The Curvelet Transform for Image Denoising

Jean-Luc Starck[*], Emmanuel J. Candès[†], David L. Donoho [‡]

November 15, 2000

Abstract

We describe approximate digital implementations of two new mathematical transforms, namely, the ridgelet transform [3] and the curvelet transform [7, 6]. Our implementations offer exact reconstruction, stability against perturbations, ease of implementation, and low computational complexity. A central tool is Fourier-domain computation of an approximate digital Radon transform. We introduce a very simple interpolation in Fourier space which takes Cartesian samples and yields samples on a rectopolar grid, which is a pseudo-polar sampling set based on a concentric squares geometry. Despite the crudeness of our interpolation, the visual performance is surprisingly good. Our ridgelet transform applies to the Radon transform a special overcomplete wavelet pyramid whose wavelets have compact support in the frequency domain. Our curvelet transform uses our ridgelet transform as a component step, and implements curvelet subbands using a filter bank of à trous wavelet filters. Our philosophy throughout is that transforms should be overcomplete, rather than critically-sampled. We apply these digital transforms to the denoising of some standard images embedded in white noise. In the tests reported here, simple thresholding of the curvelet coefficients is very competitive with 'state of the art' techniques based on wavelets, including thresholding of decimated or undecimated wavelet transforms and also including tree-based Bayesian posterior mean methods. Moreover, the curvelet reconstructions exhibit higher perceptual quality than wavelet-based reconstructions, offering visually sharper images and, in particular, higher quality recovery of edges and of faint linear and curvilinear features.

Existing theory for curvelet and ridgelet transforms suggests that these new approaches can outperform wavelet methods in certain image reconstruction problems. The empirical results reported here are in encouraging agreement.

Key Words. Radon Transform, Wavelets, Ridgelets, Curvelets, FFT, FWT, Discrete Wavelet Transform, Thresholding Rules, Filtering.

[*]J.L. Starck is with the CEA-Saclay, DAPNIA/SEI-SAP, F-91191 Gif sur Yvette, France, and the Department of Statistics, Stanford University. Email: jstarck@cea.fr.

[†]E.J. Candès is with the Department of Applied Mathematics, Mail Code 217-50, California Institute of Technology, Pasadena, CA 91125, and the Department of Statistics, Stanford University, Sequoia Hall, Stanford, CA 94305 USA. Email: emmanuel@acm.caltech.edu

[‡]D.L. Donoho is with the Department of Statistics, Stanford University, Sequoia Hall, Stanford, CA 94305 USA. Email: donoho@stat.Stanford.EDU

1 Introduction

1.1 Wavelet Image De-Noising

Over the last decade there has been abundant interest in wavelet methods for noise removal in signals and images. In many hundreds of papers published in journals throughout the scientific and engineering disciplines, a wide range of wavelet-based tools and ideas have been proposed and studied. Initial efforts included very simple ideas like thresholding of the orthogonal wavelet coefficients of the noisy data, followed by reconstruction. Later efforts found that substantial improvements in perceptual quality could be obtained by translation invariant methods based on thresholding of an undecimated wavelet transform. More recently, "tree-based" wavelet de-noising methods were developed in the context of image de-noising, which exploit the tree structure of wavelet coefficients and the so-called parent-child correlations which are present in wavelet coefficients of images with edges. Also, many investigators have experimented with variations on the basic schemes – modifications of thresholding functions, level-dependent thresholding, block thresholding, adaptive choice of threshold, Bayesian conditional expectation nonlinearities and so on. Extensive efforts by a large number of researchers have produced a body of literature which exhibits substantial progress overall, achieved by combining a sequence of incremental improvements.

1.2 A Promising New Appoach

In this paper, we report initial efforts at image de-noising based on a recently-introduced family of transforms – the ridgelet and curvelet transforms – which have been proposed as alternatives to wavelet representation of image data. These transforms, to be described further below, are new enough that the underlying theory is still under development. Software for computing these new transforms is still in a formative stage, as various trade-offs and choices are still being puzzled through.

Although we have completed an initial software development only, and although the time and effort we have expended in implementation, and in fine-tuning, is miniscule in comparison to the efforts which have been made in image de-noising by wavelets, we have been surprised at the degree of success already achieved. We present in this paper evidence that the new approach, in this early state of development, already performs as well as, or better than, mature wavelet image de-noising methods. Specifically, we exhibit higher PSNR on standard images such as `Barbara` and `Lenna`, across a range of underlying noise levels. (At our website, additional examples are provided). Our comparisons consider standard wavelet methods such as thresholding of standard undecimated wavelet transforms, thresholding of decimated wavelet transforms, and Bayesian tree-based methods.

While of course the evidence provided by a few examples is *by itself* quite limited, the evidence we present is consistent with the developing theory of curvelet de-noising, which predicts that, in recovering images which are smooth away from edges, curvelets will obtain dramatically smaller asymptotic mean square error of reconstruction than wavelet methods. The images we study are small in size, so that the asymptotic theory cannot be expected to fully 'kick in'; however, we do observe already, at these limited image sizes, noticeable improvements of the new methods over wavelet de-noising.

By combining the experiments reported here with the theory being developed elsewhere, we conclude that the new approaches offer a high degree of promise which may repay further development in appropriate image reconstruction problems.

1.3 The New Transforms

The new ridgelet and curvelet transforms were developed over several years in an attempt to break an inherent limit plaguing wavelet de-noising of images. This limit arises from the well-known and frequently depicted fact that the two-dimensional wavelet transform of images exhibits large wavelet coefficients even at fine scales, all along the important edges in the image, so that in a map of the large wavelet coefficients one sees the edges of the images repeated at scale after scale. While this effect is visually interesting, it means that many wavelet coefficients are required in order to reconstruct the edges in an image properly. With so many coefficients to estimate, denoising faces certain difficulties. There is, owing to well-known statistical principles, an imposing tradeoff between parsimony and accuracy which even in the best balancing leads to a relatively high Mean Squared Error (MSE).

While this tradeoff is intrinsic to wavelet methods (and also to Fourier and many other standard methods), there exist, on theoretical grounds, better de-noising schemes for recovering images which are smooth away from edges. For example, asymptotic arguments show that, in a certain continuum model of treating noisy images with formal noise parameter ϵ, for recovering an image which is C^2 smooth away from C^2 edges, the ideal MSE scales like $\epsilon^{4/3}$ whereas the MSE acheivable by wavelet methods scales only like ϵ. (For discussions of this white noise model, see [9, 17]).

To approach this ideal MSE, one should develop new expansions which accurately represent smooth functions using only a few nonzero coefficients, and which also accurately represent edges using only a few nonzero coefficients. Then, because so few coefficients are required either for the smooth parts or the edge parts, the balance between parsimony and accuracy will be much more favorable and a lower MSE results. The ridgelet transform and curvelet transform were developed explicitly to show that this combined sparsity in representation of both smooth functions and edges is possible.

The continuous ridgelet transform provides a sparse representation of both smooth functions and of perfectly straight edges. As introduced in [3], the *ridgelet transform* in two dimensions allows the representation of arbitrary bivariate functions $f(x_1, x_2)$ by superpositions of elements of the form $a^{-1/2}\psi((x_1 \cos\theta + x_2 \sin\theta - b)/a)$. Here ψ is a wavelet, $a > 0$ is a scale parameter, θ is an orientation parameter, and b is a location scalar parameter. These so-called ridgelets are constant along ridge lines $x_1 \cos\theta + x_2 \sin\theta$, and along the orthogonal direction they are wavelets. Because ridgelets at fine scale $a \approx 0$ are localized near lines $x_1 \cos\theta + x_2 \sin\theta = b$, it is possible to efficiently superpose several terms with common ridge lines (i.e. common θ, b and different scales a to efficiently approximate singularities along a line. But ridgelets also work well for representing smooth functions, in fact they represent functions in the Sobolev space W_2^2 of functions with two derivatives in mean-square just as efficiently as wavelets (i.e. comparable numbers of terms for same degree of approximation).

There are also various discrete ridgelet transforms – i.e. expansions into a countable discrete collection of generating elements – based on ideas of frames and orthobases. For all of these notions, one has frame/basis elements localized near lines at all locations and orientations and ranging though a variety of scales (localization widths). It has been shown that for these schemes, simple thresholding of the discrete ridgelet transform provides near-optimal N-term approximations to smooth functions with discontinuities along lines [8, 5, 18]. In short, discrete ridgelet representations solve the problem of sparse approximation to smooth objects with straight edges.

In image processing, edges are typically curved rather than straight and ridgelets alone cannot yield efficient representations. However at sufficiently fine scales, a curved edge is almost straight, and so to capture curved edges, one ought to be able to deploy ridgelets in a localized manner, at sufficiently fine scales. Two approaches to localization of ridgelets are possible:

1. *Monoscale ridgelets*. Here one thinks of the plane as partitioned into congruent squares of a given fixed sidelength and constructs a system of renormalized ridgelets smoothly localized near each such square [4].

2. *Multiscale ridgelets*. Here one thinks of the plane as subjected to an infinite series of partitions, based on dyadic scales, where each partition, like in the monoscale case, consists of squares of the given dyadic sidelength. The corresponding dictionary of generating elements is a pyramid of windowed ridgelets, renormalized and transported to a wide range of scales and locations, see sections 2.2 and 4.

Both localization approaches will play important roles in this paper.

Curvelets are based on multiscale ridgelets combined with a spatial bandpass filtering operation to isolate different scales [7, 6]. Like ridgelets, curvelets occur at all scales, locations, and orientations. However, while ridgelets all have global length and variable widths, curvelets in addition to a variable width have a variable length and so a variable anisotropy. The length and width at fine scales are related by a scaling law $width \approx length^2$ and so the anisotropy increases with decreasing scale like a power law. Recent work shows that thresholding of discrete curvelet coefficients provide near-optimal N-term representations of otherwise smooth objects with discontinuities along C^2 curves.

Quantitatively, the N-term squared approximation error by curvelet thresholding scales like $log(N)^3/N^2$. This approximation- theoretic result implies the following statistical result. By choosing a threshold so that one is reconstructing from the largest $N \approx \epsilon^{-2/3}$ noisy curvelet coefficients in a noisy image at noise level ϵ, one obtains decay of the MSE almost of order $O(\epsilon^{4/3})$. In contrast, in analyzing objects with edges, wavelets give an N-term squared approximation error only of size N^{-1}, and wavelet thresholding gives a corresponding MSE only of size $O(\epsilon)$ and no better.

1.4 This Paper

So according to theory for a certain *continuous*-space model, discrete ridgelet transforms and discrete curvelet transforms provide near-ideal sparsity of representation of both smooth objects and of objects with edges. In a certain continuous-space statistical theory, this implies that simple thresholding of noisy coefficients in these expansions is a near-optimal method of de-noising in the presence of white Gaussian noise.

In this paper we provide an initial test of these ideas in a digital image processing setting, where images are available on an n-by-n grid. We first review some basic ideas about ridgelet and curvelet representations in the continuum. We next use these to develop a series of digital ridgelet and digital curvelet transforms taking digital input data on a cartesian grid. Next we consider a model de-noising problem where we embed some standard images in white noise and apply thresholding in the digital curvelet transform domain. Finally we discuss interpretations and possibilities for future work.

Not surprisingly, other researchers have undertaken efforts to implement ridgelet and curvelet transforms, and develop applications. In addition to work mentioned in the body of the article, we would like to point out the work of Do and Vetterli [14], Donoho and

Duncan [19]. We would also like to mention the articles of Sahiner and Yagle [22, 23, 24], Olson and DeStefano [21], Zhao et al. [31], and Zuidwijk [32, 33] although these references are less directly related.

2 Continuous Ridgelet Transform

The two-dimensional continuous ridgelet transform in $\mathbf{R}^2$ can be defined as follows [3]. We pick a smooth univariate function $\psi : \mathbf{R} \to \mathbf{R}$ with sufficient decay and satisfying the admissibility condition

$$\int |\hat{\psi}(\xi)|^2/|\xi|^2 \, d\xi < \infty, \tag{1}$$

which holds if, say, ψ has a vanishing mean $\int \psi(t)dt = 0$. We will suppose that ψ is normalized so that $\int |\hat{\psi}(\xi)|^2 \xi^{-2} d\xi = 1$.

For each $a > 0$, each $b \in \mathbf{R}$ and each $\theta \in [0, 2\pi)$, we define the bivariate *ridgelet* $\psi_{a,b,\theta} : \mathbf{R}^2 \to \mathbf{R}^2$ by

$$\psi_{a,b,\theta}(x) = a^{-1/2} \cdot \psi((x_1 \cos\theta + x_2 \sin\theta - b)/a); \tag{2}$$

this function is constant along lines $x_1 \cos\theta + x_2 \sin\theta = const$. Transverse to these ridges it is a wavelet. Given an integrable bivariate function $f(x)$, we define its ridgelet coefficients by

$$\mathcal{R}_f(a, b, \theta) = \int \overline{\psi}_{a,b,\theta}(x) f(x) dx.$$

We have the exact reconstruction formula

$$f(x) = \int_0^{2\pi} \int_{-\infty}^{\infty} \int_0^{\infty} \mathcal{R}_f(a, b, \theta) \psi_{a,b,\theta}(x) \frac{da}{a^3} db \frac{d\theta}{4\pi} \tag{3}$$

valid a.e. for functions which are both integrable and square integrable. Furthermore, this formula is stable as one has a Parseval relation

$$\int |f(x)|^2 dx = \int_0^{2\pi} \int_{-\infty}^{\infty} \int_0^{\infty} |\mathcal{R}_f(a, b, \theta)|^2 \frac{da}{a^3} db \frac{d\theta}{4\pi}. \tag{4}$$

Hence, much like the wavelet or Fourier transforms, the identity (3) expresses the fact that one can represent any arbitrary function as a continuous superposition of ridgelets. Discrete analogs of (3)-(4) exist, see [3], or [18] for a slightly different approach.

2.1 The Radon Transform

A basic tool for calculating ridgelet coefficients is to view ridgelet analysis as a form of wavelet analysis in the Radon domain. We recall that the Radon transform of an object f is the collection of line integrals indexed by $(\theta, t) \in [0, 2\pi) \times \mathbf{R}$ given by

$$Rf(\theta, t) = \int f(x_1, x_2) \delta(x_1 \cos\theta + x_2 \sin\theta - t) \, dx_1 dx_2, \tag{5}$$

where δ is the Dirac distribution. The ridgelet coefficients $\mathcal{R}_f(a, b, \theta)$ of an object f are given by analysis of the Radon transform via

$$\mathcal{R}_f(a, b, \theta) = \int Rf(\theta, t) a^{-1/2} \psi((t - b)/a) \, dt.$$

Hence the ridgelet transform is precisely the application of a 1-dimensional wavelet transform to the slices of the Radon transform where the angular variable θ is constant and t is varying.

2.2 Ridgelet Pyramids

Let Q denote a dyadic square $Q = [k_1/2^s, (k_1+1)/2^s) \times [k_2/2^s, (k_2+1)/2^s)$ and let $\mathcal{Q}$ be the collection of all such dyadic squares. We write $\mathcal{Q}_s$ for the collection of all dyadic squares of scale s. Associated to the squares $Q \in \mathcal{Q}_s$ we construct a partition of energy as follows. With w a nice smooth window obeying $\sum_{k_1,k_2} w^2(x_1 - k_1, x_2 - k_2) = 1$, we dilate and transport w to all squares Q at scale s, producing a collection of windows (w_Q) such that the w_Q^2's, $Q \in \mathcal{Q}_s$, make up a partition of unity. We also let T_Q denote the transport operator acting on functions g via

$$(T_Q g)(x_1, x_2) = 2^s g(2^s x_1 - k_1, 2^s x_2 - k_2).$$

With these notations, it is not hard to see that

$$f w_Q = \int \langle f, w_Q T_Q \psi_{a,b,\theta} \rangle T_Q \psi_{a,b,\theta} \frac{da}{a^3} db \frac{d\theta}{4\pi}$$

and, therefore, summing the above equality across squares at a given scale gives

$$f = \sum_{Q \in \mathcal{Q}_s} f w_Q^2 = \sum_Q \int \langle f, w_Q T_Q \psi_{a,b,\theta} \rangle w_Q T_Q \psi_{a,b,\theta} \frac{da}{a^3} db \frac{d\theta}{4\pi}. \tag{6}$$

The identity (6) expresses the fact that one can represent any function as a superposition of elements of the form $w_Q T_Q \psi_{a,b,\theta}$; that is, of ridgelet elements localized near the squares Q. For the function $T_Q \psi_{a,b,\theta}$ is the ridgelet ψ_{a_Q,θ_Q,b_Q} (2) with parameters obeying

$$a_Q = 2^{-s} a, \qquad \theta_Q = \theta, \qquad b_Q = b + k_1 2^{-s} \cos\theta + k_2 2^{-s} \sin\theta$$

and thus $w_Q T_Q \psi_{a,b,\theta}$ is a windowed ridgelet, supported near the square Q, hence the name *local ridgelet transform*.

The previous paragraph discussed the construction of local ridgelets of fixed length, roughly 2^{-s} (s fixed). Letting the scale s vary defines the multiscale ridgelet dictionary $\{\psi^Q_{a,b,\theta} : s \geq s_0, Q \in \mathcal{Q}_s, a > 0, b \in \mathbf{R}, \theta \in [0, 2\pi)\}$ by

$$\psi^Q_{a,b,\theta} = w_Q \, T_Q \psi_{a,b,\theta};$$

that is, a whole pyramid of local ridgelets at various lengths and locations. This is, of course, a massively overcomplete representation system and no formula like (6) is available for this multiscale ridgelet pyramid, because it is highly overcomplete.

3 An Approximate Digital Ridgelet Transform

So a basic strategy for calculating the continuous ridgelet transform is first to compute the Radon transform $Rf(t, \theta)$ and second, to apply a one-dimensional wavelet transform to the slices $Rf(\cdot, \theta)$. In this section we develop a digital procedure which is inspired by this viewpoint, and is realizable on n by n numerical arrays,

3.1 Fourier Strategy for Digital Radon Transform

A fundamental fact about the Radon transform is the projection-slice formula [13]:

$$\hat{f}(\lambda \cos\theta, \lambda \sin\theta) = \int Rf(t, \theta) e^{-i\lambda t} dt.$$

This says that the Radon transform can be obtained by applying the one-dimensional inverse Fourier transform to the two-dimensional Fourier transform restricted to radial lines going through the origin.

This of course suggests that approximate Radon transforms for digital data can be based on discrete fast Fourier transforms. This is a widely used approach, in the literature of medical imaging and synthetic aperture radar imaging, for which the key approximation errors and artifacts have been widely discussed. In outline, one simply does the following, for gridded data $(f(i_1, i_2))$, $0 \le i_1, i_2 < n - 1$.

1. *2D-FFT.* Compute the two-dimensional FFT of f giving the array $(\hat{f}(k_1, k_2))$, $-n/2 \le k_1, k_2 \le n/2 - 1$.

2. *Cartesian to Polar Conversion.* Using an interpolation scheme, substitute the sampled values of the Fourier transform obtained on the square lattice with sampled values of $\hat{f}$ on a polar lattice: that is, on a lattice where the points fall on lines going through the origin.

3. *1D-IFFT.* Compute the one-dimensional IFFT on each line, i.e. for each value of the angular parameter.

The use of this strategy in connection with ridgelet transforms has been discussed in the articles [16, 15].

3.2 A Polar Sampling Scheme for Digital Data

For our implementation of the Cartesian-to-polar conversion, we have used a pseudo-polar grid, in which the pseudo-radial variable has level sets which are squares rather than circles. Starting with Oppenheim and Mersereau [20] this grid has often been called the *concentric squares* grid in the signal processing literature; in the medical tomography literature it is associated with the *linogram*, while in [2] it is called the rectopolar grid; see this last reference for a complete bibliographic treatment. The geometry of the rectopolar grid is illustrated on Figure 3.2. We select $2n$ radial lines in the frequency plane obtained by connecting the origin to the vertices (k_1, k_2) lying on the boundary of the array (k_1, k_2) , i.e. such that k_1 or $k_2 \in \{-n/2, n/2\}$. The polar grid $\xi_{\ell,m}$ (ℓ serves to index a given radial line while the position of the point on that line is indexed by m) that we shall use is the intersection between the set of radial lines and that of cartesian lines parallel to the axes. To be more specific, the sample points along a radial line $\mathcal{L}$ whose angle with the vertical axis is less or equal to $\pi/4$ are obtained by intersecting $\mathcal{L}$ with the set of horizontal lines $\{x_2 = k_2,\, k_2 = -n/2, -n/2 + 1, \ldots, n/2\}$. Similarly, the intersection with the vertical lines $\{x_1 = k_1,\, k_1 = -n/2, -n/2 + 1, \ldots, n/2\}$ defines our sample points whenever the angle between $\mathcal{L}$ and the horizontal axis is less or equal to $\pi/4$. The cardinality of the rectopolar grid is equal to $2n^2$ as there are $2n$ radial lines and n sampled values on each of these lines. As a result, data structures associated with this grid will have a rectangular format. We observe that this choice corresponds to irregularly spaced values of the angular variable θ.

3.3 Interpolation to rectopolar Grid

To obtain samples on the rectopolar grid, we should, in general, interpolate from nearby samples at the Cartesian grid. In principle, compare [2, 16], the interpolation of Fourier transforms is a very delicate matter because of the well-known fact that the Fourier transform of an image is highly oscillatory, and the phase contains crucial information about

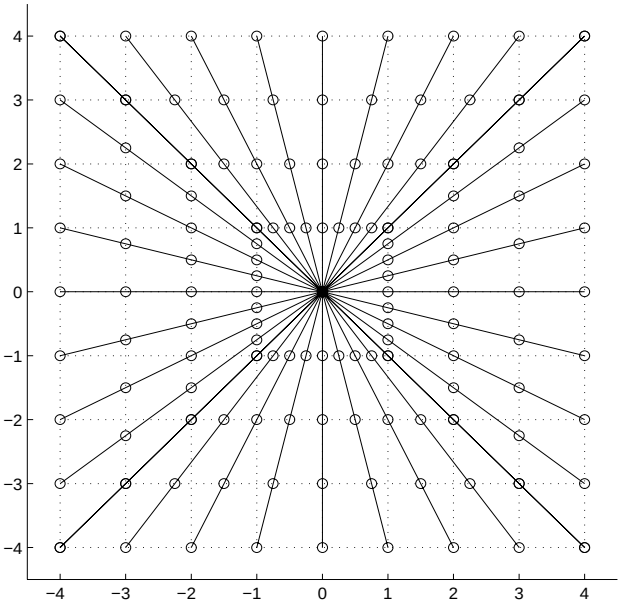

Figure 1: Illustration of the digital polar grid in the frequency domain for an n by n image ($n = 8$). The figure displays the set of radial lines joining pairs of symmetric points from the boundary of the square. The rectopolar grid is the set of points – marked with circles – at the intersection between those radial lines and those which are parallel to the axes.

the image. In our approach, however, we use a crude interpolation method: we simply impute for $\hat{f}(\xi_{\ell,m})$ the value of the Fourier transform taken at the point on the Cartesian grid nearest to $\xi_{\ell,m}$.

There are, of course, more sophisticated ways to realize the Cartesian-to-polar conversion; even simple bilinear interpolation would offer better theoretical accuracy. A very high accuracy approach used in [15] consists in viewing the data $(\hat{f}(k_1, k_2))$ as samples of the trigonometric polynomial F defined by

$$F(\omega_1, \omega_2) = \sum_{i_1=0}^{n-1} \sum_{i_2=0}^{n-1} f(i_1, i_2) \exp\{-i(\omega_1 i_1 + \omega_2 i_2)\} \tag{7}$$

on a square lattice; that is, with $\hat{f}(k_1, k_2) = F(\frac{2\pi k_1}{n}, \frac{2\pi k_2}{n})$ with $-n/2 \le k_1, k_2 < n/2$. There turns out [15, 2] to be an exact algorithm for rapidly finding the values of F on the polar grid. The high-accuracy approach can be used in reverse, allowing for exact reconstruction of the original trigonometric polynomial from its rectopolar samples.

Our nearest-neighbor interpolation, although admittedly simple-minded, happens to give good results in our application. In fact numerical experiments show that in overall system performance, it rivals the exact interpolation scheme. This is explainable as follows. Roughly speaking, the high-frequency terms in the trigonometric polynomial F are associated with pixels at the edge of the underlying n by n grid. Our crude interpolation evidently will fail at reconstructing high-frequency terms. However, in the curvelet application - see below - we use a window function to downweight the contributions of our reconstruction near the edge of the image array. So, inaccuracies in reconstruction caused

by our crude interpolation can be expected to be located mostly in regions which make little visual impact on the reconstruction.

A final point about our implementation. Since we are interested in noise removal artifact removal is very important. At the signal-to-noise ratios we consider, high-order-accuracy interpolation formulas which generate substantial artifacts (as many high-order formulas do) can be less useful than low-order-accuracy schemes which are relatively artifact-free. A known artifact of exact interpolation of trigonometric polynomials: substantial long-range disturbances can be generated by local perturbations such as discontinuities. In this sense, our crude interpolation may actually turn out to be preferable for some purposes.

3.4 Exact Reconstruction and Stability

The Cartesian-to-rectopolar conversion we have suggested here is reversible. That is to say, given the rectopolar values output from this method, one can recover the original Cartesian values exactly. To see this, take as given the following: **Claim:** *the assignment of Cartesian points as nearest neighbors of rectopolar points happens in such a way that each Cartesian point is assigned as the nearest neighbor of at least one rectopolar point.* It follows from this claim that each value in the original Cartesian input array is copied into at least one place in the the output rectopolar array. Hence, perfect reconstruction is obviously possible in principle – just keeping track of where the entries are have been copied to and undoing the process.

Our reconstruction rule obtains, for each point on the Cartesian grid, the *arithmetic mean of all the values in the rectopolar grid which have that Cartesian point as their nearest point.* This provides a numerically stable left inverse. Indeed, if applied to a perturbed set of rectopolar values, this rule gives an approximate reconstruction of the original unperturbed Cartesian values in which the approximation errors are smaller than the size of the perturbations suffered by the rectopolar values. (This final comment is reassuring in the present de-noising context, where our reconstructions will always be made by perturbing the empirical rectopolar FT of the noisy data.)

It remains to explain the italicized claim, because, as we have seen, from it flows the exact reconstruction property and stability of the inverse. Consider the rectopolar points in the hourglass region made of 'basically vertical lines', i.e. lines which make an angle less than $\pi/4$ with vertical, and more specifically those points on a single horizontal scan line. Assuming the scan line is not at the extreme top or bottom of the array, these points are spaced *strictly less than one unit apart*, where our unit is the spacing of the Cartesian grid. Therefore, when we consider a Cartesian grid point C belonging to this scan line and ask about the rectopolar points R_L and R_R which are closest to it on the left and right respectively, these two points cannot be as much as 1 unit apart: $||R_L - R_R|| < 1$. Therefore at least one of the two points must be strictly less than $1/2$ unit away from the Cartesian Point: i.e. either $||R_L - C|| < 1/2$ or $||R_R - C|| < 1/2$. Without loss of generality suppose that $||R_L - C|| < 1/2$. Then clearly R_L has C as its closest Cartesian point. In short, every Cartesian point in the strict interior of the 'hourglass' associated with the 'basically vertical' lines arises as the strict closest Cartesian point of at least one rectopolar point. Similar statements can be made about points on the boundary of the hourglass, although the arguments supporting those statements are much simpler, essentially mere inspection. Similar statements can be made about the points in the transposed hourglass. The italicized claim is established.

3.5 One-dimensional Wavelet Transform

To complete the ridgelet transform, we must take a one-dimensional wavelet transform along the radial variable in Radon space. We now discuss the choice of digital one-dimensional wavelet transform.

Experience has shown that compactly-supported wavelets can lead to many visual artifacts when used in conjunction with nonlinear processing - such as hard-thresholding of individual wavelet coefficients - particularly for decimated wavelet schemes used at critical sampling. Also, because of the lack of localization of such compactly-supported wavelets in the frequency domain, fluctuations in coarse-scale wavelet coefficients can introduce fine-scale fluctuations; this is undesirable in our setting. Here we take a frequency-domain approach, where the discrete Fourier transform is reconstructed from the inverse Radon transform. These considerations lead us to use band-limited wavelet – whose support is compact in the Fourier domain rather than the time-domain. Other implementations have made a choice of compact support in the frequency domain as well [16, 15]. However, we have chosen a specific overcomplete system, based on work of Starck et al. [27, 29], who constructed such a wavelet transform and applied it to interferometric image reconstruction. The wavelet transform algorithm is based on a scaling function ϕ such that $\hat{\phi}$ vanishes outside of the interval $[-\nu_c, \nu_c]$. We defined the scaling function $\hat{\phi}$ as a renormalized B_3-spline

$$\hat{\phi}(\nu) = \frac{3}{2} B_3(4\nu),$$

and $\hat{\psi}$ as the difference between two consecutive resolutions

$$\hat{\psi}(2\nu) = \hat{\phi}(\nu) - \hat{\phi}(2\nu).$$

Because $\hat{\psi}$ is compactly supported, the sampling theorem shows than one can easily build a pyramid of $n + n/2 + \ldots + 1 = 2n$ elements, see [29] for details.

This transform enjoys the following features:

- The wavelet coefficients are directly calculated in the Fourier space. In the context of the ridgelet transform, this allows avoiding the computation of the one-dimensional inverse Fourier transform along each radial line.
- Each subband is sampled above the Nyquist rate, hence, avoiding aliasing –a phenomenon typically encountered by critically sampled orthogonal wavelet transforms [26].
- The reconstruction is trivial. The wavelet coefficients simply need to be co-added to reconstruct the input signal at any given point. In our application, this implies that the ridgelet coefficients simply need to be co-added to reconstruct Fourier coefficients.

This wavelet transform introduces an extra redundancy factor, which might be viewed as an objection by advocates of orthogonality and critical sampling. However, we note that our goal in this implementation is not data compression/efficient coding - for which critical sampling might be relevant - but instead noise removal, for which it well-known that overcompleteness can provide substantial advantages [10].

3.6 Combining the Pieces

Figure 2 shows the flowgraph of the ridgelet transform. The ridgelet transform of an image of size $n \times n$ is an image of size $2n \times 2n$, introducing a redundancy factor equal to 4.

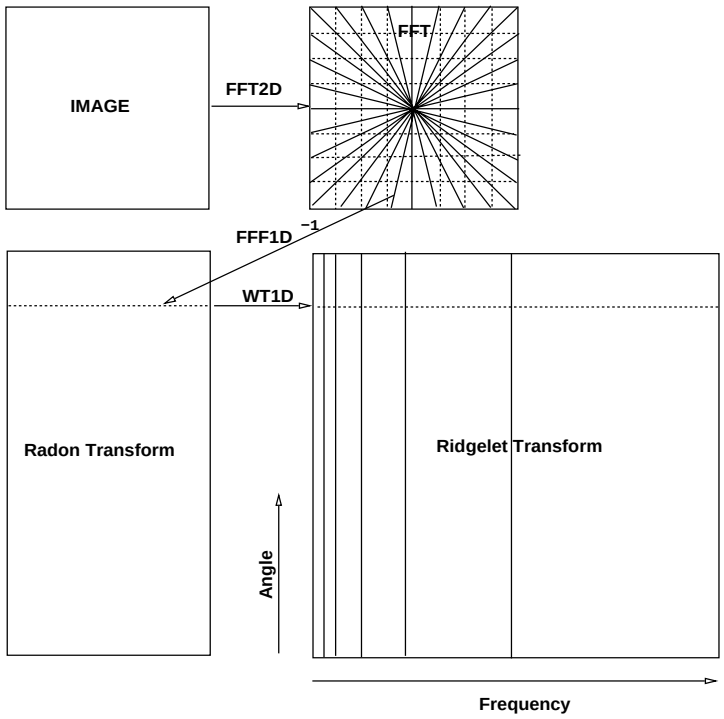

Figure 2: Ridgelet transform flowgraph. Each of the $2n$ radial lines in the Fourier domain is processed separately. The 1-D inverse FFT is calculated along each radial line followed by a 1-D nonorthogonal wavelet transform. In practice, the one-dimensional wavelet coefficients are directly calculated in the Fourier space.

We note that, because our transform is made of a chain of steps, each one of which is invertible, the whole transform is invertible, and so has the exact reconstruction property. For the same reason, the reconstruction is stable under perturbations of the coefficients.

Last but not least, our discrete transform is computationally attractive. Indeed, the algorithm we presented here has low complexity since it runs in $O(n^2 \log(n))$ flops for an $n \times n$ image.

3.7 Smooth Partitioning: Local Ridgelet Transforms

A digital version of the ideas presented in section 2.2 decomposes the original n by n image into smoothly overlapping blocks of sidelength b pixels in such a way that the overlap between two vertically adjacent blocks is a rectangular array of size b by $b/2$; we use overlap to avoid blocking artifacts. For an n by n image, we count $2n/b$ such blocks in each direction.

The partitioning introduces redundancy, as a pixel belongs to 4 neighboring blocks. We present two competing strategies to perform the analysis and synthesis:

1. The block values are weighted (analysis) in such a way that the co-addition of all blocks reproduce exactly the original pixel value (synthesis).

2. The block values are those of the image pixel values (analysis) but are weighted when the image is reconstructed (synthesis).

Of course, there are intermediate strategies and one could apply smooth windowing at both the analysis and synthesis stage as discussed in Section 2.2, for example. In the first approach, the data are smoothly windowed and this presents the advantage to limit the analysis artifacts traditionally associated with boundaries. The drawback, however, is a loss of sensitivity. Indeed, suppose for sake of simplicity that a vertical line with intensity level L intersects a given block of size b. Without loss of generality assume that the noise standard deviation is equal to 1. When the angular parameter of the Radon transform coincides with that of the line, we obtain a measurement with a signal intensity equal to bL while the noise standard deviation is equal to $\sqrt{b}$ (in this case, the value of the Signal to Noise Ratio (SNR) is $\sqrt{b}L$). If weights are applied at the analysis stage, the SNR is roughly equal to $L \sum_{i=1}^{b} w_i / \sqrt{\sum_{i=1}^{b} w_i^2} < \sqrt{b}L$. Experiments have shown that this sensitivity loss may have substantial effects in filtering applications and, therefore, the second approach seems more appropriate since our goal is image restoration.

We calculate a pixel value, $f(i,j)$ from its four corresponding block values of half-size $\ell = b/2$, namely, $B_1(i_1, j_1)$, $B_2(i_2, j_1)$, $B_3(i_1, j_2)$ and $B_4(i_2, j_2)$ with $i_1, j_1 > b/2$ and $i_2 = i_1 - \ell, j_2 = j_1 - \ell$, in the following way:

$$\begin{aligned} f_1 &= w(i_2/\ell)B_1(i_1, j_1) + w(1 - i_2/\ell)B_2(i_2, j_1) \\ f_2 &= w(i_2/\ell)B_3(i_1, j_2) + w(1 - i_2/\ell)B_4(i_2, j_2) \\ f(i,j) &= w(j_2/\ell)f_1 + w(1 - j_2/\ell)f_2 \end{aligned} \tag{8}$$

with $w(x) = \cos^2(\pi x/2)$. Of course, one might select any other smooth, nonincreasing function satisfying, $w(0) = 1$, $w(1) = 0$, $w'(0) = 0$ and obeying the symmetry property $w(x) + w(1 - x) = 1$.

It is worth mentioning that the spatial partitioning introduces a redundancy factor equal to 4.

4 Digital Curvelet Transform

4.1 Discrete Curvelet Transform of Continuum Functions

We now briefly return to the continuum viewpoint of Section 2.2. Suppose we set an initial goal to produce a decomposition using the multiscale ridgelet pyramid. The hope is that this would allow us to use thin 'brushstrokes' to reconstruct the image, with all lengths and widths available to us. In particular, this would seem allow us to trace sharp edges precisely using a few elongated elements with very narrow widths.

As mentioned in Section 2.2, the full multiscale ridgelet pyramid is highly overcomplete. As a consequence, convenient algorithms like simple thresholding will not find sparse decompositions when such good decompositions exist. An important ingredient of the curvelet transform is to restore sparsity by reducing redundancy across scales. In detail, one introduces interscale orthogonality by means of subband filtering. Roughly speaking, different levels of the multiscale ridgelet pyramid are used to represent different subbands of a filter bank output. At the same time, this subband decomposition imposes a relationship between the width and length of the important frame elements so that they are anisotropic and obey $width = length^2$.

The discrete curvelet transform of a continuum function $f(x_1, x_2)$ makes use of a dyadic sequence of scales, and a bank of filters $(P_0 f, \Delta_1 f, \Delta_2 f, \ldots)$ with the property that the

passband filter Δ_s is concentrated near the frequencies $[2^{2s}, 2^{2s+2}]$, e.g.

$$\Delta_s = \Psi_{2s} * f, \quad \widehat{\Psi_{2s}}(\xi) = \widehat{\Psi}(2^{-2s}\xi).$$

In wavelet theory, one uses a decomposition into dyadic subbands $[2^s, 2^{s+1}]$. In contrast, the subbands used in the discrete curvelet transform of continuum functions have the nonstandard form $[2^{2s}, 2^{2s+2}]$. This is nonstandard feature of the discrete curvelet transform well worth remembering.

With the notations of section 2.2, the curvelet decomposition is the sequence of the following steps:

- *Subband Decomposition.* The object f is decomposed into subbands:

$$f \mapsto (P_0 f, \Delta_1 f, \Delta_2 f, \ldots).$$

- *Smooth Partitioning.* Each subband is smoothly windowed into "squares" of an appropriate scale (of sidelength $\sim 2^{-s}$):

$$\Delta_s f \mapsto (w_Q \Delta_s f)_{Q \in \mathcal{Q}_s}.$$

- *Renormalization.* Each resulting square is renormalized to unit scale

$$g_Q = (T_Q)^{-1}(w_Q \Delta_s f), \qquad Q \in \mathcal{Q}_s. \tag{9}$$

- *Ridgelet Analysis.* Each square is analyzed via the discrete ridgelet transform.

In this definition, the two dyadic subbands $[2^{2s}, 2^{2s+1}]$ and $[2^{2s+1}, 2^{2s+2}]$ are merged before applying the ridgelet transform.

4.2 Digital Realization

In developing a transform for digital n by n data which is analogous to the discrete curvelet transform of a continuous function $f(x_1, x_2)$, we replace each of the continuum concepts with the appropriate digital concept mentioned in sections above. In general, the translation is rather obvious and direct. However, experience shows that one modification is essential; we found that, rather than merging the two the two dyadic subbands $[2^{2s}, 2^{2s+1}]$ and $[2^{2s+1}, 2^{2s+2}]$ as in the theoretical work, in the digital application, leaving these subbands separate, applyingh spatial partitioning to each subband and applying the ridgelet transform on each subband separately led to improved visual and numerical results.

We believe that the "à trous" subband filtering algorithm is especially well-adapted to the needs of the digital curvelet transform. The algorithm decomposes an n by n image I as a superposition of the form

$$I(x, y) = c_J(x, y) + \sum_{j=1}^{J} w_j(x, y),$$

where c_J is a coarse or smooth version of the original image I and w_j represents 'the details of I' at scale 2^{-j}, see [29] for more information. Thus, the algorithm outputs $J+1$ subband arrays of size $n \times n$. (The indexing is such that, here, $j = 1$ corresponds to the finest scale (high frequencies).)

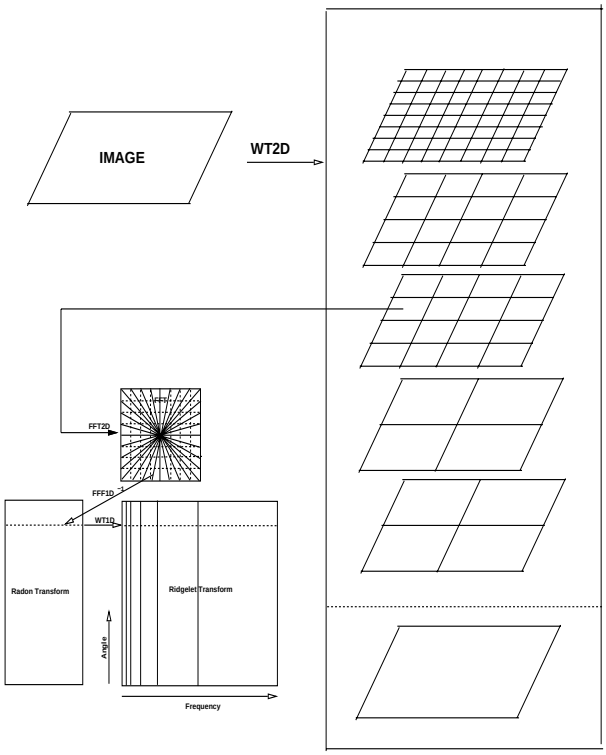

Figure 3: Curvelet transform flowgraph. The figure illustrates the decomposition of the original image into subbands followed by the spatial partitioning of each subband. The ridgelet transform is then applied to each block.

4.3 Algorithm

We now present a sketch of the discrete curvelet transform algorithm:

1. apply the à trous algorithm with J scales,
2. set $B_1 = B_{min}$,
3. for $j = 1, \ldots, J$ do,
   - partition the subband w_j with a block size B_j and apply the digital ridgelet transform to each block,
   - if j modulo $2 = 1$ then $B_{j+1} = 2B_j$,
   - else $B_{j+1} = B_j$.

The sidelength of the localizing windows is doubled *at every other* dyadic subband, hence maintaining the fundamental property of the curvelet transform which says that elements of length about $2^{-j/2}$ serve for the analysis and synthesis of the j-th subband $[2^j, 2^{j+1}]$. Note also that the coarse description of the image c_J is not processed. We used the default value $B_{min} = 16$ pixels in our implementation. Finally, Figure 3 gives an overview of the organization of the algorithm.

This implementation of the curvelet transform is also redundant. The redundancy factor is equal to $16J + 1$ whenever J scales are employed. Finally, the method enjoys exact reconstruction and stability, because these invertibility holds for each element of the processing chain.

5 Filtering

We now apply our digital transforms for removing noise from image data. The methodology is standard and is outlined mainly for the sake of clarity and self-containedness.

5.1 Gaussian Observations

Suppose that one is given noisy data of the form

$$x_{i,j} = f(i,j) + \sigma z_{i,j},$$

where f is the image to be recovered and z is white noise, i.e. $z_{i,j} \overset{i.i.d.}{\sim} N(0,1)$. Unlike FFT's or FWT's, our discrete ridgelet (resp. curvelet) transform is not norm-preserving and, therefore, the variance of the noisy ridgelet (resp. curvelet) coefficients will depend on the ridgelet (resp. curvelet) index λ. For instance, letting F denote the discrete curvelet transform matrix, we have $Fz \overset{i.i.d.}{\sim} N(0, FF^T)$. Because the computation of FF^T is prohibitively expensive, we calculated an approximate value $\tilde{\sigma}_\lambda^2$ of the individual variances using Monte-Carlo simulations where the diagonal elements of FF^T are simply estimated by evaluating the curvelet transforms of a few standard white noise images.

Let y_λ be the noisy curvelet coefficients ($y = Fx$). We use the following hard-thresholding rule for estimating the unknown curvelet coefficients:

$$\hat{y}_\lambda = y_\lambda \quad \text{if} \quad |y_\lambda|/\sigma \geq k\tilde{\sigma}_\lambda \tag{10}$$

$$\hat{y}_\lambda = 0 \quad \text{if} \quad |y_\lambda|/\sigma < k\tilde{\sigma}_\lambda. \tag{11}$$

In our experiments, we actually chose a scale-dependent value for k; we have $k = 4$ for the first scale ($j = 1$) while $k = 3$ for the others ($j > 1$).

5.2 Poisson Observations

Assume now that we have Poisson data $x_{i,j}$ with unknown mean $f(i,j)$. The Anscombe transformation [1]

$$\tilde{x} = 2\sqrt{x + \frac{3}{8}} \tag{12}$$

stabilizes the variance and we have $\tilde{x} = 2\sqrt{f} + \epsilon$ where ϵ is a vector with independent and approximately standard normal components. In practice, this is a good approximation whenever the number of counts is large enough, greater than 30 per pixel, say.

For small number of counts, a possibility is to compute the Radon transform of the image, and then to apply the Anscombe transformation to the Radon data. The rationale being that, roughly speaking, the Radon transform corresponds to a summation of pixel values over lines and that the sum of independent Poisson random variables is a Poisson random variable with intensity equal to the sum of the individual intensities. Hence, the intensity of the sum may be quite large (hence validating the Gaussian approximation) even though the individual intensities may be small. This might be viewed as an interesting feature as unlike wavelet transforms, the ridgelet and curvelet transforms tend to average data over elongated and rather large neighborhoods.

6 Filtering Experiments

6.1 Who Else?

In our first example, a Gaussian noise with a standard deviation equal to 20 was added to the classical `Lenna` image (512 by 512). Several methods were used to filter the noisy image:

1. Thresholding of Monoscale ridgelet transforms with scale (= block size) (8, 16, 32 and 64),
2. Thresholding of Curvelet transform, and
3. Wavelet de-noising methods in the following four families:
   (a) Bi-orthogonal wavelet transform using the Dauchechies-Antonini 7/9 filters (FWT-7/9) and hard thresholding.
   (b) Undecimated bi-orthogonal wavelet transform (UWT-7/9) with hard thresholding; we used $k = 4$ for the finest scale, and 3 for the others.
   (c) Multiscale entropy processing using the undecimated wavelet transform. This method is discussed in [30, 28].
   (d) Wavelet-domain Hidden Markov Models (WHMM) using Daubechies orthonormal wavelets of length 8. This method [11] attempts to model the joint probability density of the wavelet coefficients and derives the filtered coefficients using an empirical Bayesian approach. We used this rather than a competing method of Simoncelli [25] owing to availability of a convenient software implementation.

We use the PSNR as an 'objective' measure of performance. In addition, we used our own visual capabilities to identify artifacts whose effects may not be well-quantified by the PSNR value. The sort of artifacts we are particularly concerned about may be seen on display in the upper right panel of Figure 4, which displays a wavelet reconstruction. This image has a number of problems near edges. In reconstructing some edges which should follow smooth curves one gets edges which are poorly defined and very choppy in reconstruction (for example in the crown of the hat); also some edges which are accurately reconstructed exhibit oscillatory structure along the edge which is not present in the underlying image (for example in the shoulder and the hat brim). We refer to all such effects as artifacts.

Our experiments are reported on Figures 4 and 5. The latter figure represents a detail of the original image and helps the reader observe the qualitative differences between the different methods. We observe that:

- The curvelet transform enjoys superior performance over local ridgelet transforms, regardless of the the block size, and
- The undecimated wavelet transform approach outputs a PSNR comparable to that obtained via the curvelet transform (the PSNR is slightly better for the multiscale entropy method).
- The curvelet reconstruction does not contain the quantity of disturbing artifacts along edges that one sees in wavelet reconstructions. An examination of the details of the restored images (Figure 5) is instructive. One notices that the decimated wavelet

Method	PSNR	Comments
Noisy image	22.13	
FWT7-9 + Universal Hard thresh.	28.35	many artifacts
UWT7-9 + ksigma Hard thresh.	31.94	very few artifact
UWT7-9 + Multiscale entropy	32.10	very few artifact
WHMM	30.80	some noise remains
Local ridgelets ($B = 8$)	29.99	artifacts
Local ridgelets ($B = 16$)	30.87	few artifacts
Local ridgelets ($B = 32$)	30.97	few artifacts
Local ridgelets ($B = 64$)	30.79	few artifacts
Curvelets ($B = 16$)	31.95	very few artifact

Table 1: Table of PSNR values after filtering the noisy image (`Lenna` + Gaussian white noise (sigma = 20)). Images are available at "http://www-stat.stanford.edu/~jstarck/lena.html.

transform exhibits distortions of the boundaries and suffers substantial loss of important detail. The undecimated wavelet transform gives better boundaries, but completely omits to reconstruct certain ridges in the hatband. In addition, it exhibits numerous small-scale embedded blemishes; setting higher thresholds to avoid these blemishes would cause even more of the intrinsic structure to be missed.

- The curvelet reconstructions display higher sensitivity than the wavelet-based reconstructions. In fact both wavelet reconstructions obscure structure in the hatband which was visually detectible in the noisy panel at upper left. In comparison, every structure in the image which is visually detectible in the noisy image is clearly displayed in the curvelet reconstruction.

These observations are not limited to the particular experiment displayed here. We have observed similar characteristics in many other experiments; see Figure 9 for another example. Further results are visible at the website with URL

`http://www-stat.stanford.edu/~jstarck`

To study the dependency of the curvelet denoising procedure on the noise level, we generated a set of noisy images (the noise standard deviation varies from 5 to 100) from both `Lenna` and `Barbara`. We then compared the three different filtering procedures based respectively on the curvelet transform and on the undecimated/decimated wavelet transforms. This series of experiments is summarized in Figure 6 which displays the PSNR versus the noise standard deviation. These experimental results show that the curvelet transform outperforms wavelets for removing noise from those images, as the curvelet PSNR is systematically higher than the wavelet PSNR's – and this, across a broad range of noise levels. Other experiments with color images led to similar results.

6.2 Recovery of Linear Features

The next experiment (Figure 7) consists of an artificial image containing a few bars, lines and a square. The intensity is constant along each individual bar; from left to right, the intensities of the ten vertical bars (these are in fact thin rectangles which are 4 pixels

wide and 170 pixels long) are equal to $\frac{32}{2^i}$, $i = 0, \ldots 9$. The intensity along all the other lines is equal to 1, and the noise standard deviation is 1/2. Displayed images have been log-transformed in order to better see the results at low signal to noise ratio.

The curvelet reconstruction of the nonvertical lines is obviously sharper than that obtained using wavelets. The curvelet transform also seems to go one step further as far as the reconstruction of the vertical lines is concerned. Roughly speaking, for those templates, the wavelet transforms stops detecting signal at a SNR equal to 1 (we defined here the SNR as the intensity level of the pixels on the line, divided by the noise standard deviation of the noise) while the cut-off value equals 0.5 for the curvelet approach. It is important to note that the horizontal and vertical lines correspond to privileged directions for the wavelet transform, because the underlying basis functions are direct products of functions varying solely in the horizontal and vertical directions. Wavelet methods will given even poorer results on lines of the same intensity but tilting substantially away from the cartesian axes. Compare the reconstructions of the faint diagonal lines in the image.

6.3 Denoising of a Color Image

In a wavelet based denoising scenario, color RGB images are generally mapped into the YUV space, and each YUV band is then filtered independently from the others. The goal here to see whether the curvelet transform would give improved results. We used four of the classical color images, namely `Lenna`, `Peppers`, `Baboon`, and `Barbara` (all images except perhaps `Barbara` are available from the USC-SIPI Image Database [12]. We performed the series of experiments described in section 6.1 and summarized our findings on Figure 8 which again displays the PSNR versus the noise standard deviation for the four images. In all cases, the curvelet transform outperforms the wavelet transforms in terms of PSNR – at least for moderate and large values of the noise level. In addition, the curvelet transform outputs images that are visually more pleasant. Figure 9 illustrates this last point. For other examples, please check `http://www-stat.stanford.edu/~jstarck`.

7 Conclusion

In this paper, we presented a strategy for digitally implementing both the ridgelet and the curvelet transforms. The resulting implementations have the exact reconstruction property, give stable reconstruction under perturbations of the coefficients, and as deployed in practice, partial reconstructions seem not to suffer from visual artifacts.

There are, of course, many competing strategies to translate the theoretical results on ridgelets and curvelets into digital representations. Guided by a series of experiments, we arrived at several innovative choices which we now highlight.

1. *Subband Definition.* We split the nonstandard frequency subband $[2^{2s}, 2^{2s+2}]$ – arising in theoretical treatments of curvelets [6]– into the two standard dyadic frequency subbands $[2^{2s}, 2^{2s+1}]$ and $[2^{2s+1}, 2^{2s+2}]$ and we processed each of them individually. This seems to give better result.

2. *Subband Filtering.* The à trous algorithm is well-adapted to the decomposition into subbands. For instance, an alternative strategy using a decimated 2-dimensional wavelet transform introduces visual artifacts near strong edges, in the form of curve fragments at 90 degree orientation to the underlying edge direction.

3. *Wavelet underlying the Ridgelet Transform.* Our ridgelet transform uses a 1-dimensional wavelet transform based on wavelets which are compactly supported in the Fourier domain. If a wavelet, compactly supported in space, is used instead, it appears that thresholding of ridgelet/curvelet coefficients may introduce many visual artifacts in the form of ghost oscillations parallel to strong linear features.

A remark about *which principles are important*: because we are working in a de-noising setting, the attraction of traditional transform desiderata – critical sampling and orthogonality – is weak. Instead, redundancy, and overcompleteness seem to offer advantages, particularly in avoiding visual artifacts.

The work presented here is an initial attempt to address the problem of image de-noising using digital analogs of some new mathematical transforms. Our experiments show that curvelet thresholding rivals sophisticated techniques that have been the object of extensive development over the last decade. We find this encouraging, particularly as there seem to be numerous opportunities for further improvement. Areas for further work clearly include improved interpolation schemes, and improved folding strategies for space partitioning, to mention a few. On the other hand, the digital curvelet transform is nonorthogonal, quite redundant and as a consequence, the noisy coefficients are correlated and one should clearly design thresholding rules taking into account this dependency. There is an obvious tree structure with parent and children curvelet coefficients that might also be used effectively in this setting.

We also look forward to testing our transforms on larger datasets in order to fully exploit the multiscale nature of the curvelet transform. Images of size 2048 by 2048 or 4096 by 4096 would be a reasonable target, as those resolutions will undoubtedly become standard over the next few years. As images scale up, the asymptotic theory which suggests that curvelets outperform wavelets may become increasingly relevant. The quality of the local reconstructions as illustrated on the 'zoomed restored images' obtained via the curvelet transform are especially promising. We hope to report on this issue in a forthcoming paper.

Acknowledgments

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.

References

[1] F.J. Anscombe. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 15:246–254, 1948.

[2] A. Averbuch, R.R. Coifman, D.L. Donoho, M. Israeli, and J. Waldén. Polar fft, rectopolar fft, and applications. Technical report, Stanford University, 2000.

[3] E. J. Candès. Harmonic analysis of neural netwoks. *Applied and Computational Harmonic Analysis*, 6:197–218, 1999.

[4] E. J. Candès. *Monoscale ridgelets for the representation of images with edges.* Technical report, Department of Statistics, Stanford University, 1999. Submitted for publication.

[5] E. J. Candès. *On the representation of mutilated Sobolev functions*. Technical report, Department of Statistics, Stanford University, 1999. Submitted for publication.

[6] E. J. Candès and D. L. Donoho. *Curvelets*. Manuscript. *http://www-stat.stanford.edu/˜donoho/Reports/1999/curvelets.pdf*, 1999.

[7] E. J. Candès and D. L. Donoho. Curvelets – a surprisingly effective nonadaptive representation for objects with edges. In A. Cohen, C. Rabut, and L.L. Schumaker, editors, *Curve and Surface Fitting: Saint-Malo 1999*, Nashville, TN, 1999. Vanderbilt University Press.

[8] E. J. Candès and D. L. Donoho. Ridgelets: the Key to Higher-dimensional Intermittency? *Phil. Trans. R. Soc. Lond. A.*, 1999.

[9] E.J. Candès and D.L. Donoho. Edge-preserving denoising in linear inverse problems: Optimality of curvelet frames. Technical report, Department of Statistics, Stanford Univeristy, 2000.

[10] R.R. Coifman and D.L. Donoho. Translation invariant de-noising. In A. Antoniadis and G. Oppenheim, editors, *Wavelets and Statistics*, pages 125–150, New York, 1995. Springer-Verlag.

[11] M. Crouse, R. Nowak, and R. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Transactions on Signal Processing*, 46:886–902, 1998.

[12] USC-SIPI Image Database. `http://sipi.usc.edu/services/database/-Database.html`.

[13] S. R. Deans. *The Radon transform and some of its applications*. John Wiley & Sons, 1983.

[14] M. N. Do and M. Vetterli. Orthonormal finite ridgelet transform for image compression. In *Proc. of IEEE International Conference on Image Processing (ICIP)*, September 2000.

[15] D. L. Donoho. Fast ridgelet transforms in dimension 2. Technical report, Stanford University, Department of Statistics, Stanford CA 94305–4065, 1997.

[16] D. L. Donoho. Digital ridgelet transform via rectopolar coordinate transform. Technical report, Stanford University, 1998.

[17] D.L Donoho. Wedgelets: nearly-minimax estimation of edges. *Ann. Statist*, 27:859–897, 1999.

[18] D.L Donoho. Orthonormal ridgelets and linear singularities. *Siam J. Math Anal.*, 31(5):1062–1099, 2000.

[19] D.L. Donoho and M.R. Duncan. Digital curvelet transform: strategy, implementation and experiments. In H.H. Szu, M. Vetterli, W. Campbell, and J.R. Buss, editors, *Proc. Aerosense 2000, Wavelet Applications VII*, volume 4056, pages 12–29, Bellingham Washington, 2000. SPIE.

[20] R.M. Mersereau and A.V. Oppenheim. Digital reconstruction of multidimensional signals from their projections. *Proc. IEEE*, 62(10):1319–1338, 1974.

[21] T. Olson and J. DeStefano. Wavelet localization of the Radon transform. *IEEE Trans. on Signal Process.*, 42(8):2055–2067, 1994.

[22] B. Sahiner and A.E. Yagle. On the use of wavelets in inverting the radon transform. In *Nuclear Science Symposium and Medical Imaging Conference, 1992., Conference Record of the 1992*, volume 2, pages 1129–1131, 1992.

[23] B. Sahiner and A.E. Yagle. Iterative inversion of the radon transform using image-adaptive wavelet constraints. In *Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine., 18th Annual International Conference of the IEEE*, volume 2, pages 722–723, 1997.

[24] B. Sahiner and A.E. Yagle. Iterative inversion of the radon transform using image-adaptive wavelet constraints. In *Image Processing, 1998. ICIP 98. Proceedings.*, volume 2, pages 709–713, 1998.

[25] E. P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. In P Muller and B. Vidakovic, editors, *Bayesian Inference in Wavelet Based Models*. Springer-Verlag, 1999.

[26] E.P. Simoncelli, W.T Freeman, E.H. Adelson, and D.J. Heeger. Shiftable multi-scale transforms [or "what's wrong with orthonormal wavelets"]. *IEEE Trans. Information Theory*, 1992.

[27] J.L. Starck, A. Bijaoui, B. Lopez, and C. Perrier. Image reconstruction by the wavelet transform applied to aperture synthesis. *Astronomy and Astrophysics*, 283:349–360, 1994.

[28] J.L. Starck and F. Murtagh. Multiscale entropy filtering. *Signal Processing*, 76(2):147–165, 1999.

[29] J.L. Starck, F. Murtagh, and A. Bijaoui. *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge University Press, Cambridge (GB), 1998.

[30] 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.

[31] S. Zhao, G. Welland, and G. Wang. Wavelet sampling and localization schemes for the radon transform in two dimensions. *SIAM Journal on Applied Mathematics*, 57(6):1749–1762, 1997.

[32] R.A. Zuidwijk. The wavelet X-ray transform. Technical Report PNA-R9703, ISSN 1386-3711, Centre Math. Computer Sci., 1997.

[33] R.A. Zuidwijk and Paul M. de Zeeuw. The fast wavelet X-ray transform. Technical Report PNA-R9908, ISSN 1386-3711, Centre Math. Computer Sci., 1999.

Figure 4: Noisy image (top left) and filtered images using the decimated wavelet transform (top right), the undecimated wavelet transform (bottom left) and the curvelet transform (bottom right).

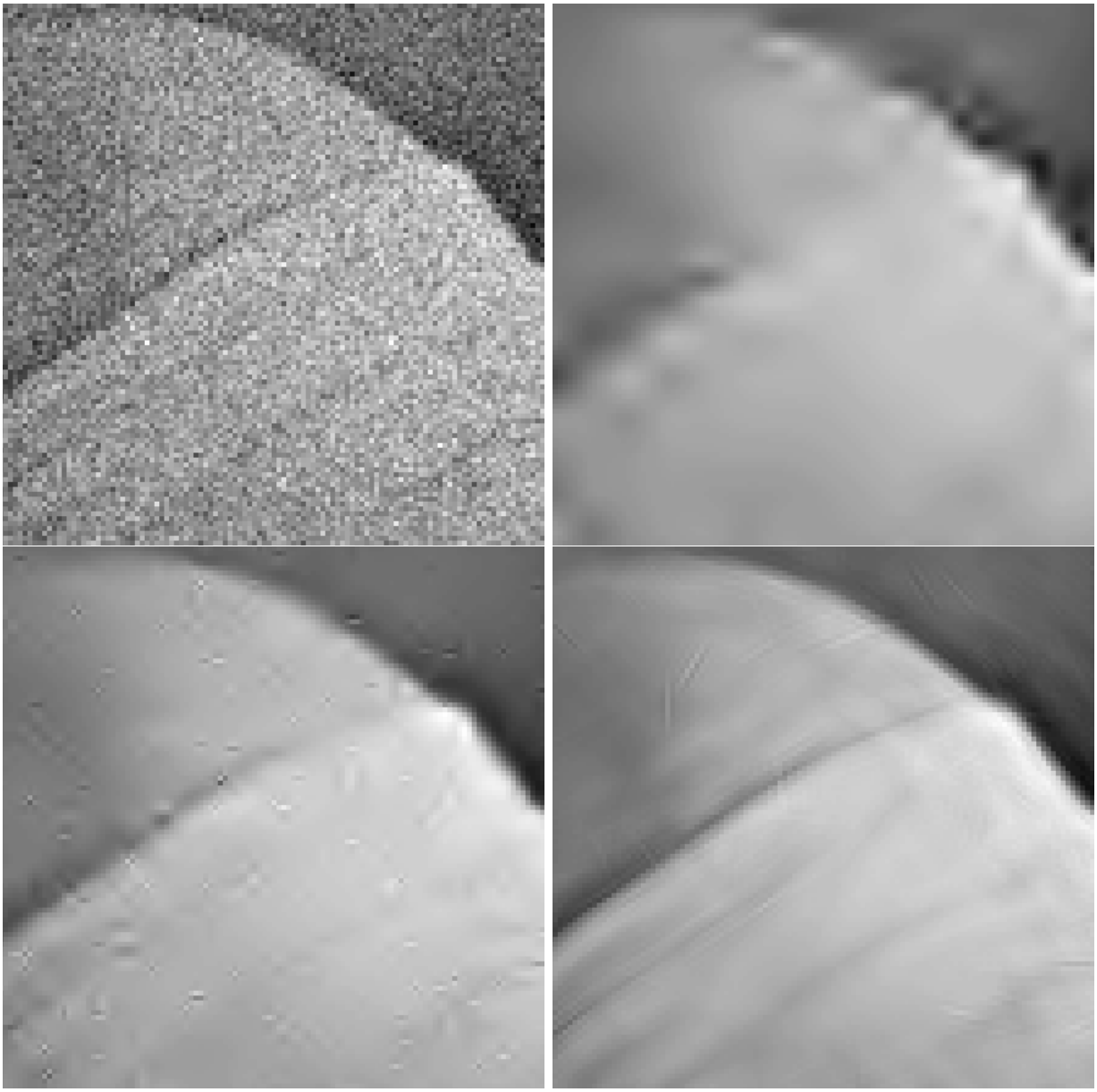

Figure 5: The figure displays the noisy image (top left), and the restored images after denoising by means of the DWT (top right), UWT (bottom left), and the curvelet transform (bottom right). The diagonal lines of the hat have been recovered with much greater fidelity in the curvelet approach.

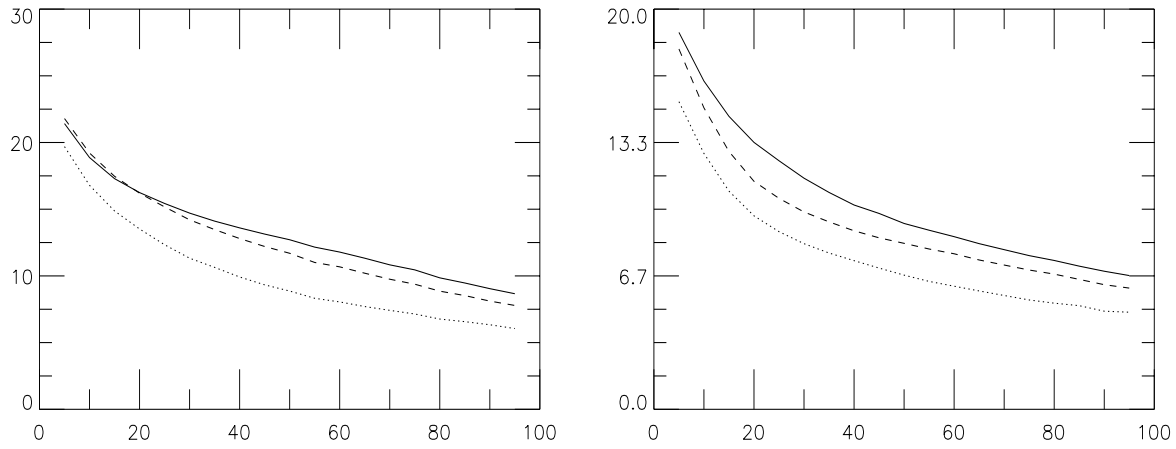

Figure 6: PSNR versus noise standard deviation for different denoising methods. The three methods based on the curvelet, undecimated and decimated wavelet transforms are represented with a continuous, dashed, and dotted line respectively. The left panel corresponds to `Lenna`, and the right to `Barbara`.

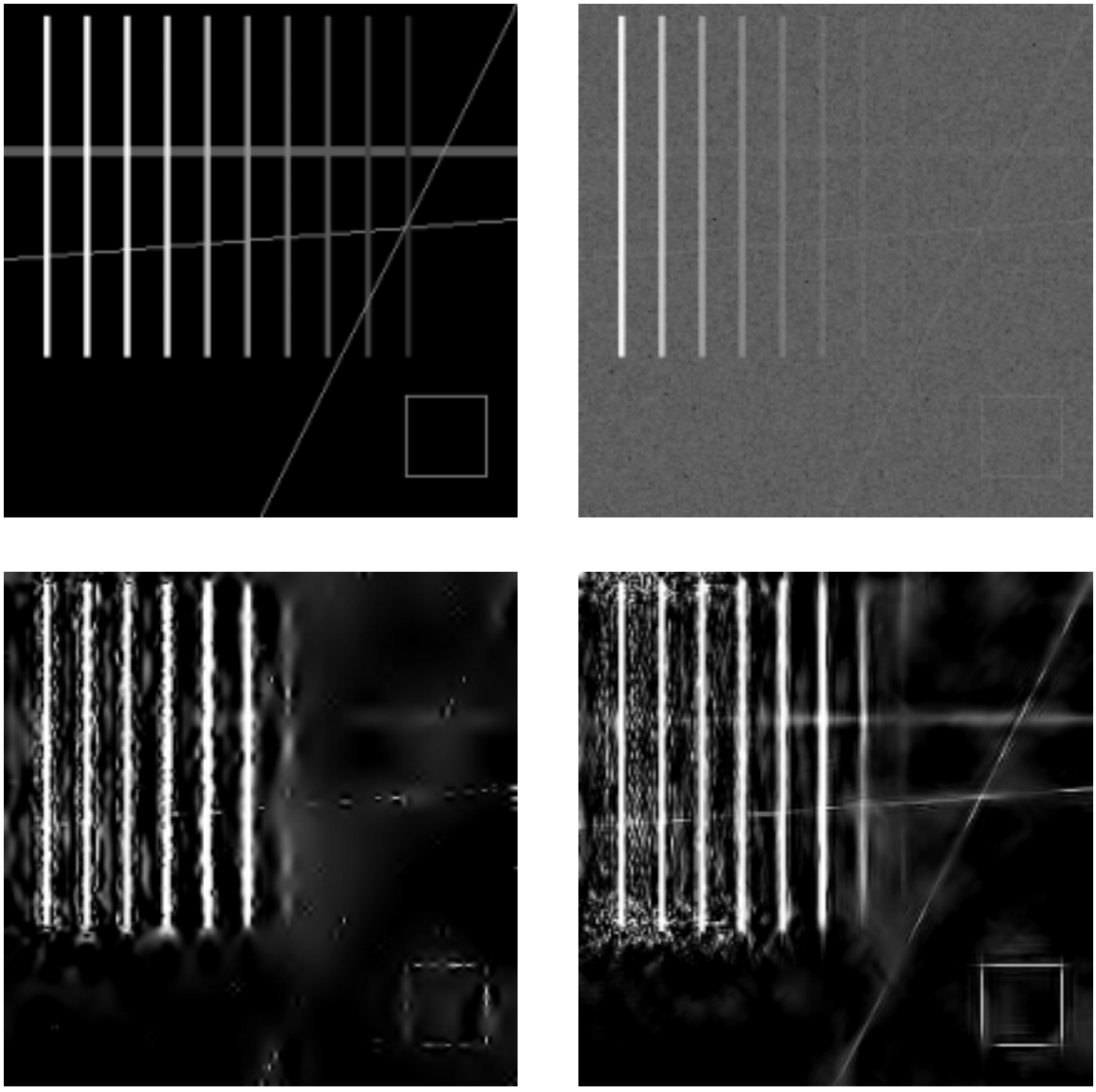

Figure 7: The top panels display a geometric image and that same image contaminated with a Gaussian white noise. The bottom left and right panels display the restored images using the undecimated wavelet transform and the curvelet transform respectively.

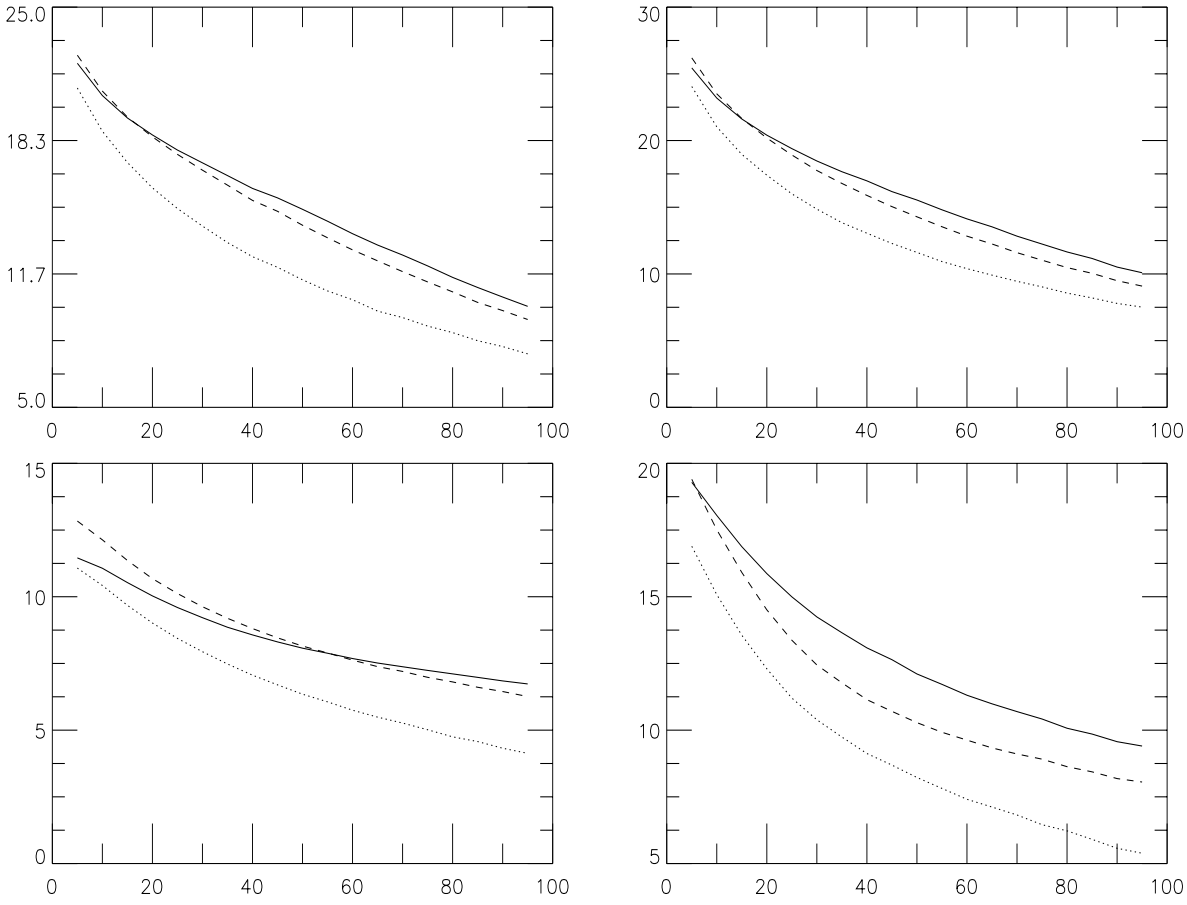

Figure 8: PSNR versus noise standard deviation using different filtering methods. YUV and curvelet, YUV and undecimated wavelet, and YUV and decimated wavelet transforms are represented respectively with a continuous, dashed, and dotted line. The upper left panel corresponds to `Lenna` (RGB), the upper right to `pepper` (RGB), the bottom left to `Baboon` (RGB), and the bottom right to `Barbara` (RGB).

Figure 9: Upper left: noisy `Barbara` image. Upper right: restored image after applying the curvelet transform. Details of the restored images are shown on the bottom left panel (undecimated wavelet transform) and right (curvelet transform) panel.