.

$$I(x,y) = (O * P)(x,y) + N(x,y). \tag{16}$$

$P(x, y)$ is the point spread function (PSF) of the imaging system, and $N(x, y)$ is an additive noise. In practice $O * P$ is subject to nonstationary noise which one can tackle by simultaneous object estimation and restoration [18]. The issue of more extensive statistical modeling will not be further addressed here (see [22, 23, 30]), beyond noting that multiresolution frequently represents a useful framework, allowing the user to introduce a priori knowledge of objects of interest.

Equation (16) is always an ill-posed problem. If the noise is modeled as a Gaussian or Poisson process, then an iterative approach for computing maximum likelihood estimates may be used. The Richardson–Lucy method ([24, 34]; see also [1, 18]) uses such an iterative approach,

$$\begin{aligned} O^{(n+1)} &\leftarrow O^{(n)}[(I/I^{(n)}) * P^*] \\ I^{(n)} &\leftarrow P * O^{(n)}, \end{aligned} \tag{17}$$

where P^* is the transpose of the PSF, and $O^{(n)}$ is the current estimate of the desired real image.

4.2. *Noise Suppression Based on the Wavelet Transform Decomposition*

In using an iterative deconvolution algorithm such as Van Cittert or Richardson–Lucy, we define $R^{(n)}(x, y)$, the residual at iteration n:

$$R^{(n)}(x,y) = I(x,y) - P(x,y) * O^{(n)}(x,y). \tag{18}$$

By using the *à trous* wavelet transform algorithm [40–42], $R^{(n)}$ can be defined by the sum of its p wavelet planes and the last smooth plane

$$R^{(n)}(x,y) = c_p(x,y) + \sum_{j=1}^{p} w_j(x,y), \tag{19}$$

where the first term on the right-hand side is the last smoothed plane, and w denotes a wavelet plane.

The wavelet coefficients provide a mechanism to extract only the significant structures from the residuals at each iteration. Normally, a large part of these residuals are statistically nonsignificant. The significant residual is then

$$\overline{R}^{(n)}(x,y) = c_p(x,y) + \sum_{j=1}^{p} g(w_j(x,y), \sigma_j)\, w_j(x,y). \tag{20}$$

σ_j is the standard deviation of the noise at scale j, and g is a function which is defined by

$$g(a,\sigma) = \begin{cases} 1 & \text{if } |a| \geq k\sigma\,(a \text{ significant}) \\ 0 & \text{if } |a| < k\sigma\,(a \text{ nonsignificant}). \end{cases} \tag{21}$$

The standard deviation of the noise σ_j is estimated from the standard deviation of the noise in the image as discussed above in Section 2.3.2.

4.3. *Noise Suppression Based on the Multiresolution Support*

In the approach presented in the preceding section, a wavelet coefficient is significant if it is above a threshold. Therefore, a coefficient which is less than this threshold is not considered, even if a significant coefficient had been found at the same scale as this coefficient, during previous iterations; and consequently we were justified in thinking that we had found signal at this scale, and at this position. Arising out of this approach, it follows that the wavelet coefficients of the residual image could contain signal, above the set threshold, which is ignored.

In order to conserve such signal, we use the notion of multiresolution support. Whenever we find signal at a scale j and at a position (x, y), we will consider that this position in the wavelet space belongs to the multiresolution support of the image.

Equation (20) becomes

$$\overline{R}^{(n)}(x,y) = c_p(x,y) + \sum_{j=1}^{p} M(j,x,y) w_j(x,y). \tag{22}$$

An alternative approach was outlined in [32]: the support was initialized to zero and built up at each iteration of the restoration algorithm. Thus in Eq. (22) above, $M(j, x, y)$ was additionally indexed by n, the iteration number. In this case, the support was specified in terms of significant pixels at each scale, j, and in addition pixels could become significant as the iterations proceeded, but could not be made nonsignificant. In practice, we have found both of these strategies to be equally acceptable.

4.4 *Regularization of the Richardson–Lucy Algorithm*

From Eq. (18), we seek $I^{(n)}(x, y) = P(x, y) * O^{(n)}(x, y)$. Then $R^{(n)}(x, y) = I(x, y) - I^{(n)}(x, y)$, and hence $I(x, y) = I^{(n)}(x, y) + R^{(n)}(x, y)$. The Richardson–Lucy equation is

$$O^{(n+1)}(x,y) = O^{(n)}(x,y)\left[\frac{I^{(n)}(x,y) + R^{(n)}(x,y)}{I^{(n)}(x,y)} * P(-x,-y)\right] \tag{23}$$

and regularization leads to

$$O^{(n+1)}(x,y) = O^{(n)}(x,y)\left[\frac{I^{(n)}(x,y) + \overline{R}^{(n)}(x,y)}{I^{(n)}(x,y)} * P(-x,-y)\right]. \tag{24}$$

The standard deviation of the residual decreases until no more significant structures are found. Convergence can be estimated from the residual. The algorithm stops when a user-specified threshold is reached:

$$(\sigma_{R^{(n-1)}} - \sigma_{R^{(n)}})/(\sigma_{R^{(n)}}) < \varepsilon. \tag{25}$$

Regularization of other iterative restoration methods, e.g., Van Cittert or One-Step Gradient, can be carried out in an analogous fashion.

4.5. *Example 1*

A simulated Hubble Space Telescope Wide Field Camera image of a distant cluster of galaxies was used to assess how well the suppression of noise, inherent in the wavelet-based method, aids object detection. The image used was one of a number described in [5, 12]. A spatially invariant PSF was used; the approximation to the known spatial variance which was involved in doing this was mitigated by use of a 256 × 256 subimage for test purposes. The simulated image allowed us to bypass certain problems, such as cosmic ray hits and CCD detector faults, and to concentrate on the general benefits of regularization of the type described in this article.

The procedure followed was to detect objects in the simulated image, and also in the images restored by the wavelet-based (or regularized) Richardson–Lucy method and the basic Richardson–Lucy method. The Inventory package in MIDAS (*Munich Image Data Analysis System*, a large image processing system, developed at the European Southern Observatory) was used for this. Inventory detects objects by means of a local background threshold, which was varied. Various other parameters were not used.

A set of 122 objects was found, using Inventory, in the original, unaberrated, noise-free image (Fig. 3, upper left). This agrees well with the fact that 124 objects were used in the simulation (121 galaxies, three stars). With a somewhat different threshold in the case of the wavelet-based Richardson–Lucy method, 165 objects were obtained. With a very much raised threshold (to exclude noise peaks) in the case of the basic Richardson–Lucy method, 159 objects were obtained.

Detections of spurious objects were made in the case of both restorations. Given that we have "ground truth" in this case, we simply selected the real objects among them. This was done by seeking good matches (less than 1 pixel separation) between objects found in the restored images, and the objects found in the original, unaberrated noise-free image. This led to 69 close matches, in the case of the wavelet-based Richardson–Lucy method, and to 53 close matches, in the case of the basic Richardson–Lucy method.

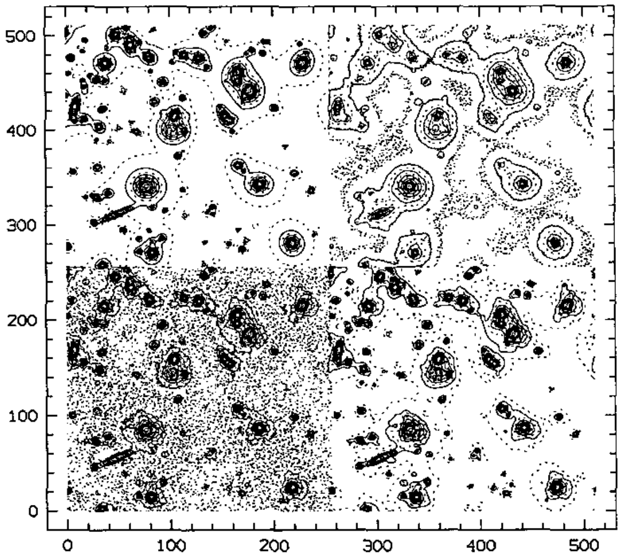

FIG. 3. Simulated HST wide field camera image of a distant cluster of galaxies. Four quadrants. (Upper Left) Original, unaberrated, and noise-free. (Upper Right) Input, aberrated, noise added. (Lower Left) Restoration, Richardson–Lucy method without noise suppression, 60 iterations. (Lower Right) Restoration, Richardson–Lucy method with noise suppression, 60 iterations. Intensities logarithmically transformed.

There was thus a greater number of object detections, obtained with the wavelet-based Richardson–Lucy method. These were also more accurate: the mean square error was 0.349 pixel units versus 0.379 for the smaller number of detections obtained from the basic Richardson–Lucy method. For bright objects, photometric plots using aperture magnitudes were relatively similar in both cases, and for fainter objects neither were good. While the wavelet-based Richardson–Lucy method acquitted itself well in these respects, its regularization property is clearly advantageous for object detection.

4.6. *Example 2*

We used the simulated elliptical galaxy available in the test image suite at anonymous ftp address `stsci.edu: / software/stsdas/testdata/restore`. It is briefly described in [14]. This image is referred to there as "Galaxy Number 2." It has a simple elliptical shape. The brightness profile includes both bulge and exponential disk components. It has additional distortions introduced in isophote

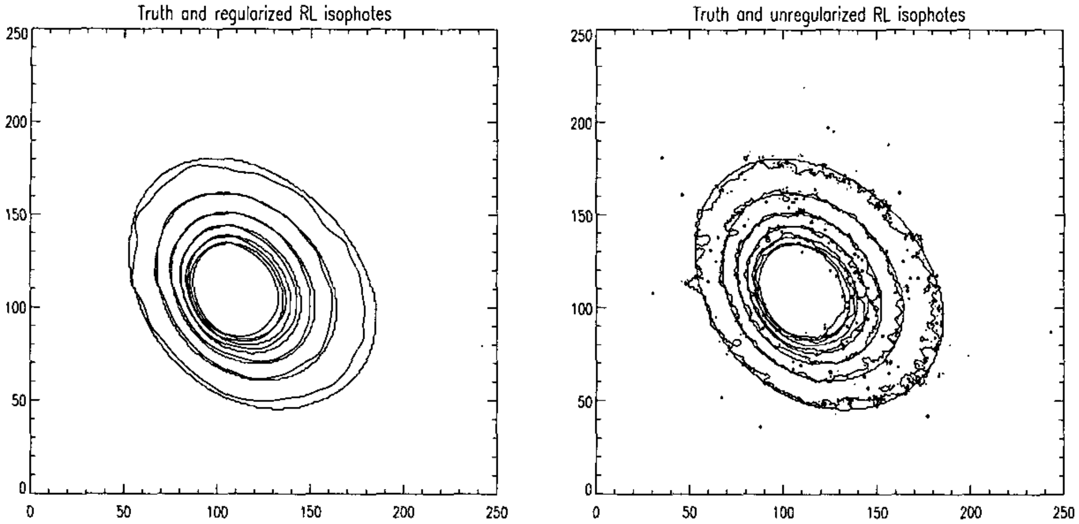

FIG. 4. Isophotal contours corresponding to (left) "truth image" and regularized Richardson–Lucy restoration; and (right) "truth image" and unregularized Richard–Lucy restoration.

center, ellipticity, and position angle. This image was convolved with a Hubble Space Telescope Wide Field Camera (WF/PC-1) PSF, and Poisson and readout noise (Gaussian) were added.

Under the assumption that the readout noise was small, we used a Poisson model for all noise in the image. We set negative values in the blurred, noisy input image to zero. This was the case in the background only and was necessitated by the Anscombe transformation used.

Figure 4 (left) shows contours formed in the truth image, overplotted with contours yielded by the regularized Richardson–Lucy method. Note that the truth image was not the one used as input for restoration; rather, it was the image on the basis of which the blurred, noisy input image was created. All contours in Fig. 4 relate to identical intensity values (4, 8, 12, 16, 20, 24). For the regularized restoration, a Poisson model was used for clipping wavelet coefficients. A 4σ threshold was chosen, above which (in wavelet space) a value was taken as significant. The multiresolution support algorithm was used, in order to prevent any untoward alteration to the galaxy. The plot shown in Fig. 4 (left) corresponds to just five iterations (unaccelerated) of the regularized Richardson–Lucy restoration method. Figure 4 (right) shows the same isophotes for the truth image, and those obtained by restoration following five iterations of the unregularized Richardson–Lucy method. Allowing further iterations (to convergence in the case of the regularized Richardson–Lucy method) yielded relatively similar results in the case of the regularized restoration, but in the case of the unregularized restoration, the fitting of a PSF to every noise spike made for a very unsmooth image.

5. CONCLUSION

The wavelet transform, and noise suppression strategies, must take properties of the input image into account. It may even be necessary to take into account aspects related to the physical nature of that which is imaged. We have studied the case of astronomical images and have proposed an effective framework for tackling problems related to restoration and filtering. As a byproduct, this framework also helps in object detection (and this is now the topic of ongoing work). The multiresolution support data structure is an important image processing tool.

The wavelet transform used could be replaced with some other multiresolution algorithm. However the *à trous* algorithm has acquitted itself well. The experimental results demonstrate the usefulness of this broad framework.

APPENDIX 1: THE "À TROUS" WAVELET TRANSFORM ALGORITHM

In a wavelet transform, a series of transformations of an image is generated, providing a resolution-related set of "views" of the image. The properties satisfied by a wavelet transform, and in particular the *à trous* wavelet transform ("with holes," so called because of the interlaced convolution used in successive levels: see step 2 of the algorithm below), are further discussed in [4]. Extensive literature exists on the wavelet transform and its applications [6, 9, 27, 28, 36]. The discrete *à trous* algorithm is described in [15, 37].

For the scaling function, the B-spline of degree 3 was used in our calculations. See [38, 39] for discussion of linear and other scaling functions. We have derived a simple algorithm in order to compute the associated wavelet transform:

1. We initialize j to 0 and we start with the data $c_j(k)$.
2. We increment j, and we carry out a discrete convolution of the data $c_{j-1}(k)$ using the filter h. The distance between the central pixel and those adjacent is 2^{j-1}.
3. After this smoothing, we obtain the discrete wavelet transform from the difference $c_{j-1}(k) - c_j(k)$.
4. If j is less than the number p of resolutions we want to compute, then go to step 2.
5. The set $\mathcal{W} = \{w_1, \ldots, w_p, c_p\}$ represents the wavelet transform of the data.

The above *à trous* algorithm has been discussed in terms of a single index, x, but is easily extendable to two-dimensional space. The use of the B_3 spline leads to a convolution with a mask of 5×5:

$$\begin{pmatrix} \frac{1}{256} & \frac{1}{64} & \frac{3}{128} & \frac{1}{64} & \frac{1}{156} \\ \frac{1}{64} & \frac{1}{16} & \frac{3}{32} & \frac{1}{16} & \frac{1}{64} \\ \frac{3}{128} & \frac{3}{32} & \frac{9}{64} & \frac{3}{32} & \frac{3}{128} \\ \frac{1}{64} & \frac{1}{16} & \frac{3}{32} & \frac{1}{16} & \frac{1}{64} \\ \frac{1}{256} & \frac{1}{64} & \frac{3}{128} & \frac{1}{64} & \frac{1}{256} \end{pmatrix}$$

In one dimension, this mask is $(\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16})$.

To facilitate computation, a simplification of this wavelet is to assume separability in the 2-dimensional case. In the case of the B_3 spline, this leads to a row by row convolution with $(\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16})$, followed by column by column convolution.

The most general way to handle the boundaries is to consider that $c(k + N) = c(N - k)$. But other methods can be used such as periodicity ($c(k + N) = c(k)$), or continuity ($c(k + N) = c(N)$). Choosing one of these methods has little influence on our general restoration strategy. We used continuity.

A series expansion of the original image, c_0, in terms of the wavelet coefficients is now given as follows. The final smoothed array $c_p(x)$ is added to all the differences w_j:

$$c_0(k) = c_p + \sum_{j=1}^{p} w_j(k). \tag{26}$$

This equation provides a reconstruction formula for the original image.

At each scale j, we obtain a set $\{w_j\}$ which we call a wavelet plane. This has the same number of pixels as the image (which therefore is a limitation on the use of this particular wavelet transform approach for image compression).

APPENDIX 2: FILTERING BASED ON POISSON NOISE

If the noise associated with image I is Poisson, the following transformation acts as if the data came from a Gaussian process with a noise of standard deviation 1, subject to a sufficiently large mean value of image I:

$$T(I(x, y)) = 2\sqrt{I(x, y) + \tfrac{3}{8}}. \tag{27}$$

Therefore the noise contained in $e^{(n)} = T(I) - T(I^{(n)})$ can be suppressed using the same principle as the suppression of noise in $E^{(n)} = I - I^{(n)}$. Image $e^{(n)}$ is decomposed into multiresolution coefficients (in the case of the multiresolution strategy), and only the significant coefficients, or the coefficients associated with the multiresolution support, are retained. The support is, of course, determined from $T(I)$ and not from I. Reconstruction then gives $\tilde{e}^{(n)}$. We have the relations

$$e^{(n)}(x, y) = T(I(x, y)) - T(I^{(n)}(x, y)) \tag{28}$$

$$E^{(n)}(x, y) = I(x, y) - I^{(n)}(x, y). \tag{29}$$

Hence we have

$$[T(I(x, y))]^2 = [e^{(n)}(x, y) + T(I^{(n)}(x, y))]^2 \tag{30}$$

$$= (e^{(n)}(x, y))^2 + 4(I^{(n)}(x, y) + \tfrac{3}{8}) + 4e^{(n)}(x, y)\sqrt{I^{(n)}(x, y) + \tfrac{3}{8}} \tag{31}$$

and

$$[T(I(x, y))]^2 = 4(I(x, y) + \tfrac{3}{8}. \tag{32}$$

From these two equations, we deduce that $I(x, y)$ can be expressed by

$$I(x, y) = \tfrac{1}{4}[(e^{(n)}(x, y))^2 + 4(I^{(n)}(x, y) + \tfrac{3}{8}) + 4e^{(n)}(x, y)\sqrt{I^{(n)}(x, y) + \tfrac{3}{8}]} - \tfrac{3}{8}. \tag{33}$$

Now, replacing I by its expression in $E^{(n)} = I(x, y) - I^{(n)}(x, y)$, we have

$$E^{(n)}(x, y) = e^{(n)}(x, y)\left[\frac{e^{(n)}(x, y)}{4} + \sqrt{I^{(n)}(x, y) + \tfrac{3}{8}}\right]. \tag{34}$$

Filtering $e^{(n)}$ by thresholding nonsignificant coefficients, or coefficients which are not contained in the support, we obtain $\tilde{e}^{(n)}$, and we then have

$$\tilde{E}^{(n)}(x, y) = \tilde{e}^{(n)}(x, y)\left[\frac{\tilde{e}^{(n)}(x, y)}{4} + \sqrt{I^{(n)}(x, y) + \tfrac{3}{8}}\right]. \tag{35}$$

In image restoration with a Poisson model, a similar analysis can be carried out. In this case, the right-hand side of Eq. (29) defines the residual image, $R^{(n)}(x, y)$. The right-hand side of Eq. (28) provides the image from which noise is suppressed.

While this appendix deals with Poisson noise, the case of combined Poisson and Gaussian noise (see Eq. (9)) is handled in a similar way.

APPENDIX 3: NOMENCLATURE

(x, y) Pixel; position in image (integers).

j Multiresolution level or scale (integer).

$M^{(I)}(j, x, y)$ Multiresolution support. Sequence of Boolean images. Boolean image at each scale or level j is of same dimensions as input image, with which the multiresolution support is associated. When nonambiguous, the superscript is not used.

c_j Result of convolving wavelet with image at level j.

w_j, $w_j(x, y)$ Wavelet coefficient. We have $w_j = c_j - 1 - c_j$. Wavelet coefficients at level j define a wavelet plane. A superscript on w is used to indicate the image with which the wavelet plane is associated, e.g., $w_j^{(I)}(x, y)$.

σ_j For a given image, proportional to the standard deviation of wavelet coefficients at scale j. See Section 2.3.2.

$\sigma_j(x, y)$ Standard deviation of wavelet coefficients at scale j and at location (x, y) in the case of multiple images.

σ_I Standard deviation of values in the image, I.

g A filtering function. See, e.g., Section 3.1.

I, $I(x, y)$ Image.

$\tilde{I}$ Filtered image.

E Error image. Difference between image and its filtered version.

T An element-wise image transformation, defined in Eq. (8) or (9).

ε Small convergence threshold constant.

k, K Constants.

O, $O(x, y)$ Observed, degraded image.

P, $P(x, y)$ Point spread function.

α Convergence parameter.

R, $R^{(n)}(x, y)$ Residual image at iteration n. See, e.g., Eq. (18).

$\overline{R}$ "Significant residual." See Eqs. (20) and (22).

$h(x)$ Low pass filter.

REFERENCES

1. H.-M. Adorf, HST image restoration—Recent developments, in *Science with the Hubble Space Telescope* (P. Benvenuti and E. Schreier, Eds.), pp. 227–238, European Southern Observatory, Garching bei München, 1992.
2. F. J. Anscombe, The transformation of Poisson, binomial and negative-binomial data, *Biometrika* **15**, 1948, 246–254.
3. A. Bijaoui, Algorithmes de la transformation en ondelettes. Applications en astronomie, *Ondelettes et Paquets d'Ondes* (P. J. Lions, Ed.), Cours CEA/EdF/INRIA, Rocquencourt, 1991.
4. A. Bijaoui, J.-L. Starck, and F. Murtagh, Restauration des images multi-échelles par l'algorithme à trous, *Trait. Signal* **11**, 1994, 229–243.
5. A. Caulet and W. Freudling, Distant galaxy cluster simulations—HST and groundbased, *ST-ECF Newsletter,* No. 20, 1993, pp. 5–7.
6. C. H. Chui, *Wavelet Analysis and its Application,* Academic Press, New York, 1992.
7. A. Cohen, I. Daubechies, and J. C. Feauveau, Biorthogonal bases of compactly supported wavelets, *Comm. Pure Appl. Math.* **45**, 1992, 485–560.
8. T. J. Cornwell, Image restoration, in *Proceedings, NATO Advanced Studies Institute on Diffraction-Limited Image with Very Large Telescopes, Cargese, 1988,* pp. 273–292.
9. I. Daubechies, Orthonormal bases of compactly supported wavelets, *Comm. Pure Appl. Math.* **41,** 1988, 909–916.
10. D. L. Donoho and I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Stanford University, Technical Report 400, 1993 (available by anonymous ftp from playfair.stanford.edu:/pub/donoho).
11. J. C. Feauveau, Analyse multirésolution par ondelettes non-orthogonales et bancs de filtres numériques, Thèse en Sciences de l'Université Paris Sud, 1990.
12. W. Freudling and A. Caulet, Simulated HST observations of distant clusters of galaxies, in *5th ESO/ST-ECF Data Analysis Workshop* (P. Grosbøl, Ed.), pp. 63–68, European Southern Observatory, Garching bei München, 1993.
13. N. P. Galatsanos and A. K. Katsaggelos, Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation, *IEEE Trans. Image Process.* 1, 1992, 322–336.
14. R. Hanisch (Ed.), *Restoration—Newsletter of STScI's Image Restoration Project,* Space Telescope Science Institute, Baltimore, 1993.
15. M. Holdschneider, R. Kronland-Martinet, J. Morlet, and Ph. Tchamitchian, A real-time algorithm for signal analysis with the help of the wavelet transform, in *Wavelets* (J. M. Combes, A. Grossmann, and Ph. Tchamitchian, Eds.), pp. 286–297, Springer-Verlag, Berlin, 1989.
16. M. G. Kang and A. K. Katsaggelos, Simultaneous iterative image restoration and evaluation of the regularization parameter, *IEEE Trans. Signal Process.* **40**, 1992, 2329–2334.
17. A. K. Katsaggelos, J. Biemond, R. W. Schafer, and R. M. Mersereau, A regularized iterative image restoration algorithm, *IEEE Trans. Signal Process.* **39**, 1991, 914–929.
18. A. K. Katsaggelos (Ed.), *Digital Image Restoration,* Springer-Verlag, New York, 1991.
19. L. Landweber, An iteration formula for Fredholm integral equations of the first kind, *Am. J. Math.* **73,** 1951, 615–624.
20. T. Lindeberg, *Scale-Space Theory in Computer Vision,* Kluwer, Dordrecht, 1994.
21. R. L. Lagendijk, J. Biemond, and D. E. Boekee, Regularized iterative image restoration with ringing reduction, *IEEE Trans. Acoust. Speech Signal Process.* **36,** 1988, 1874–1888.

22. J. Llacer and J. Núñez, Iterative maximum likelihood estimator and Bayesian algorithms for image reconstruction in astronomy, in *The Restoration of HST Images and Spectra* (R. L. White and R. J. Allen, Eds.), pp. 67–70, Space Telescope Science Institute, Baltimore, 1990.
23. H. Lorenz and G. M. Richter, Adaptive filtering of HST images: Preprocessing for deconvolution, in *Science with the Hubble Space Telescope* (P. Benvenuti and E. Schreier, Eds.), pp. 203–206, European Southern Observatory, Garching bei München, 1993.
24. L. B. Lucy, An iterative technique for the rectification of observed distributions, *Astron. J.* **79,** 1974, 745–754.
25. S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pattern Anal. Mach. Intell.* **11,** 1989, 674–693.
26. P. Meer, R.-H., Park, and K. Cho, Multiresolution adaptive image smoothing, *CVGIP: Graphical Models Image Process.* **56,** 1994, 140–148.
27. Y. Meyer, in *Ondelettes et Paquets d'Ondes* (P. J. Lions, Ed.), INRIA, Rocquencourt, 1991.
28. Y. Meyer, *Ondelettes: Algorithms et Applications,* Collection Acquis Avancés de l'Informatique, 1992.
29. Y. Meyer, *Ondelettes et Algorithmes Concurrents,* Hermann, Paris, 1992.
30. R. Molina, On the hierarchical Bayesian approach to image restoration. Applications to astronomical images, *IEEE Trans. Pattern Anal. Mach. Intell.* **16,** 1994, 1122–1128.
31. J. Morlet, G. Arens, E. Fourgeau, and D. Giard, Wave propagation and sampling theory I, II, *Geophysics* **47,** 1982, 203–236.
32. F. Murtagh, J.-L. Starck, and A. Bijaoui, Image restoration with noise suppression using a multiresolution support, *Astron. Astrophys.* 1995, in press.
33. G. P. Nason, The discrete wavelet transform in S, Version 2.1, 1993. (Software and documentation available from Statlib repository, http://lib.stat.cmu.edu/.)
34. W. H. Richardson, Bayesian-based iterative method of image restoration, *J. Opt. Soc. Am.* **62,** 1972, 55–59.
35. S. Ranganath, Image filtering using multiresolution representations, *IEEE Trans. Pattern Anal. Mach. Intell.* **13,** 1991, 426–440.
36. M. B. Ruskai, G. Beylkin, R. Coifman, I. Daubechies, S. Mallat, Y. Meyer, and L. Raphael, *Wavelets and their Applications,* Jones and Barlett, Boston, 1992.
37. M. J. Shensa, Discrete wavelet transforms: Wedding the à trous and Mallat algorithms, *IEEE Trans. Signal Process.* **40,** 1992, 2464–2482.
38. J. L. Starck, Ph.D. Thesis, Université de Nice-Sophia Antipolis, 1992.
39. J. L. Starck, in MIDAS Manual, Release 93NOV, Image Processing Group, European Southern Observatory, Garching, 1993.
40. J. L. Starck and A. Bijaoui, Filtering and deconvolution by the wavelet transform, *Signal Process.* **35,** 1994, 195–211.
41. J. L. Starck and F. Murtagh, Image restoration with noise suppression using the wavelet transform, *Astron. Astrophys.* **288,** 1994, 342–348.
42. J. L. Starck, Les Transformées Multiresolutions, technical report, CISI, Valbonne, France, 1994.
43. P. H. Van Cittert, Zum Einfluß der Spaltbreite auf die Intensitätsverteilung in Spektrallinien II, *Z. Phys.* **69,** 1931, 298–308.
44. R. L. White and R. J. Allen (Eds.), *The Restoration of HST Images and Spectra,* STScI, Baltimore, 1990.
45. Y. Xu, J. B. Weaver, D. M. Healy Jr., and J. Lu, Wavelet transform domain filters: A spatially selective noise filtration technique, *IEEE Trans. Image Process.* **3,** 1994, 747–758.