Non Linear Multiscale Transforms

J.-L. Starck

[1] Statistics Department, Stanford University,
Sequoia Hall, Stanford, CA 94305 USA
[2] DAPNIA/SEI-SAP, CEA-Saclay, F-91191 Gif-sur-Yvette Cedex, France

Abstract. We present in this paper several approaches to perform a non linear multiscale transform to an image. Both redundant, and non redundant transformations are discussed. The median based multiscale transforms and some of their applications are detailed. Finally we show how a multiscale vision model can used to decompose an image, and how several transformations can be combined in order to benefit of the advantages of each of them.

Keywords. Discrete Wavelet Transform, non linear transform, lifting scheme, combined transform, Filtering.

1 Introduction

Multiscale methods have become very popular with the development of the wavelets in last ten years. Background texts on the wavelet transform include [13,18,47,32,6,44]. The most used wavelet transform algorithm is certainly the decimated bi-orthogonal wavelet transform (OWT). Using the OWT, a signal s can be decomposed by:

$$s(l) = \sum_k c_{J,k} \phi_{J,l}(k) + \sum_k \sum_{j=1}^{J} \psi_{j,l}(k) w_{j,k} \tag{1}$$

with $\phi_{j,l}(x) = 2^{-j}\phi(2^{-j}x - l)$ and $\psi_{j,l}(x) = 2^{-j}\psi(2^{-j}x - l)$, where ϕ and ψ are respectively the scaling function and the wavelet function. J is the number of resolutions used in the decomposition, w_j the wavelet (or details) coefficients at scale j, and c_J is a coarse or smooth version of the original signal s. Thus, the algorithm outputs $J+1$ subband arrays. The indexing is such that, here, $j = 1$ corresponds to the finest scale (high frequencies). Coefficients $c_{j,k}$ and $w_{j,k}$ are obtained by means of the filters h and g:

$$\begin{aligned} c_{j+1,l} &= \sum_k h(k-2l) c_{j,k} \\ w_{j+1,l} &= \sum_k g(k-2l) c_{j,k} \end{aligned} \tag{2}$$

where h and g verify:

$$\begin{aligned} \frac{1}{2}\phi(\frac{x}{2}) &= \sum_k h(k)\phi(x-k) \\ \frac{1}{2}\psi(\frac{x}{2}) &= \sum_k g(k)\phi(x-k) \end{aligned} \tag{3}$$

and the reconstruction of the signal is performed with:

$$c_{j,l} = 2\sum_k [\tilde{h}(k+2l) c_{j+1,k} + \tilde{g}(k+2l) w_{j+1,k}] \tag{4}$$

where the filters $\tilde{h}$ and $\tilde{g}$ must verify the conditions of dealiasing and exact reconstruction:

$$\hat{h}(\nu + \frac{1}{2})\hat{\tilde{h}}(\nu) + \hat{g}(\nu + \frac{1}{2})\hat{\tilde{g}}(\nu) = 0$$
$$\hat{h}(\nu)\hat{\tilde{h}} + \hat{g}(\nu)\hat{\tilde{g}}(\nu) = 1 \tag{5}$$

The application of the OWT to image compression, using the 7-9 filters [2] and the zerotree coding [41,38] has lead to impressive results, compared to previous methods like JPEG. However some problems related to the OWT may impact on their use in some applications, include the following.

1. Edges representation: If the OWT performs better than the FFT to represent edges in an image, it is still not optimal. There is only a fixed number of directional elements independent of scales, and there is no highly anisotropic elements [11]. For instance, the Haar 2D wavelet transform is optimal to find features with a ratio $length/width = 2$, and a horizontal,vertical, or diagonal orientation.
2. Isotropic features representation: the 2D OWT [33] leads to a wavelet transform with three wavelet functions (at each scale there are three wavelet coefficient sub-images) which does not simplify the analysis and the interpretation of the wavelet coefficients. An isotropic transform seems more appropriate for images containing features or objects with no favored orientation (astronomical images, ...).
3. Negative values: By definition, the wavelet coefficient mean is null. Every time we have a positive structure at a scale, we have negative values surrounding it. These negative values often create artifacts during the restoration process, or complicate the analysis.
4. Point artifacts: For example, cosmic ray hits in optical astronomy can "pollute" all the scales of the wavelet transform. The wavelet transform is non-robust relative to such real or detector faults.
5. Integer Values: The OWT produce floating values which are not easy to handle for lossless image compression.

These problems have lead to the development of other multiscale representations. Some of them are still linear transforms, like the ridgelet transform or the curvelet transform [11,10]. (both of them have been designed in order to better represent the edges in an image), while some others are non linear. We will describe in this article the second class.

Section 2 introduces the non linear multiscale transforms, in particular the lifting scheme [49] approach, which generalizes the standard filter bank decomposition. Using the lifting scheme, non linearity can be introduced in a strait-forward way, allowing us to perform an integer wavelet transform, or a wavelet transform on a irregular sampled grid. Section 3 presents the median based multiscale representations [21,46] which discriminates well outliers in the data (non Gaussian noise, pixels with high intensity values, ...). Sections 4 and 5 show how a signal can decomposed into several components, either using a Multiscale Vision Model, or by combining several transforms.

2 Multiscale Image Decomposition

2.1 The Lifting Scheme

The lifting scheme [49] is a flexible technique that has been used in several different settings, for easy construction and implementation of traditional wavelets [49], and of second generation wavelets [48] such as spherical wavelets [39].

Its principle is to compute the difference between a true coefficient and its prediction:

$$w_{j+1,l} = c_{j,2l+1} - \mathcal{P}(c_{j,2l-2L}, ..., c_{j,2l-2}, c_{j,2l}, c_{j,2l+2}, ..., c_{j,2l+2L}) \tag{6}$$

A pixel at an odd location 2l+1 is then predicted using pixels at even locations.

The transformation is done in three steps:

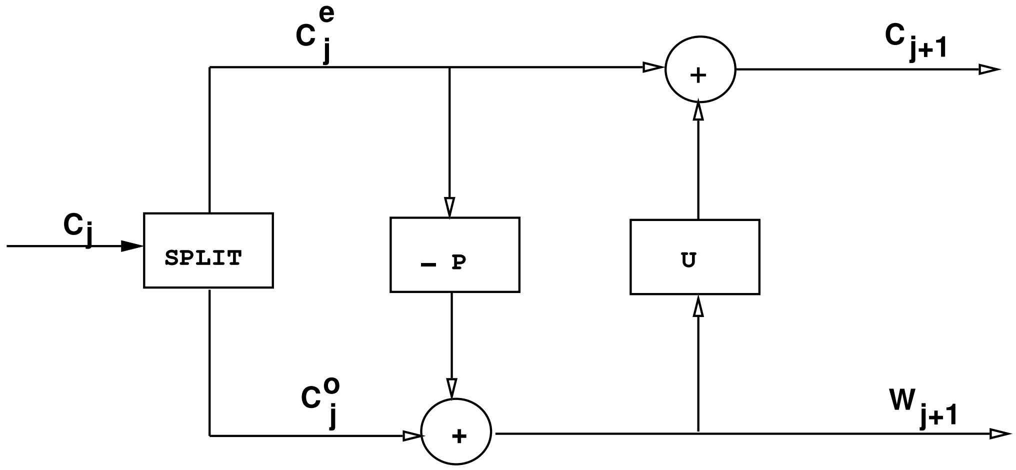

Fig. 1. The lifting scheme – forward direction.

1. **Split:** Split the signal into even and odd number samples:

$$\begin{aligned} c^e_{j,l} &= c_{j,2l} \\ c^o_{j,l} &= c_{j,2l+1} \end{aligned} \tag{7}$$

2. **Predict:** Calculate the wavelet coefficient $w_{j+1,l}$ as the prediction error of $c^o_{j,l}$ from $c^e_{j,l}$ using the prediction operator $\mathcal{P}$:

$$w_{j+1,l} = c^o_{j,l} - \mathcal{P}(c^e_{j,l}) \tag{8}$$

3. **Update:** The coarse approximation c_{j+1} of the signal is obtained by using $c^e_{j,l}$ and $w_{j+1,l}$ and the update operator $\mathcal{U}$:

$$c_{j+1,l} = c^e_{j,l} + \mathcal{U}(w_{j+1,l}) \tag{9}$$

The lifting steps are easily inverted by:

$$\begin{aligned} c_{j,2l} &= c^e_{j,l} = c_{j+1,l} - \mathcal{U}(w_{j+1,l}) \\ c_{j,2l+1} &= c^o_{j,l} = w_{j+1,l} + \mathcal{P}(c^e_{j,l}) \end{aligned} \tag{10}$$

Some examples of wavelet transform via the lifting scheme are:

- Haar wavelet via lifting: the Haar transform can be performed via the lifting scheme by taking the predict operator equal to the identity, and an update operator which halves the difference. The transform becomes:

$$\begin{aligned} w_{j+1,l} &= c^o_{j,l} - c^e_{j,l} \\ c_{j+1,l} &= c^e_{j,l} + \frac{w_{j+1,l}}{2} \end{aligned}$$

  All computation can be done in place.
- Linear wavelets via lifting: the identity predictor used before is correct when the signal is constant. In the same way, we can use a linear predictor which is correct when the signal is linear. The predictor and update operators are now:

$$\begin{aligned} \mathcal{P}(c^e_{j,l}) &= \frac{1}{2}(c^e_{j,l} + c^e_{j,l+1}) \\ \mathcal{U}(w_{j+1,l}) &= \frac{1}{4}(w_{j+1,l-1} + w_{j+1,l}) \end{aligned}$$

It is easy to verify that:

$$c_{j+1,l} = -\frac{1}{8}c_{j,2l-2} + \frac{1}{4}c_{j,2l-1} + \frac{3}{4}c_{j,2l} + \frac{1}{4}c_{j,2l+1} - \frac{1}{8}c_{j,2l+2}$$

which is the bi-orthogonal Cohen-Daubechies-Feauveau [16] wavelet transform.

- 9/7 filters: the lifting factorization of popular (9/7) filter pair leads to the following implementation [20]:

$$\begin{aligned}
s_l^{(0)} &= c_{j,2l} \\
d_l^{(0)} &= c_{j,2l+1} \\
d_l^{(1)} &= d_l^{(0)} + \alpha(s_l^{(0)} + s_{l+1}^{(0)}) \\
s_l^{(1)} &= s_l^{(0)} + \beta(d_l^{(1)} + d_{l-1}^{(1)}) \\
d_l^{(2)} &= d_l^{(1)} + \gamma(s_l^{(1)} + s_{l+1}^{(1)}) \\
s_l^{(2)} &= s_l^{(1)} + \delta(d_l^{(2)} + d_{l-1}^{(2)}) \\
c_{j+1,l} &= u s_l^{(2)} \\
c_{j+1,l} &= u d_l^{(2)}
\end{aligned} \tag{11}$$

with

$$\begin{aligned}
\alpha &= -1.586134342 \\
\beta &= -0.05298011854 \\
\gamma &= 0.8829110762 \\
\delta &= 0.4435068522 \\
u &= 1.149604398
\end{aligned} \tag{12}$$

Every wavelet transform can be written via lifting.

Integer wavelet transform When the input data are integer values, the wavelet transform no longer consists of integers. For lossless coding, it is useful to have a wavelet transform which produces integer values. We can build an integer version of every wavelet transform [8]. For instance, denoting $\lfloor x \rfloor$ as the largest integer not exceeding x, the integer Haar transform (also called "S" transform) can be calculated by:

$$\begin{aligned}
w_{j+1,l} &= c_{j,l}^o - c_{j,l}^e \\
c_{j+1,l} &= c_{j,l}^e + \lfloor \frac{w_{j+1,l}}{2} \rfloor
\end{aligned} \tag{13}$$

while the reconstruction is

$$\begin{aligned}
c_{j,2l} &= c_{j+1,l} - \lfloor \frac{w_{j+1,l}}{2} \rfloor \\
c_{j,2l+1} &= w_{j+1,l} + c_{j,2l}
\end{aligned} \tag{14}$$

More generally, the lifting operators for an integer version of the wavelet transform are:

$$\begin{aligned}
\mathcal{P}(c_{j,l}^e) &= \lfloor \sum_k p_k c_{j,l-k}^e + \frac{1}{2} \rfloor \\
\mathcal{U}(w_{j+1,l}) &= \lfloor \sum_k u_k w_{j+1,l-k} + \frac{1}{2} \rfloor
\end{aligned} \tag{15}$$

The linear integer wavelet transform is

$$
\begin{aligned}
w_{j+1,l} &= c^o_{j,l} - \lfloor \frac{1}{2}(c^e_{j,l} + c^e_{j,l+1}) + \frac{1}{2} \rfloor \\
c_{j+1,l} &= c^e_{j,l} + \lfloor \frac{1}{4}(w_{j+1,l-1} + w_{j+1,l}) + \frac{1}{2} \rfloor
\end{aligned} \tag{16}
$$

Even if there is no filter that consistently performs better than all the other filters on all images, the linear integer wavelet transform performs generally better than others [8]: More filters can be found in [8].

Wavelet transform on irregular grid A wavelet transform on irregular samples can be performed by introducing weighted inner products [49,19], assuming that we are given weighted averages of some unknown function over interval:

$$
c_{0,l} = \frac{1}{I_{0,l}} \int_{x_l}^{x_{l+1}} u(x) f(x) dx \tag{17}
$$

where $[x_l, x_{l+1}]$ defines an interval, $u(x)$ is a positive weighting function, and $I_{0,l}$ is

$$
I_{0,l} = \int_{x_l}^{x_{l+1}} u(x) dx \tag{18}
$$

In its simplest form, $u(x)$ can take the value 1 when a sample at position x is available, and 0 otherwise. If error measurements are also available, $u(x)$ can be derived from them.

In the case of the Haar wavelet transform, it leads to the unbalanced Haar transform which is obtained by [49]:

$$
\begin{aligned}
I_{j+1,l} &= I_{j,2l} + I_{j,2l+1} \\
w_{j+1,l} &= c^o_{j,l} - c^e_{j,l} \\
c_{j+1,l} &= c^e_{j,l} + \frac{I_{j,2l+1}}{I_{j+1,l}} w_{j+1,l}
\end{aligned} \tag{19}
$$

Other kind of wavelets can be performed in a similar way. See [49] for more details.

Adaptive Wavelet Transform Adaptativity can be introduced in the wavelet transform by reversing the order of the predict and the update steps in the lifting scheme [14]. When the update operator is first applied, the prediction is based on the low-pass coefficients that are computed as in the standard wavelet transform. In the update-first approach, as illustrated in Figure 2, the detail coefficients are not in the loop for calculating the coefficients at a coarser scale. Hence, we can start the prediction process at the coarser scale, and working from coarse to fine scales. The idea is now to make the predictor data dependent. The prediction operator is chosen, based on the local properties of the data at a coarser scale. If an edge is detected, the order of the predictor is reduced, while if the data are smooth, a larger order is preferred.

2.2 Multiscale Morphological Transform

Mathematical Morphology Originally developed by Matheron [35,36] and Serra [40], the mathematical morphology is based on two operators: the *infimum* (denoted $\wedge$) and the *supremum* (denoted $\vee$). The infimum of a set of images is defined as the greatest lower bound while the *supremum* is defined as the least upper bound. The basic morphological transformations are erosion, dilation, opening and closing. For gray-level images, they can be defined in the following way:

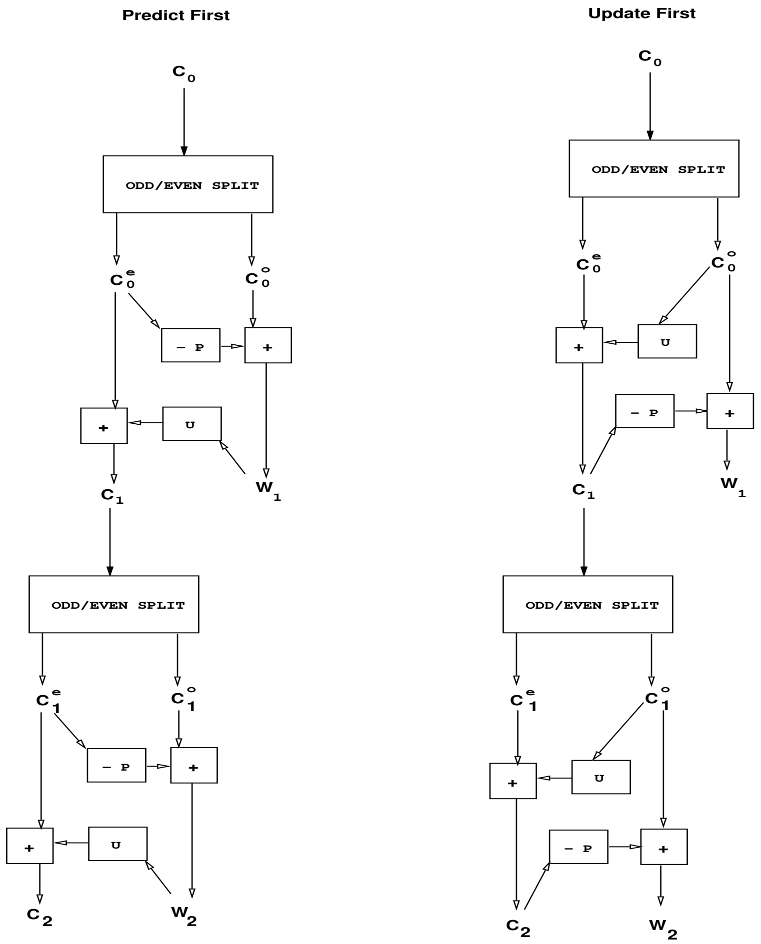

Fig. 2. The lifting scheme with two iterations: left, predict-first approach, and right, update-first approach.

- *erosion* consists of replacing each pixel of an image by the minimum of its neighbors:

$$\epsilon_B(f) = \bigwedge_{b \in B} f_{-b}$$

  where f stands for the image, and B denote the structuring element, typically a small convex set such a square or disc.
- *dilation* consists of replacing each pixel of an image by the maximum of its neighbors.

$$\delta_B(f) = \bigvee_{b \in B} f_b$$

- *opening* consists of doing an erosion followed by a dilations.

$$\alpha_B = \delta_B \epsilon_B \text{ and } \alpha_B(f) = f \circ B$$

- *closing* consists of doing a dilation followed by an erosion.

$$\beta_B = \epsilon_B \delta_B \text{ and } \beta_B(f) = f \bullet B$$

In a more general way, *opening* and *closing* referred to morphological filters which respect some specific properties [5].

Mathematical morphology have been until now considered as another way to analyze data, in competition with linear methods. But from a multiscale point of view [44,25,27], mathematical morphology or linear methods are just filters allowing us to go from a resolution to a coarser one, and the multiscale coefficients are then analyzed in the same way.

Undecimated Morphological Transform By choosing a set of structuring elements B_j having a size increasing with j, we can define an undecimated morphological multiscale transform by

$$\begin{aligned} c_{j+1,l} &= \mathcal{M}_j(c_j)(l) \\ w_{j+1,l} &= c_{j,l} - c_{j+1,l} \end{aligned} \tag{20}$$

where $\mathcal{M}_j$ is a morphological filter (erosion, opening, ...) using the structuring element B_j. An example of B_j is a box of size $(2^j + 1) \times (2^j + 1)$. As the detail signal w_{j+1} is obtained by calculating a simple difference between the c_j and c_{j+1}, the reconstruction is straitforward, and is identical to the reconstruction relative to the "à trous" wavelet transform [44]. An exact reconstruction of the image c_0 is obtained by:

$$c_{0,l} = c_{J,l} + \sum_{j=1}^{J} w_{j,l} \tag{21}$$

where J is the number of scales used in the decomposition. Each scale has the same number N of samples as the original data. The total number of pixels in the transformation is $(J+1)N$. Such a decomposition is obviously not dedicated to image compression. It is however interesting for the following reasons:

- The computational requirement is reasonable.
- The reconstruction algorithm is trivial.
- The transform is known at each pixel, allowing position detection without any error, and without interpolation.
- We can follow the evolution of the transform from one scale to the next.
- Invariance under translation is completely verified.
- The transform is isotropic (if the structuring element B is isotropic).

The last point is important if the image contain isotropic features, like cells in biologic images, or stars in astronomical images.

Pyramidal Transform A decimation can be introduced in order produce a pyramidal transform similar to the Burt and Adelson pyramidal [1], but by using morphological filters. We introduce two operators $\beta^{\downarrow}$, and $\beta^{\uparrow}$: $\beta^{\downarrow}$ (analysis operator) consists in filtering first the image and then to decimate it, and $\beta^{\uparrow}$ (synthesis operator) consists in interpolating the data.

$$\begin{aligned} c_{j+1} &= \beta^{\downarrow}(c_j) \\ w_{j+1} &= c_j - \beta^{\uparrow}(c_{j+1}) \end{aligned} \tag{22}$$

The reconstruction is obtained by:

$$c_j = w_{j+1} + \beta^{\uparrow}(c_{j+1}) \tag{23}$$

An example of operators is:

$$\begin{aligned} \beta^{\downarrow}(c_j)(l) &= c_{j,2l} \wedge c_{j,2l+1} \\ \beta^{\uparrow}(c_j)(2l) &= \beta^{\uparrow}(c_j)(2l+1) = c_{j+1,l} \end{aligned} \tag{24}$$

$$\tag{25}$$

In this scheme, $\beta^{\downarrow}$ and $\beta^{\uparrow}$ can be both linear or non linear. For example, choosing

$$\begin{aligned} \beta^{\downarrow}(c_j)(l) &= c_{j,2l} \wedge c_{j,2l+1} \\ \beta^{\uparrow}(c_j)(2l) &= c_{j+1,2l} \\ \beta^{\uparrow}(c_j)(2l+1) &= \frac{1}{2}(c_{j+1,l} + c_{j+1,l+1}) \end{aligned} \tag{26}$$

leads to a morphological analysis operator, and a linear synthesis operator.
As for the Burt and Adelson pyramid, the redundancy factor, after transformation of the image, is $\frac{4}{3}N$ where N is the number of pixels.

Non Redundant Multiscale Transform A non redundant multiscale morphological transform can easily be built via the lifting scheme, by introducing morphological operators. For example, the Haar morphological transform, also named "G" transform, is:

$$\begin{aligned} w_{j+1,l} &= c^o_{j,l} - \mathcal{P}(c^e_{j,l}) \\ c_{j+1,l} &= c^e_{j,l} + \mathcal{U}(w_{j+1,l}) \end{aligned} \tag{27}$$

where

$$\begin{aligned} \mathcal{P}(c^e_{j,l}) &= c_{j,2l} \\ \mathcal{U}(w_{j+1,l}) &= 0 \wedge w_{j+1,l} \end{aligned} \tag{28}$$

It is easy to verify that:

$$\begin{aligned} w_{j+1,l} &= c_{j,2l+1} - c_{j,2l} \\ c_{j+1,l} &= c_{j,2l} + (0 \wedge (c_{j,2l+1} - c_{j,2l})) = c_{j,2l} \wedge c_{j,2l+1} \end{aligned} \tag{29}$$

The Multiscale Operators Goutsias [25,27] has recently proposed a general multiscale decomposition scheme, by means of the *signal analysis operator* $\beta^{\downarrow}$ and the *detail analysis operator* $\gamma^{\downarrow}$:

$$\begin{aligned} w_{j+1,l} &= \gamma^{\downarrow}(c_j)(l) \\ c_{j+1,l} &= \beta^{\downarrow}(c_j)(l) \end{aligned} \tag{30}$$

and the reconstruction is done by means of the *synthesis operator* $\Psi^{\uparrow}$:

$$c_j = \Psi^{\uparrow}(c_{j+1}, w_{j+1}) \tag{31}$$

The perfect reconstruction condition is [27]:

$$c_{j,l} = \Psi^{\uparrow}(\beta^{\downarrow}(c_j), \gamma^{\downarrow}(c_j))(l) \tag{32}$$

We distinguish the case of the *uncoupled multiscale decomposition* when the operator $\Psi^{\uparrow}$ verifies the condition:

$$\Psi^{\uparrow}(c_{j+1}, w_{j+1}) = \beta^{\uparrow}(c_{j+1}) + \gamma^{\uparrow}(w_{j+1}) \tag{33}$$

where $\beta^{\uparrow}$ is the *signal synthesis operator*, and $\gamma^{\uparrow}$ is the *detail synthesis operator*.

Furthermore, the transformation is not redundant if the two following conditions are verified [27]:

$$\begin{cases} \beta^{\downarrow}_j(\Psi^{\uparrow}_j(c,w)) = c & \text{if } c \in V_{j+1} \text{ and } w \in W_{j+1} \\ \gamma^{\downarrow}_j(\Psi^{\uparrow}_j(c,w)) = w & \text{if } c \in V_{j+1} \text{ and } w \in W_{j+1} \end{cases}$$

where V_{j+1} and W_{j+1} are respectively the signal and the detail space at resolution level j.

For example, the undecimated morphological multiscale transform corresponds to the uncoupled multiscale decomposition, where

$$\begin{cases} \beta^{\downarrow} = \mathcal{M}_j \\ \gamma^{\downarrow} = Id - \beta^{\downarrow} \\ \beta^{\uparrow} = Id \\ \gamma^{\uparrow} = Id \end{cases} \tag{34}$$

and the standard bi-orthogonal decimated wavelet transform to:

$$\begin{cases} \beta^{\downarrow}(l) = \sum_k h(k-2l) c_{j,k} \\ \gamma^{\downarrow}(l) = \sum_k g(k-2l) c_{j,k} \\ \beta^{\uparrow}(l) = 2\sum_k \tilde{h}(k+2l) c_{j+1,k} \\ \gamma^{\uparrow}(l) = 2\sum_k \tilde{g}(k+2l) w_{j+1,k} \end{cases} \tag{35}$$

Many morphological filters can be designed using the previously described scheme, but no solution has been proposed until now for choosing the non linear filters for a given application.

3 Median Based Multiscale Transforms

3.1 Multiscale Median Transform

The median transform is nonlinear, and offers advantages for robust smoothing (i.e. the effects of outliers pixel values are mitigated). Define the median transform of image f, with square kernel of dimensions $n \times n$, as $med_n(f)$. Let $n = 2s + 1$; initially $s = 1$ or 2. The iteration counter will be denoted by j, and p is the user-specified number of resolution scales. the Multiscale Median Transform (MMT) algorithm is:

1. Let $c_j = f$ with $j = 0$
2. Determine $c_{j+1} = med_{2s+1}(c_j)$.
3. The multiresolution coefficients w_{j+1} are defined as: $w_{j+1} = c_j - c_{j+1}$.
4. Let $j \longleftarrow j + 1$; $s \longleftarrow 2s$. Return to Step 2 if $j < J$.

A straightforward expansion formula for the original image is given by:

$$f = c_J + \sum_{j=1}^{J} w_j \tag{36}$$

where c_J is the residual image.

In Step 3, the set of resolution levels associated with s lead to a dyadic analysis (i.e. the resolution is degraded in steps which are increasing integer exponents of 2). Other possibilities involving intermediate scales (e.g. $s \longleftarrow 1.5s$) can also be considered.

The multiresolution coefficient values, w_j, are evidently not necessarily of zero mean, and so the potential artifact-creation difficulties related to this aspect of wavelet transforms do not arise. Note of course that values of w can be negative.

The multiscale median transform corresponds to the uncoupled multiscale decomposition, where

$$\begin{cases} \beta^{\downarrow} = Med_{2^j+1} \\ \gamma^{\downarrow} = Id - \beta^{\downarrow} \\ \beta^{\uparrow} = Id \\ \gamma^{\uparrow} = Id \end{cases} \tag{37}$$

For integer image input values, this transform can be carried out in integer arithmetic only which may lead to computational savings.

Figure 3 shows the galaxy NGC2997, and figure 4 its multiscale decomposition by the multiscale median transform, with 6 scales (5 details images and the smoothed image). The coaddition of the six images reproduce exactly the original image. We can see that even the brightest stars have do not "pollute" the largest scales (from 4 to 6). This would not be true with a wavelet transform. Another point to notice is that there is no negative ring around the star in the first scales. This point is of great importance, as discussed in the next example.

Figure 3.1 shows the comparison between the "à trous" wavelet transform (on the left) and the MMT (on the right) of the input signal displayed in Figure 5. In these diagrams, the scale is represented as a function of the time. The data were obtained with an infrared camera, on board of the ISO satellite (Infrared Space Observatory). A faint source is observed between pixels 120 to 150. Cosmic ray impacts appears around pixels 50 and 230. The second one has introduced a variation in the detector gain. Due to this kind of cosmic ray impacts (glitches), sources are not easy to identified. Figure 3.1 shows the decomposition of the signal with both the undecimated wavelet transform (UWT), and the MMT. The glitch and the source are clearly separated in the MMT and they merge in the UWT. Therefore, the MMT allows us a better separation between sources and glitches.

Computational requirements of the MMT are high, and these can be reduced by decimation: one pixel out of two is retained at each scale. Here the transform kernel does not change from one iteration to the next, but the image to which this transform is applied does. This pyramidal algorithm is looked at next.

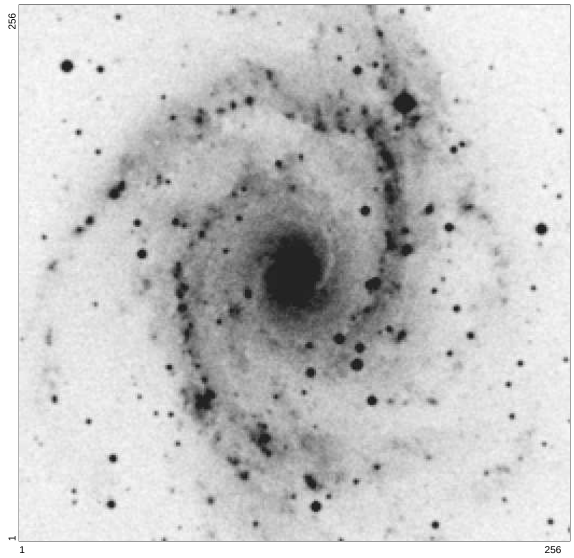

Fig. 3. Galaxy NGC2997.

3.2 Pyramidal Median Transform

The Pyramidal Median Transform (PMT) is obtained by the following algorithm:

1. Let $c_j = f$ with $j = 1$.
2. Determine $c^*_{j+1} = med_{2s+1}(c_j)$ with $s = 1$.
3. The pyramidal multiresolution coefficients w_{j+1} are defined as: $w_{j+1} = c_j - c^*_{j+1}$.
4. Let $c_{j+1} = dec(c^*_{j+1})$ where the decimation operation, dec, entails 1 pixel replacing each 2×2 sub-image.
5. Let $j \longleftarrow j + 1$. Return to Step 2 so long as $j < J$.

Here the kernel or mask of dimensions $(2s + 1) \times (2s + 1)$ remains the same during the iterations. The image itself, to which this kernel is applied, becomes smaller.

While this algorithm aids computationally, the reconstruction formula 36 is no longer valid. Instead we use the following algorithm based on B-spline interpolation:

1. Take the lowest scale image, c_j.
2. Interpolate c_j to determine the next resolution image (of twice the dimensionality in x and y). Call the interpolated image c'_j.
3. Calculate $c_{j-1} \longleftarrow c'_j + w_j$.
4. Set $j \longleftarrow j - 1$. Go to Step 2 if $j > 0$.

This reconstruction procedure takes account of the pyramidal sequence of images containing the multiresolution transform coefficients, w_j.

It presupposes, though, that a good reconstruction is possible. We ensure that by use of the following refined version of the Pyramidal Median Transform. Using iteration, the coefficients, $w_{j+1} = c_j - c_{j+1}$, are improved relative to their potential for reconstructing the input image.

3.3 Iterative Pyramidal Median Transform

An iterative scheme can be proposed for reconstructing an image, based on pyramidal multi-median transform coefficients. Alternatively, the PMT algorithm, itself, can be enhanced to allow for better estimates of coefficient values. The following is an iterative algorithm for this objective:

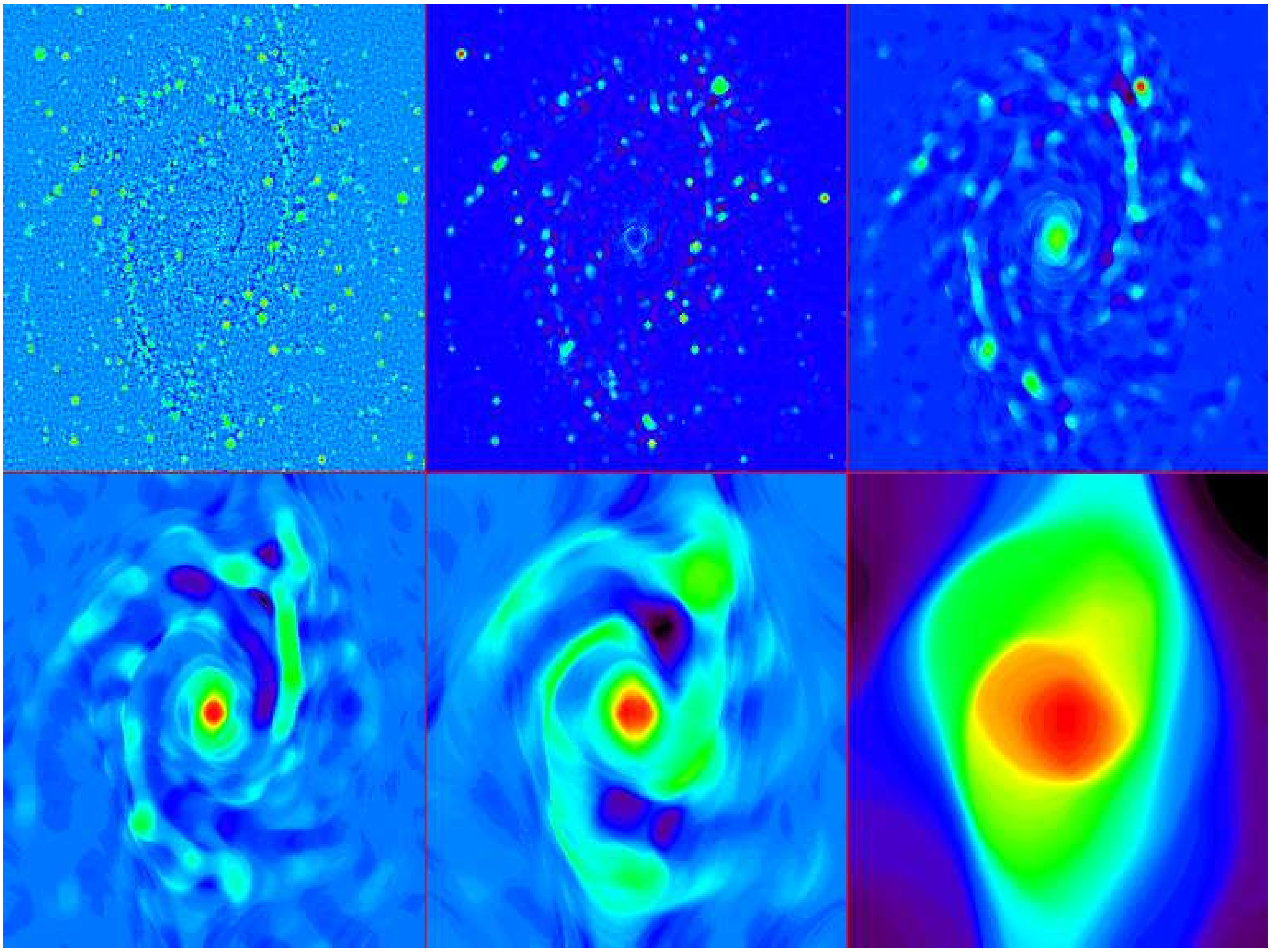

Fig. 4. Multiresolution median transform of NGC2997.

1. $i \longleftarrow 0$. Initialize f^i with the given image, f. Initialize the multiresolution coefficients at each scale j, w_j^f, to 0.
2. Using the Pyramidal Median Transform, determine the set of transform coefficients, $w_j^{f_i}$.
3. $w_j^f \longleftarrow w_j^f + w_j^{f_i}$.
4. Reconstruct image f_{i+1} from w_j^f (using the interpolation algorithm described in the previous section).
5. Determine the image component which is still not reconstructible from the wavelet coefficients: $f_{i+1} \longleftarrow f - f_{i+1}$.
6. Set $i \longleftarrow i + 1$, and return to Step 2.

The number of iterations is governed by when f_{i+1} in Step 5 approaches a null image. Normally 4 or 5 iterations suffice. Note that the additivity of the wavelet coefficients in Step 3 is justified by additivity of the image decomposition in Step 5 and the reconstruction formula used in Step 4, both of which are based on additive operations.

3.4 Non-Iterative Pyramidal Transform with Exact Reconstruction

A non-iterative version of the pyramidal median transform can be performed by decimating and interpolating the median images during the transform:

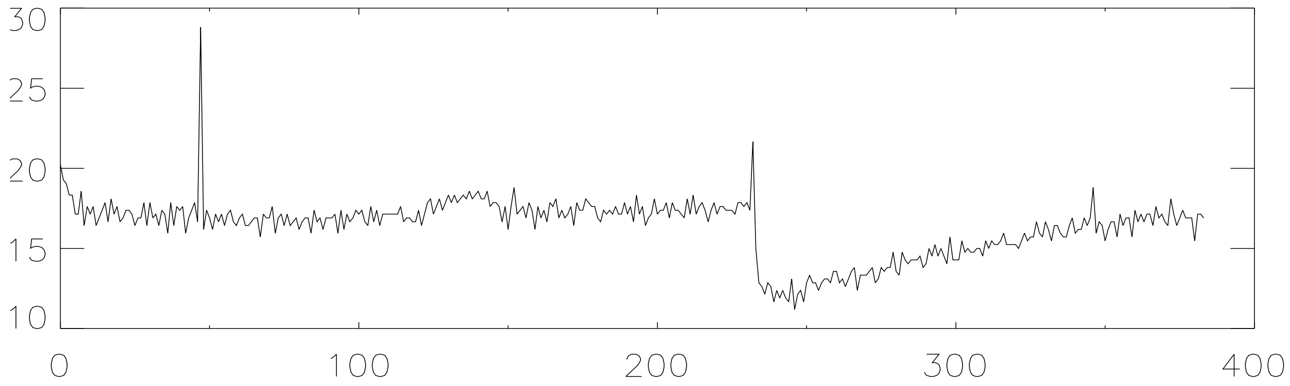

Fig. 5. Example of signal obtained with the ISOCAM infrared camera. A source is observed between pixels 120 to 150. Cosmic ray impacts appears around pixels 50 and 230. The second one has introduced a variation in the detector gain. Due to this kind of cosmic ray impacts (glitches), sources are not easy to identified.

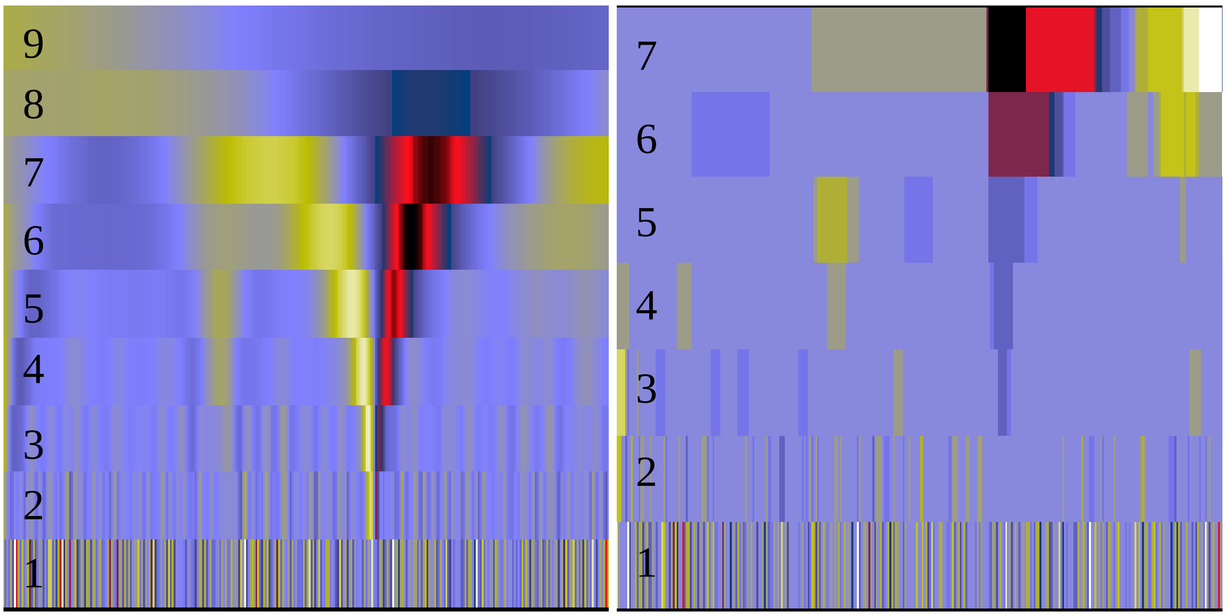

Fig. 6. Comparison between the “à trous” wavelet transform (left) and the multiresolution median transform (right) of above signal. Resolution scale is represented versus the time. We note that the separation between the source and the glitch is improved using the MMT.

1. Let $c_j = f$ with $j = 1$.
2. Determine $c_{j+1} = dec[med_{2s+1}(c_j)]$
3. Determine c^*_{j+1} = interpolation of c_{j+1} to size of c_j
4. The pyramidal multiresolution coefficients w_{j+1} are defined as: $w_{j+1} = c_j - c^*_{j+1}$.
5. Let $j \longleftarrow j + 1$. Return to Step 2 so long as $j < J$.

This saves computation time in two ways. First, there is no need to iterate. Secondly, in step (2) one does not really calculate the median for all pixels and then decimate it; rather, one just calculates the median for the pixels to be left after decimation. Thus the medians are 4 times fewer. This algorithm is very close to the Laplacian pyramid developed by Burt and Adelson [7]. The reconstruction algorithm is the same as before, but the reconstructed image has no error. In the following, we will indicate this version by referring to PMT.

The PMT corresponds to the uncoupled multiscale decomposition:

$$\begin{cases} \beta^{\downarrow} = Med^{\downarrow}_{2j+1} \\ \gamma^{\downarrow} = Id - \beta^{\uparrow}\beta^{\downarrow} \\ \gamma^{\uparrow} = Id \\ \beta^{\uparrow} = \mathcal{I} \end{cases} \tag{38}$$

where $Med^{\downarrow}_{2j+1}$ is the median operator followed by a decimation, and $\mathcal{I}$ is an interpolation operator. A bilinear interpolation can be used [46,44], or an interpolation based on the morphological operator $\vee$ [25]:

$$\mathcal{I}(c)(2l) = c(l) \text{ and } \mathcal{I}(c)(2l+1) = c(l) \vee c(l+1)$$

We note in passing one particularly interesting use of the PMT: the final scale often provides a good estimate of the image background. This is especially so for images containing small structures, e.g. astronomical wide-field images.

3.5 Merging Wavelets and the PMT

One of the advantages of the pyramidal wavelet transform over the pyramidal median transform (PMT) is the ability to have robust noise estimation in the different scales, while the advantage of the PMT is a better separation of the structures in the scales. Using the PMT, a strong structure (like a bright star, cosmic rays, bad pixels, etc.) will not be spread over all scales as when using a wavelet transform. In fact, when there is no signal in a given region, a wavelet transform would be better, and if a strong signal appears, it is the PMT that we would like to use. So the idea arises to try to merge both transforms, and to adapt the analysis at each position and at each scale, depending on the amplitude of the coefficient we measure.

A possible algorithm to perform this on an image I is the following:

1. Let $c_j = I$, and $j = 0$.
2. Filter the image c_j by the median: we get m
3. Set $c^* = c_j$ and $d = c_j - m$
4. For all pixels k do
   if $\mid d(k) \mid > k\sigma_j$ then $c^*(k) = m(k)$
5. Smooth c^* by a B_3-spline: we get b
6. Set $w_{j+1} = c_j - b$
7. $c_{j+1} = dec(b)$ (where the decimation operation, dec, entails 1 pixel replacing each 2 $\times$2 sub-image). Let $j = j + 1$ and return to step 2 if $j < J$ (J being the number of scales).

In this algorithm, the linear filtering relative to the wavelet transform is not applied to the strong features contained in the image. Indeed, significant pixel values are detected at step 4, and are replaced by the median. Regions containing no bright object are treated as if the pyramidal wavelet transform is used. The parameter k used at step 4 must be large enough in order to be sure that noise is not filtered by the median ($k = 5$ seems high enough). As this transform merges the wavelet transform and the PMT, we call it the PMWT transform. PMWT takes more time than the PMT, but this algorithm has the advantages of the PMT without the drawbacks. The reconstruction is the same as for the PMT.

3.6 Non Redundant Multiscale Median Transform

A non redundant multiscale median transform has been proposed independently by Egger and Li [23], and Florêncio and Schafer [24], which can easily be described using the lifting scheme by:

$$\begin{aligned} \mathcal{P}(c^e_{j,l}) &= med_N(c^e_j)(l) \\ \mathcal{U}(w_{j+1,l}) &= 0 \end{aligned} \tag{39}$$

It has also been suggested [26,15] that the update operator could also be defined using the median operator. This leads to the following equations:

$$\begin{aligned}\mathcal{P}(c^e_{j,l}) &= med_N(c^e)(l) \\ \mathcal{U}(w_{j+1,l}) &= med_{\tilde{N}}(w_{j+1})(l)\end{aligned} \tag{40}$$

$N = \tilde{N} = 4$ was used in [26], and $N = 5$ and $\tilde{N} = 3$ in [15].

3.7 Example 1 : filtering

Figure 7 shows the results of the MMT-multiscale entropy method on simulated data (2048 pixels). From top to bottom, the simulated data, the noisy data, the filtered data, and both noisy and filtered data overplotted.

The multiscale entropy method [45,43] consists of measuring the information h relative to wavelet coefficients, and of separating this into two parts h_s, and h_n. The expression h_s is called the signal information and represents the part of h which is certainly not contaminated by the noise. The expression h_n is called the noise information and represents the part of h which may be contaminated by the noise. We have $h = h_s + h_n$. Following this notation, the corrected coefficient $\tilde{w}$ should minimize:

$$J(\tilde{w}_j) = h_s(w_j - \tilde{w}_j) + \alpha h_n(\tilde{w}_j) \tag{41}$$

i.e. there is a minimum of information in the residual $(w - \tilde{w})$ which can be due to the significant signal, and a minimum of information which could be due to the noise in the solution $\tilde{w}_j$. In order to verify a number of properties, the following functions have been proposed for h_s and h_n in the case of Gaussian noise [45]:

$$\begin{aligned}h_s(w_j) &= \frac{1}{\sigma_j^2}\int_0^{|w_j|} u \ \mathrm{erf}\left(\frac{\mid w_j \mid - u}{\sqrt{2}\sigma_j}\right) du \\ h_n(w_j) &= \frac{1}{\sigma_j^2}\int_0^{|w_j|} u \ \mathrm{erfc}\left(\frac{\mid w_j \mid - u}{\sqrt{2}\sigma_j}\right) du\end{aligned} \tag{42}$$

3.8 Example 2 : Astronomical image compression

The principle of the compression method is to select the information we want to keep, by using the PMT, and to code this information without any loss. Thus the first phase searches for the minimum set of quantized multiresolution coefficients which produce an image of "high quality". Criteria which can be retained for estimating the quality of a compression method in astronomy are

1. Visual aspect
2. Signal to noise ratio
3. Photometry (i.e. integrated pixel values in a given object)
4. Astrometry (i.e. object positions)
5. Detection of real and faint objects
6. Objects morphology

Lost information cannot be recovered, so if we do not accept any loss, we have to compress what we take as noise too, and the compression ratio will be low (3 or 4 only). The PMT compression method employed involves the following sequence of operations:

1. Determination of the significant coefficients.
2. Determination of the quantized multiresolution coefficients which give the filtered image.
3. Coding of each resolution level using the Huang-Bijaoui method [28]. This consists of quadtree-coding each image, followed by Huffman-coding the quadtree representation. There is no information lost during this phase.

4. Compression of the noise if this is wished.
5. Decompression consists of reconstituting the noise-filtered image (plus the compressed noise if this was specified).

The decompression is carried out scale by scale, starting from the low resolution, so it is not necessary to decompress the entire file if one is just interested in having a look at the image. Noise is decompressed and added at the end, if this is wanted. See the full description of the method in [46,37].

Various studies have been carried out in order to determine the best compression method for astronomical images, especially in the framework of the ALADIN project [4]. It has been shown, using a sample of nearly 2000 stars from an ESO Schmidt plate centered on the globular cluster M5, that the PMT can give compression ratios of up to 5 times the maximum ratio obtained from other methods [30], such JPEG or wavelet based methods.

Figure 8 top shows a subimage of a Schmidt photographic plate of the region of M5 (numbered ESO 7992v). This photographic plate was digitized using the MAMA ("Machine Automatique à mesurer pour l'Astronomie") machine of Paris Observatory (Paris, France).
Figures 8 bottom left and right allow the visual quality of both the JPEG (compression ratio 40:1) and PMT (compression ratio 260:1) to be compared.

3.9 Conclusion

The median based multiscale transforms present the advantage to be very robust to strong singularities. These singularities can be due to non white noise, or from the observed data themself like stars in astronomical images. In both cases, the median based multiscale transform is an adequate tool to separate these singularities from the rest of the signal. This is the reason of the success of the PMT for the compression of astronomical images.

Finally we would like to point out the work of Donoho [22] about a median based pyramidal transform, in which a decimation of 3 is used, instead of the standard dyadic scheme, in order to preserve the independence of the coefficients.

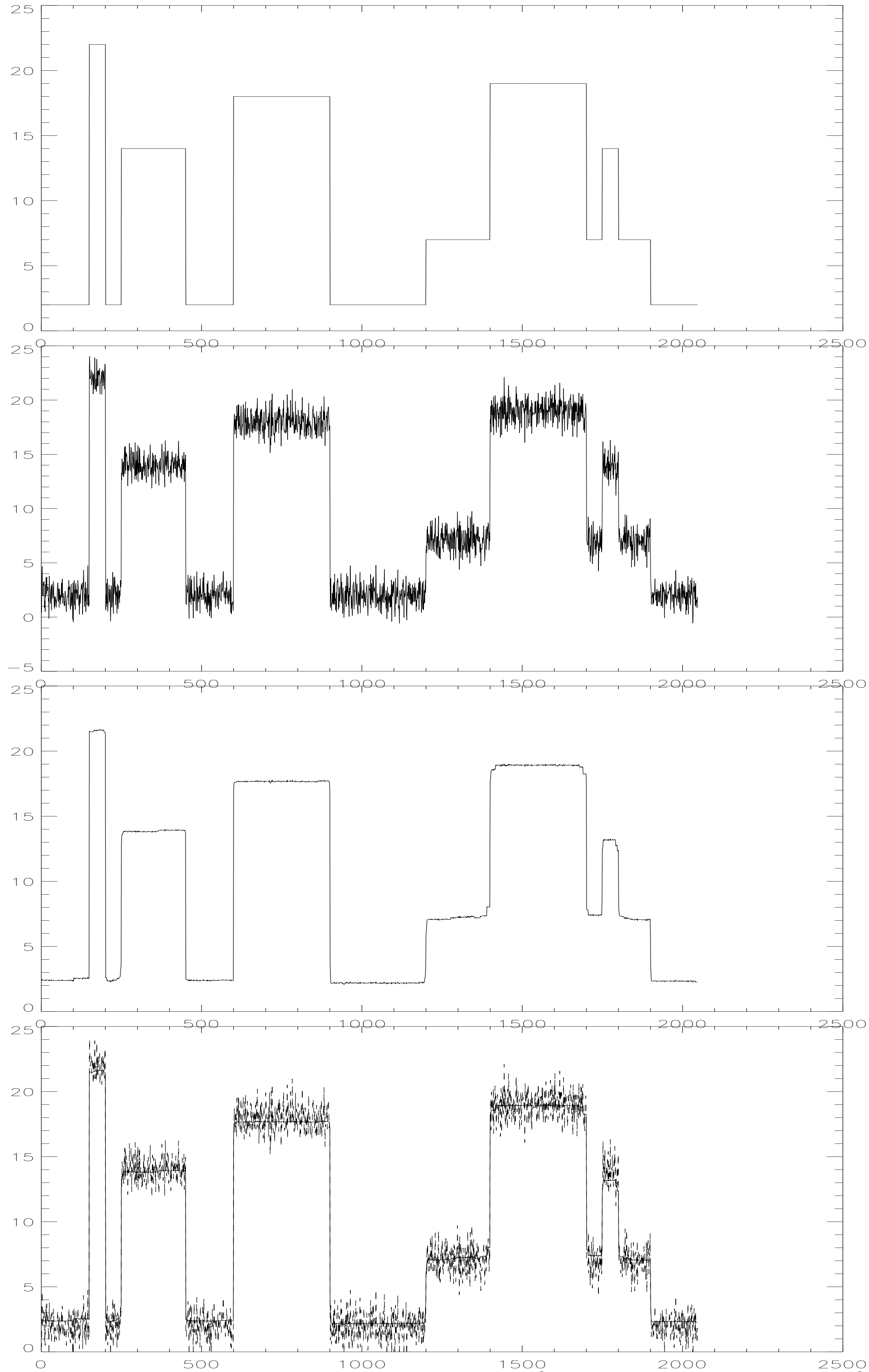

Fig. 7. From top to bottom, simulated block data, noise blocks, filtered blocks, and both noisy and filtered blocks overplotted.

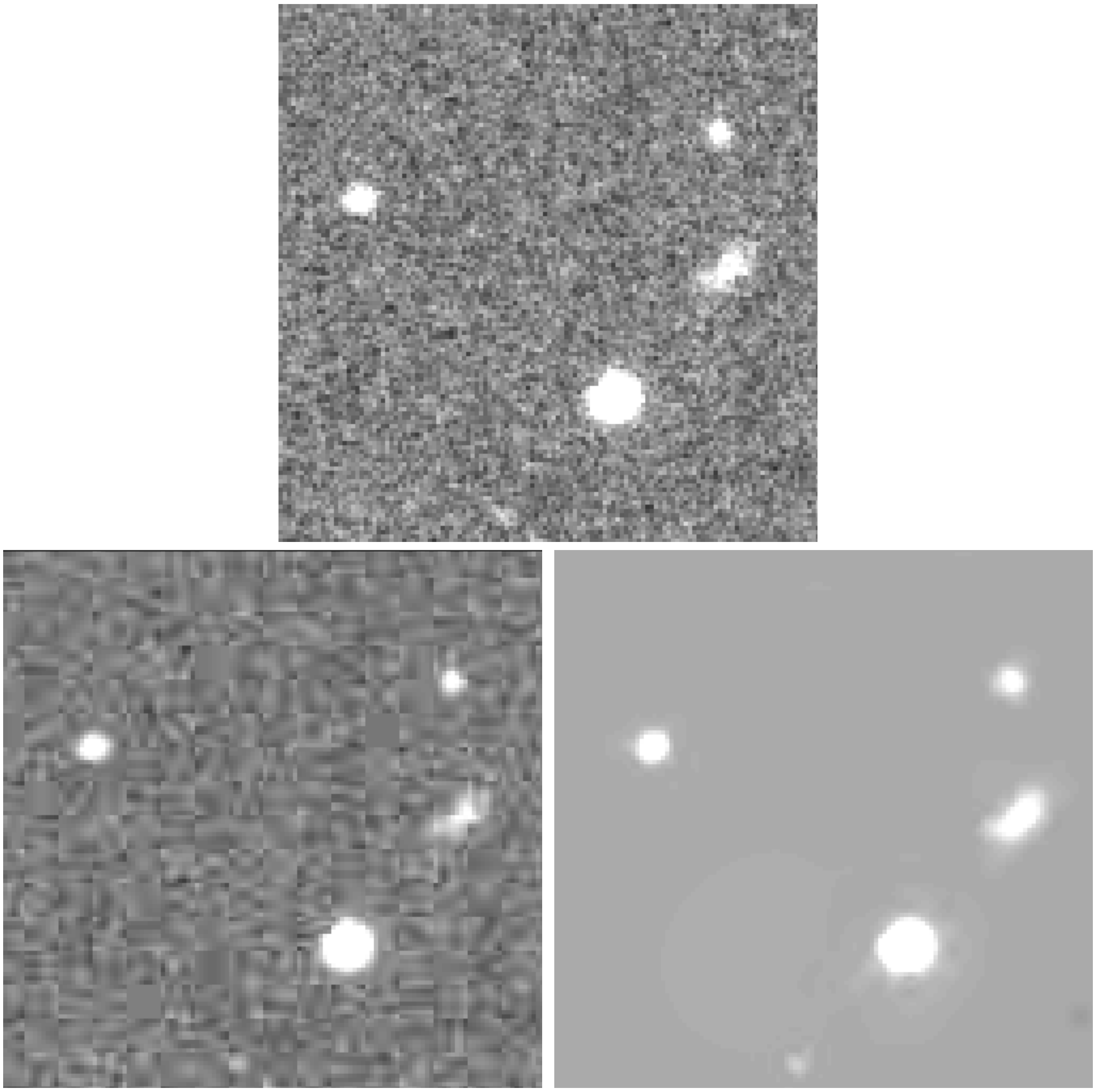

Fig. 8. Top, Original image: sub-image extracted from 1024x1024 patch, extracted in turn from the central region of ESO7992v. Bottom left, JPEG compressed image at 40:1 compression rate: small image patch extracted from central region of ESO7992v. Bottom right, PMT compressed image at 260:1 compression rate: small image patch extracted from central region of ESO7992v.

4 Decomposition of a signal into its main components

4.1 Introduction

The multiscale transform of an image by the à trous algorithm, or the Multiscale Median transform, produces, at each scale j, a set $\{w_j\}$. 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 scales and the smoothed array c_J by the expression

$$c_{0,l} = c_{J,l} + \sum_{j=1}^{J} w_{j,l} \tag{43}$$

Hence, we have a *multiscale pixel representation*, i.e. each pixel of the input image is associated to a set of pixels of the multiscale transform. A further step is to consider a *multiscale object representation*, which would associate to an object contained in the data, a volume in the multiscale transform. Such a representation obviously depends on the kind of image we need to analyze, and we present here a model which has been developed for astronomical data. It may however be used for other kind of data, if they present some similarities with astronomical data. We assume that an image I can be decomposed into a set of components:

$$c_0(k,l) = \sum_{i=1}^{N_o} O_i(k,l) + B(k,l) + N(k,l) \tag{44}$$

where N_o is the number of object, O_i are the objects contained in the data (stars galaxies, ...), B is the background image, and N is the noise.

To perform such a decomposition, we have to detect, to extract, to measure and to recognize the significant structures. This is done by first computing the multiresolution support of the image, and by applying a segmentation scale by scale. The wavelet space of a 2D direct space is a 3D one. An object has to be defined in this space. A general idea for object definition lies in the connectivity property. An object occupies a physical region, and in this region we can join any pixel to other ones. Connectivity in direct space has to be transported to wavelet transform space (WTS). In order to define the objects we have to identify the WTS pixels we can attribute to the objects. We describe in this section the different steps of this method.

4.2 Multiscale Vision Model Definition

The Multiscale Vision Model (MVM) [3,44] described an object as a hierarchical set of structures. It uses the following definitions:

- *significant wavelet coefficient*: a wavelet coefficient is significant when its absolute value is above a given detection limit. The detection limit depends on the noise model (Gaussian noise, Poisson noise, and so on). See Starck et al. [44] for a full description of the noise modeling.
- *structure*: a structure $\mathcal{S}_j$ is a set of significant connected wavelet coefficients at the same scale j.
- *object*: an object is a set of structures.
- *object scale*: the scale of an object is given by the scale of the maximum of its wavelet coefficients.
- *interscale relation*: the criterion allowing us to connect two structures into a single object is called the "interscale relation".
- *sub-object*: a sub-object is a part of an object. It appears when an object has a local wavelet maximum. Hence, an object can be composed of several sub-objects. Each sub-object can also be analyzed.

4.3 The multiresolution support and its segmentation

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). M depends on several parameters:

- The input image.
- The algorithm used for the multiresolution decomposition.
- The noise.
- All additional constraints we want the support to satisfy.

Such a support results from the data, the treatment (noise estimation, etc.), and from knowledge on our part of the objects contained in the data (size of objects, linearity, etc.). In the most general case, a priori information is not available to us.

The multiresolution support of an image is computed in several steps:

- Step one is to compute the wavelet transform of the image.
- Binarization of each scale leads to the multiresolution support (the binarization of an image consists of assigning to each pixel a value only equal to 0 or 1).
- A priori knowledge can be introduced by modifying the support.

This 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.

The multiresolution support will be obtained by detecting at each scale the significant coefficients. The multiresolution support is defined by:

$$M(j,x,y)=\begin{cases} 1 & \text{if } w_j(x,y) \text{ is significant} \\ 0 & \text{if } w_j(x,y) \text{ is not significant} \end{cases} \tag{45}$$

In the case of Gaussian noise, it suffices to compare the wavelet coefficients $w_j(x,y)$ to a threshold level t_j. t_j is generally taken equal to $k\sigma_j$, where σ_j is the noise standard deviation at scale j, and k is chosen between 3 and 5. The value of 3 corresponds to a probability of false detection of 0.0027 for Gaussian statistics. If $w_j(x,y)$ is small, then it is not significant and could be due to noise. If $w_j(x,y)$ is large, it is significant:

$$\begin{array}{ll} \text{if } \mid w_j \mid \geq t_j & \text{then } w_j \text{ is significant} \\ \text{if } \mid w_j \mid < t_j & \text{then } w_j \text{ is not significant} \end{array} \tag{46}$$

Many other kinds of noise can be considered in the wavelet space. See [44] for a review.

Multiresolution Support Segmentation The segmentation consists of labeling a boolean image (0 or 1). Each group of connected pixels having a "1" value gets a label value between 1 and L_{max}, L_{max} being the number of groups. This process is repeated at each scale of the multiresolution support. We define a "structure" S_j^i as the group of connected significant pixels which has the label i at a given scale j.

4.4 Interscale connectivity graph

An object is described as a hierarchical set of structures. The rule which allows us to connect two structures into a single object is called "interscale relation". Figure 9 shows how several structures at different scales are linked together, and form objects. We have now to define the interscale relation: let us consider two structures at two successive scales, S_j^k and S_{j+1}^l. Each structure is located in one of the individual images of the decomposition and corresponds to a region in this image where the signal is significant. Denoting p_m the pixel position of the maximum wavelet coefficient value of S_j^k, S_j^k is said to be connected to S_{j+1}^l if S_{j+1}^l contains the pixel position p_m (i.e. the maximum position of the structure S_j^k must also be contained in the structure S_{j+1}^l). Several structures appearing in successive wavelet coefficient images can be connected in such a way, which we call an object in the interscale connectivity graph.

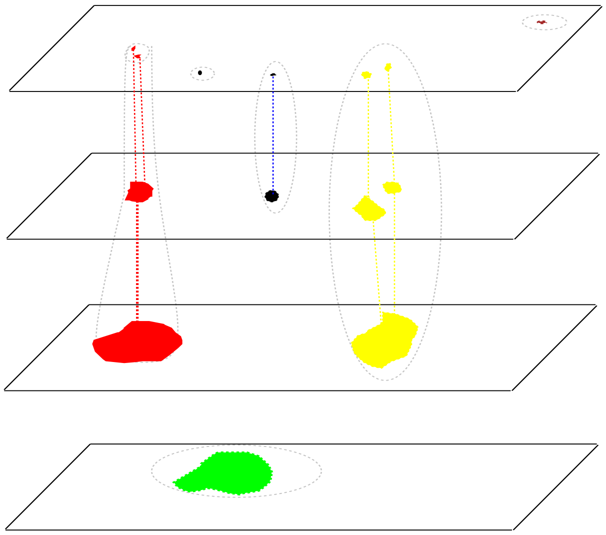

Fig. 9. Example of connectivity in wavelet space: contiguous significant wavelet coefficients form a structure, and following an interscale relation, a set of structures form an object. Two structures S_j, S_{j+1} at two successive scales belong to the same object if the position pixel of the maximum wavelet coefficient value of S_j is included in S_{j+1}.

4.5 Reconstruction

Hence, a set of structures defines an object $W = \{S_j^k, ...S_{j'}^{k'}\}$ which can be reconstructed separately from other objects. The coaddition of all reconstructed objects is a filtered version of the input data.

The reconstruction problem consists of searching for a signal O such that its wavelet coefficients are the same as those of the detected structures. If $\mathcal{T}$ describes the wavelet transform operator, and P_w the projection operator in the subspace of the detected coefficients (i.e. having set to zero all coefficients at scales and positions where nothing was detected), the solution is found by minimization of

$$J(O) = \parallel W - (P_w \circ \mathcal{T})O \parallel \tag{47}$$

where W represents the detected wavelet coefficients of the data. The object image size is generally relatively small, and this minimization does not present any computation problem. Furthermore, there is also no problem if instability, because only coefficients with high signal to noise ratio are used (generally larger than 3), and we do not need to add any regularization term.

Another interesting point to notice is the fact that the coaddition of all detected and reconstructed objects furnishes a filtered image of the input data. This restoration concept has been extended to the deconvolution problem [42]. Indeed, we can associate to an object its point spread function (PSF), and apply the following minimization:

$$J(O) = \parallel W - (P_w \circ \mathcal{T})(P * O) \parallel \tag{48}$$

where P is the point spread function. This combined detection-deconvolution method presents the following advantages:

1. Faint extended objects can be detected as well as point sources.
2. The analysis does not require background estimation.
3. Objects are easier to identify in the deconvolved map.
4. Morphological parameters (galaxy ellipticity and so on) are more accurate.
5. Since each object is deconvolved separately, a spatially variable point spread function can easily be taken into account.
6. Very large images can be deconvolved (each object has a small size).

Point 5 is of great interest. Deconvolution when the PSF is space variant is always problematic, while in this approach, we just have to associate to each object, the PSF at its center.

4.6 Example 1: Band Extraction

We have simulated a spectrum which contains an emission band at 3.50 μm and non stationary noise superimposed on a smooth continuum. The band is a Gaussian of width FWHM = 0.01 μm (FWHM = Full Width Half maximum), and normalized such that its maximum value equals ten times the local noise standard deviation.

Figure 10 (top) contains the simulated spectrum. The wavelet analysis results in the detection of an emission band at 3.50 μm above 3σ. Figure 10 (middle) shows the reconstruction of the detected band in the simulated spectrum. The real feature is over-plotted as a dashed line. Figure 10 (bottom) contains the original simulation with the reconstructed band subtracted. It can be seen that there are no strong residuals near the location of the band, which indicates that the band is well reconstructed. The center position of the band, its FWHM, and its maximum, can then be estimated via a Gaussian fit.

4.7 Example 2: Star subtraction in astronomical images

We applied the MVM to the galaxy NGC2997 (fig. 11 top left). Two images have been created by coadding objects detected from scale 1 to 2, and from scale 3 to 6. They are displayed respectively in figure 11 top right and bottom left. Figure 11 bottom right shows the difference between the input data and the image which contained the objects from scale 1 to 2. As we can see, all small objects have removed, and the galaxy can be better analyzed.

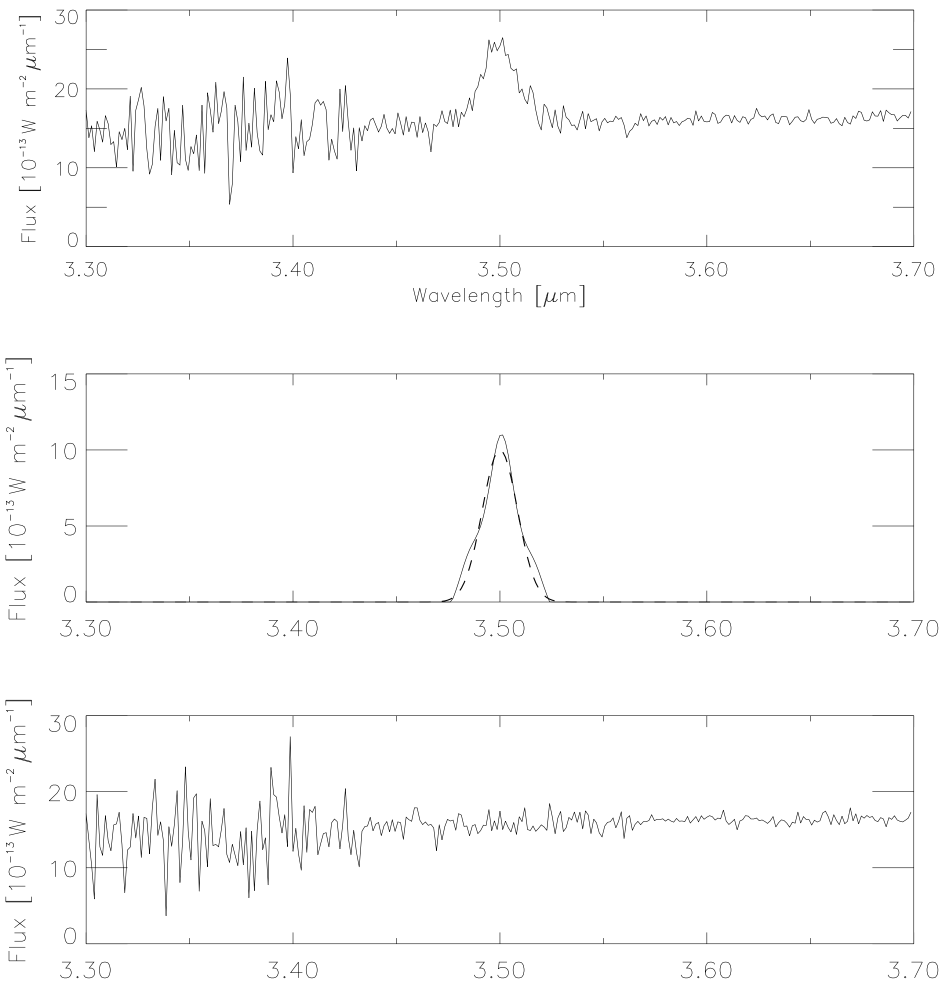

Fig. 10. Top: simulated spectrum. Middle: reconstructed simulated band (full line) and original band (dashed line). Bottom: simulated spectrum minus the reconstructed band.

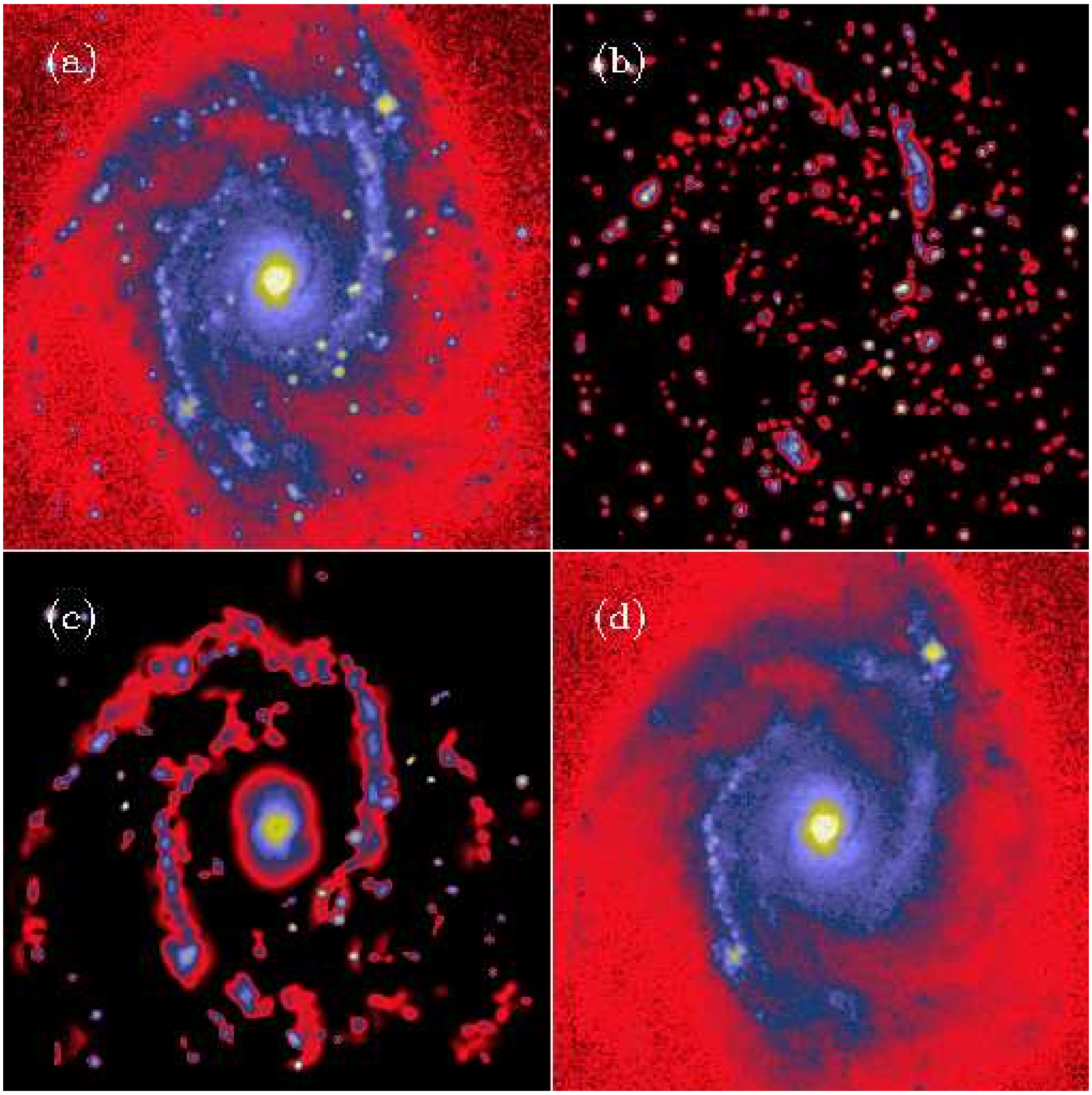

Fig. 11. (a) galaxy NGC2997, (b) objects detected from scale 1 to 2, (c) objects detected from scale 3 to 6, and (d) difference between (a) and (b).

5 Combined Transforms and Applications

5.1 Introduction

The content of an image is often complex, and there is not a single transform which is optimal to represent all the contained features. For example, the Fourier transform better represents some textures, while the wavelet transform better represents singularities. Even if we limit our class of transforms to the wavelet one, decision have to be taken between an isotropic wavelet transform which produce good results for isotropic objects (such stars and galaxies in astronomical images, cells in biological images, ...), or an orthogonal wavelet transform, which is better for images with edges. This has motivated the development of methods, such *matching pursuit* [34], or *basis pursuit* [12], which allow us to represent a signal with overcomplete dictionaries.

A dictionary $\mathcal{D}$ is defined as a collection of waveforms $(\phi_\gamma)_{\gamma \in \Gamma}$, and the goal is to obtain a representation of a signal s with a linear combination of a small number of basis such that:

$$s = \sum_\gamma a_\gamma \phi_\gamma \tag{49}$$

or an approximate decomposition

$$s = \sum_{i=1}^{m} a_{\gamma_i} \phi_{\gamma_i} + R^{(m)} \tag{50}$$

Matching pursuit [34,31] method (MP) uses a greedy algorithm which adaptively refines the signal approximation with an iterative procedure:

- set $s^0 = 0$ and $R^0 = 0$.
- find the element $\alpha_k \phi_{\gamma_k}$ which best correlates with the residual.
- Update s and R:

$$\begin{aligned} s^{k+1} &= s^k + \alpha_k \phi_{\gamma_k} \\ R^{k+1} &= s - s^k \end{aligned} \tag{51}$$

In case of non orthogonal dictionaries, it has been shown [12] that MP may spend most of the time correcting mistakes made in the first few terms, and therefore is suboptimal in term of sparsity.

Basis pursuit method [12] (BP) consists in minimizing the functional:

$$J(\alpha) = \parallel s - \Phi\alpha \parallel_2^2 + \lambda \parallel \alpha \parallel_1 \tag{52}$$

Between all possible solutions, the chosen one has the minimum l^1 norm. This choice of l_1 norm is very important. A l_2 norm, as used in the method of frames [17], does not preserve the sparsity [12]. The minimization of equation 52, which involves linear programming, is however much more time consuming than MP, and remains prohibitive for large signals [31].

We present in the following an alternative approach, that we call *Combined Transforms Method* (CTM), which combines the different available transforms in order to benefit of the advantages of each of them.

5.2 Combined Transformation

Multiscale transforms which may be considered for a combined transformation are:

- Orthogonal wavelet transform. For filtering application, an undecimated wavelet transform must be preferred.
- A trous algorithm (isotropic objects)
- Ridgelet transform (lines or edges of a given size, and of any width)

- Curvelet transform (lines or edges of different sizes, and of any width)
- Fourier transform (oscillary pattern, ...)
- Multiresolution Median transform (strong singularities, such cosmic rays, very bright stars, saturated pixels, ...)
- Multiscale Morphological transform, wavelet packets, ...

Depending of the content of the data, some of these transforms can be combined in order to get an optimal representation of all features contained in our data set. Compared to MP and BP, we are limited to the use of waveforms relative to the existing transforms, but in practice, these transforms furnish a very large panel of waveforms.

Two iterative methods, *soft-CTM* and *hard-CTM*, allowing us to realize such a combined transform, are describe in this section.

5.3 Soft-CTM

Noting $\mathcal{T}_1, ..., \mathcal{T}_{N_t}$ the N_t transform operators, a solution α is obtained by minimizing a functional of the form:

$$J(\alpha) = \parallel s - \sum_{k=0}^{N_t} \mathcal{T}_k^{-1} \alpha_k \parallel_2^2 + \lambda \sum_k \parallel \alpha_k \parallel_1 \tag{53}$$

where s is the original signal, and α_k are the coefficient obtained with the transform $\mathcal{T}_k$.

An simple algorithm to achieve such an solution is:

1. set $\lambda = L_{\max}$
2. while $\lambda >= 0$ do
3. for k = 1, .., N_t do
   - calculate the residual $R = s - \sum_k \mathcal{T}_k^{-1} \alpha_k$
   - calculate the transform $\mathcal{T}_k$ of the residual: $r_k = \mathcal{T}_k R$.
   - add the residual to α_k: $\alpha_k = \alpha_k + r_k$
   - soft threshold the coefficient α_k with the λ threshold.
4. $\lambda = \lambda - \delta$, and goto 2.

Figure 12 illustrates the result in the case where the input image contains only lines and Gaussians. In this experiment, we have initialized $L_{\max}$ to 20, and δ to 2 (10 iterations). Two transform operators were used, the à trous wavelet transform and the ridgelet transform. The first is well adapted to the detection of Gaussian due to the isotropy of the wavelet function [44], while the second is optimal to represent lines [11]. Figure 12 top, bottom left, and bottom right represents respectively the original image, the reconstructed image from the à trous wavelet coefficient, and the reconstructed image from the ridgelet coefficient. The addition of both reconstructed images reproduces the original one.

In some specific cases where the data are sparse in all bases, it has been shown [29] that the solution is identical to the solution when using a $\parallel . \parallel_0$ penalty term. This is however generally not the case. The problem we met in image restoration applications, when minimizing equation 53, is that both the signal and noise are splitted into the bases. The way the noise is distributed in the coefficients α_k is not known, and leads to the problem that we do not know at which level we should threshold the coefficients. Using the threshold we would have used with a single transform makes a strong over-filtering of the data. Using the l^1 optimization for data restoration implies to first study how the noise is distributed in the coefficients. The hard-CTM method does not present this drawback.

5.4 Hard-CTM

The following algorithm consists in hard thresholding the residual successively on the different bases.

1. initialize $L_{\max}$ and the number of iterations Ni.

2. For noise filtering, estimate the noise standard deviation σ, and set $L_{\min} = k$. Otherwise, set $\sigma = 1$ and $L_{\min} = 0$.
3. set $\delta_\lambda = \frac{L_{\max} - L_{\min}}{Ni}$
4. set $\lambda = L_{\max}$
5. set all coefficients α_k to 0.
6. while $\lambda >= L_{\min}$ do
7. for k = 1, .., N_t do
   - calculate the residual $R = s - \sum_k \mathcal{T}_k^{-1} \alpha_k$
   - calculate the transform $\mathcal{T}_k$ of the residual: $r_k = \mathcal{T}_k R$.
   - for all coefficients $r_{k,i}$ do
     - Update the coefficients: if $\alpha_{k,i} \neq 0$ or $\mid r_{k,i} \mid > \lambda\sigma$ then $\alpha_{k,i} = \alpha_{k,i} + r_{k,i}$
8. $\lambda = \lambda - \delta_\lambda$, and goto 6.

For an exact representation of the data, k must be set to 0. Choosing $k > 0$ introduces a filtering. If a single transform is used, it corresponds to the standard $k\sigma$ hard thresholding.

It seems that starting with a high enough $L_{\max}$ and a high number of iterations would lead to the l^0 optimization solution, but this remains to be proved.

Figure 13 shows the decomposition of the classical `Lenna` image (512 by 512). Fig. 13 bottom left and right show respectively the reconstructed image from the wavelet and the ridgelet coefficients. The ridgelet transform [9] was used with a partitioning box of size 64. All the information relative to the long edges is in the ridgelet coefficients, while information relative to small edges or smooth regions is in the wavelet coefficients. The coaddition of both images reproduces the original one. This method does not present the drawback of the previous relative to the distribution of the noise in the different bases. As a hard thresholding is used, when a feature is detected in the residual by a given base, it is extracted. Hence, it cannot be diluted in several bases. This important for a restoration purpose. Indeed, it means that the detection level in a given base is unchanged by the fact that we use a set of bases. In our filtering simulations, we have set $L_{\max} = 40$, $Ni = 40$ and $L_{\min} = 4$.

5.5 Applications

Elongated - point like object separation in astronomical images Figure 14 shows the result of a decomposition of a spiral galaxy (NGC2997). This image (figure 14 top left) contains many compact structures (stars and HII region), more or less isotropic, and large scale elongated features (NGC2997 spiral part). Compact objects are well represented by isotropic wavelets, and the elongated features are better represented by a ridgelet basis. In order to benefit of the optimal data representation of both transforms, the image has been decomposed on both the à trous wavelet transform and on the ridgelet transform by using the same method as described in section 5.4. When the functional is minimized, we get two images, and their coaddition is the filtered version of the original image. The reconstructions from the à trous coefficient, and from the ridgelet the ridgelet coefficient can be seen in figure 14 top right and bottom left. The addition of both images is presented in figure 14 bottom right.

Filtering A simulation has been done with the classical `Lenna` 512×512 image. A white Gaussian noise with a standard deviation equal to 20 has been added to the original data. The noisy image has been filtered by the undecimated wavelet transform, the curvelet transform, and a combination of three methods (ridgelet, curvelet, and undecimated wavelet). Results are given in table 1

Figure 15 shows the noisy image (top left), the undecimated wavelet transform (UWT) filtering (top right), the curvelet filtering (bottom left), and the combined filtering (bottom right). Figure 16 shows a zoom of the same images. The residuals of the three methods are shown in figure 17.

6 Conclusion

From these simulations, we can conclude that (i) the CTM-filtering leads to a real improvement for both the PSNR and the visual aspect, and (ii) only a combination of methods allows us to clean properly the residual.

The CTM approach cost obviously much more than a single transform, for both the computation time and the memory, but presents several advantages:

- it is easy to implement.
- noise modeling can be addressed in the same way as when using a single transform.
- the reconstruction is straitforward.

We have seen in this paper that a separation point-like/extended objects can be done either by the use of a multiscale vision model, either by the combined transform method. But the first approach presents the drawback to need an explicit model of an object definition.

In our examples, we have built our combined transform from linear transforms (ridgelet, wavelets, curvelets), but any kind of non linear transform can be used in the same way.

Method	PSNR	Comments
Noisy image	22.13	
UWT7-9 + ksigma Hard thresh.	31.94	very few artifact
Curvelet (B=16)	31.95	no artifact
Combined filtering	32.17	no artifact

Table 1. PSNR after filtering the simulated image (Lena + Gaussian noise (sigma=20)). In the combined filtering, a ridgelet, a curvelet, and a undecimated wavelet transform have been used.

Acknowledgments. We wish to thank David Donoho and Emmanuel Candès for useful discussions and comments. This research was supported by National Science Foundation grants DMS 98–72890 (KDI) and DMS 95–05151; and by AFOSR MURI-95-P49620-96-1-0028.

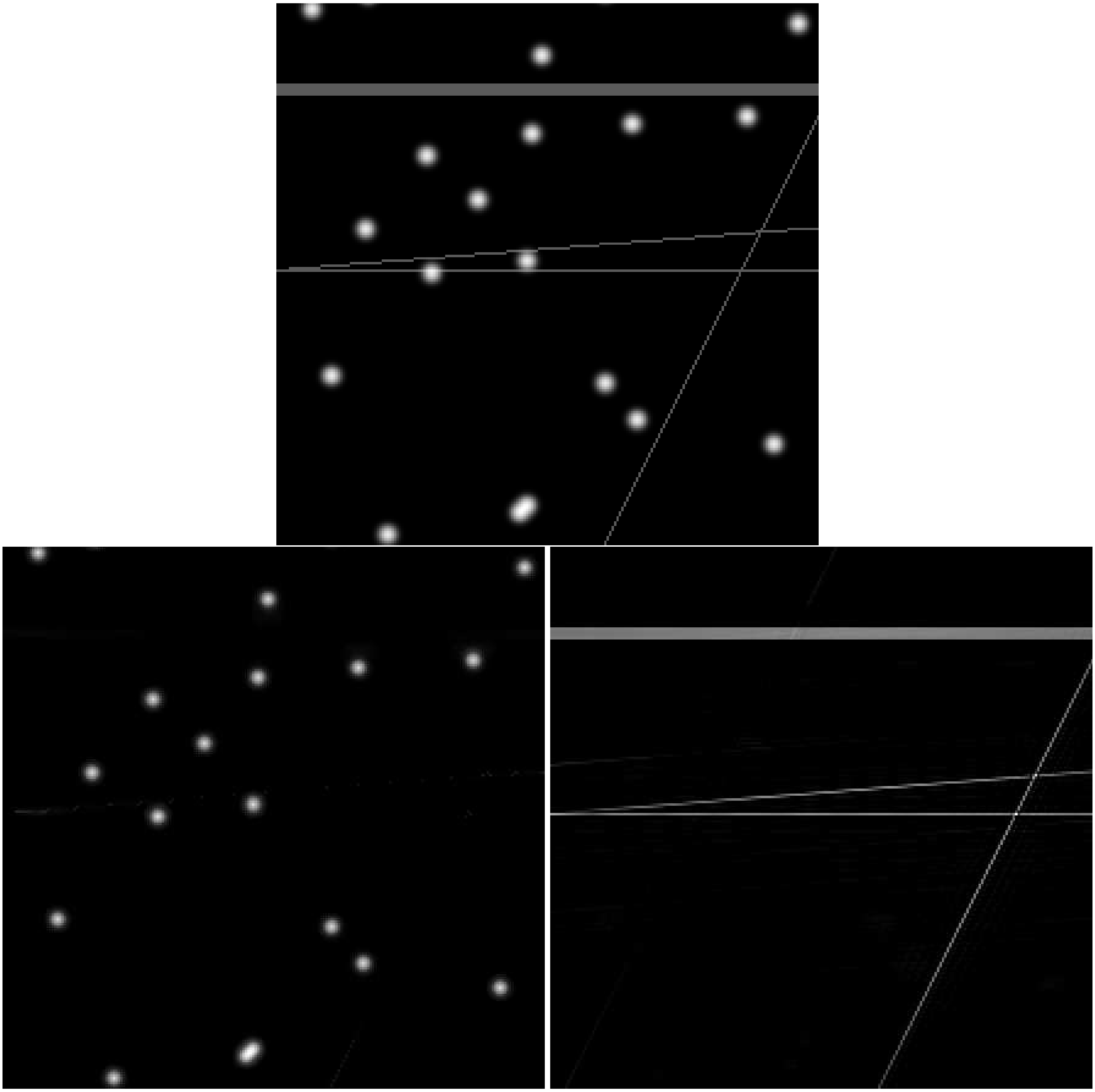

Fig. 12. Top, original image containing lines and gaussians. Botton left, reconstructed image for the à trous wavelet coefficient, bottom right, reconstructed image from the ridgelet coefficients.

Fig. 13. Combined transform using a ridgelet transform and an undecimated wavelet transform. Top, the classical `Lenna` image, bottom left, reconstructed image from the wavelet coefficients, and bottom right, reconstructed image from the ridgelet coefficients.

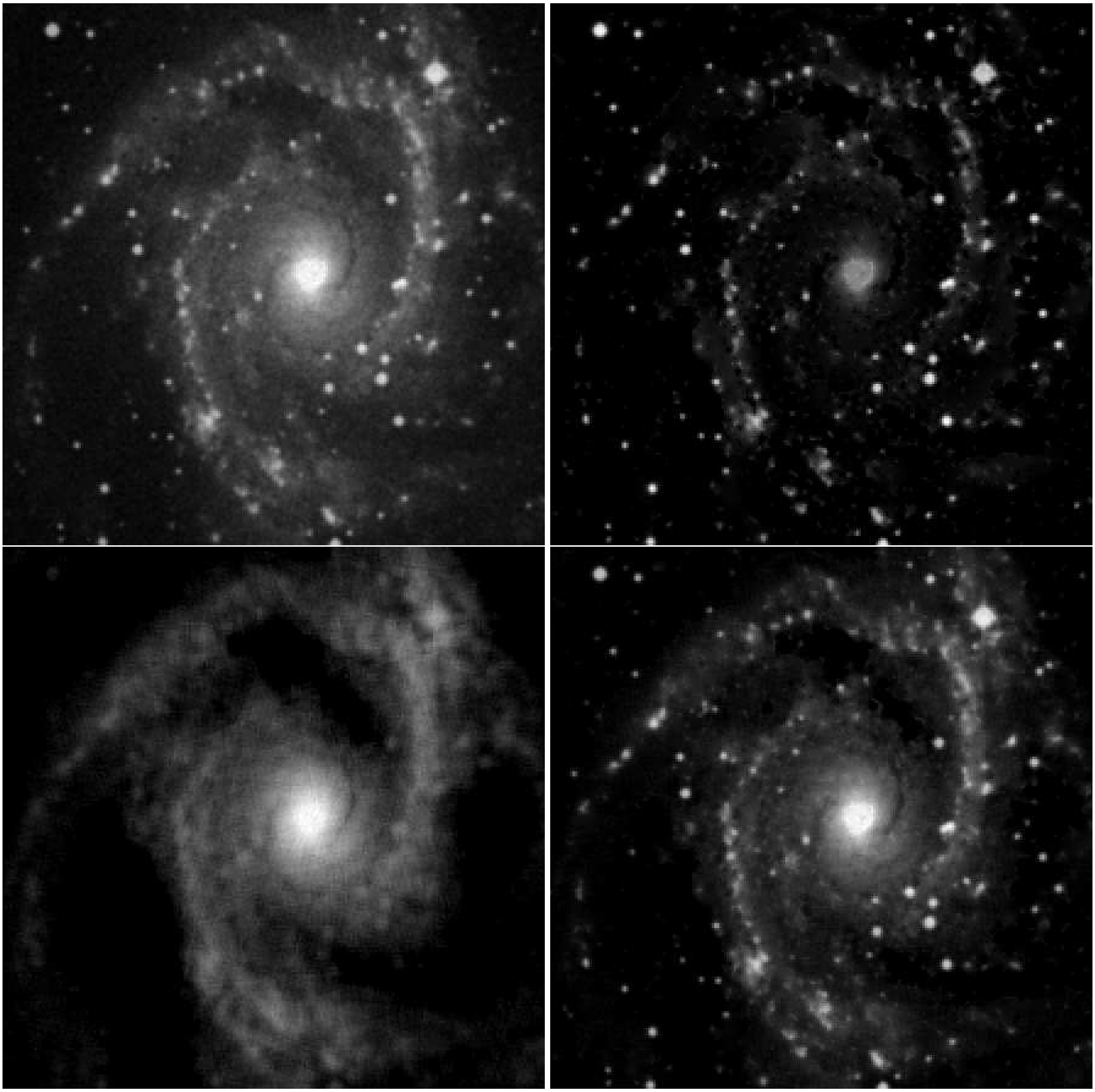

Fig. 14. Top left, galaxy NGC2997, top right reconstructed image from the à trous wavelet coefficients, bottom left, reconstruction from the ridgelet coefficients, and bottom right addition of both reconstructed images.

Fig. 15. Noisy image, and filtered images by the undecimated wavelet transform, the curvelet transform, and the combined method.

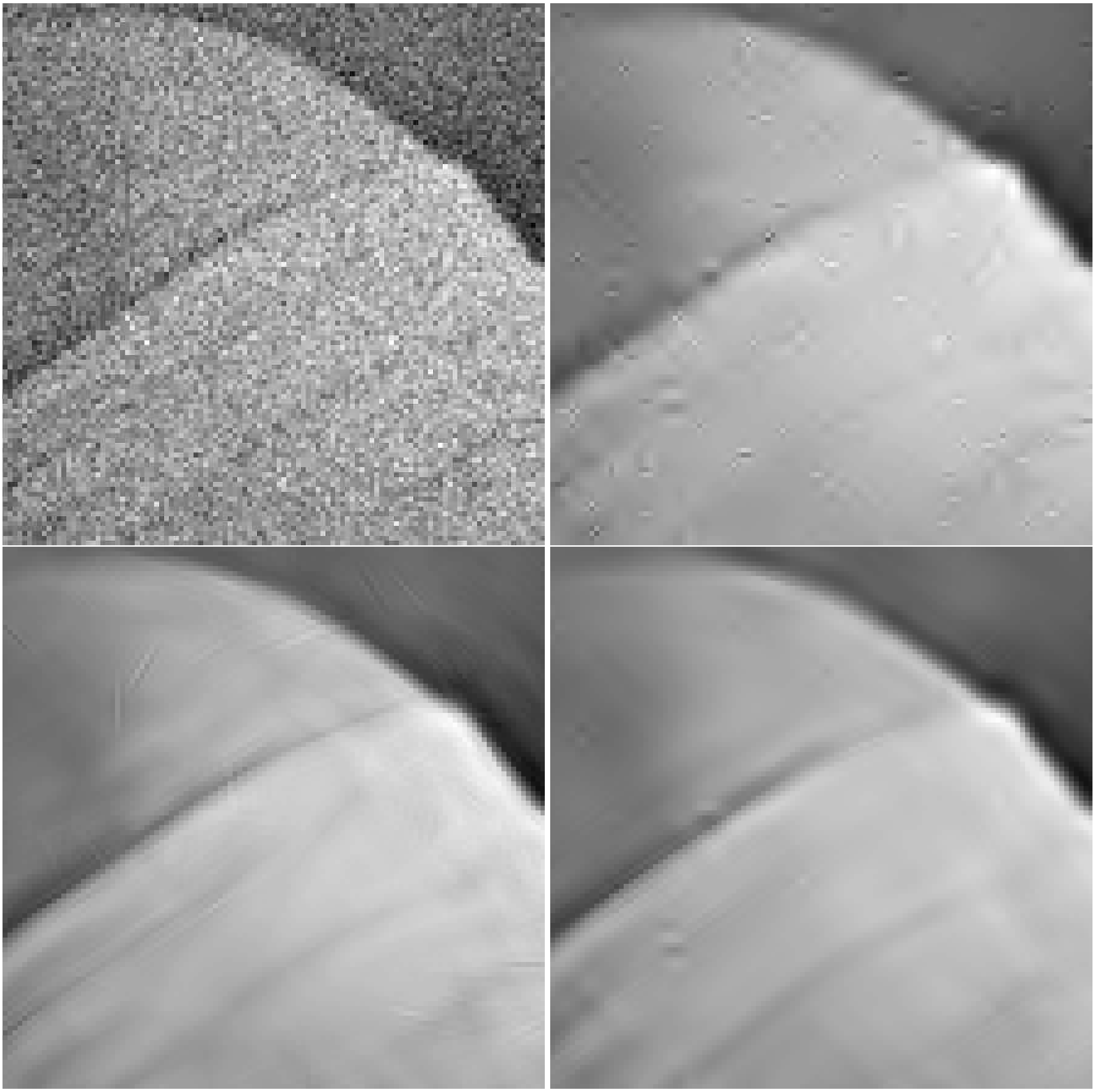

Fig. 16. Noisy image, and filtered images by the undecimated wavelet transform, the curvelet transform, and the combined method.

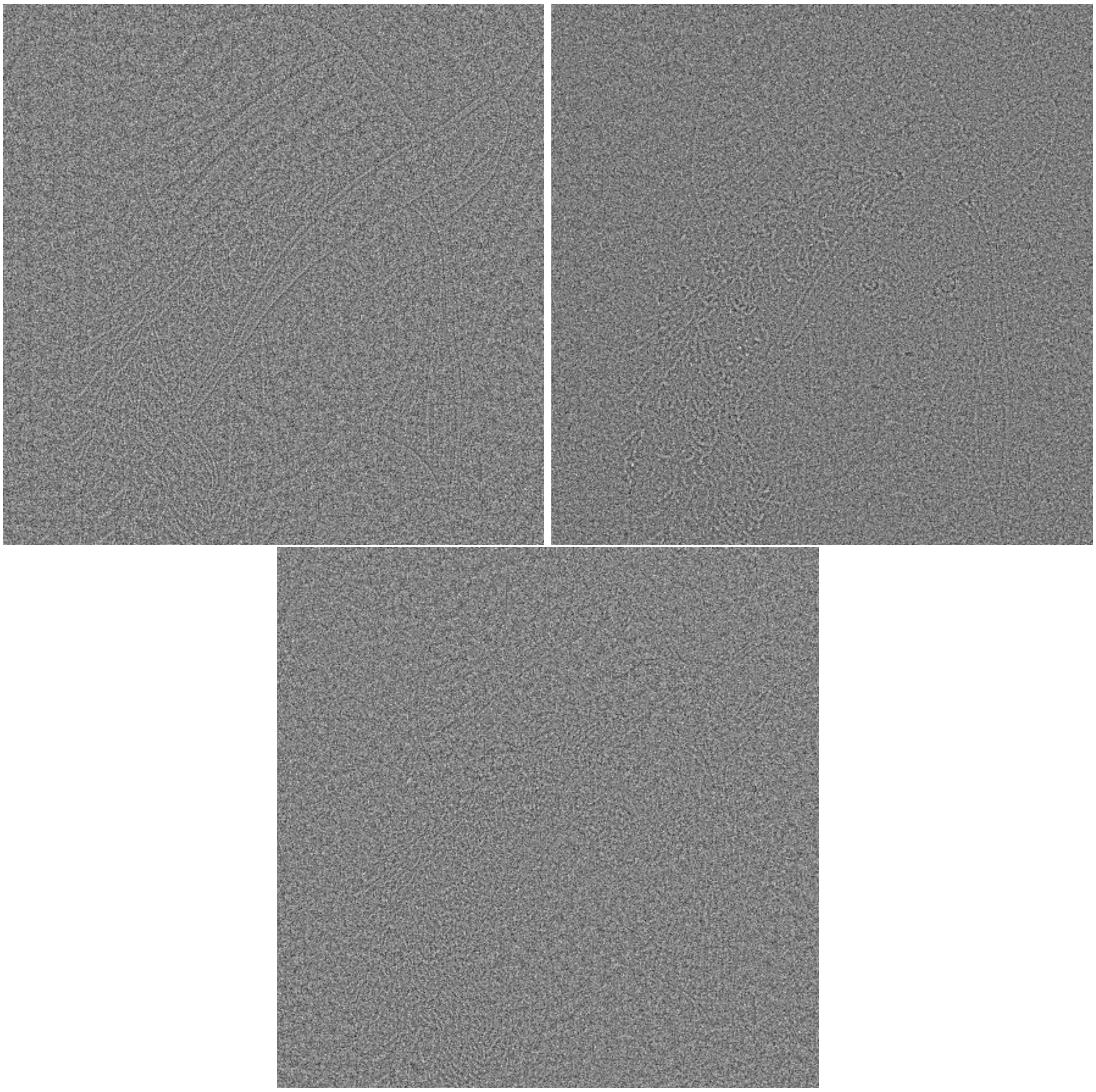

Fig. 17. Top, residual for the undecimated wavelet transform, and the curvelet transform, bottom, residual for the combined method.

References

1. E.H. Adelson, E. Simoncelli, and R. Hingorani. Optimal image addition using the wavelet transform. *SPIE Visual Communication and Image Processing II*, 845:50–58, 1987.
2. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Transactions on Image Processing*, 1(2):205–220, 1992.
3. A. Bijaoui and F. Rué. A multiscale vision model adapted to astronomical images. *Signal Processing*, 46:229–243, 1995.
4. F. Bonnarel, P. Fernique, F. Genova, J. G. Bartlett, O. Bienaymé, D. Egret, J. Florsch, H. Ziaeepour, and M. Louys. ALADIN: A reference tool for identification of astronomical sources. *Astronomical Data Analysis Software and Systems VIII, ASP Conference Series, Vol. 172. Ed. David M. Mehringer, Raymond L. Plante, and Douglas A. Roberts. ISBN: 1-886733-94-5 (1999), p. 229.*, 8:229–232, 1999.
5. E.J. Breen, R.J. Jones, and H. Talbot. Mathematical morphology: A usefull set of tools for image analysis. *Statistics and Computing journal*, 10:105–120, 2000.
6. C.S. Burrus, R.A. Gopinath, and H. Guo. *Introduction to Wavelets and Wavelet Transforms.* Prentice Hall, 1998.
7. P.J. Burt and A.E. Adelson. The laplacian pyramid as a compact image code. *IEEE Tranactions on Communications*, 31:532–540, 1983.
8. R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo. Wavelet transforms that map integers to integers. *Appl. Comput. Harmon. Anal.*, 5(3):332–369, 1998.
9. E. J. Candes. Harmonic analysis of neural netwoks. *Applied and Computational Harmonic Analysis*, 6:197–218, 1999.
10. E.J. Candès and D. Donoho. Curvelets. Technical report, Statistics, Stanford Univeristy, 1999.
11. E.J. Candès and D. Donoho. Ridgelets: the key to high dimensional intermittency? *Phil. trans; R. Soc. Lond. A*, 357:2495–2509, 1999.
12. S.S. Chen, D.L. Donoho, and M.A. Saunder. Atomic decomposition by basis pursuit. *SIAM J. Math. Anal.*, 20(1):33–61, 1998.
13. C.H. Chui. *Wavelet Analysis and Its Applications.* Academic Press, 1992.
14. R. Claypoole, G.M. Davis, W. Sweldens, and R. Baraniuk. Nonlinear wavelet transforms for image coding via lifting. *IEEE Transactions on Image Processing*, 2000. submitted.
15. R.L. Claypoole, R.G. Baraniuk, and R.D. Nowak. Lifting construction of non-linear wavelet transforms. In *IEEE International Conference IEEE-SP Time-Frequency and Time-Scale Analysis*, pages 49–52, 1998.
16. A. Cohen, I. Daubechies, and J.C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Communications in Pure and Applied Mathematics*, 45:485–560, 1992.
17. I. Daubechies. Time-frequency localization operators: A geometric phase space approach. *IEEE Transactions on Information Theory*, 34(1):605–612, 1988.
18. I. Daubechies. *Ten Lectures on Wavelets.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1992.
19. I. Daubechies, I. Guskov, P. Schröder, and W. Sweldens. Wavelets on irregular point sets. *Phil. Trans. R. Soc. Lond. A*, To be published.
20. I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 4(3):245–267, 1998.
21. D.L. Donoho. Nonlinear wavelet methods for recovery of signals, densities, and spectra from indirect and noisy data. *Proceedings of Symposia in Applied Mathematics*, 47, 1993.
22. D.L. Donoho. Nonlinear pyramid transforms based on median-interpolation. *SIAM J. Math. Anal.*, 0(0):000–000, 2000.
23. O. Egger and W. li. Very low bit rate image coding using morphological operators and adaptive decompositions. In *IEEE International Conference ICIP-94 on Image Processing*, volume 2, pages 326–330, 1994.
24. D.A.F. Florêncio and R.W. Schafer. Perfect reconstructing nonlinear filter banks. In *IEEE International Conference ICASSP-96 on Acoustics, Speech, and Signal Processing*, volume 3, pages 1814–1817, 1996.
25. J. Goutsias and H. J. A. M. Heijmans. Nonlinear multiresolution signal decomposition schemes. part 1: Morphological pyramids. *IEEE Transactions on Image Processing*, to appear.
26. F.J. Hampson and J.-C. Pesquet. A nonlinear subband decomposition with perfect reconstruction. In *IEEE International Conference ICASSP-96 on Acoustics, Speech, and Signal Processing*, volume 3, pages 1523–1526, 1996.
27. H. J. A. M. Heijmans and J. Goutsias. Multiresolution signal decomposition schemes. part 2: Morphological wavelets. Research Report PNA-R9905, CWI, 1999.

28. L. Huang and A. Bijaoui. Astronomical image data compression by morphological skeleton transformation. *Experimental Astronomy*, 1:311–327, 1991.
29. Xiaoming Huo. *Sparse Image Representation via Combined Transforms.* PhD thesis, Stanford Univesity, August 1999.
30. M. Louys, J.-L. Starck, S. Mei, F. Bonnarel, and F. Murtagh. Astronomical image compression. *Astronomy and Astrophysics, Suppl. Ser.*, 136:579–590, 1999.
31. S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1998.
32. S. Mallat and F. Falzon. Analysis of low bit rate image transform coding. *IEEE Transactions on Signal Processing*, 46(4):1027–42, 1998.
33. S. G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.
34. S.G. Mallat and Z. Zhang. Atomic decomposition by basis pursuit. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.
35. G. Matheron. *Elements pour une théorie des milieux poreux.* Masson, Paris, 1967.
36. G. Matheron. *Random Sets and Integral Geometry.* Willey, New York, 1975.
37. F. Murtagh, J.L. Starck, and M. Louys. Very high quality image compression based on noise modeling. *International Journal of Imaging Systems and Technology*, 9:38–45, 1998.
38. A. Said and W.A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchival trees. *IEEE Trans. on Circ. and Syst. for Video Tech.*, 6(3):243–250, 1996.
39. Peter Schröder and Wim Sweldens. Spherical wavelets: Efficiently representing functions on the sphere. *Computer Graphics Proceedings (SIGGRAPH 95)*, pages 161–172, 1995.
40. J. Serra. *Image Analysis and Mathematical Morphology.* Academic Press, London, 1982.
41. J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans. on Signal Process.*, 41(12):3445–3462, 1993.
42. J.L. Starck, A. Bijaoui, I. Vatchanov, and F. Murtagh. A combined approach for object detection and deconvolution. *Astronomy and Astrophysics*, 2000. to appear.
43. J.L. Starck and F. Murtagh. Multiscale entropy filtering. *Signal Processing*, 76(2):147–165, 1999.
44. J.L. Starck, F. Murtagh, and A. Bijaoui. *Image Processing and Data Analysis: The Multiscale Approach.* Cambridge University Press, Cambridge (GB), 1998.
45. J.L. Starck, F. Murtagh, and R. Gastaud. A new entropy measure based on the wavelet transform and noise modeling. Special Issue on Multirate Systems, Filter Banks, Wavelets, and Applications of *IEEE Transactions on CAS II*, 45(8), 1998.
46. J.L. Starck, F. Murtagh, B. Pirenne, and M. Albrecht. Astronomical image compression based on noise suppression. *Publications of the Astronomical Society of the Pacific*, 108:446–455, 1996.
47. G. Strang and T. Nguyen. *Wavelet and Filter Banks.* Wellesley-Cambridge Press, Box 812060, Wellesley MA 02181, fax 617-253-4358, 1996.
48. W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1997.
49. W. Sweldens and P. Schröder. Building your own wavelets at home. In *Wavelets in Computer Graphics*, pages 15–87. ACM SIGGRAPH Course notes, 1996.