IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Slicedridgelet Transform for Image Denoising

V.Krishnanaik[*1], Dr.G.Manoj Someswar[2]

[*1] Assistant Professor, Department of Electrical & Computer Engineering, College of Engineering & Tech, Aksum University,Axsum, Ethiopia

[2] Professor in Computer Science and Engineering, Anwarul- uloom College of Engineering and Technology, Hyderabad, India

krishnanaik.ece@gmail.com

Abstract

In this paper the image denoising based on ridgelet transforms gives better result in imagedenoising than standard wavelet transforms. In this project we introduces an new approach for image denoising that is based on ridgelets computed in a localized manner and that is computationally less intensive than curvelets, but similar donising performance. The projection of image at a certain angle is computed at a certain angle, but only on a defined slice of the noisy image. After that ridgelet transform of each slice is computed , to produce the ridgelet coefficients for an image.The denoising operation corresponds to a simple thresholding of these ridgelet coefficients.

The new method for image denoising technique is based on two operations: one is the redundant directional wavelet transform based on the radon transform, and thresholding of the ridgelet coefficient.

The image denoising algorithm with the ridgelet transform can be described by the following operations. First , add the noise to an image and than projection(radon transform) is computed at a certain angle of the noise image. After that, the ridgelet transform of this projection of the noise image is computed and the noise component is reduced by simple thresholding of the ridgelet coefficient. Then, the inverse ridgelet transform is computed to get back the denoised version of that projection of slice at the same angle. Although the shape of the reconstructed object can be seen, the reconstructed image is heavily blurred. To counteract this effect, a high pass filter is applied to the sinogram data in the frequency domain. This is achieved by applying a 1-D DFT to the sinogram data for each angle, multiply by the filter, and then using the inverse DFT to reconstruct the data. The simplest form of high pass filter is a ramp. Applying the ramp filter significantly improves the quality of the reconstructed image. However, because the ramp filter emphasises high frequency components of the image, it can cause unwanted noise. To counteract this, several other high-pass filters are commonly used. In this project we are using Adaptive Filtering. The wiener2 function applies a Wiener filter (a type of linear filter) to an image adaptively, tailoring itself to the local image variance. Where the variance is large, wiener2 performs little smoothing. Where the variance is small, wiener2 performs more smoothing. This approach often produces better results than linear filtering. The adaptive filter is more selective than a comparable linear filter, preserving edges and other high-frequency parts of an image. In addition, there are no design tasks; the wiener2 function handles all preliminary computations and implements the filter for an input image. wiener2.

However, it does require more computation time than linear filtering. Wiener2 works best when the noise is constant-power ("white") additive noise, such as Gaussian noise.

Keywords: Denoising, Sliced Ridgelet Transform, Wiener2, DFT, Radon Transform, Projection.

Introduction

Image denoising is one of the most popular research fields in image processing due to fact that it is extremely difficult to form a general global denoising scheme effective for all types of noise as well as all types images. A common problem in image denoising is the blurring of the prominent edges in the image which can cause discrepancies when the denoising operation is combined with other operations such as image edge detection and segmentation. Notewothy schemes for image denoising based on transforms such as wavelets, curvelets, exploit redundancy and thresholding to remove the noise without blurring the edges.

The important characteristic of the denoising technique introduced in this project is that it can reduce considerably the noise without destroying the edges of the objects in the image. That is , the noise is well attenuated but the edge information is preserved. The new method for image denoising technique is based on two operations: one is the redundant directional wavelet transform based on the ridgelet transform, and thresholding of the ridgelet coefficient.

The image denoising algorithm with the ridgelet transform can be described by the following operations. First, add the noise to an image and thanprojection(ridgelet transform) is computed at a certain angle, but only on a defined silce of the noise image. After that, the ridgelet transform of this projection of the noise image is computed and the noise component is reduced by simple thresholding of the ridgeletcoefficient.Then, the inverse ridgelet transform is computed to get back the denoised version of that projection of the defined slice at the same angle. Although the shape of the reconstructed object can be seen, the reconstructed image is heavily blurred. To counteract this effect, a high pass filter is applied to the sinogram data in the frequency domain. The wiener2 function applies a Wiener filter (a type of linear filter) to an image adaptively, tailoring itself to the local image variance. Where the variance is large, wiener2 performs little smoothing. Where the variance issmall, wiener2 performs more smoothing. This approach often produces better results than linear filtering.

The adaptive filter is more selective than a comparable linear filter, preserving edges and other high-frequency parts of an image. In addition, there are no design tasks; the wiener2 function handles all preliminary computations and implements the filter for an input image. wiener2, however, does require more computation time than linear filtering. wiener2 works best when the noise is constant-power ("white") additive noise, such as Gaussian noise.

Experimental Design and Setups

Radon Transform: The Radon transform of an image is the sum of the Radon transforms of each individual pixel. The algorithm first divides pixels in the image into four subpixels and projects each subpixel separately.

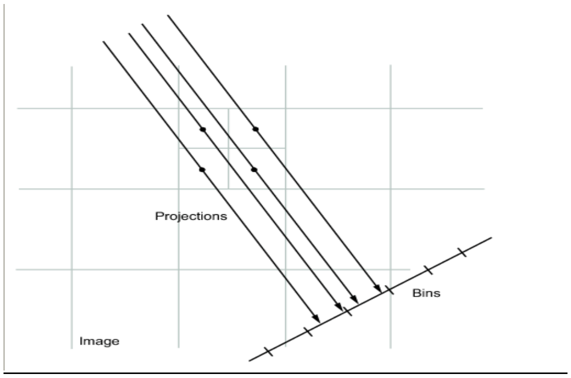


Figure 1.Subpixel Projection

Each subpixel's contribution is proportionally split into the two nearest bins, according to the distance between the projected location and the bin centers. If the subpixel projection hits the center point of a bin, the bin on the axes gets the full value of the subpixel, or one-fourth the value of the pixel. If the subpixel projection hits the border between two bins, the subpixel value is split evenly between the bins.

Radon Transformation Definition

The radon function computes projections of an image matrix along specified directions. A projection of a two-dimensional function f(x,y) is a set of line integrals. The radon function computes the line integrals from multiple sources along parallel paths, or beams, in a certain direction. The beams are spaced 1 pixel unit apart. To represent an image, the radon function takes multiple, parallel-beam projections of the image from different angles by rotating the source around the center of the image. The following figure shows a single projection at a specified rotation angle.

Parallel-Beam Projection at Rotation Angle Theta

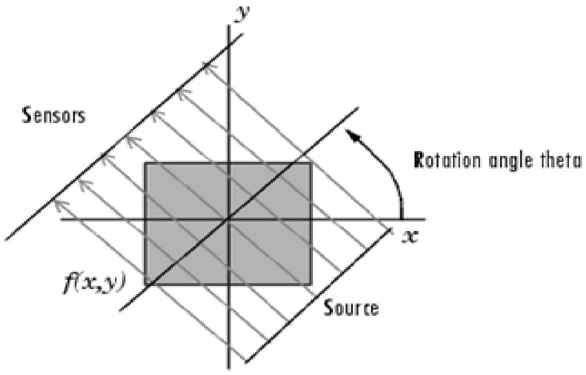


Figure 2.Parallel beam Projection.

For example, the line integral of *f(x,y)* in the vertical direction is the projection of *f(x,y)* onto the *x*-

axis; the line integral in the horizontaldirection is the projection of *f(x,y)* onto the *y*-axis. The following figure shows horizontal and vertical projections for a simple two-dimensional function.

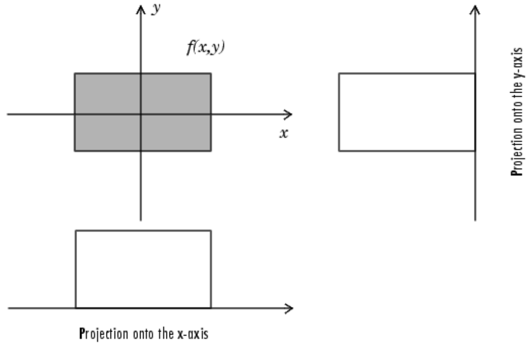


Figure 3. Horizontal and vertical Projection.

Projections can be computed along any angle θ. In general, the Radon transform of *f(x,y)* is the line integral of *f* parallel to the *y´*-axis

$$R_\theta(x') = \int_{-\infty}^{\infty} f(x'\cos\theta - y'\sin\theta,\ x'\sin\theta + y'\cos\theta)\, dy'$$

Where

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

The following figure illustrates the geometry of the Radon transform.

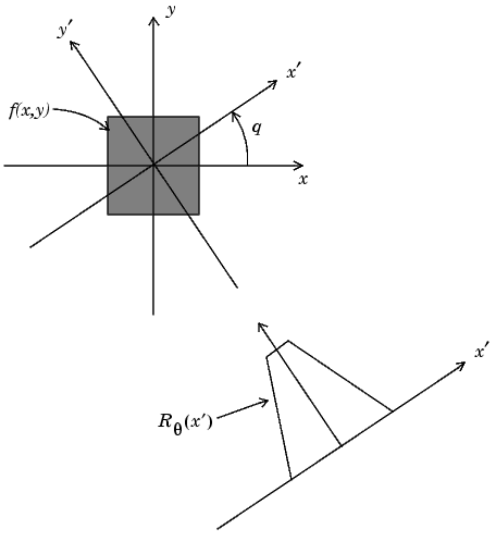


Figure 4.Randon transform projection

Geometry of the Radon Transform

This command computes the Radon transform of I for the angles specified in the vector theta.

- [R,xp] = radon(I,theta);

The columns of R contain the Radon transform for each angle in theta. The vector xp contains the corresponding coordinates along the *x´*-axis. The center pixel of I is defined to be floor((size(I)+1)/2); this is the pixel on the *x´*-axiscorresponding to $x' = 0$.

The Radon transform of an image represented by the function f(x,y) can be defined as a series of line integrals through f(x,y) at different offsets from the origin. It can be defined mathematically as

$$R(p,\tau) = \int_{-\infty}^{\infty} f(x, px+\tau)\, dx$$

where p and tau are the slope and intercepts of the line.

A more directly applicable form of the transform can be defined by using a delta function:

$$R(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\,\delta(x\cos\theta + y\sin\theta - r)\,dx\,dy$$

where theta is the angle of the line, and r is the perpendicular offset of the line.

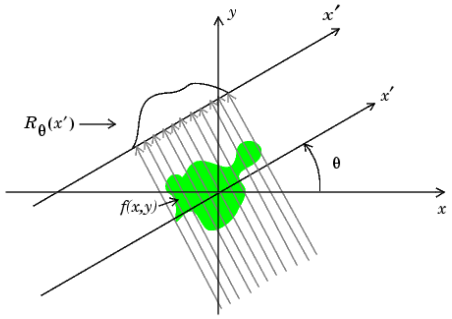


Figure 5. Radon transform projection angle theta,tau

Radon Transform

The acquisition of data in medical imaging techniques such as MRI, CT and PET scanners involves a similar method of projecting a beam through an object, and the data is in a similar form to that described in the second equation above. The plot of the Radon transform, or scanner data, is referred to as a *sinogram* due to its characteristic sinusoid shape. The *Figure* shows a simple non-homogeneous shape and the sinogram created by taking the Radon transform at intervals of one degree from 0 to 180 degrees.The sinogram produced by applying the Radon Transform

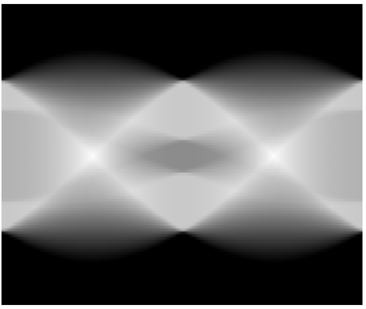


Figure 6.Radon transform scanner data (Sinogram)

Invers Radon TransformUsing filtered back Projection

To reconstruct the image from the sinogram, the inverse Radon transform is applied to the image. There are several techniques by which the inverse transform can be calculated but the most common is Filtered Back Projection. The filtered back projection algorithm is split into two phases, filtration and projection.

Projection

The projection phase is very similar to the Radon transform described above, and shown in*Figure* except now the line integrals are projected back onto the plane at their respective angles. The projection phase of the Filtered Back Projection, using the data format described above:

$$f(x,y) = \int_0^{\pi} f'(x\cos\theta + y\sin\theta, \theta)$$

where f' is the filtered data.
A common discrete approximation of this is:

$$f(x_m, y_n) \approx \Delta\theta \sum_{t=0}^{T-1} f'(x_m\cos\theta_t + y_n\sin\theta_t, \theta_t)$$

This equation can be used to determine the pixel values at a given point. The exact values depend on the chosen interpolation method, e.g. nearest-neighbour, linear interpolation etc. As more projections are added, the quality of the reconstruction will increase. This is clearly not enough to accurately reconstruct the image, but the individual projections can be seen. the result of the back projection algorithm applied as described and using all available data. Although the shape of the reconstructed object can be seen, the reconstructed image is heavily blurred. To counteract this effect, a high pass filter is applied to the sinogram data in the frequency domain.

This is achieved by applying a 1-D DFT to the sinogram data for each angle, multiply by the filter, and then using the inverse DFT to reconstruct the data. The simplest form of high pass filter is a ramp. Applying the ramp filter significantly improves the quality of the reconstructed image. However, because the ramp filter emphasises high frequency components of the image, it can cause unwanted noise. To counteract this, several other high-pass filters are commonly used. See the reference for a full discussion, including derivations, on the use of filtration.

Adaptive filter

The wiener2 function applies a Wiener filter (a type of linear filter) to an image adaptively, tailoring itself to the local image variance. Where the variance is large, wiener2 performs little smoothing. Where the variance is small, wiener2 performs more smoothing.This approach often produces better results than linear filtering. The adaptive filter is more selective than a comparable linear filter, preserving edges and other high-frequency parts of an image.

In addition, there are no design tasks; the wiener2 function handles all preliminary computations and implements the filter for an input image. wiener2, however, does require more computation time than linear filtering. wiener2 works best when the noise is constant-power ("white") additive noise, such as Gaussian noise.

Discrete Wavelet Transform

Introduction: The Discrete Wavelet Transform (DWT) involves choosing scales and positions based on powers of two. So called dyadic scales and positions. The mother wavelet is rescaled or dilated by powers of two and translated by integers. Specifically, a function f(t) $\in$ L2(R) (defines space of square integrable functions) can be represented as

$$f(t) = \sum_{j=1}^{L} \sum_{k=-\infty}^{\infty} d(j,k)\psi(2^{-j}t-k) + \sum_{k=-\infty}^{\infty} a(L,k)\phi(2^{-L}t-k)$$

The function ψ(t) is known as the mother wavelet, while φ(t) is known as the scaling Function. The set of functions

$$\{\sqrt{2^{-l}}\phi(2^{-L}t-k), \sqrt{2^{-j}}\psi(2^{-j}t-k) \mid j \le L, j,k,L \in Z\},$$

Where Z is the set of integers is an orthonormal basis for L2(R).

The numbers a(L, k) are known as the approximation coefficients at scale L, while d(j,k) are known as the detail coefficients at scale j.

The approximation and detail coefficients can be expressed as:

$$a(L,k)=\frac{1}{\sqrt{2^{L}}}\int_{-\infty}^{\infty} f(t)\phi(2^{-L}t-k)dt$$

$$d(j,k)=\frac{1}{\sqrt{2^{j}}}\int_{-\infty}^{\infty} f(t)\psi(2^{-j}t-k)dt$$

To provide some understanding of the above coefficients consider a projection fl(t) of the function f(t) that provides the best approximation (in the sense of minimum error energy) to f(t) at a scale l. This projection can be constructed from the coefficients a(L, k), using the equation

$$f_l(t)=\sum_{k=-\infty}^{\infty} a(l,k)\phi(2^{-l}t-k).$$

As the scale l decreases, the approximation becomes finer, converging to f(t) as l → 0. The difference between the approximation at scale l + 1 and that at l, fl+1(t) - fl(t), is completely described by the coefficients d(j, k) using the equation

$$f_{l+1}(t)-f_l(t)=\sum_{k=-\infty}^{\infty} d(l,k)\psi(2^{-l}t-k).$$

Using these relations, given a(L, k) and {d(j, k) | j ≤ L}, it is clear that we can build the approximation at any scale. Hence, the wavelet transform breaks the signal up into a coarse approximation fL(t) (given a(L, k)) and a number of layers of detail {fj+1(t)-fj(t)| j < L} (given by {d(j, k) | j ≤ L}). As each layer of detail is added, the approximation at the next finer scale is achieved.

Vanishing Moments

The number of vanishing moments of a wavelet indicates the smoothness of the wavelet function as well as the flatness of the frequency response of the wavelet filters (filters used to compute the DWT).Typically a wavelet with p vanishing moments satisfies the following equation .

$$\int_{-\infty}^{\infty} t^{m}\psi(t)dt=0 \qquad \text{for m = 0,…., p-1,}$$

or equivalently,

$$\sum_{k}(-1)^{k}k^{m}c(k)=0 \quad \text{for m = 0,…., p-1.}$$

For the representation of smooth signals, a higher number of vanishing moments leads to a faster decay rate of wavelet coefficients. Thus, wavelets with a high number of vanishing moments lead to a more compact signal representation and are hence useful in coding applications.

However, in general, the length of the filters increases with the number of vanishing moments and the complexity of computing the DWT coefficients increases with the size of the wavelet filters.

The Fast Wavelet Transform Algorithm:

The Discrete Wavelet Transform (DWT) coefficients can be computed by usingMallat.s Fast Wavelet Transform algorithm. This algorithm is sometimes referred to asthe two-channel sub-band coder and involves filtering the input signal based on thewavelet function used.

Implementation Using Filters

To explain the implementation of the Fast Wavelet Transform algorithm consider the following equations:

$$\phi(t)=\sum_{k} c(k)\phi(2t-k)$$

$$\psi(t)=\sum_{k}(-1)^{k}c(1-k)\phi(2t-k)$$

$$\sum_{k} c_{k}c_{k-2m}=2\delta_{0,m}$$

The first equation is known as the twin-scale relation (or the dilation equation) and defines the scaling function φ. The next equation expresses the wavelet ψ in terms of the scaling function φ. The third equation is the condition required for the wavelet to beOrthogonal to the scaling function and its translates.

The coefficients c(k) or {c0, .., c2N-1} in the above equations represent the impulse response coefficients for a low pass filter of length 2N, with a sum of 1 and a norm of1/2.

The high pass filter is obtained from the low pass filter using the relationshipg ( )k c(k) k = −1 1− , where k varies over the range (1 . (2N . 1)) to 1.

The first Equation shows that the scaling function is essentially a low pass filter and is used to define the approximations. The wavelet function defined by equation (second) is a highpass filter and defines the details.Starting with a discrete input signal vector s, the first stage of the FWT algorithm decomposes the signal into two sets of coefficients. These are the approximation coefficients cA1 (low frequency information) and the detail coefficients cD1 (high frequency information), as shown in the figure below.

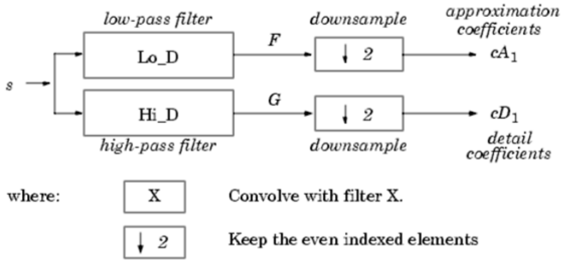


Figure 7.FWT decomposing of coefficients

The coefficient vectors are obtained by convolving s with the low-pass filter Lo_D for approximation and with the high-pass filter Hi_D for details. This filtering operation is then followed by dyadic decimation or down sampling by a factor of 2.Mathematically the two-channel filtering of the discrete signal s is represented by theexpressions:

$$cA_1 = \sum_k c_k s_{2i-k}\,, \qquad cD_1 = \sum_k g_k s_{2i-k}$$

These equations implement a convolution plus down sampling by a factor 2 and give the forward fast wavelet transform.

If the length of each filter is equal to 2N and the length of the original signal s is equal to n, then the corresponding lengths of the coefficients cA1 and cD1 are given by the formula:

$$\text{floor}\left(\frac{n-1}{2}\right) + N$$

This shows that the total length of the wavelet coefficients is always slightly greater than the length of the original signal due to the filtering process used.

Multilevel Decomposition:

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolutionComponents. This is called the wavelet decomposition tree.

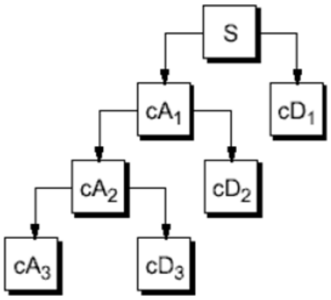


Figure 8. Multilevel Decomposition

The wavelet decomposition of the signal s analysed at level j has the following structure[cAj, cDj, ..., cD1].Looking at a signals wavelet decomposition tree can reveal valuable information. The diagram below hows the waveletdecomposition to level 3 of a sample signal S.

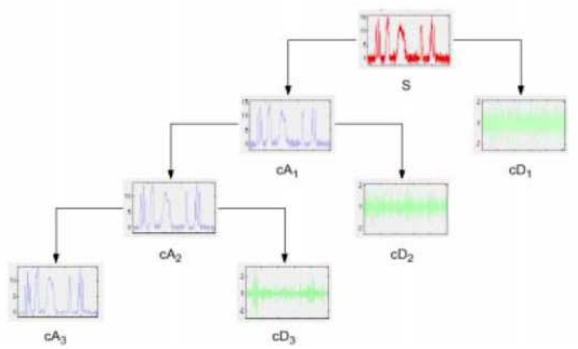


Figure 9.Multilevel Decomposition level 3 of a sample signal S

Since the analysis process is iterative, in theory it can be continued indefinitely. In reality, the decomposition can only proceed until the vector consists of a single sample. Normally, however there is little or no advantage gained in decomposing a signal beyond a certain level. The selection of the optimal decomposition level in the hierarchy depends on the nature of the signal being analysed or some other suitable criterion, such as the low-pass filter cut-off.

Signal Reconstruction:

The original signal can be reconstructed or synthesised using the inverse discrete wavelet transform (IDWT). The synthesis starts with the approximation and detail coefficients cAj and cDj, and then reconstructs cAj-1 by up sampling and filtering with the reconstruction filters.

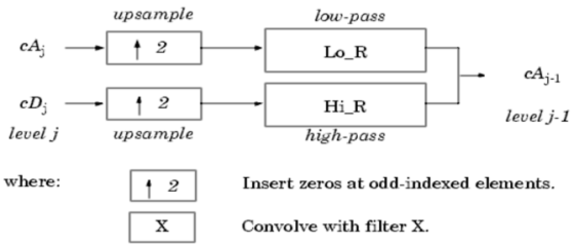


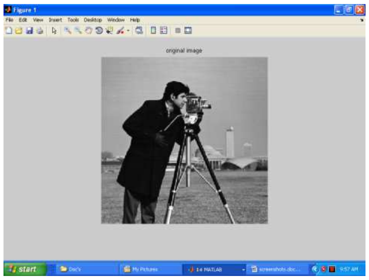
Figure 10. Signal synthesizer (Signal Reconstruction)

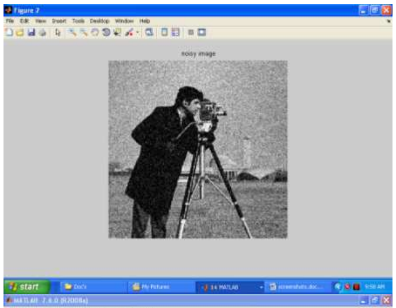
The reconstruction filters are designed in such a way to cancel out the effects of aliasing introduced in the wavelet decomposition phase. The reconstruction filters (Lo_R and Hi_R) together with the low and high pass decomposition filters, forms a system known as quadrature mirror filters (QMF).

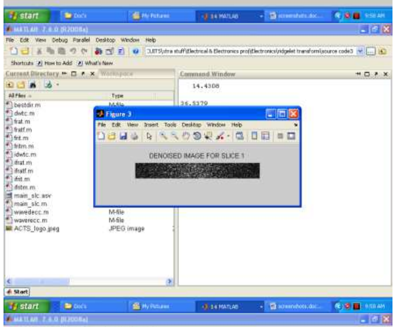
For a multilevel analysis, the reconstruction process can itself be iterated producing successive approximations at finer resolutions and finally synthesising the original signal.

Results & Discussion

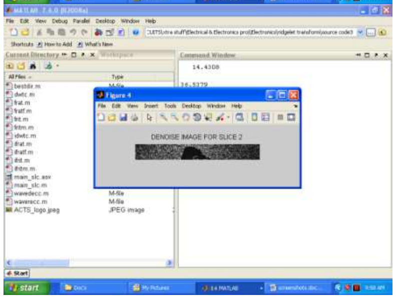
The wiener2 function applies a Wiener filter (a type of linear filter) to an image adaptively, tailoring itself to the local image variance. Where the variance is large, wiener2 performs little smoothing. Where the variance is small, wiener2 performs more smoothing.This approach often produces better results than linear filtering. The adaptive filter is more selective than a comparable linear filter, preserving edges and other high-frequency parts of an image. In addition, there are no design tasks; the wiener2 function handles all preliminary computations and implements the filter for an input image. wiener2, however, does require more computation time than linear filtering. wiener2 works best when the noise is constant-power ("white") additive noise, such as Gaussian noise.



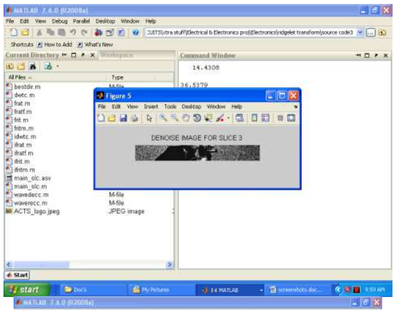


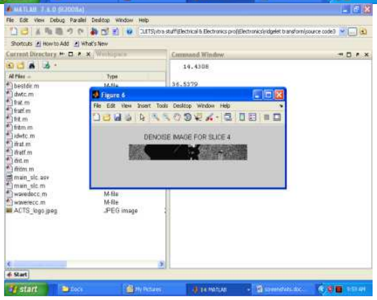


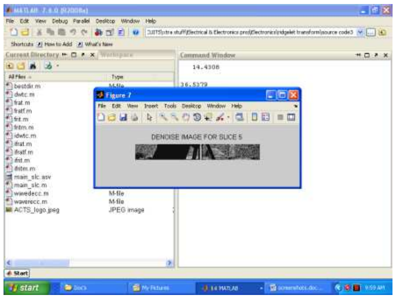
```
   14.4308

36.5379
```

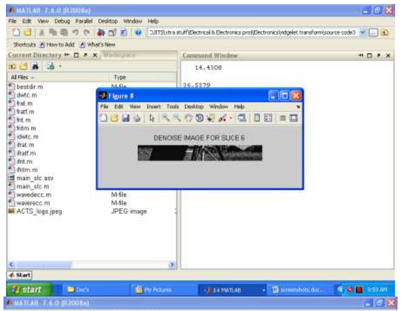


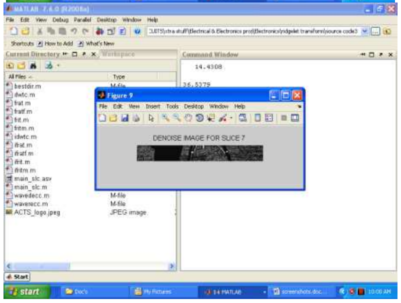
```
   14.4308

36.5379
```

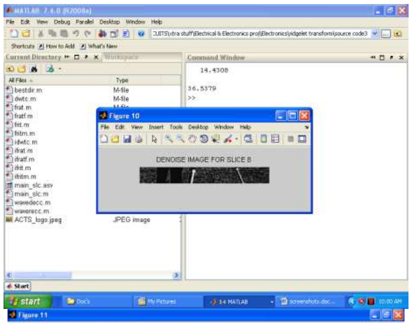












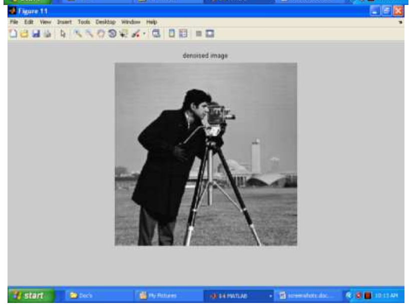


Figure 11.a).Original Image, b). Noisy Image, c)Denoised Image for slice, 1 d).Denoised Image for slice2, e).Denoised Image for slice 3, f).Denoised Image for slice 4, g).Denoised Image for slice, 5 h).Denoised Image for slice 6, i).Denoised Image for slice 7, j).Denoised Image for slice 8, k).Denoised Image

Conclusion

In this Thesis, we presented a strategy for digitally implementing the sliced ridgelet 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 into digital representations. Guided by a series of experiments, we arrived at several innovative choices which we have highlighted in this thesis.

References

[1] Averbuch, R. R. Coifman, D. L. Donoho, M. Israeli, and J. Waldén, “Polar FFT, rectopolar FFT, and applications,” Stanford Univ., Stanford, CA, Tech. Rep., 2000.

[2] E. J. Candès, “Harmonic analysis of neural netwoks,” *Appl. Comput.Harmon. Anal.*, vol. 6, pp. 197–218, 1999.

[3] “Monoscaleridgelets for the representation of images with edges,”Dept. Statist., Stanford Univ., Stanford, CA, Tech. Rep., 1999, submitted for publication.

[4] "On the representation of mutilated Sobolev functions," Dept. Statist., Stanford Univ., Stanford, CA, Tech. Rep., 1999.

[5] E. J. Candès and D. L. Donoho, "Curvelets," [Online] Available:http://www-tat.stanford.edu/~donoho/Reports/1999/curvelets.pdf, 1999.

[6] "Curvelets—A surprisingly effective nonadaptiverepresentationfor objects with edges," in *Curve and Surface Fitting: Saint-Malo 1999*,A. Cohen, C. Rabut, and L. L. Schumaker, Eds. Nashville, TN: Vanderbilt Univ. Press, 1999.

[7] "Ridgelets: The key to higher-dimensional intermittency?," *Phil.Trans. R. Soc. Lond. A.*, vol. 357, pp. 2495–2509, 1999.

[8] G.Y. Chen, T.D. Bui, A. Krzy˙zak, Image denoising with neighbour dependency and customized wavelet and threshold, Pattern Recognition 38 (1) (2005) 115–124.

[9] L. Sendur, I.W. Selesnick, Bivariate shrinkage functions for waveletbaseddenoising exploiting interscale dependency, IEEE Trans. Signal Process. 50 (11) (2002) 2744–2756.

[10] L. Sendur, I.W. Selesnick, Bivariate shrinkage with local variance estimation, IEEE Signal Process. Lett. 9 (12) (2002) 438–441.

[11] D.L. Donoho, A.G. Flesia, in: J. Stoecker, G.V. Welland (Eds.), Digital ridgelet transform based on true ridge functions, Beyond Wavelets, Academic Press, New York, 2001.

[12] G.Y. Chen, T.D. Bui, A. Krzy˙zak, Rotation invariant pattern recognition using ridgelet, wavelet cycle-spinning and Fourierfeatures, Pattern Recognition 38 (12) (2005) 2314–2322.

[13] J.L. Starck, E.J. Candes, D.L. Donoho, Astronomical image representation by the curvelet transform, Astron. Astrophys. 2 (2003) 785–800.

[14] J.L. Starck, E.J. Candes, D.L. Donoho, The curvelet transform for image denoising, IEEE Trans. Image Process. 11 (6) (2002)670–684.