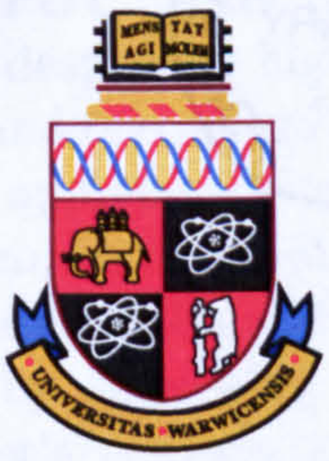
Directional Edge and Texture Representations for Image Processing

Zhen Yao, B.Sc.

A thesis submitted to
the University of Warwick
for admission to the degree of
Doctor of Philosophy

October, 2007



Directional Edge and Texture Representations for Image Processing

Zhen Yao, B.Sc.

A thesis submitted to
The University of Warwick
for the admission to the degree of
Doctor of Philosophy

October 2007

Summary

An efficient representation for natural images is of fundamental importance in image processing and analysis. The commonly used separable transforms such as wavelets are not best suited for images due to their inability to exploit directional regularities such as edges and oriented textural patterns; while most of the recently proposed directional schemes cannot represent these two types of features in a unified transform. This thesis focuses on the development of directional representations for images which can capture both edges and textures in a multiresolution manner.

The thesis first considers the problem of extracting linear features with the multiresolution Fourier transform (MFT). Based on a previous MFT-based linear feature model, the work extends the extraction method into the situation when the image is corrupted by noise. The problem is tackled by the combination of a "Signal+Noise" frequency model, a refinement stage and a robust classification scheme. As a result, the MFT is able to perform linear feature analysis on noisy images on which previous methods failed.

A new set of transforms called the multiscale polar cosine transforms (MPCT) are also proposed in order to represent textures. The MPCT can be regarded as real-valued MFT with similar basis functions of oriented sinusoids. It is shown that the transform can represent textural patches more efficiently than the conventional Fourier basis. With a directional best cosine basis, the MPCT packet (MPCPT) is shown to be an efficient representation for edges and textures, despite its high computational burden.

The problem of representing edges and textures in a fixed transform with less complexity is then considered. This is achieved by applying a Gaussian frequency filter, which matches the disperson of the magnitude spectrum, on the local MFT coefficients. This is particularly effective in denoising natural images, due to its ability to preserve both types of feature. Further improvements can be made by employing the information given by the linear feature extraction process in the filter's configuration. The denoising results compare favourably against other state-of-the-art directional representations.

Key Words

Directional representation; multiresolution; polar cosine transform; linear feature extraction; lapped transform; directional filtering; denoising.

Acknowledgements

This work was mainly supported by an Oversea Research Student Award and a Warwick Postgraduate Research Fellowship and conducted within the Image and Signal Processing Research Group in the Department of Computer Science at the University of Warwick.

First, I would like to thank my parents for their support for my eight-year study in the United Kingdom, which made this thesis possible. I would also like to thank my supervisor Dr. Nasir Rajpoot for his mentoring and friendship throughout my years in Warwick, even when I was a foundation student in the Warwick HEPF programme. Same gratitude also goes to Prof. Roland Wilson for many enlightening discussions and other help for my PhD.

I also want to thank my other friends and colleagues with whom I had great company, especially Heechan Park who shared many technical and social conversations with me.

Thanks also goes to Prof. Justin Romberg at Georgia Tech., for his generosity in sharing his GHMT implementation and Prof. Popovici and Prof. Withers at U.S. Naval Academy for the custom-built moment algorithm's source code and sharing some unpublished materials on their recent work.

Finally, I wish to thank my two examiners, Prof. Mark Nixon and Dr. Ahhir Bhalerao for their suggestions which helped me to improve the quality of this thesis.

Declaration

I declare that, except where acknowledged, the material contained in this thesis is my own work and that it has neither been previously published nor submitted elsewhere for the purpose of obtaining an academic degree.

Zhen Yao

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Image Representations and Analysis

At the heart of many digital signal processing tasks, there is an efficient representation that can capture significant information in the signal. The term *representation* can be analogously understood as a certain description from which the original entity that is being described can be reproduced. A useful representation in this sense should be able to describe the signal in a concise manner, with as small a vocabulary as possible. The process of transcribing from the signal's original form into the domain of the representation with respect to the set of vocabularies, or formally the *basis*, is often referred to as the *transform*. In other words, the representation should be *sparse*. That is to say the transformed signal has only a small set of coefficients with high amplitude. This is a highly desirable property in applications like compression, denoising or feature extraction.

In recent years, the wavelet basis [93] has emerged as the prominent representation for signal and image analysis and processing. The wavelet basis is built by shifting and scaling

a single mother wavelet function $\psi(\cdot)$. The wavelet basis $\{\psi_{\eta,s}(\xi)\}$ on $L^2(\mathbb{R})$ is given by:

$$\psi_{\eta,s}(\xi) = 2^{-s/2}\psi\left(\frac{\xi - 2^s\eta}{2^s}\right), \qquad \eta, s \in \mathbb{Z}. \tag{1.1}$$

The ability to detect discontinuities at different resolutions in the signal makes the wavelet basis a better choice than the conventional Fourier basis and its real versions (cosine and sine transforms [4]), which have precise frequency resolution but approximate singularities poorly. Further, it was also shown that the wavelet basis is an optimal basis for representing one-dimensional piecewise smooth signals with a finite number of singularities [44, 51]. However, in two-dimensional cases, the situation is more complicated. To seek a suitable representation for natural images, it is vital to consider two important questions:

1. **How are the natural images structured?**
2. **How does the Human Visual System (HVS) perceive the images?**

A recently developed image-dependent linear transformation method called Independent Component Analysis (ICA) [9, 81], which minimises the statistical dependence of the components of the representation, has gained a wide attention. A blockwise basis with reasonable sparsity can be automatically computed via ICA. Therefore, by looking at the basis functions obtained by applying ICA on natural images, we can shed light on the question of natural image structure, and how basis vectors look like in a sparse representation.

Figure 1.1 shows one example of the dictionary of basis vectors of 16×16 pixel blocks from an image of natural scene (see [110]) generated by ICA. It is clear that two types of vectors are present in the dictionary. One is directional linear type features and the other is directional oscillatory waveforms. These two types of features can be viewed as the building blocks for natural images. Therefore, in the light of Question 1, for an efficient 2-D image representation, similar basis functions should be expected.

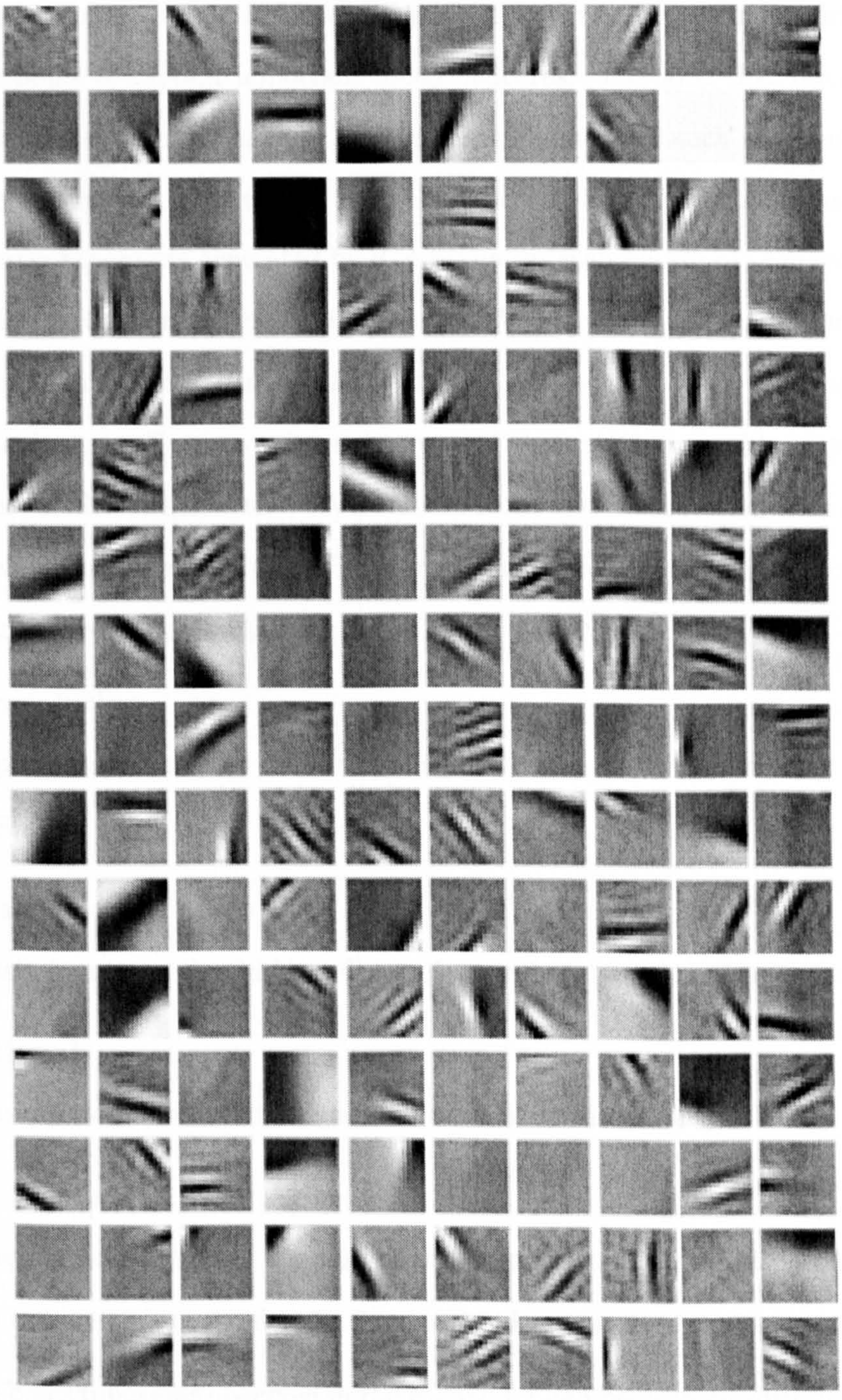


Figure 1.1: The basis functions generated by ICA from natural images

It is believed that during the millennia of evolution, the external visual stimuli of complex natural scenarios have influenced the HVS, so that the cells in the primary visual cortex can respond to certain important image features. The idea that biological visual systems in mammals actually analyse images along dimensions such as orientation, scale and frequency dates back to the work by Hubel and Wiesel [80]. Later it was also found that the sparse components generated by ICA actually resemble simple-cell receptive fields [110, 111]. This matches the hypothesis that the HVS captures essential information with a minimum number of excited neurons, which can be understood as a biological form of sparse representation. However, since the type of feature that would excite the retinal neuron is fixed, it also suggests that HVS is more likely a representation with a fixed basis, instead of being adaptive like ICA. Therefore a list of desiderata can be compiled for a desirable sparse image representation :

- **Multiresolution**

  The representation should allow the image to be analysed from coarser scale to finer resolutions.

- **Localised**

  The basis functions should be well localised in both space and frequency.

- **Directional**

  The basis functions should be oriented at various directions.

- **Fixed**

  Although it is not strictly a requirement, in order to allow a fast decomposition, the library of bases is expected to be fixed instead of image dependent, contrary to ICA.

Since the basis functions observed in the Figure 1.1 can be both anisotropic and isotropic directional waveforms, this suggests that another major requirement in deriving appropriate bases to represent a visual scene should include meaningful features including lines,

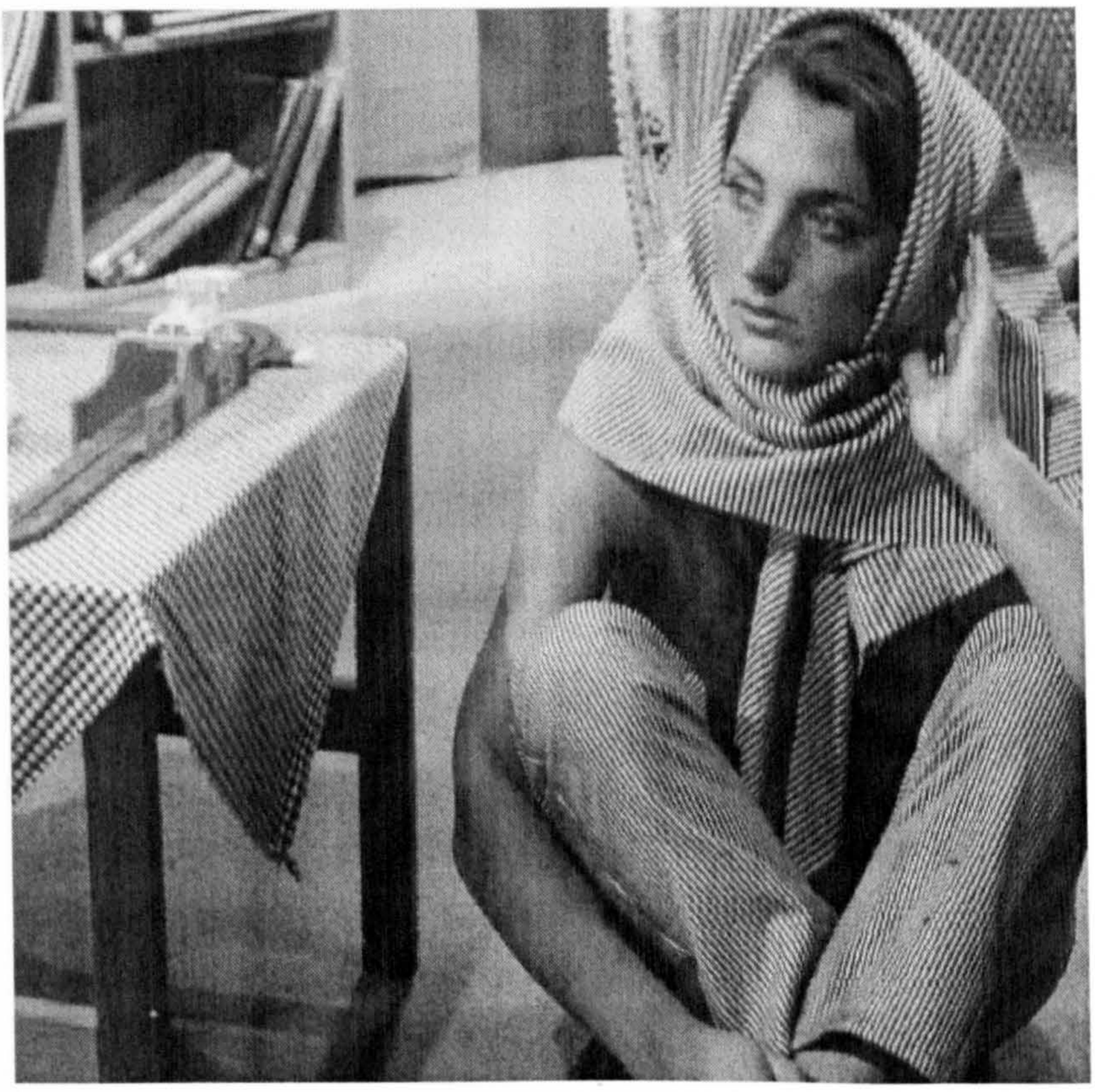


Figure 1.2: The 512×512 `barbara` image

edges and textural patches. The problem of finding such a representation is considered in greater detail in the remainder of this thesis.

1.2 Edges and Textures

The image `barbara` shown in Figure 1.2, which is frequently used in image processing research experiments, can be regarded as a classic example of a natural image. The image contains some significant textural patterns in the form of oriented stripes (on the trousers and kerchief) and non-oriented "chessboard" texture (on the table cloth), as well as edge discontinuities, for example the contours of her face and arm. It is, therefore, a good specimen by which to study the concept of a *natural image*.

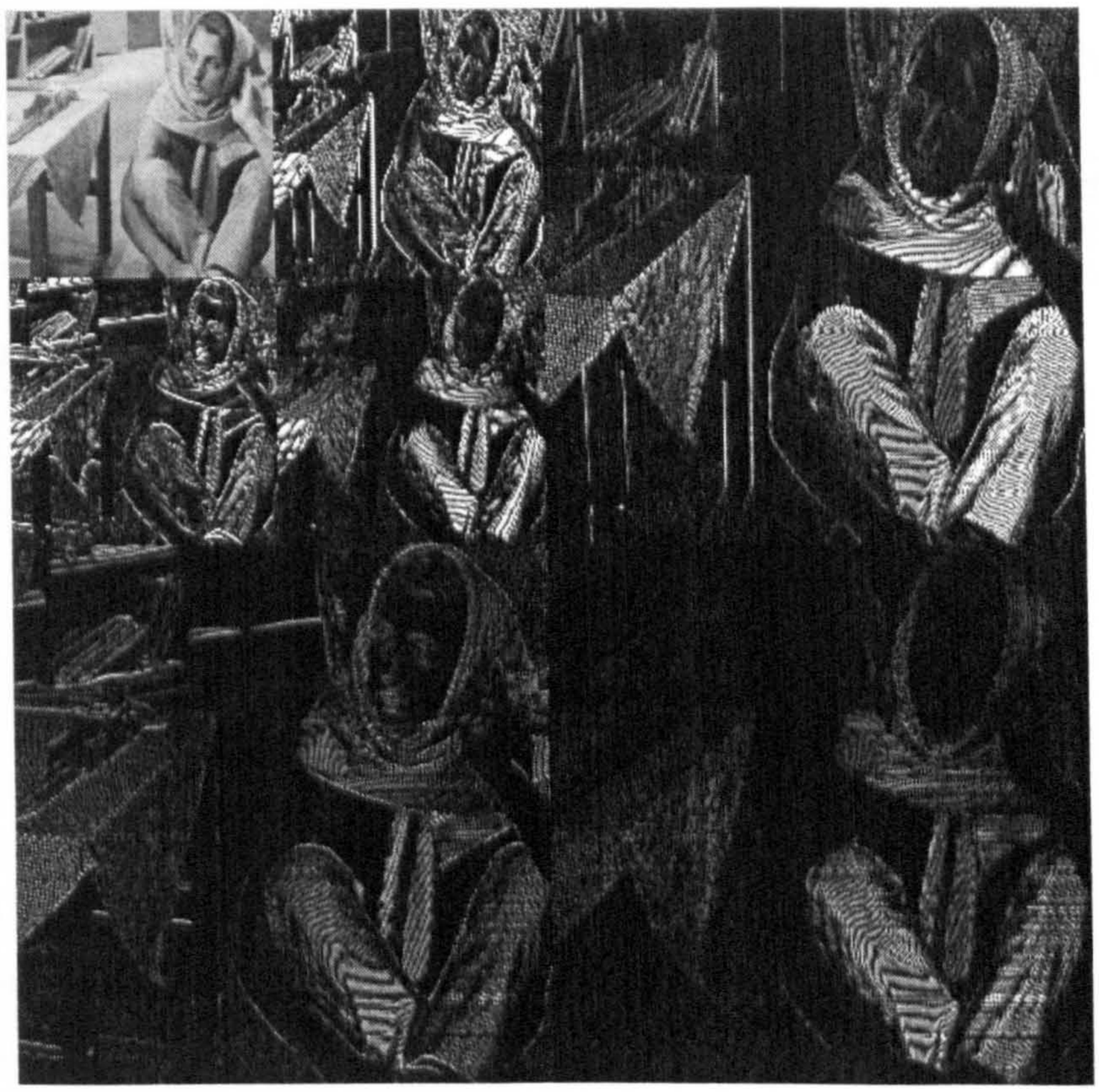


Figure 1.3: The 2-level decomposition of barbara with Haar wavelets

If the barbara image is decomposed by the wavelet basis in a Cartesian separable manner, i.e. taking the 1-D transform horizontally and then vertically, the resulting wavelet coefficients, as illustrated in Figure 1.3, have high amplitudes around both edges and textural regions, which by no means fulfils the sparsity requirement. The reason for this phenomenon is the fact that the 2D wavelet basis functions formed by a tensor product are isotropic, and correspond to point singularities in 2-D. Also, the orientation resolution of the basis consists only 3 directions (at $0^o, 45^o, 90^o$ degrees), far from being practically useful. These factors make the wavelet representation unable to exploit the coherence along oriented features.

In order to overcome this deficiency in representing directional features by the orthonor-

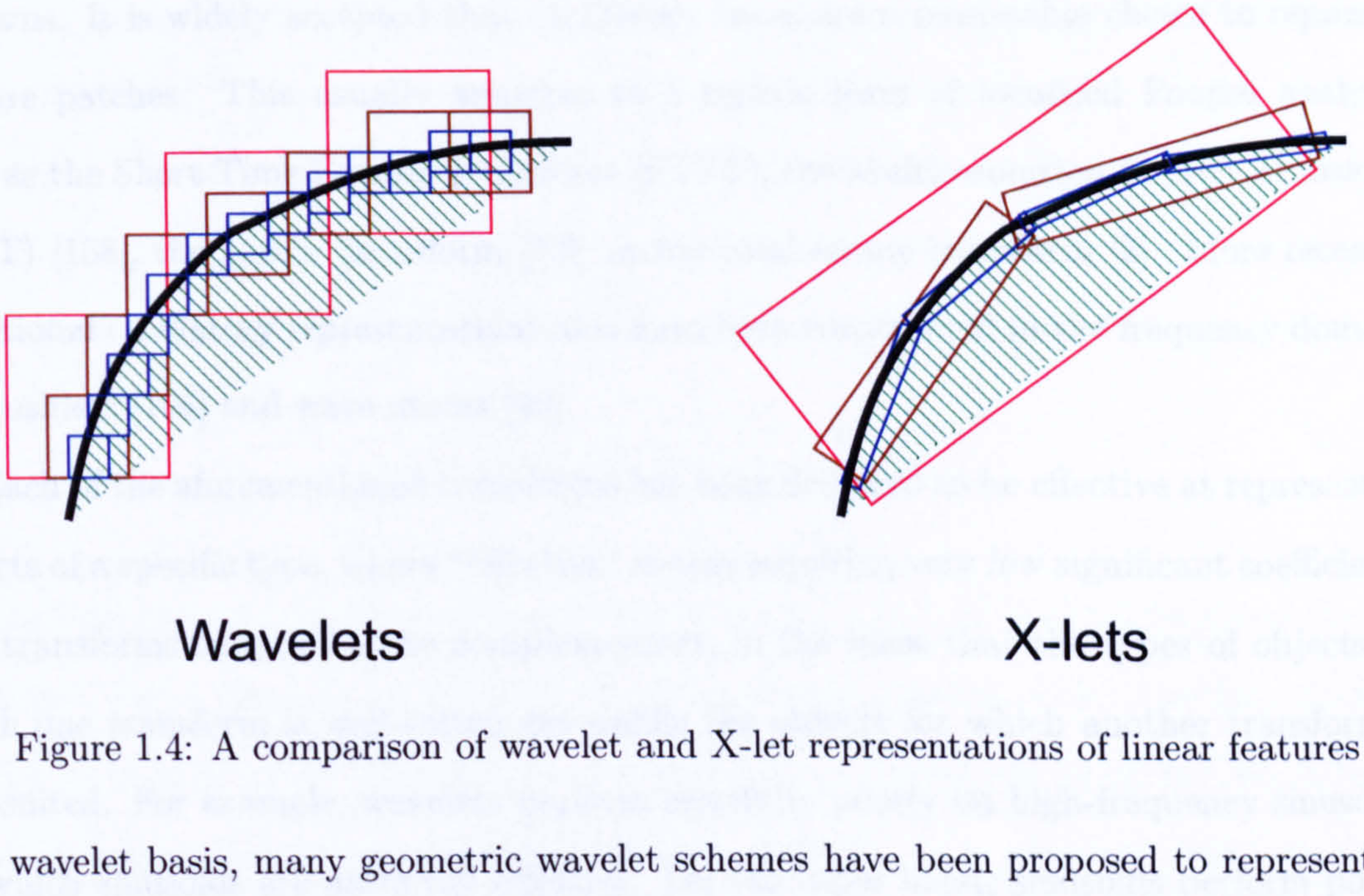


Figure 1.4: A comparison of wavelet and X-let representations of linear features

mal wavelet basis, many geometric wavelet schemes have been proposed to represent the edge structures, such as wedgelets [55], curvelets [28], contourlets [54], and shearlets [63]. The motivation behind these new representations is the fact that the image boundaries are critical in recognising an object as a form of "primal sketch", as noted by Marr [98]. They are commonly referred to as the *"X-lets"* in the literature. The bases of these X-let representations are dictionaries of anisotropic waveforms enlongated at different orientation, scale and position. This makes such bases significantly advantageous in representing linear edges compared to the wavelet basis. A schematic illustration of how wavelets and X-lets may represent a curved boundary is shown in Figure 1.4. While the wavelet basis uses point singularities to approximate the curve, the X-lets use linear singularities which are better suited to express the geometric nature of the boundary. The number of basis atoms used to express the boundary is therefore greatly reduced, compared to wavelets.

However, these geometric wavelets make the explicit assumption that the natural image is composed of geometrically regular and smooth objects, which on its own is not a good model for realistic scenarios. The missing component is a good representation for texture

patterns. It is widely accepted that oscillatory bases are a reasonable choice to represent texture patches. This usually amounts to a certain form of localised Fourier analysis, such as the Short Time Fourier Transform (STFT), the Multiresolution Fourier Transform (MFT) [158], the Gabor transform [73], or the local cosine transform [2]. More recently, directional oscillatory representations also have been constructed in the frequency domain, as brushlets [106] and wave atoms [49].

Each of the aforementioned transforms has been designed to be effective at representing objects of a specific type, where "effective" means requiring very few significant coefficients. The transforms turn out to be complementary, in the sense that the types of objects for which one transform is well-suited are unlike the objects for which another transform is well-suited. For example, wavelets perform relatively poorly on high-frequency sinusoids, for which sinusoids are naturally effective. On the other hand, sinusoids perform poorly on singular discontinuities, for which wavelets are very effective. In 2-D, the same analogy applies to representations such as curvelets and brushlets.

In order to cope with the variety of different features, the authors in [42] have devised a dynamic programming strategy to extract the best basis from a large library of functions. However, if the signal in composed of highly dependent components, then the best basis rarely provides an optimal representation. Another approach is to combine the different bases into an overcomplete dictionary and seek a representation of the signal by selecting a subset of the dictionary. Such an approach is called an *atomic decomposition.* Several methods for selecting the atoms from the dictionary have been proposed, such as Matching Pursuit (MP) [95] and Basis Pursuit (BP) [34]. Also, as discussed in the beginning of this chapter, ICA [81] follows a *basis from image* approach which naturally provides a sparse representation for the image, but unfortunately does not use a multiresolution notion. Bhalerao and Wilson also proposed an image-dependent representation named warplets [12]. The warplets are based on an affine model consisting self-similar image blocks and by

clustering in the warplet domain various types of features can be represented efficiently.

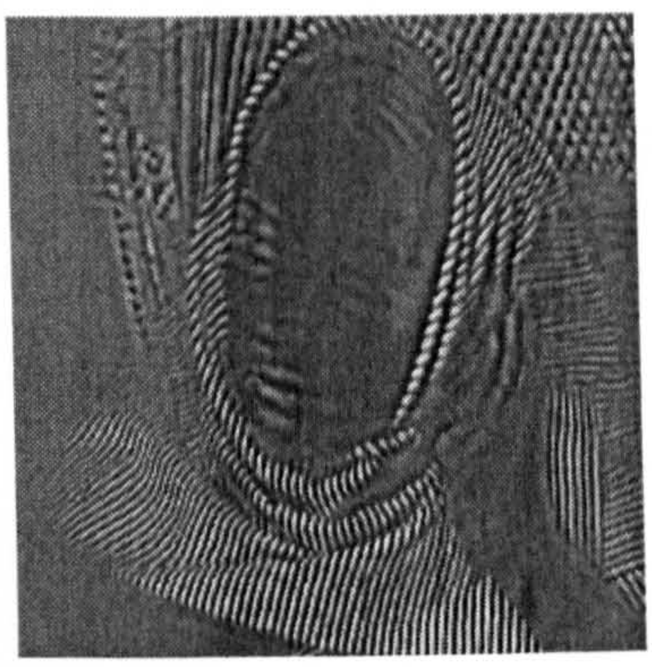
There is also a trend of research which interprets the hypothesis of *"image = edge + texture"* literally, by decomposing the image into two additive layers. This class of representation is named as the "$u + v$" model in [107], which assumes the image $f(\boldsymbol{\xi})$ (where $\boldsymbol{\xi} = [\xi_1, \xi_2]^T$ denotes the spatial coordinate) being a sum of two components :

$$f(\boldsymbol{\xi}) = u(\boldsymbol{\xi}) + v(\boldsymbol{\xi}). \tag{1.2}$$

The first component $u(\boldsymbol{\xi})$ is aimed to model the geometric objects which are present in the image, while the $v(\boldsymbol{\xi})$ is responsible for texture and possible noise. The hypothesis is that by separating these components, they can be represented efficiently by two different bases, instead of having a combined dictionary of atoms. This approach was adopted in [102] for the task of image compression. The u component used in [102] is just a reconstruction from a low bit-rate coded wavelet coefficients, the residual v is then compressed by an adaptive local cosine basis [103]. Although in the context of compression, the wavelet reconstructed u component is a suitable choice, it does not faithfully reflect the geometric regularity of the image. The variational methods proposed by Vese and Osher use a mathematical model for the texture that extends the notion of Total Variation [151] that the resulting separation of u is geometrically regular and smooth, while v retains only the oscillatory patterns. Starck et al. proposed a basis pursuit denoising (BPDN) method [145] to split the layers by choosing two appropriate bases. The basis for representing geometric part is curvelets and a DCT is used for the representation of textures. The BPDN will align these sparse representations for each part to perform the separation. An example of the decomposition on the image `barbara` is shown in Figure 1.5, where the smooth objects are separated from oscillatory patterns using the method presented in [145].

A portion of `barbara`

Geometric component $u(\boldsymbol{\xi})$



Texture componenet $v(\boldsymbol{\xi})$

Figure 1.5: The separated components of `barbara` using the $u + v$ model

1.3 Aims and Contributions

A brief overview of image representation research highlights the importance of the *"image = edge + texture"* model in constructing an efficient two dimensional representation. It was also suggested that the directional selectivity of the bases is vital for the representation to reflect the structure of natural images. Therefore, a directional representation that can cope with both linear type edges and oscillating textures is of fundamental interest of this thesis.

Originally proposed as a general purpose tool for image analysis, the multiresolution Fourier transform (MFT)[18] has been used in analysing the edge and textural features in a unified setting. It has been successfully applied in parametric linear feature extraction [20] and texture analysis and synthesis [79], which make the transform an appropriate choice for our study.

However, the MFT implementations previously reported in [18, 20, 79, 45] are all based on applying scale-dependent variable sized windows over a pre-whitened version of the image, which is less suitable for the purpose of image representation. Also, the MFT has certain drawbacks which limit its usage in certain image processing tasks. First, the frequency linear feature model used with the MFT is extremely sensitive to noise, making

the linear feature extraction algorithm very ineffective in the presence of noise. Secondly, the Fourier basis assumes a periodic extension of the image boundaries, which seriously affects the approximation rate due to the potential boundary discontinuities. Although its real versions, such as the DCT, can remedy this shortcoming, its basis functions are not oriented as in the Fourier basis. Thirdly, it should also be pointed out that there does not exist an orthonormal overlapping local Fourier basis, while such local bases exist for cosine and sine transforms. This seriously restricts the MFT's use in tasks such as compression. Finally, there has not been a study on combining the edge and texture modelling in a coherent manner in the MFT domain, nor has any denoising algorithm been proposed with the MFT representation. It is under these considerations that the research of this thesis was conducted.

The contribution of this thesis is therefore threefold.

- First, it tackles the issue of how to incorporate the MFT linear feature analysis in noisy situations, which can potentially widen its domain of potential applications.

- Secondly, it presents a new multiscale directional transform which has the directional selectivity of the Fourier basis and good convergence properties of the DCT, and can represent directional textures efficiently.

- Thirdly, the thesis presents two possible approaches for joint "edge + texture" representation, one of which leads to very efficient denoising algorithms that can preserve both edge and textural patterns.

1.4 Thesis Outline

The rest of the thesis is organised as follows:

A detailed review on different directional representations and their applications in image processing will be presented in Chapter 2. Also, the problem of image denoising by transform domain thresholding will be discussed, due to its relevance to the problem of approximation by sparse representations.

Chapter 3 addresses the problem of extracting edge information (linear feature extraction) in a parametric fashion from noisy images with the MFT. This algorithm first uses a magnitude spectrum model for the mixture of signal and noise and then uses a Kalman filter [86] to refine the parameter estimates. A voting-based classification scheme will also be discussed in order to select significant features from the estimate.

A new directional transform, the *multiscale polar cosine transform* is presented in Chapter 4. The transform has oriented basis functions like the MFT, but the construction is in fact a form of discrete cosine transform. The MFT and the proposed transform both belong to the family of multiscale directional trigonometric transforms. Also, by seeking a directional adaptive cosine packet basis, the transform can be extended to accommodate both edges and textures. Comparative results for image denoising will be presented at the end of the chapter. Part of the work presented in this chapter has been reported in [164, 165].

The thesis continues to explore the notion of *"image=edge + texture"* in Chapter 5, in the form of applying Gaussian frequency filters on MFT's local spectrum. Based on this idea, two denoising algorithms are discussed and compared with some of the state-of-the-art directional representations. The proposed schemes seem to compare well with other methods both in terms of visual reconstruction and computational cost. Some of the results from this chapter has appeared in [165].

The thesis is concluded in Chapter 6 with a synopsis and a discussion of limitations of the work which can be further improved or extended.

Chapter 2

Directional Image Representations and Analysis

Over the years, various attempts have been made in finding suitable representations for natural images with different underlying models and constraints, making the construction of geometric and directional image representations a very active research area. This chapter is intended to give a brief overview of this growing body of literature, with an introduction to the basic concepts of approximation. Furthermore, in order to demonstrate the effectiveness of these endeavours, a section is devoted to the discussion of noise removal by transform domain coefficient shrinkage [59].

2.1 Background

Sparse signal approximations from certain representation bases are useful for applications such as compression or denoising. A non-linear approximation of an image f in a certain

basis $\mathcal{B} = \{b_m\}_m$ is defined as:

$$f_M = \sum_{m \in I_M} \langle f, b_m \rangle b_m \tag{2.1}$$

where I_M is the index set of the M largest coefficients in the transform domain. The resulting approximation error is

$$\|f - f_M\|^2 = \sum_{m \notin I_M} |\langle f, b_m \rangle|^2. \tag{2.2}$$

Given a prior model on the image f, it is desirable to find a basis $\mathcal{B}$ such that $\|f - f_M\|^2$ converges quickly to zero as M increases, in which case there is a small constant C and a large exponent $\beta > 1$ with

$$\|f - f_M\|^2 \leq CM^{-\beta}. \tag{2.3}$$

If the image is smooth, which means that it is $\mathbf{C}^\alpha$ uniformly smooth (α times continously differentiable) and is represented using an orthonormal wavelet with vanishing moment $p > \alpha$, it is proven [51] that there exists a constant C for

$$\|f - f_M\|^2 \leq CM^{-\alpha} \tag{2.4}$$

to hold. However, if f is *piecewise smooth*, which means f is $\mathbf{C}^\alpha$ ($\alpha > 1$) inside and outside of certain finite discontinous boundaries as depicted in Figure 2.1, then the discontinuities result in many wavelet coefficients of large amplitude. Then equation (2.4) no longer holds, but the decay rate is still $O(M^{-1})$:

$$\|f - f_M\|^2 \leq CM^{-1}. \tag{2.5}$$

However, the approximation rate can be improved by exploiting the regularity of such

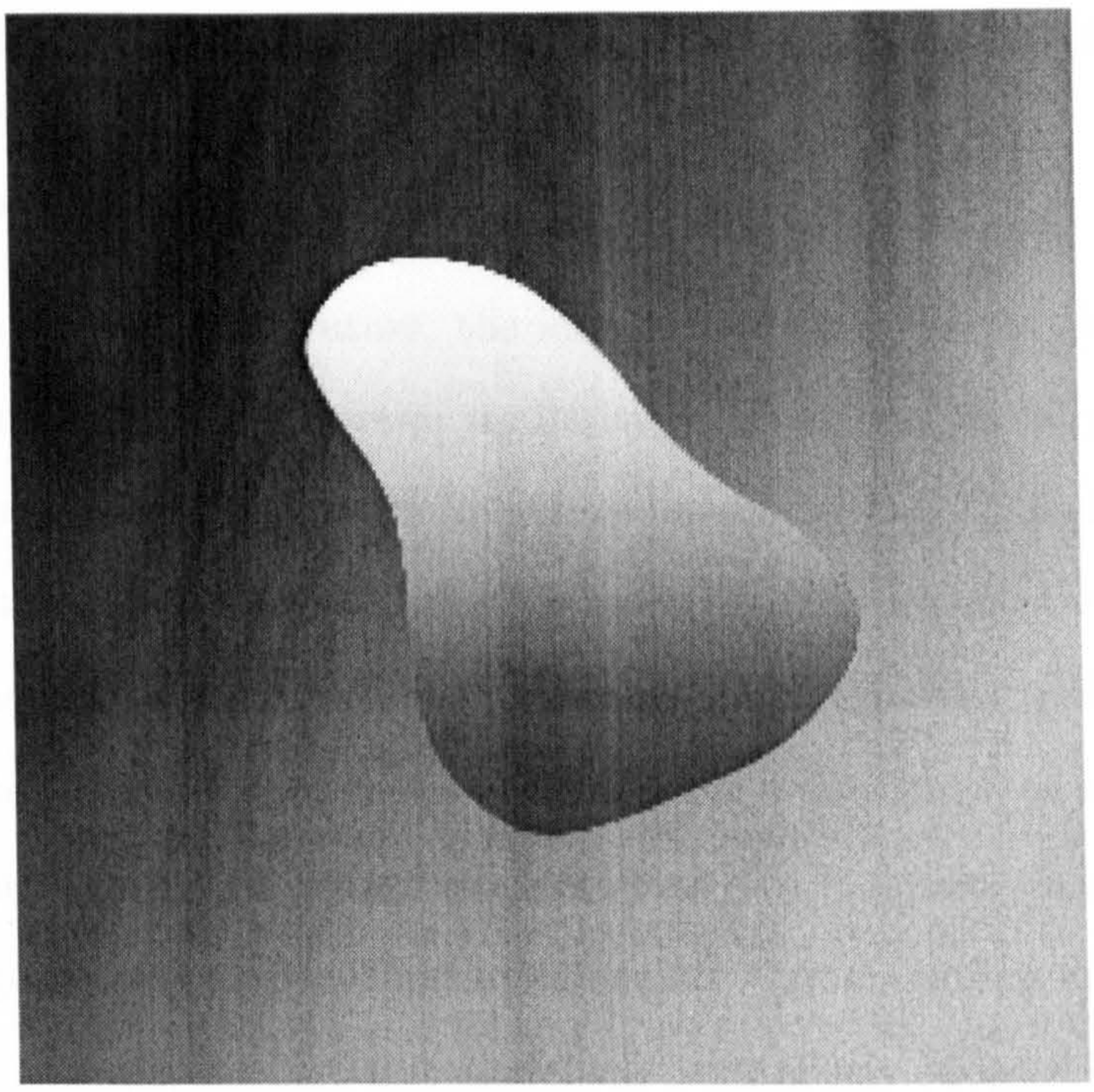


Figure 2.1: A typical piecewise smooth image

discontinuities as regular geometric curves. Let $\Omega \subset [0,1]$ with a boundary $\partial\Omega$ that is a $\mathbf{C}^2$ curve, which means that there is a twice continuously differentiable parameterisation of $\partial\Omega$. For this type of image, it was theoretically estimated that the optimal approximation rate will be of $O(M^{-2})$ [38], which the ordinary 2D wavelet basis formed by tensor-product is clearly unable to achieve. The effort to overcome this usually amounts to certain forms of edge-adaptive models or directional transforms with bases of elongated waveforms, which will be discussed in detail in the following sections.

2.2 Edge-based Geometric Representations

2.2.1 Non-parametric Models

In the computational vision literature, the idea of analysing images along multiple orientations in order to exploit geometric regularity of edges was initially explored at the beginning of the seventies with the Binford-Horn line-finder [13, 77] and in later work by Granlund [75]. Carlsson [31] proposed an edge-based image representation which measures intensity jumps across boundary curves. An image approximation is then calculated by imposing the same jumps along the edge by computing values between edges with a diffusion process. Many edge-based image representations [65, 162] have then been elaborated, with different edge detection procedures using similar "jump" models along these edges.

In the wavelet domain, the zero-tree model has been widely adopted in various applications such as *Embedded Zerotree Wavelet* (EZW) [137] and *Set Partitioning In Hierarchical Trees* (SPIHT) [136]. Although it does not explicitly employ an edge-based model, the attempt at exploiting multiscale statistical correlation of significant wavelet coefficients is effectively close to the idea of identifying edge discontinuities. Such ideas were refined in the multiscale wavelet maxima [96] and another wavelet edge representation was reported in [162]. Edge-based image representations with complete orthonormal families of foveal wavelets in [114] and footprints [61] have been introduced and studied to reconstruct the main image edge structures.

The aforementioned schemes all share a common theme: that the edge representations are based on a statistical model of intensity or coefficient magnitudes. Although multiscale coherence allows the edges to be exploited to a certain extent, such approaches do not consider the coherence *along* the edge structures. These algorithms do not, therefore, seem to outperform separable orthonormal wavelets significantly.

2.2.2 Linear Feature Parametric Models

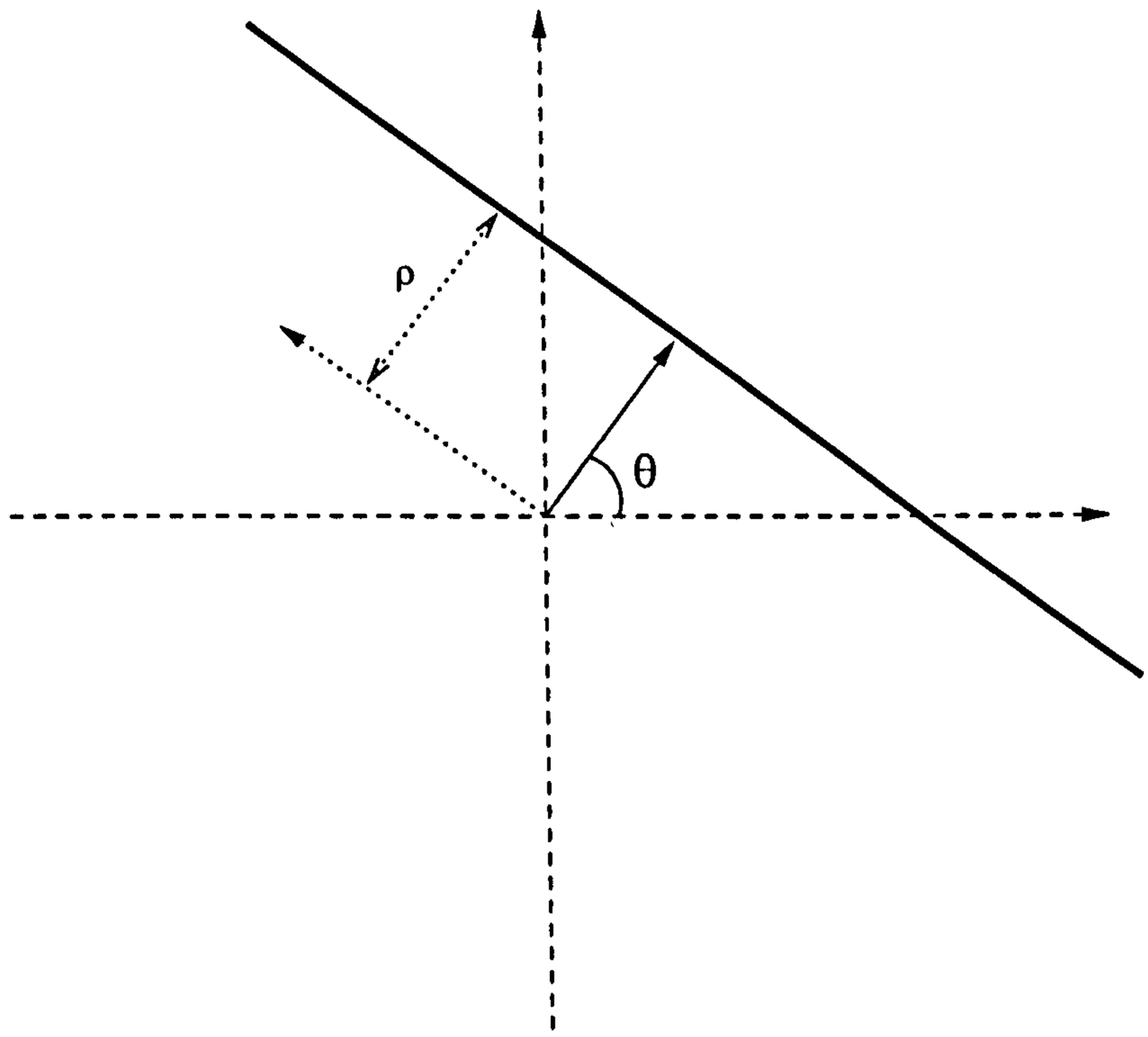


Figure 2.2: A straight line and its parameters

Another stream of research tries to model the curvilinear edges as a combination of simple prototypical features, which can be characterised by a small set of parameters. The common parametric model for edges usually uses the local prototype of a straight line positioned at certain angle θ with a centroid distance ρ relative to the centre of the spatial window origin (see Figure 2.2).

Wedgelets were proposed by Donoho [55] as a means of approximating piecewise constant images with smooth boundaries. The wedgelet dictionary by definition is given by the characteristic functions of wedge-shaped sets obtained by splitting the image dyadically into straight line step edges. The implementation of such representations was further discussed by Claypool and Baraniuk [36], Shukla et al. [140] and Wakin et al. [153].

A closely related representation, called the *Beamlet* [56], which uses simple straight

lines as the basis dictionary atoms instead of step edges, can be used in detecting linear singularities in the presence of noise. Such forms of representation are similar to the approach of applying local Hough transforms [62, 82] on image edge maps generated by edge detectors such as Canny [30]. In [130] a hierarchical version of the Hough transform based edge parametric representation was proposed, which combines local and more global information by integrating line segments detected in small spatial regions into larger segments where possible.

A similar curve representation based on the *Multiresolution Fourier Transform* (MFT) [158] with a linear feature frequency model was studied in [18, 20]. Later the model was extended by Davies [45] to accommodate multiple line segments within one block and further optimisation was reported in [101]. The main merit of the MFT-based scheme is that it does not require an edge detection step, which is a prerequisite for the Hough transform and the parameter estimation is computationally fast.

All these line-type parametric representations are hierarchical and adaptive, which can be depicted schematically in Figure 2.3 (from [18]). There are also cases where the representation is fixed. A multiscale geometric hidden Markov tree model was developed [135] for representing edge structures in the complex wavelet domain [87]. Popovici and Withers developed a moment-based method for linear edge detection and parameter estimation [127] directly from image patches of fixed size. They also reported another estimation algorithm [128] which operates in the Discrete Cosine Transform (DCT) domain, which allows the edge representation to be directly extracted from JPEG [154] compressed images, with improvements over other similar DCT-domain edge models [138, 142].

2.2.3 Other Parametric Models

Besides using simple straight lines for representing edges, there have also been attempts to use other types of feature profiles to capture the curvilinearity of object contours in

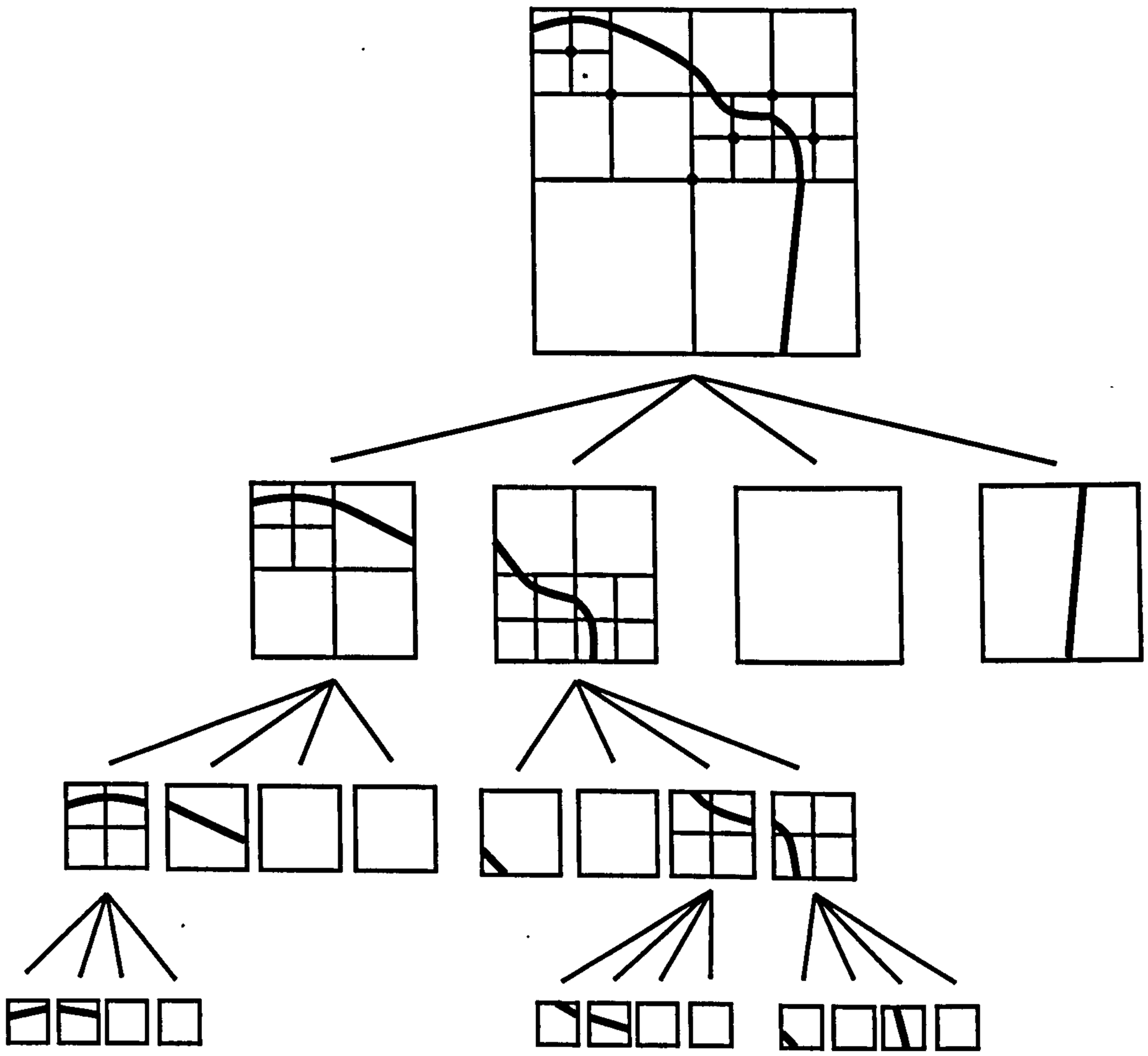


Figure 2.3: Hierarchical parametric representation of a curve as line segments

order to obtain a compact representation. The Hough transform, for instance, has been generalised to consider other parametric [8] or non-analytic [7] curves, and performance improvements have also been considered for the transform in [109, 108].

In the MFT framework, Davies also discussed an arc (a portion of a circle) feature model [45] in the frequency domain, which was combined with the linear feature model to provide a joint representation of the discontinuities around edges. It was more than a decade later when Pongpiyapaiboon [125] proposed the *arclet* decomposition which uses the same underlying arc model but with a construction akin to the wedgelets. For a function $f \in \mathbf{C}^\alpha$, the resulting approximation rate for the arclets is $O(M^{-\alpha})$ for $2 < \alpha < 3$, instead of $O(M^{-\alpha}), 1 < \alpha < 2$ for the wedgelets. It is, therefore, not surprising to see that in [125] Pongpiyapaiboon also combined arclets with wedgelets for representing natural images as two complimentary parts.

In order to unify the underlying curve models, Popovici and Withers [126] studied a quadratic curve parametric model in the form of :

$$p_1 + p_2 x + p_3 y + p_4 x^2 + p_5 y^2 + p_6 xy = 0. \tag{2.6}$$

This results in a six-dimensional parameter vector $(p_1, p_2, p_3, p_4, p_5, p_6)$ which can be estimated efficiently from the moments of the image. The model can be seen as an extension and generalisation of the combined *"line+arc"* representation, but it is not yet clear how the connectivity between the estimated features can be established.

2.3 Directional Transforms for Linear Singularities

Besides the geometry based edge representations, another paradigm for such image representation is to design new directional transforms whose basis functions resemble anisotropic

waveforms. Such a basis can, in turn, provide a sparse image representation by exploiting the smoothness of the contours in a variety of directions. This section gives a detailed overview on some of these directional fixed transforms which are designed to represent linear singularities.

2.3.1 Ridgelets

In order to overcome the inefficiency of representing linear discontinuities of 2D orthogonal wavelets formed by tensor product, Candés [24] proposed a directional multiresolution representation named *ridgelets*, which deal effectively with linear singularities in 2D. The idea is to map a linear singularity into a point singularity via the Radon transform [46, 47] and then use a 1D wavelet transform to handle the point singularities in the Radon domain at different scales. Formally, a multivariate ridgelet of dimension d is defined as

$$\gamma_{\theta,\eta,s}(\xi) = \frac{1}{\sqrt{s}}\psi\left(\frac{\theta\cdot\xi-\eta}{s}\right). \tag{2.7}$$

where $\theta \in \mathbf{S}^{d-1}$, and $\mathbf{S}^{d-1}$ denotes a unit sphere of dimension d, and s, η are the same parameters as in the wavelet function, denoting scaling and dilation. The corresponding transform of a function $f(\xi) \in \mathbb{R}^d$ is defined as

$$\Gamma f(\theta,\eta,s) = \langle f, \gamma_{\theta,\eta,s}\rangle = \int_{-\infty}^{+\infty} f(\xi)\gamma_{\theta,\eta,s}(\xi)d\xi. \tag{2.8}$$

In dimension $d = 2$, $\xi = [\xi_1, \xi_2]^T$. The above equation 2.8 can be simplified to

$$\Gamma f(\theta,\eta,s) = \int_{\mathbb{R}^2} \psi_{\eta,s}(\xi_1\cos\theta + \xi_2\sin\theta) f(\xi_1,\xi_2)d\xi_1 d\xi_2. \tag{2.9}$$

where the ridge profile ψ is a wavelet function.

While ridgelets enjoy certain theoretical elegance [27, 25, 26], the major problem lies in

their digital implementation, because the underlying Radon transform is a polar transform, and therefore ill-defined on a Cartesian grid. In order to remedy this problem, Donoho [57, 58] modified the ridgelet construction by essentially replacing the discretisation of the angular variable with a periodic wavelet transform, resulting in an orthonormal basis. He called these new basis elements *orthonormal ridgelets.* A version of discrete Radon transform called the *Fast Slant Stack* [6], which is algebraically exact, geometrically faithful and invertible, was used in [143] in the construction of the ridgelet transform. Do and Vetterli [52] also proposed a method for orthonormal ridgelet analysis based on the use of the Radon transform for Z_p^2, the Cartesian product of two copies of the integers mod p, where p is a prime. Its orthogonality has much to recommend it, but since the Z_p^2 Radon transform [100] integrates over lines which are defined algebraically, it is not geometrically faithful and suffers from textural artifacts. Its application has been limited to very simple geometrical shapes. Others reported some variations on the transform by substituting the 1D wavelet transform on Radon slices by complex wavelets [33] (complex ridgelets) and biorthogonal wavelets [149] (ridgelet bi-frame).

Since the basis functions of the ridgelet frame are global linear singularities, it is by no means a suitable representation for natural images. However, it has found its way into certain applications in image processing. Yao and Rajpoot [163] proposed a ridgelet based image signature, which can be used to perform robust image authentication and tampering detection. Le Borgne and O'Connor [14] also used the ridgelet transform as the signature in association with a support vector machine, in order to retrieve images in a Content-Based Image Retrieval (CBIR) system. A robust watermarking scheme was introduced in the work by Campisi et al. [21] which uses the finite ridgelet transform [52] as the underlying transform.

2.3.2 Curvelets

As previously mentioned, edges on images are typically curved and ridgelets alone cannot yield efficient representations. However at a sufficiently fine scale, a portion of a curved edges is almost straight. This curvilinearity is essential in edge-based representations such as wedgelets discussed in Section 2.2. Therefore one could employ the ridgelet transform in a localised manner to capture curved edges. This resulted in the earliest implementation of the curvelet tranform by combining multiscale local ridgelet transforms with a spatial bandpass filtering operation [28]. This particular approach was later referred to as the *curvelet-99* transform and a digital implementation was reported in [143] by employing the undecimated *algorithme á trous* wavelet transform [146] which has a high redundancy factor of $16J + 1$, where J is the total number of scales in the decomposition. The same implementation was also used in enhancing image contrast [147] and image fusion [35]. A step-by-step illustration of the transform is shown in Figure 2.4.

Conceptually, the curvelet transform decomposes an image using a dictionary of atoms which have elongated shape and different directions, spatial locations and scales, obeying a parabolic scaling relation. This relation states that at scale 2^{-s}, each element in the dictionary has an envelope which is aligned along a 'ridge' of length $2^{-s/2}$ and width 2^{-s}. It was shown in [28] that for the class of piecewise smooth images $f \in \mathrm{C}^2$, the largest M-term approximation provided by the curvelet frame is almost optimally sparse:

$$\|f - f_M\|^2 \leq C\,(\log M)^3\, M^{-2}. \tag{2.10}$$

Since it is based on the local ridgelet transform, the curvelet-99 transform suffers from two major drawbacks. First, the partial reconstruction contains blocking artefacts. Secondly, the extreme redundancy of the transform not only requires larger storage, but also means the transform is expensive to compute. Therefore in [29], an alternative construction

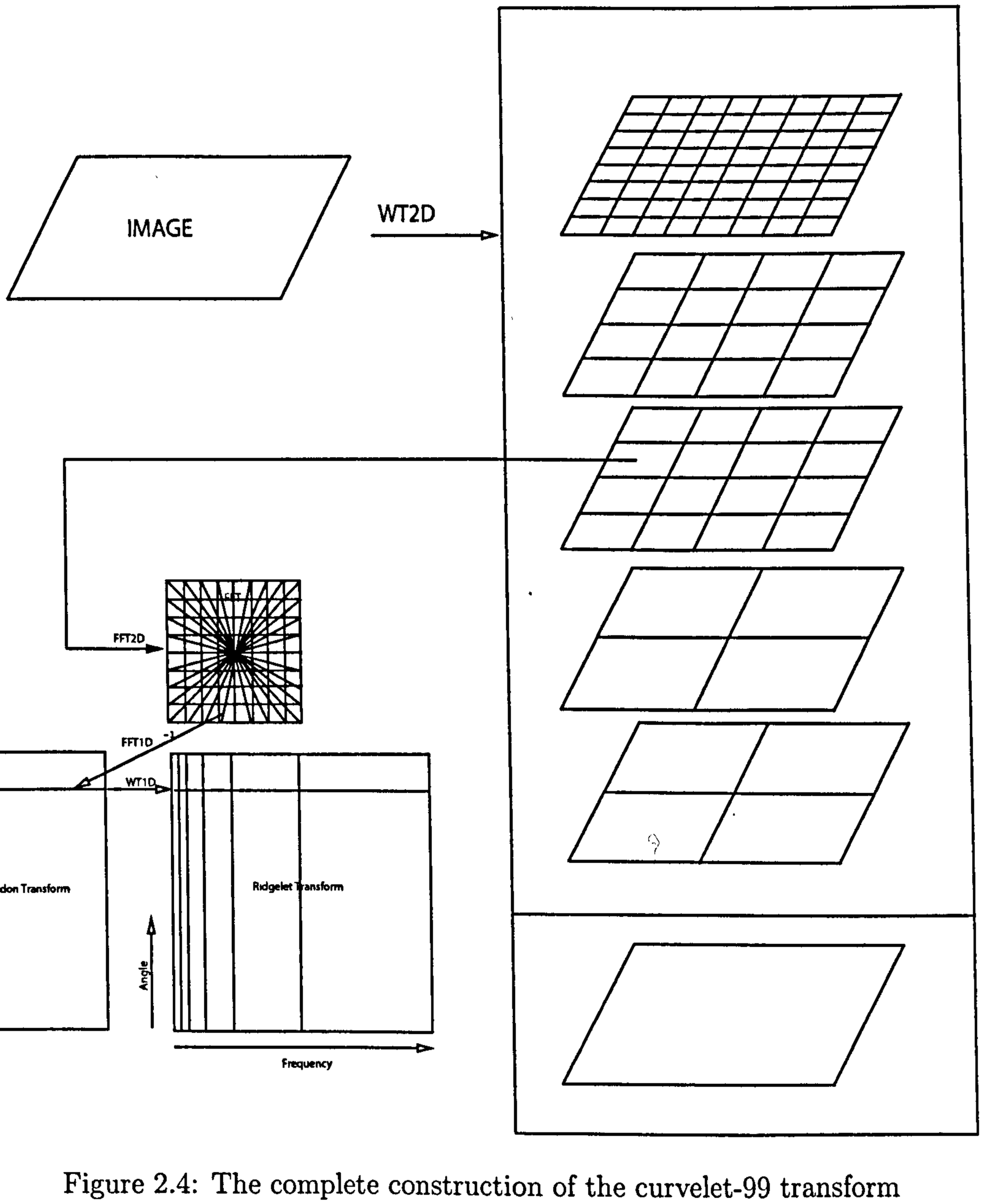


Figure 2.4: The complete construction of the curvelet-99 transform

of the curvelet transform was considered directly from defining the transform in the frequency domain, and a multivariate statistical model in this domain was considered in [15] to characterise the dependency of the curvelet coefficients. In [22], Candés et al. discussed two digital implementations of the second generation curvelet transform by windowing the Fourier spectrum by admissible polar windows, termed as the *Fast Discrete Curvelet Transform* (FDCT). The FDCT can be obtained by either windowing the Unequispaced FFT [71] or via frequency wrapping. Effectively the FDCT's computational complexity is 6 to 10 times that of an FFT operating on an array of the same size. However, the use of the FFT has one side effect: it implicitly treats the image as periodic across the boundaries. As a result, basis functions that are localised near the image boundary wrap around to the opposite boundary by periodicity. In order to remedy this, Demanet and Ying [48] constructed the FDCT in the discrete cosine domain instead of the Fourier domain, which obtained improved approximation rate on features placed near the boundaries.

2.3.3 Contourlets

Apart from attempts of constructing curvelet transforms in a block-based fashion and via frequency windowing, Do and Vetterli [53, 54] proposed another curvelet-type transform called the *"contourlet"* transform, by marrying the Laplacian pyramid and a directional filter bank as illustrated in Figure 2.5. A similar filtering process can be found in the work of Wilson and Spann [161]. Such a discrete-domain multiresolution and multidirection expansion, using non-separable filter banks, is similar to a wavelet expansion derived from filter banks. Since the transform is defined in the discrete domain, its digital implementation is straightforward and hence more applicable than other types of construction. The decimation in the directional filter bank gives the transform very little redundancy. In the frequency domain, the contourlet transform provides a multiscale and directional decomposition very similar to curvelets. It was also shown in [54] that the best M-term contourlet

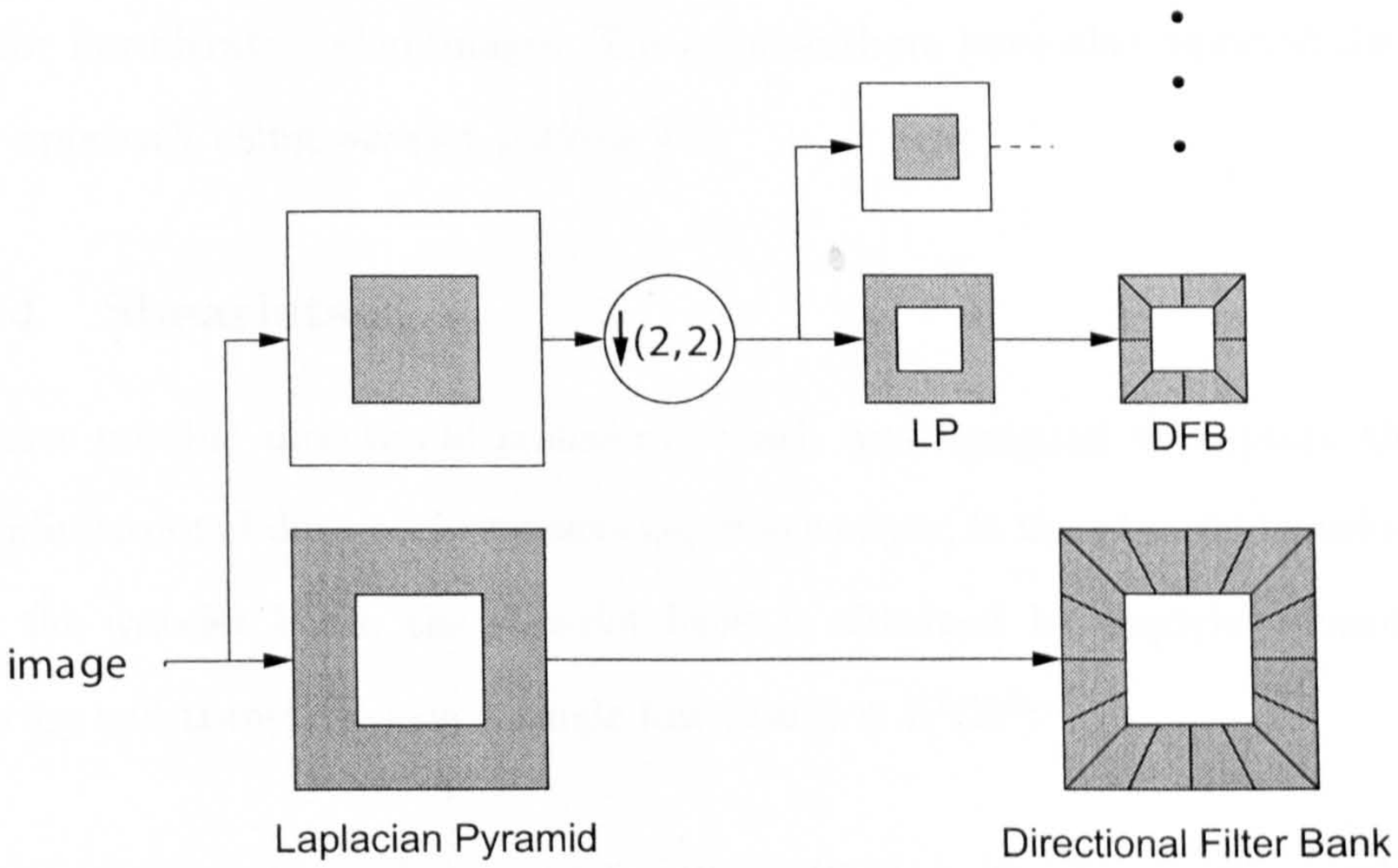


Figure 2.5: The double filter bank construction of the contourlet transform

expansion also has $O((\log M)^3 M^{-2})$ approximation rate for $\mathbf{C}^2$ piecewise smooth images like the curvelets.

The double filter bank construction of the contourlet transform also allows other variations by choosing different filter banks. In [92], Lu and Do developed a critically sampled contourlet transform called *"CRISP-contourlet"*, using a combined, iterated, non-separable filter bank for both multiscale and directional decomposition. Cunha et al. [43] considered an undecimated construction of the transform using a non-subsampled pyramid similar to the *á trous* wavelet which is shift invariant and applied it in image denoising and enhancement tasks. As in [15], Po and Do [124] also conducted a statistical study on the multiresolution dependencies among the contourlet coefficients and developed a Hidden Markov Tree (HMT) model for the contourlet transform which shows a certain effectiveness in denoising and texture retrieval. Eslami and Radha [67] replaced the Laplacian Pyramid with the conventional wavelet transform in the contourlet construction, which makes the transform non-redundant. Although little theoretical justification was given for such a choice, it was shown that the wavelet-based contourlet gives better visual reconstruc-

tion for low bitrate coded images. The same authors have also reported the application of their approach using wavelet packets [68].

2.3.4 Shearlets

Another notable directional transform which was designed to capture the geometry of multidimensional data and representing image edges, is the *shearlet* transform [63, 76, 90]. Like the wavelet basis, the shearlet basis is obtained by applying anisotropic dilation, shearing and translation on a single function $\psi \in L^2(\mathbb{R}^2)$:

$$\mathcal{A}_{AB}(\psi) = \left\{ \psi_{i,j,k} = |\det A|^{i/2} \psi(B^j A^i x - k) \ : \ i, j \in \mathbb{Z}, k \in \mathbb{Z}^2 \right\}, \tag{2.11}$$

where

$$A = \begin{pmatrix} 2 & 0 \\ 0 & \sqrt{2} \end{pmatrix}, \text{ and } B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}. \tag{2.12}$$

The approximation property of the shearlet expansion is the same as curvelets, and contourlets as stated in the equation (2.10). However, compared to curvelets, shearlets have a much simpler mathematical construction. Curvelets are a system consisting of translations, and rotation of a sequence of basic functions depending on a parabolic scaling parameter, which does not form an affine system like shearlets.

In [63] two implementations of the discrete shearlet transform were reported, from both frequency and spatial domains. Compared with the implementation of contourlets, an important advantage of the shearlet transform is that there are no restrictions on the number of directions for the shearing. In addition, there are no constraints on the size of the image supports for the shearing, unlike the underlying directional filter banks in contourlets. Finally, the inversion of the shearlet transform only requires a summation of the shearing filters, rather than inverting a directional filter bank. These advantages make

the implementation of the transform highly efficient.

2.4 Representation of 2D Oriented Harmonics

A variety of representations have been proposed in the context of the *piecewise smooth* image model, despite the presence of textures. The reason that less attention is given to developing efficient texture representations is possibly due to the lack of a rigorous definition for image textures [161]. However, it is a common simplification to model the texture as oriented oscillatory harmonics. The justification for this is due to the biological fact that directional patterns are more prominent than other features to human vision [80]. Certain bases can provide reasonable sparse expansions, such as local cosines [2, 103], wavelet packets [104] and Gabor [73], but these representations either lack directional selectivity or do not possess a multiscale structure. This section reviews a few representations that can efficiently represent directional texture patterns.

2.4.1 Multiresolution Fourier Transform

The basis functions of the 2D Fourier transform are directional periodic complex sinusoidal patterns, making the transform a natural choice for representing oriented textures. However, the transform needs to be localised at a certain spatial location and scale. In order to overcome this, the MFT [158, 18] has been proposed as a combination of Windowed Fourier Transform and wavelet methods, which inherits many of the desired features of both. With the windowing function $w(\xi)$, the transform of a function $f \in L^2(\mathbb{R})$ at position η, frequency k, and scale s is defined as:

$$\hat{f}(\eta, k, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(\eta) w\left(\frac{\xi - \eta}{s}\right) e^{-ik\eta} d\eta. \tag{2.13}$$

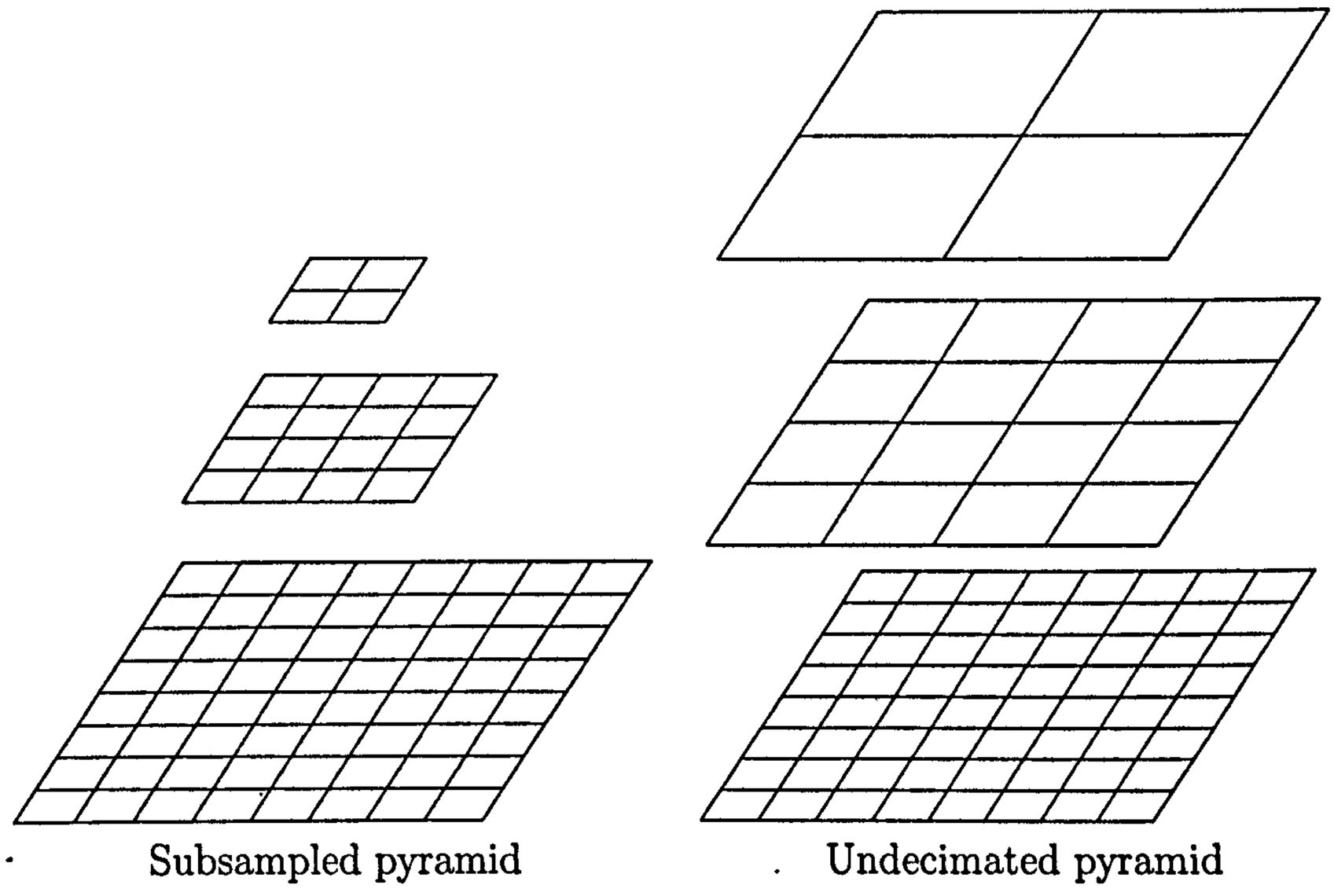


Figure 2.6: The MFT on different types of pyramids

In effect, its discrete version is simply a stack of windowed Fourier transforms, in which the scale of the analysis window is varied systematically with the stack index as in the curvelet-99 implementation, only the block based ridgelet transform is replaced by the FFT. The implementation of the transform can be either in a subsampled pyramid with constant window size, or in the undecimated case, dyadically increasing window size (see Figure 2.6). In [79], Hsu and Wilson proposed a two-component model in the MFT domain for texture representation based on affine transformation estimations. Later, it was extended as an image representation by combining the structural texton with a stochastic residual [19]. Bhalerao and Wilson [11, 12] applied a Gaussian mixture model for the local Fourier spectrum for affine invariant image segmentation using the affine invariant texture synthesis approach.

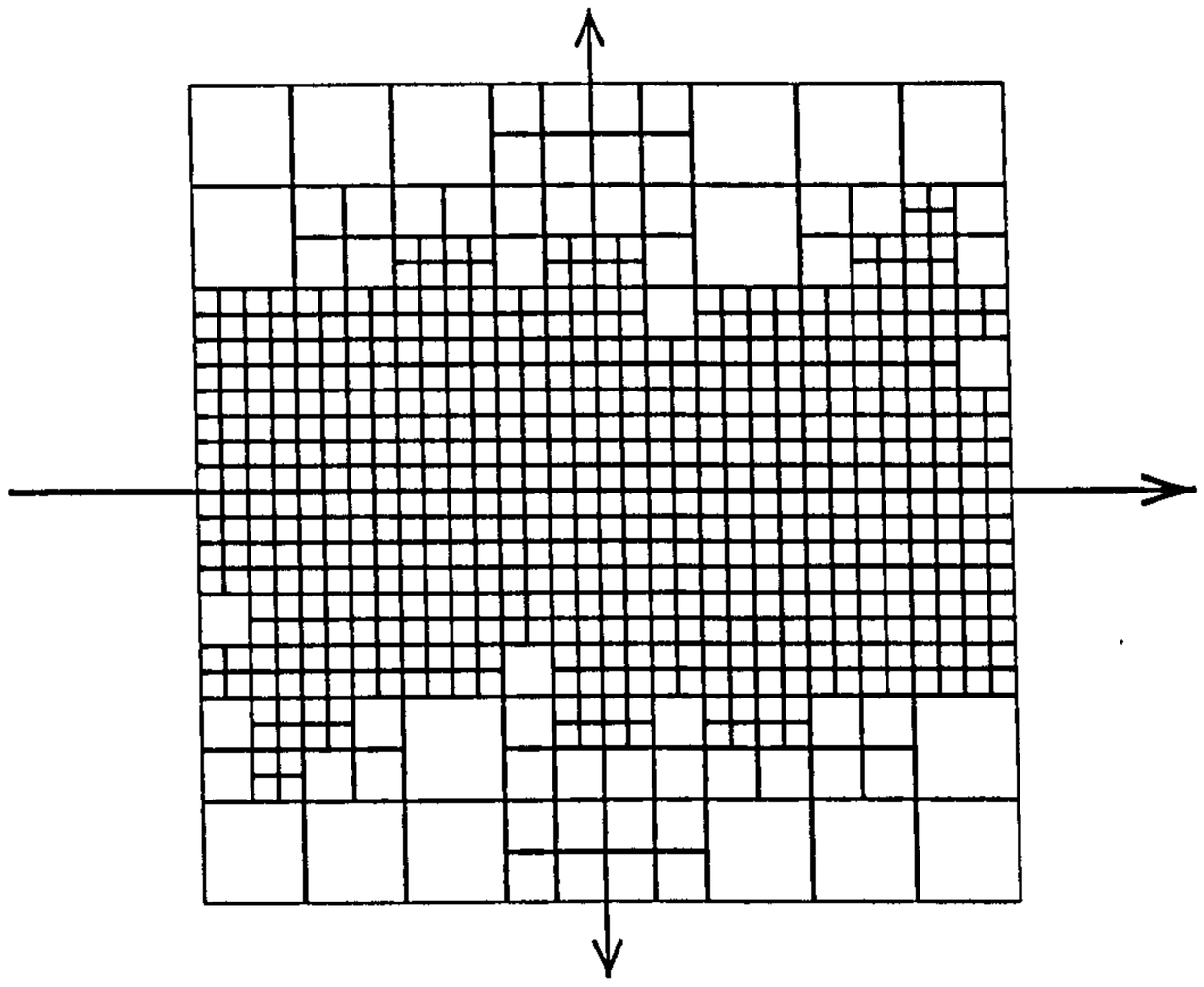


Figure 2.7: An example of brushlet tiling on the frequency plane [105]

2.4.2 Brushlets

While the MFT takes a redundant, block-based approach, which has certain advantages in conducting multiresolution analysis on images, Meyer and Coifman [105, 106] developed a similar transform with oriented periodic oscillating atoms named *"brushlets"*.

The construction of the brushlet transform, however, is completely different. The brushlets are obtained by applying orthonormal windowing in the frequency domain. With an appropriate choice of the windowing function, the overlapping part can be folded without introducing extra redundancy, making the brushlet transform orthonormal. In essence, the brushlet transform adaptively tiles the frequency domain according to the amplitude of the Fourier coefficients. An example of the tiling is shown in Figure 2.7. The authors also reported its application to image compression in [105]. While the brushlet transform achieves good compression rates for highly textured images, it suffers from serious artifacts in the image reconstruction, since its basis functions are less well localised in space.

2.4.3 Wave Atoms

Another basis, named *"wave atoms"*, proposed by Demanet and Ying in [49], and performs local windowing in the frequency plane, like brushlets. Wave atoms are waveforms whose period of oscillation (wavelength) is linked to the size of the support (diameter) of the wave by the parabolic scaling law

$$\text{wavelength} \approx (\text{diameter})^2.$$

In order to achieve a good localisation in frequency and space, the wave atom transform adopted the Villemoes wavelet packet [152] in the frequency domain. Its discrete transform also uses the wrapping strategy used in the Fast Discrete Curvelet Transform (FDCT) implementation [22]. The resulting expansion of the wave atom frame is two times that of the original signal but an orthonormal construction was also reported to be possible.

Compared with brushlets, wave atoms have better spatial localisation: the basis functions only have two bumps in space while the brushlets have more. Another significant difference between the two is that while the brushlet decomposition is adaptive, the wave atoms basis is a fixed transform and, therefore, not image-dependent. In the same paper where the mirror-extended curvelet is reported, a version of DCT-based mirror-extended wave atoms is also presented [48].

2.4.4 Bandelets

Instead of decomposing the images using a fixed dictionary of basis functions, the *"bandelet"* [115, 116] is constructed from a *geometric flow* of vectors which indicate the local directions in which the image intensity have regular variations (see Figure 2.8). Although it is not specifically designed to represent oriented textures, the adaptive nature of the transform provides a unified treatment for both edges and directional patterns.

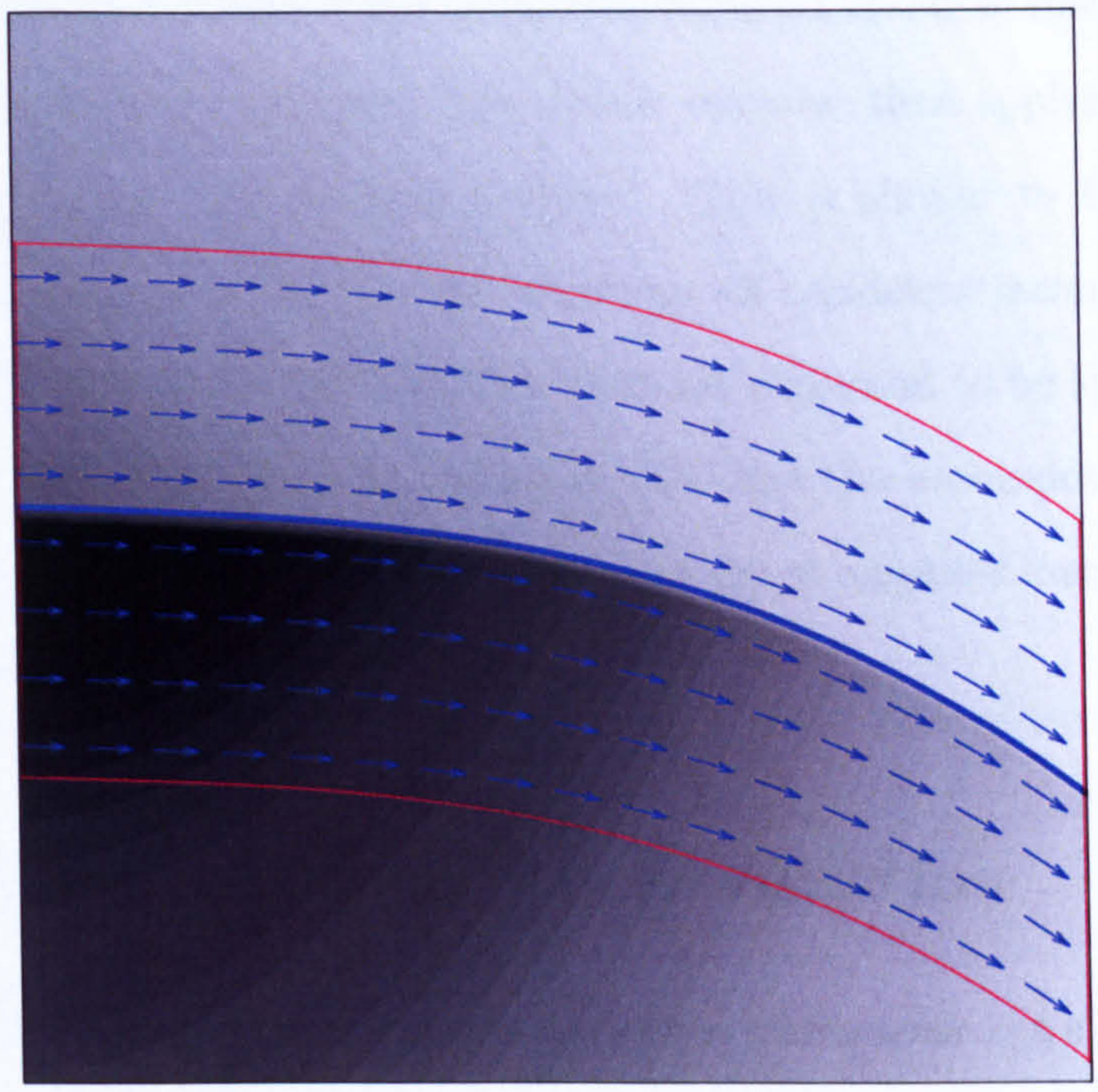


Figure 2.8: The geometric flow along a directional feature

Like wedgelets, the bandelet transform uses a dyadic square segmentation to approximate the images. In each block, the geometry is not defined by finding an edge location, but an orientation along which the image has regular variations. Then the block is warped into a rectangle where the geometric flow will be horizontal and a 1-D wavelet is then applied. This ability to exploit image geometry makes its approximation error decay asymptotically optimal for piecewise regular images. In contrast to curvelet-type systems, where the approximation rate $O((\log M)^3 M^{-2})$ only holds for $\mathbf{C}^2$ curves, bandelets can achieve the optimal decay rate

$$\|f - f_M\|^2 \leq CM^{-\alpha} \tag{2.14}$$

for all $\mathbf{C}^\alpha$ cases, where M is composed of the number of coefficients retained as well as the parameters that specify the geometric flow and segmentation [115].

Peyré and Mallat [119, 120] later constructed an orthogonal version of the bandelet.

The orthogonal bandelet is constructed in the conventional 2D separable wavelet domain, by adaptively segmenting the subbands into dyadic squares, then applying an orthogonal directional filter bank on each individual square. This is similar to the wavelet-based contourlet implementation in [67]. Other variations of bandelets include grouplets [94], which were inspired by a set of grouping laws that are supposed to be applied recursively during the human perception of a natural scene, allowing the decomposition to take into account long range regularities. For a complete coverage of bandelet theory and literature, the reader is referred to [121].

2.5 Image Denoising by Approximation

Image restoration within the framework of an additive white noise is a classical problem in image processing and statistics. The goal is to recover an image f from a noisy observation y:

$$y = f + n, \tag{2.15}$$

where n represents the Gaussian white noise component with variation σ^2. Denoising the image f from y implies the computation of an estimate f' of the original signal which depends only on the observed data y.

2.5.1 Transform Domain Thresholding

Under a linear transform with a certain basis $\mathcal{B}$, the image in the transform domain can be written as follows:

$$\hat{y} = \hat{f} + \hat{n}. \tag{2.16}$$

The $\hat{y}$, $\hat{f}$ and $\hat{n}$ denote respectively transformed observation, original image and noise. The *shrinkage* or *thresholding* refers to the process whereby the estimate f' is reconstructed from

only a few selected coefficients in $\hat{y}$ whose magnitude may have been shrunk. In practice, the method works in three steps:

1. Transform the noisy data y into a specific representation domain $\hat{y}$.

2. Apply thresholding to the resulting coefficients, thereby suppressing those coefficients smaller than a certain amplitude $\mathcal{T}$ (the threshold).

3. Transform back into the original domain as the estimated f'.

Two thresholding methods are commonly used: *hard* and *soft* thresholding. The hard-thresholding function is defined as

$$\Xi_{\mathcal{T}}(x) = x \cdot 1\{|x| > \mathcal{T}\}. \tag{2.17}$$

which keeps the input if it is larger than the threshold $\mathcal{T}$; otherwise it is set to zero. The other popular alternative is the soft-threshold function, also called the *shrinkage* function

$$\Sigma_{\mathcal{T}}(x) = \mathrm{sgn}(x) \cdot \max(|x| - \mathcal{T}, 0). \tag{2.18}$$

The performance of the estimate is measured using the risk $E\{\|f - f'\|^2\}$, where $E\{\cdot\}$ denotes expectation, which similar to the commonly used measures for image reconstruction in denoising and coding, namely SNR and PSNR :

$$SNR(f, f') = 20 \log_{10} \left(\frac{\|f\|}{\|f - f'\|} \right), \tag{2.19}$$

$$PSNR(f, f') = 20 \log_{10} \left(\frac{FMAX}{\sqrt{E\{\|f - f'\|^2\}}} \right), \tag{2.20}$$

where the $FMAX$ corresponds to the constant of the maximum value of the pixel intensity range, typically 255 for a grey-scale image. Suppose the discussion is restricted to hard

thresholding only and with a given threshold $\mathcal{T}$, the remaining number of coefficients is M, then

$$f' = \sum_{m \in I_M} \langle y, b_m \rangle b_m \tag{2.21}$$

which is very similar to f_M defined in equation (2.1). Therefore, the effectiveness of thresholding with respect to the basis $\mathcal{B}$ is directly related to its approximation efficiency. A good representation basis, therefore, can substantially reduce the risk in the estimation, yielding better denoising results. It was shown in [59] that if the basis $\mathcal{B}$ defines a sparse signal representation, then a thresholding (shrinkage) estimator is nearly optimal for all types of signal.

2.5.2 Wavelet Shrinkage

Noise removal by wavelet shrinkage has been extensively studied in literature. The focal point of discussion is on deriving an appropiate threshold. Donoho and Johnson [59] pioneered an adaptive threshold estimation scheme known as the *VisuShrink*, whereby threshold

$$\mathcal{T} = \sigma\sqrt{2\log N}, \tag{2.22}$$

where N is the number of data points, is called the *universal threshold.* The estimation is asymptotically optimal but yields overly smoothed images. Another notable threshold method is the SURE threshold [60], derived from minimising Stein's unbiased risk estimate [148] when soft-thresholding is used. However, the threshold has a drawback in situations of extreme sparsity of wavelet coefficients. The *SUREShrink* is a hybrid of the universal and the SURE thresholds, with the choice being dependent on the energy of a particular subband. It has yielded good image denoising performance and has been used as a standard benchmark in image denoising.

BayesShrink [32] was proposed in order to formulate a Bayesian estimate of the best thresholding rule under the *Generalised Gaussian Distribution* (GGD) assumption. The *BayesShrink* threshold for a subband $\mathcal{H}$ is then computed as

$$T_{\mathcal{H}}^{B} = \frac{\sigma^2}{\sqrt{\max(\sigma_{\mathcal{H}}^2 - \sigma^2, 0)}} \tag{2.23}$$

where $\sigma_{\mathcal{H}}$ is the standard deviation of the subband $\mathcal{H}$. It was demonstrated that *BayesShrink* consistently outperforms other threshold estimation methods.

Compared to traditional Minimum Mean Squared Error (MMSE) methods, such as the Wiener filter [157] and its adaptive variations [88], the wavelet shrinkage has its obvious advantage in the simplicity while the denoising performance is largely limited by the bases approximation power.

2.5.3 Translation Invariant Denoising

Thresholding in a shift-variant basis such as orthonormal wavelets can exhibit pseudo-Gibbs phenomena in the neighbourhood of discontinuities by suppressing too many coefficients. In order to alleviate this problem, Coifman and Donoho [40] introduced the technique of "*cycle spinning*". The idea of denoising via cycle spinning is to apply denoising not only to the observation y, but also to all possible unique circularly shifted versions of it, and to average the results. Suppose If $\mathcal{D}^l$ corresponds to a circular 'delay by l-point' transform, and f_l' is the estimate of f resulting from applying a discrete wavelet transform denoising procedure to $\mathcal{D}^l y$, the cycle spinning denoising estimate of f is given by

$$f' = \frac{1}{L}\sum_{l=0}^{L} \mathcal{D}^{-l} f_l' \tag{2.24}$$

where L is the maximum number of shifts. This treatment can significantly improve the denoising results by shift-variant transforms. It was pointed out by Percival and Walden [117] that this is equivalent to applying standard thresholding to the wavelet coefficients of the maximal overlap discrete wavelet transform, which is also referred to as the *stationary wavelet transform*.

2.5.4 Image Denoising by Directional Representations

Although without explicitly involving a transform, the use of multiscale information in denoising can be traced back to [37] where the Kalman filter is combined with a scale-space model. The idea of exploiting directional coherence was introduced in [88] where an anisotropic image estimation method was incorporated to remove noise. Perona and Malik [118] combined the scale-space model with anisotropic diffusion for noise removal. Since denoising is similar to non-linear approximation, employing a simple thresholding with certain transform can effectively demonstrate its representation sparsity. Applying coefficient shrinkage on a multiscale directional bases therefore naturally can be regarded as a form of multiscale anisotropic smoothing. For this reason, most of the citations in Sections 2.3 and 2.4 adopt the thresholding type denoising algorithm in their numerical experiments.

There is also dedicated literature on denoising by using directional representations. Starck et al. [143] employed the curvelet-99 transform in denoising. The same authors also proposed a combined approach with curvelets and wavelets in denoising [144]. Candés and Guo combined the curvelet-99 with Total Variant (TV) minimisation [23] and obtained remarkable results on natural images. A variant of the curvelet-99 transform with block-based complex ridgelet transform was also used to denoise images by shrinkage of the transform domain coefficients [33]. A diffusion based curvelet shrinkage was reported by using the FDCT in [22].

Image denoising experiments using the cycle spinning on the contourlet transform were reported in [66]. Matalon et al. [99] considered the basis-pursuit denoising approach with the non-subsampled contourlet transform. Jiang et al. [84] proposed a thresholding algorithm using interscale correlations. The approach and results are similar to the work reported by Po and Do [124].

Federico and Kaufmann [69] was also reported application of applying the wave atoms thresholding in denoising the Digital Speckle Pattern Interferometry (DSPI), which are fringe patterns that can be efficiently represented by the wave atoms.

Finally, a study on using the orthogonal bandelet bases to estimate f' from noisy observation y was given in [122]. It was shown that the translation invariant bandelet estimator gives better results than translation invariant wavelet estimators and the Gaussian Scale Mixture model with Steerable Pyramid [129] on geometric images. For complex natural images, the scheme gives results similar to [129].

2.6 Summary and Discussion

The attempt to find a sparse representation for natural images has resulted in various methods with different approaches. This chapter has given an overview on some recent developments in directional image representation. These representations were discussed in three categories. The first is geometric edge-based models, which characterise the edge location. The second type of representations is a set of transforms whose basis functions are linear singularities. It was recognised that exploiting geometric regularities alone does not capture the characteristics of complex natural images. The third class of transforms can provide sparse representations for directional textures which can be considered as the complementary part to piecewise smooth geometrical structures. The chapter also gave a brief account on the use of sparse representations in image denoising by coefficient

Representation	Fixed/Adaptive	Targeted Feature
Curvelets [143]	Fixed	edges
Contourlets [54]	Fixed	edges
Shearlets [63]	Fixed	edges
Wedgelets [55]	Adaptive	edges (parametric)
Brushlets [105]	Adaptive	textures
Wave Atoms [49]	Fixed	textures
MFT [20]	Adaptive	edges (parametric)
MFT [78]	Fixed	texture
Bandelet [116]	Adaptive	edges, textures

Table 2.1: A comparison of different directional representations

thresholding. Comparing with other methods, which typically blur object edges or have high complexity, thresholding in directional image representations is an efficient way to remove noise while preserving prominent directional features.

These aforementioned directional representations are briefly summarised and compared in Table 2.1[1]. Among the listed transforms, although the bandelet can potentially deal with directional patterns other than edges, its ability to represent textures is limited, especially with non-directional textures, which cannot be characterised by the geometric flow. With the MFT, there have been separate investigations into applying it to analyse edges and textures, but no attempts have been made on combining them into a uniform setting. In the following three chapters, a method for linear feature extraction from noisy images and new directional transforms for representing directional textures will be presented, which can be regarded as "upgrades" for MFT-based edge and texture representation. Also the joint "edge + texture" directional representations by the new transform and MFT will be discussed in the context of denoising, which bridges the gap between geometry-oriented and harmonic-oriented directional representations.

[1]Since the complexity and redundancy factor of these representations can be different depending on implementation and construction, readers are referred to the original citations for details.

Chapter 3

Linear Feature Extraction

3.1 Introduction

The importance of local linear features has been recognised both in computer vision and image processing, leading to a wide range of approaches which are based upon different underlying models. The traditional way to characterise image boundaries is to use edge detection methods, such as the Canny detector [30]. Moreover, it is of particular interest to locally represent the contours as groups of line segments, instead of describing the image structure pixel-by-pixel. Such parametric description according to a line model has certain advantages:

- **Compactness.** Since the contour information is locally characterised by only the orientation and offset location of the line, the storage requirement is greatly reduced for such semantic representation. Evidence for this can be the chain code [72] used in MPEG-4 standard video compression and, more recently, the wedgelet representation [55].

- **Resolution independence.** Such parametric representation does not depend on the resolution of the original data, so it is possible to process the information at

any desired scale which can possibly be sub-pixel resolution, making it very useful in image interpolation.

- **Affine invariance.** Since the space of straight lines is closed under the affine transform, the contours described in terms of line parameters can be easily processed in an affine invariant manner. This property can be used in designing an image hash for content-based retrieval purposes.

While the advantage of using a parametric representation instead of simple edge detection is clear, estimation of the line parameters is a non-trivial problem. A common parameter estimator used after the edge detection is the Hough transform [62]. However, the major drawbacks of the *"edge detection + Hough transform"* approach are the high computational burden, and the fact that most of the edge detection methods proposed to date are prone to noise. The work of Calway *et al.* [18] has established a spectrum model for such linear feature extraction tasks with the MFT in a multiscale fashion, followed by Davies [45] and Meulemans [101]. It simplifies the feature extraction process into a matter of estimating the model parameters, without using the Hough transform. With a multiresolution quadtree model, the parameters can be grouped into semantically meaningful clusters corresponding to curvilinear edge contours. However, it does not consider noise distortion, where as in many vision-related tasks, robustness against noise is much needed. With this motivation, the model needs to be extended to deal with the effects of noise, which is the major consideration here.

As a guideline, image analysis in the presence of noise usually consists of three key ingredients:

1. **A Suitable Model**

   When the signal is observed with noise, it is important that a suitable model is used to recognise the fact that the observation is a mixture of both noise and signal. The

model can be about the nature of the noise, how the signal and noise are combined, and more importantly the domain where the model is built. The choice of domain can be a particular transform like wavelets, or in the distribution of coefficients.

2. **Multiresolution**

   It is clear that multiresolution analysis is particularly useful with noisy images. This is because natural images usually consist of a high proportion of lowpass energy, where the effect of the noise is less significant than in highpass. Therefore, it is natural to choose a multiresolution approach for analysis of noisy images.

3. **Redundancy**

   It is well known that transform redundancy can offer better robustness against noise than other critically-sampled approaches [64]. This could be by simply over-sampling the signal or using an over-complete frame for image representation to improve the analysis accuracy.

This chapter presents a method which can effectively extract curvilinear structures from noisy images in a parametric fashion. While underlying estimator is essentially the same as other MFT-based methods, the main merit of this chapter lies in the extension to the estimation in the presence of noise by employing the above-mentioned guidelines, which will be discussed in detail.

3.2 Local Linear Features

3.2.1 The Single Linear Feature Model

A perfect linear feature of infinite length in the continuous domain with a feature profile function $f(\cdot)$, it can be represented as

$$x(\xi) = f(\xi \cdot v_\theta), \tag{3.1}$$

where

$$v_\theta = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}. \tag{3.2}$$

is a vector of unit length in the direction of the line feature and $\xi = [\xi_1, \xi_2]^T$ is the spatial coordinate vector. Its Fourier spectrum is concentrated in another line perpendicular to the feature orientation [16]:

$$\hat{x}(\omega) = \hat{f}(\omega \cdot v_\theta)\delta(\omega \cdot v_{\theta_\perp}). \tag{3.3}$$

and $v_{\theta_\perp} = v_{\pi/2+\theta}$, and $\omega = [\omega_1, \omega_2]^T$ is the frequency coordinate vector.

It can also be shown that the phase response $\phi(\omega)$ of such a singularity is linear, with a gradient depending on the centroid distance $\rho = \|\rho\|$ of $x(\cdot)$, where ρ is the centroid vector:

$$\hat{f}(\omega) = |\hat{f}(\omega)|e^{j\phi(\omega)}. \tag{3.4}$$

$$\phi(\omega) = \rho\omega + \epsilon(\omega). \tag{3.5}$$

The phase constant $\epsilon(\omega)$ depends on the type of feature, such as a line or a step edge [18].

In reality, the feature of interest is usually confined within a spatial window. This

gives the *local* linear feature as a form of multiplication of the infinite line with a window function $w(\xi)$,

$$x_w(\xi) = x(\xi)w(\xi). \tag{3.6}$$

The Fourier transform would result in the frequency domain in a convolution of the spectrum of the window and the feature.

$$\hat{x}_w(\omega) = \hat{x}(\omega) * \hat{w}(\omega). \tag{3.7}$$

A previous study [112] has concluded that with a real and symmetric smooth window function centred at the origin, the phase response of the windowed feature is still close to linear. Therefore, with an appropriate choice of the window function, the model's simplicity is retained despite the confinement.

The rest of the model deals with the location and scale of the linear feature in the image. In other words, parameters for a line must be associated with a spatial coordinate η at a certain scale s. These two parameters form a wavelet-type index (η, s). Combined with the line description parameters (θ, ρ), these four parameters form the parameter space of the model. Thus a linear feature parameter set can be defined as :

$$\zeta = (\eta, s, \theta, \rho). \tag{3.8}$$

For the sake of convenience, we can denote a feature's line parameter at scale s and location η as

$$\zeta_{\eta,s} = (\theta_{\eta,s}, \rho_{\eta,s}). \tag{3.9}$$

3.2.2 Parameter Estimation

For a single linear feature, the scale and location are usually determined by the image model instead of being estimated, and can be directly taken from the indices of the estimator, despite the choice of the implementation being either a quadtree or a pyramid. The complete feature description will need an estimate for the orientation and the centroid position.

Both Calway [18] and Davies [45] used a *moment of inertia tensor* to estimate the principal orientation of the feature, initially used to analyse a system of point mass rotation about a fixed axis. The tensor covariance matrix is defined as

$$\Gamma_f = \sum_{\omega} |\hat{f}(\omega)|^2 \omega\omega^T. \tag{3.10}$$

where the superscript T denotes a matrix transpose.

Performing Principal Components Analysis (PCA) on the 2×2 tensor matrix Γ_f yields 2 eigenvalues $\lambda_1 \leq \lambda_2$ and the corresponding eigenvectors e_1 and e_2. The two eignenvectors represent the major and minor axes of an elliptical approximation of the spectrum, which are normal to each other. The orientation of the line can be estimated by

$$\theta = \pi/2 + \arctan\left(\frac{e_2(2)}{e_2(1)}\right). \tag{3.11}$$

The position of the centroid of the linear feature can be estimated from the phase of the frequency spectrum. This is done by taking the pairwise average correlations between neighbouring coefficients in each of the two directions. The conjugate products between neighbouring samples are defined as:

$$d_m(\omega) = \hat{f}(\omega)\hat{f}^*(\omega + \Omega_m), \tag{3.12}$$

where $m \in \{1, 2\}$ for the x and y dimensions respectively, and

$$\Omega_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \Omega_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{3.13}$$

The estimated position ρ can be calculated as below:

$$\rho = \frac{M}{2\pi} \arg \begin{bmatrix} \sum_\omega d_1(\omega) \\ \sum_\omega d_2(\omega) \end{bmatrix}. \tag{3.14}$$

where M is the size of the window.

3.3 Modelling Noisy Linear Features

The above mentioned frequency linear feature model has achieved considerable success in linear feature extraction [158] as well as other applications such as motion estimation and analysis [101]. However, the major drawback of the model is that it assumes the data consist of only the desired signal. Unfortunately, the physical nature of the sensor and the transmission channel does not guarantee the data to be clean, and the inertia tensor which is used for estimating the feature orientation is extremely sensitive to noise. This is due to the fact that the tensor matrix is a sum of all weighted tensors. If the noise is assumed to be white and Gaussian, with noise energy spread evenly in the spectrum and the signal energy being a relatively small part of the whole, the dispersion of the spectrum magnitude is more isotropic, making finding the correct orientation difficult.

However, in order to analyse the local linear features on noisy images, one cannot rely on a denoising procedure as a pre-processing stage. Instead, one of the goals of deriving an appropriate linear feature model for noisy data is to help design a better denoising algorithm, as we will discuss in the next chapter. In this section, a "Signal+Noise" model

is presented in order to accommodate the case when the linear feature is corrupted with noise.

3.3.1 The Signal+Noise Model

We have the original signal $f(\xi)$ which contains a linear feature and the noise function $n(\xi)$ is assumed to be Gaussian additive and white. The observed signal is then :

$$y(\xi) = f(\xi) + n(\xi). \tag{3.15}$$

It is known that under a linear transform, such as the Fourier transform, the transformed signal will still be the combination of transformed signal and noise. The transformed data can be written as

$$\hat{y}(\omega) = \hat{f}(\omega) + \hat{n}(\omega). \tag{3.16}$$

The transformed noise component $\hat{n}(\omega)$ will still be Gaussian white noise.

It is already known that the magnitude of $\hat{f}(\omega)$ should be a line through the origin. In reality, due to the imperfection of the linear feature itself and the windowing, the magnitude energy has a certain spectrum spreading, making the shape anisotropic. Such spreading is reflected in the tensor covariance matrix Γ_y and the corresponding Gaussian function is a simple approximation of the shape of the magnitude spectrum. The observed signal $|\hat{y}(\omega)|$ can be treated as a Gaussian mixture of two components:

$$|\hat{y}(\omega)| \approx w_f g_f(\omega) + w_n g_n(\omega). \tag{3.17}$$

From the symmetry property of the Fourier spectrum, the components are always positioned at the origin, which means their corresponding Gaussian functions are zero-mean.

Then $g(\cdot)$ can be simplified without the mean vector:

$$g(\omega) = \exp\left(-\frac{\omega^T \Gamma^{-1} \omega}{2}\right). \tag{3.18}$$

Because $\hat{n}(\omega)$ is still Gaussian white noise, we can use an isotropic circular Gaussian blob to model this. Then the covariance matrix for $g_n(\omega)$ is simply:

$$\Gamma_n = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3.19}$$

3.3.2 Parameter Estimation

A modified *Expectation-Maximisation* (EM) algorithm [50] is used to estimate the parameters for this mixture model. The N-point magnitude spectrum is treated as the incomplete data to be fitted, and the magnitude is the weight of each data point. In the process, we do not alter Γ_n. The whole process can be described as below:

- **E-Step (Expectation)** The expectation for the signal component at iteration t is estimated as

$$E_f^t(\omega) = \frac{w_f^t g_f^t(\omega)}{w_f^t g_f^t(\omega) + w_n^t g_n^t(\omega)}. \tag{3.20}$$

- **M-Step (Maximisation)** In this step, the parameters are determined according to the estimate of the expectation E^t. For Gaussian mixture models, this corresponds to re-estimating the w_f^{t+1}, w_n^{t+1} and in this case, only the covariance matrix Γ_f^{t+1} according to

$$w_f^{t+1} = \frac{1}{N} \sum_{\omega} E_f^t(\omega), \tag{3.21}$$

$$w_n^{t+1} = 1 - w_f^{t+1}, \tag{3.22}$$

$$\Gamma_f^{t+1} = \frac{\sum_\omega |\hat{y}(\omega)| E_f^t(\omega) \omega\omega^T}{\sum_\omega E_f^t(\omega)}. \tag{3.23}$$

- The E-M steps are repeated until convergence or within an acceptable numerical margin.

After the estimate for Γ_f is finalised, the orientation then can be estimated using PCA as described in equation (3.11).

The estimation of the centroid position relies on the phase spectrum of the data. However, it does not seem to be feasible to derive a similar mixture model which can separate the signal and the noise in the phase spectrum. Under the assumption that the noise is zero-mean Gaussian white with variance σ_n^2, the power spectrum of the noise should be equal to its variance. The noise has the same power everywhere in the frequency domain. However, its effect introduced on the phase is largely determined by the magnitude of a particular complex number. This is illustrated in Figure 3.1, where the vector represents certain complex number and the dimension of the box can be understood as the noise standard deviation. It is clear that the angle of vectors with larger magnitude is less affected by the noise. Therefore, phase difference estimated from those coefficients is more reliable.

This motivates us to introduce a weight function on the conjugate product $d_m(\omega)$ to minimise the disruption caused by noise. Because the correlation is taken pairwise, two neighbouring coefficients both with large magnitudes should receive considerably more weight as compared to when only one of them is significant. Furthermore, due to the fact that the signal under consideration contains linear features, the weight function should select a narrow oriented band at the direction approximately v_θ. With the magnitude

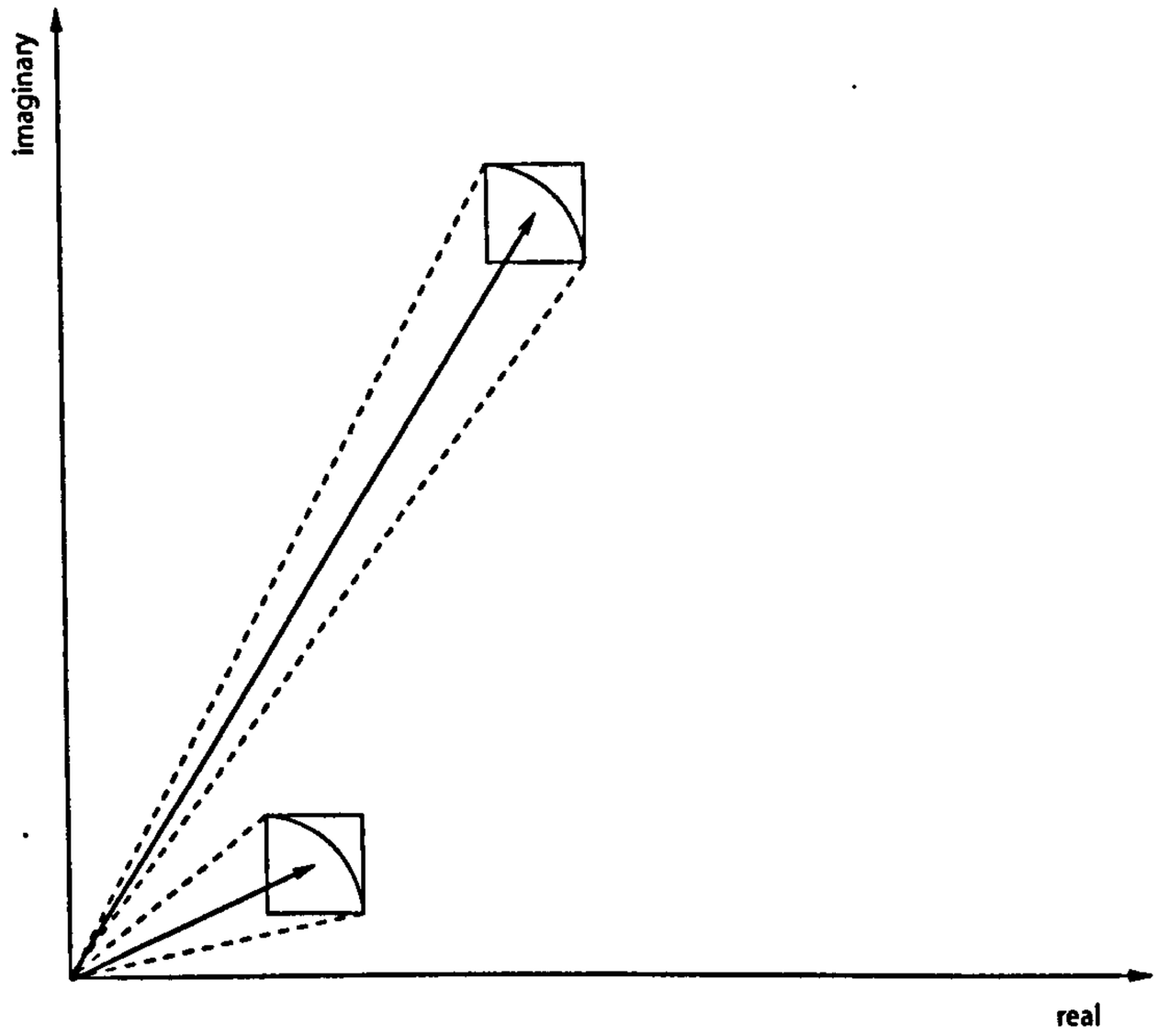


Figure 3.1: The effect of noise changing the angle of phase

model established, a band-limiting function can be introduced as:

$$h(\omega) = \frac{g_f(\omega)}{g_f(\omega) + g_n(\omega)}. \tag{3.24}$$

The "Signal/(Signal+Noise)" form is akin to the well-known Wiener filter [157], allowing us to discard most of the smaller coefficients. The conjugate product weight function is defined as:

$$s_m(\omega) = h(\omega)(|\hat{f}(\omega)| * |\hat{f}(\omega + \Omega_m)|)^2. \tag{3.25}$$

The position vector ρ is modified as:

$$\rho = \frac{M}{2\pi} \arg \begin{bmatrix} \sum_\omega s_1(\omega) d_1(\omega) \\ \sum_\omega s_2(\omega) d_2(\omega) \end{bmatrix}. \tag{3.26}$$

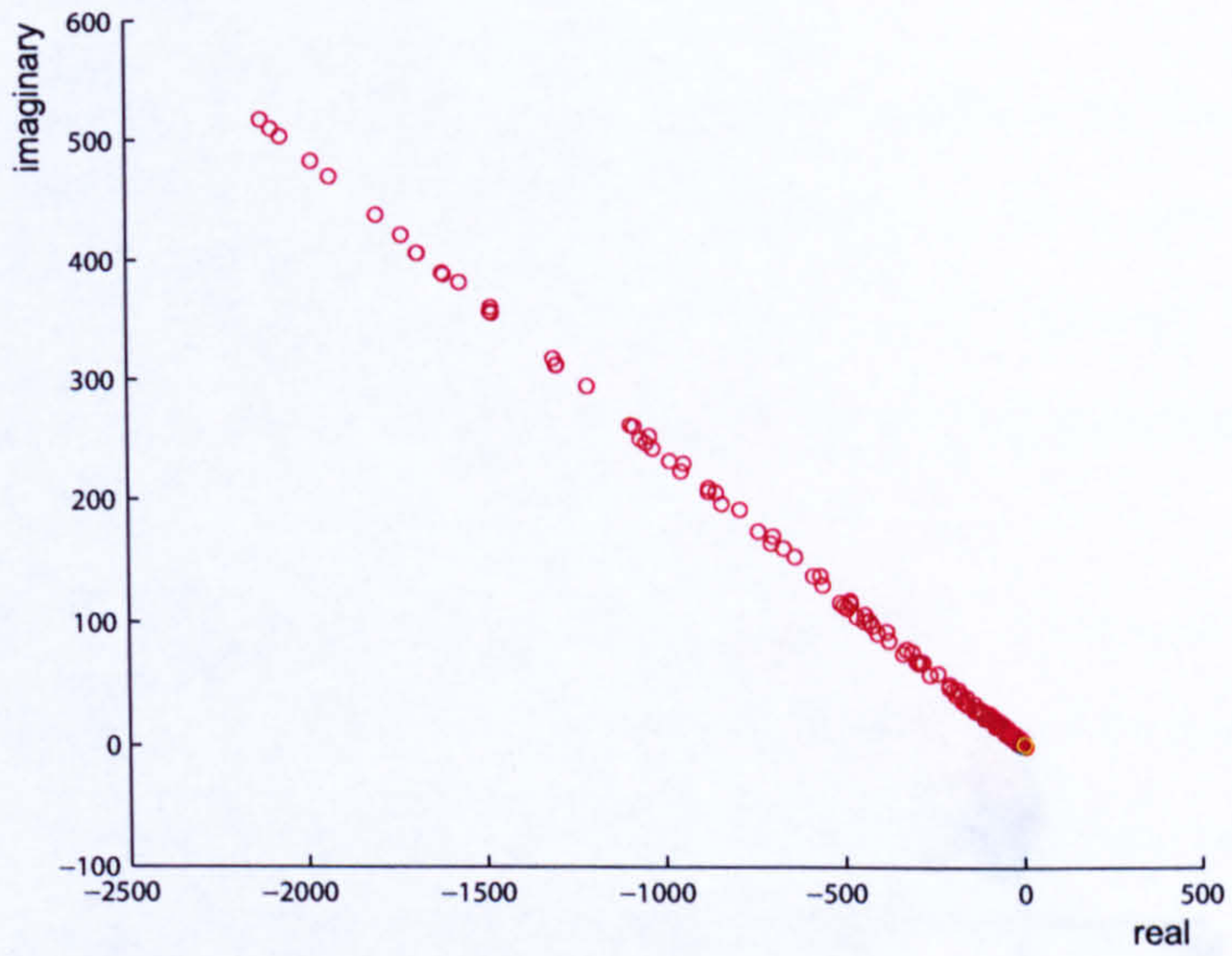


Figure 3.2: Scatter plot of $d_1(\boldsymbol{\omega})$ on clean linear feature

The effectiveness of the weight function $s_m(\boldsymbol{\omega})$ can be verified by visualising the difference between $d_m(\boldsymbol{\omega})$ and $s_m(\boldsymbol{\omega})d_m(\boldsymbol{\omega})$. Figure 3.2 is a scatter plot of all $d_1(\boldsymbol{\omega})$ taken from a 64×64 image containing a clean straight line. As expected, the points are aligned along a straight line of which the slope is related to the feature offset in x-axis. With the noise (15dB) added, the plot of $d_1(\boldsymbol{\omega})$ (Figure 3.3) is significantly scattered, introducing error in the estimation. Figure 3.4 shows how $s_1(\boldsymbol{\omega})d_1(\boldsymbol{\omega})$ can improve the estimate by assigning different weights to the data points which significantly enlongate the distribution of the points by placing greater distance between certain data points and the origin. The resulting scatter plot is closer to a line therefore the centroid position can be more accurately estimated.

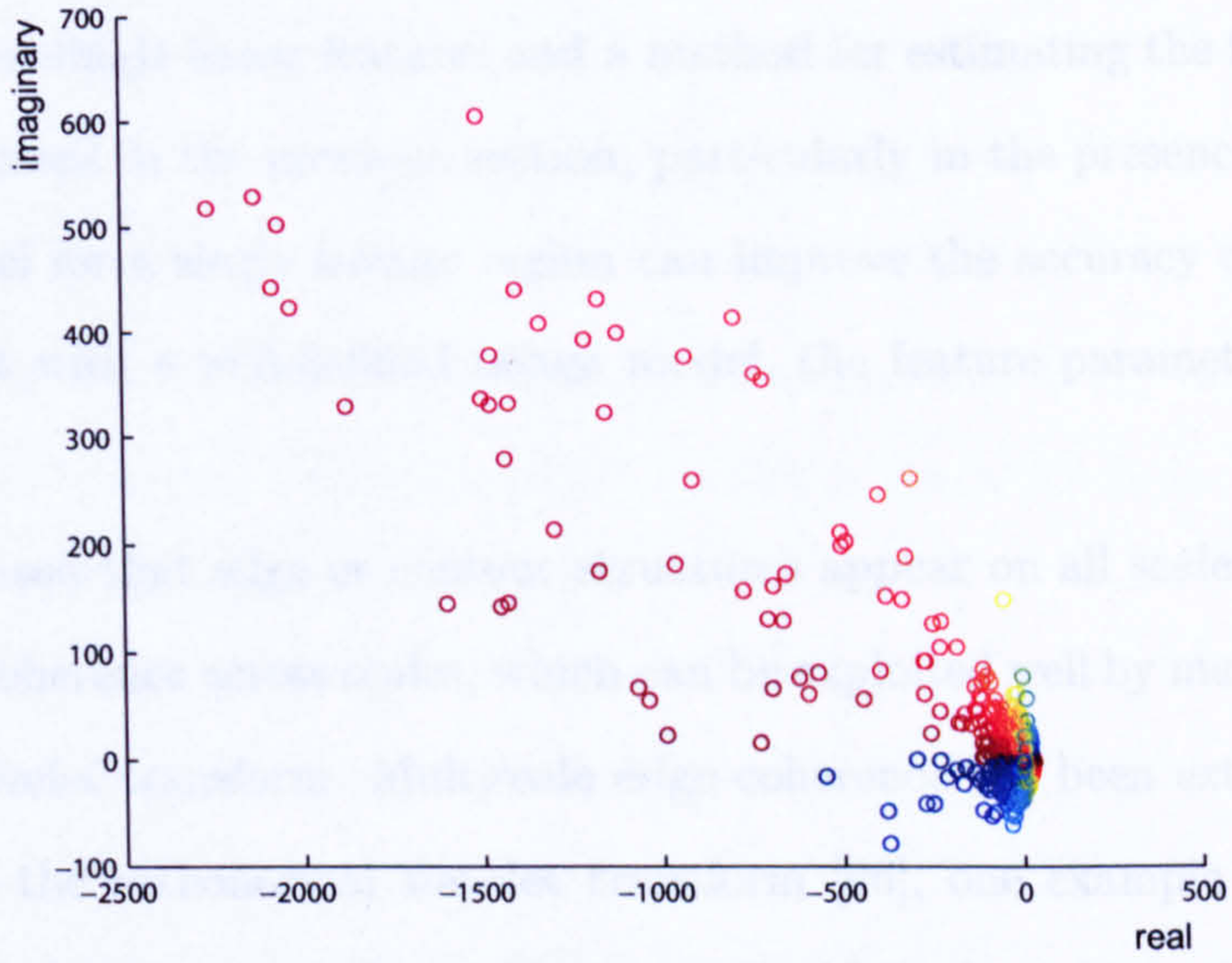


Figure 3.3: Scatter plot of $d_1(\boldsymbol{\omega})$ on noisy linear feature

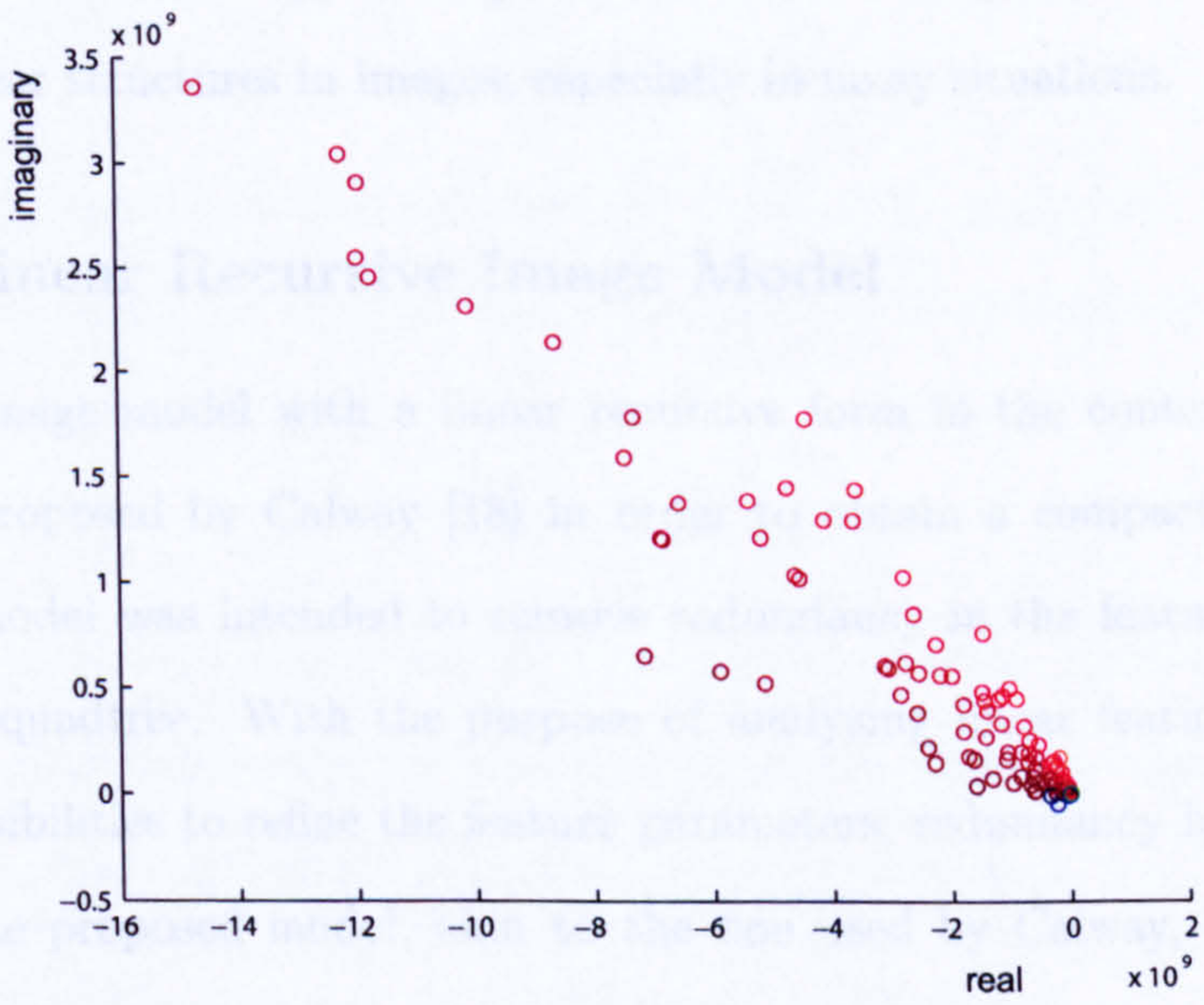


Figure 3.4: Scatter plot of $s_1(\boldsymbol{\omega})d_1(\boldsymbol{\omega})$ on noisy linear feature

3.4 Image Model and Parameter Refinement

The modelling of single linear features and a method for estimating the feature parameters have been discussed in the previous section, particularly in the presence of noise. Indeed, a mixture model for a single feature region can improve the accuracy of parameters from noisy data, but with a well-defined image model, the feature parameters can be further refined.

It is recognised that edge or contour structures appear on all scales of the image and have a strong coherence across scales, which can be exploited well by multiscale transforms, such as the wavelet transform. Multiscale edge coherence has been extensively studied in the context of the orthonormal wavelet transform [96], one example being the zerotree model [137] used in image coding. Also, some published work on statistical properties of contourlet transform [124] as well as the fast discrete curvelet transform [15] shows that such inter-scale correlation is the major characteristic of the corresponding transform domain coefficients. This suggests a general multiscale image model would be helpful to analyse the linear structures in images, especially in noisy situations.

3.4.1 A Linear Recursive Image Model

A multiscale image model with a linear recursive form in the context of linear feature analysis was proposed by Calway [18] in order to obtain a compact representation for curves. The model was intended to remove redundancy in the feature data and was in the form of a quadtree. With the purpose of analysing linear features in noise as well as seeking possibilities to refine the feature parameters, redundancy in the representation is needed. The proposed model, akin to the one used by Calway, is an over-complete pyramidal representation.

Suppose the set of all linear structures at certain level s is denoted as $X_s = \{x^M_{\eta,s}(\xi)\}, \forall \eta$

which can be seen as certain states, where M is the configured window size. It obeys the following stochastic evolution:

$$X_s = A_s X_{s-1} + B_s U_s + W_{s-1}. \tag{3.27}$$

where the matrix U_s represents the feature innovations and W_{s-1} can be understood as certain *interscale residual difference*, while smaller s corresponds to coarser scale. The model's initial condition is set to:

$$X_0 = B_0 U_0 + W_0. \tag{3.28}$$

where B_0, U_0 and W_0 are the initial vectors for the model.

In order to make the model general, one cannot assume that such model can always be directly measured. The states from the model could be observed with distortion, ie. the observation Z_s:

$$Z_s = H_s X_s + N_s. \tag{3.29}$$

where H_s corresponds to certain operation on X_s and N_s can be the noise. Suppose that the dimension of each scale is equal, which means there is no decimation, the parent-child relation is defined as :

$$x^M_{\eta,s}(\eta) \mapsto x^{2M}_{\eta,s-1}(\eta). \tag{3.30}$$

which means a local feature centered at η has a parent on the previous scale which shares the same origin, but with twice the window size. In case that a previous scale is decimated by a factor of 2, this relationship is equivalently expressed as:

$$x^M_{\eta,s}(\eta) \mapsto x^M_{\lceil \eta/2 \rceil,s-1}(\eta). \tag{3.31}$$

3.4.2 The Kalman Filter

The Kalman filter [86] is an efficient recursive filter which estimates the state of a dynamic system from a series of incomplete and noisy measurements. The image model presented above meets the assumption of the Kalman filter exactly, but the terminology is slightly different. X_s are understood as *states* and the observed Z_s are the *measurements* from the states. W_s and N_s are called *process noise* and *measurement noise* respectively, and assumed to be independent from each other. This is similar to the work described in [37] where the Kalman filter is performed on the scale-space.

In the context of the image model, the measurement Z_s, being either the orientations or the centroid positions, are observed from the "Signal+Noise" model previously discussed. The *hidden* states X_s are the refined version of the feature parameters which can be computed by the Kalman filter. The filter essentially consists of two steps:

1. **Scale Update ("Predict")**

$$\tilde{X}_s^- = A_s \tilde{X}_{s-1} + B_s U_s. \tag{3.32}$$

$$P_s^- = A_s P_{s-1} A_s^T + Q_s. \tag{3.33}$$

2. **Measurement Update ("Correct")**

$$K_s = P_s^- H_s^T (H_s P_s^- H_s^T + R_s)^{-1}. \tag{3.34}$$

$$\tilde{X}_s = \tilde{X}_s^- + K_s(Z_s - H_s \tilde{X}_s^-). \tag{3.35}$$

$$P_s = (I - K_s H_s) P_s^-. \tag{3.36}$$

Essentially, the filtering can be considered as a "denoising" process for the measured

noisy states.

3.4.3 Filter Configuration

Although there are a number of variables in the image model, without loss of generality, the model can be simplified a great deal by making reasonable assumptions for the task of parameter estimation.

First, since $x^M_{\xi,s}$ is a scalar value, the innovation term U_s is not necessary. Also, it is safe to set $A_s = I$ since the error of this simplification can be accommodated by the residual W_{s-1} which ensures the model is still complete. Therefore equation (3.27) reduces to:

$$X_s = X_{s-1} + W_{s-1}. \tag{3.37}$$

The initial condition reduces to

$$X_0 = Z_0. \tag{3.38}$$

For the measurement, since noise is the only distortion being considered, this means $H_s = I$:

$$Z_s = X_s + N_s. \tag{3.39}$$

The Q_s are the covariance matrices for the process noise W_s. As it proceeds to finer scales, the residual W_s is expected to have higher variance. A simple model can be :

$$Q_s = sI. \tag{3.40}$$

The formulation for the measurement noise covariance matrices R_s is a bit more complicated, due to the fact that the noise variance in the states is not equal to the noise variance on the image σ_n^2. For a particular scale, the standard deviation $\hat{\sigma}$ of the observation Z_s is

estimated by the robust median filter (RMF) [59] commonly used in wavelet-based denoising, only that instead of taking the HH band in the wavelet domain, the 2D DCT is used instead:

$$\hat{\sigma} = \frac{Median(|\text{DCT}(Z_s)|)}{0.6745}. \tag{3.41}$$

Then R_s is estimated as :

$$R_s = (s\hat{\sigma})^2 I. \tag{3.42}$$

Since the lowpass frequency coefficients are not taken away from the DCT, this causes inaccuracy in the estimation of $\hat{\sigma}$. However, due to the statistical nature of the RMF, sufficient data samples are required in order to make the estimation reliable. The term s is multiplied in order to compensate for the presence of the lowpass signal's effect on the median.

The above configuration for the filter is general despite the nature of measurement being collections of orientations or centroid positions. However, one exception should be noted: when the model uses decimation at coarser scales, as described in equation (3.31), due to the constant window size M, the prediction of next level's centroid position should be increased by a factor of 2, therefore setting $A_s = 2I$.

3.4.4 Filter Verification

It is known that the Kalman filter is optimal in term of Minimum Mean Squared Error (MMSE) when the noise is Gaussian white. In order to verify the above configuration for the Kalman filter indeed retain the optimality, we applied the filter on a 4-level Laplacian pyramid decomposition of a 15dB noisy `lena` image where each level is regarded as a state.

To test the filter, we take the residual between the original noisy image and the filtered version. The hypothesis of its optimality is true if the residual has a Gaussian distribution. This can be verified by looking at the autocorrelation of the residual. If the distribution

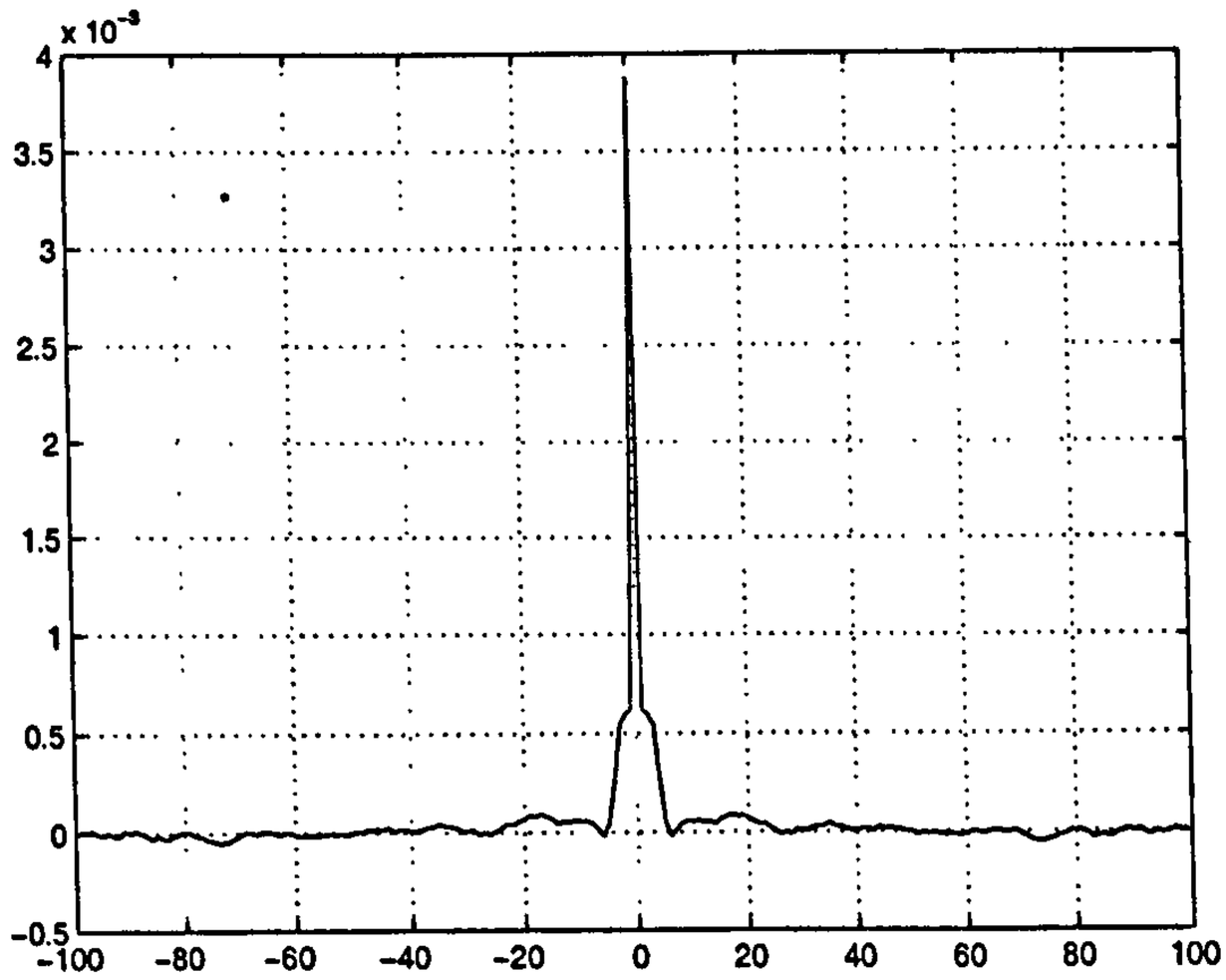


Figure 3.5: The autocorrelation of the Kalman filter's residual

of the residual is indeed Gaussian, the autocorrelation gives a high spike and everywhere else is 0.

The plot of the autocorrelation on the residual from our test was presented in Figure 3.5 which suggests that the residual indeed has a Gaussian distribution. Therefore the configuration of the Kalman filter is verified for its optimality.

3.5 Feature Significance and Selection

So far, we have discussed local frequency domain model for linear features as well as an image model which takes advantage of multiscale coherence of edges. The goal of building these models is to obtain accurate estimates of local linear feature parameters in the presence of noise. For every window at a given scale, feature parameters will be estimated regardless of the fitness between the data and the model. This makes it necessary to measure the significance of estimated features otherwise the analysis process would be

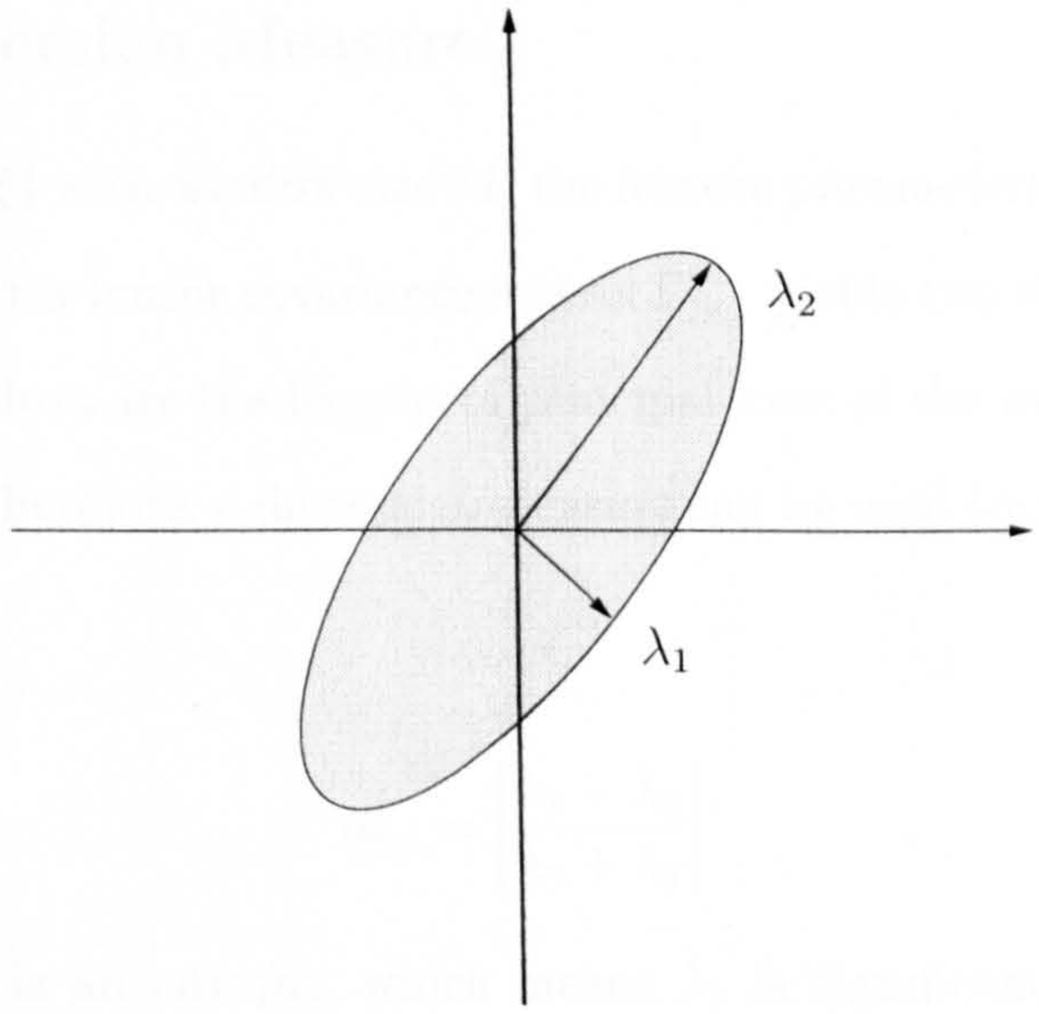


Figure 3.6: The eigenvalues of a spectrum inertia tensor covariance matrix

blind. The measure of feature significance could be a measure of anisotropy or of intensity contrast. In any case, such a significance measure should denote a degree of signal fit to the model and eventually can be used in classifying regions to be either containing linear features or not.

With noise added to the image, finding a working significance measure is more difficult than doing so for a clean image. This is because most computations on the noisy data are usually noisy as well. Noise in the significance measure means that there is a risk that false features will be included and also likely to miss true features. Instead, the approach taken here is to use several measures from different aspects and then use a voting scheme to determine the importance of a feature. This should introduce robustness in the selection of features from noisy images.

3.5.1 The Dispersion Measure

For a local region $f^M_{\eta,s}(\boldsymbol{\xi})$ with window size M, the feature parameters $\zeta_{\eta,s}$ can be estimated. In the process, the inertia tensor covariance matrix $\Gamma_{f_{\eta,s}}$ yields two eigenvalues, λ_1, λ_2 after PCA. The two eigenvalues are the lengths of principal axes of the magnitude spectrum, as shown in Figure 3.6. Therefore, a dispersion measure can be used to measure the anisotropy of the linear feature.

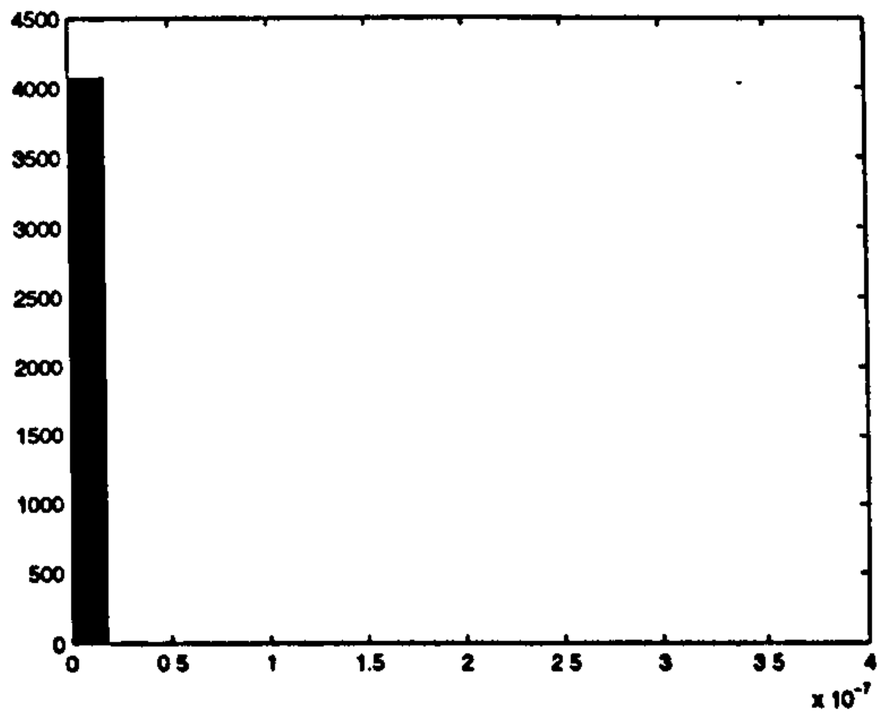
$$a_{\eta,s} = \left| \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} \right| \tag{3.43}$$

When the feature is anisotropic, which means λ_1 is significantly larger than λ_2, the measure will be closer to 1. Otherwise, if $\lambda_1 \approx \lambda_2$, the measure will be close to 0. Therefore a higher $a_{\eta,s}$ indicates a more significant linear feature.

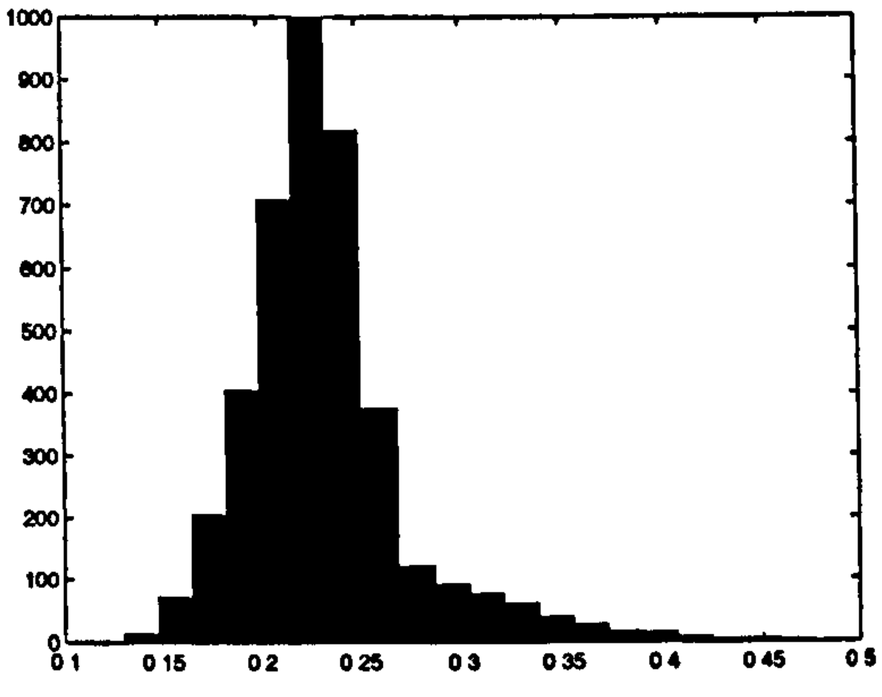
For a particular scale s, the set which contains dispersion measurements for all $f^M_{\eta,s}(\xi)$ is denoted as

$$\mathcal{A}_s = \{a_{\eta,s}\}\,,\ \forall \eta. \tag{3.44}$$

The dispersion measure gives decent results with clean images. However in the presence of noise, even when the inertia tensor is computed from the "Signal+Noise" model, the resulting Γ_f is still nearly isotropic if there is too much noise. A problem also raises with empty regions, when the noise spectrum appears to be oriented. With the noise component in the Signal+Noise model representing most of the energy, the 'signal' component will then become anisotropic, resulting in a high dispersion measure. This would create a *"phantom feature"* in the final selected feature map.



(a) Histogram of $\{w_{\eta,5}\}$, $\forall \eta$



(b) Histogram of $\mathcal{W}_s$

Figure 3.7: Distribution alteration for signal weights, 10dB noisy `lena`, finest scale

3.5.2 The Signal Weight

In equation (3.17), the components are weighted by a scalar value, where w_f corresponds to the signal Gaussian weight and w_n is the weight of the noise Gaussian. The reason for using the w_f as an indication for feature significance is clear: it is the ratio of the signal component energy. Since $w_f + w_n = 1$, the signal weight w_f is always between 0 and 1. In a similar way to the dispersion measure, the $w_{\eta,s}$ shall denote the signal weight for a particular local block, and

$$\mathcal{W}_s = \{w_{\eta,s}^{0.1}\},\ \forall \eta \tag{3.45}$$

is the set of signal weights for scale s. The exponential term 0.1 is introduced to alter the value distribution $\mathcal{W}_s$, because there will be a few weight values in $\mathcal{W}_s$ which are significantly larger than others. Figure 3.7 is a distribution comparison between $\mathcal{W}_s$ and unmodified signal weights, showing effectiveness of such a manipulation.

In order to be consistent and for ease of manipulation, the weights at a given scale are normalised to $[0, 1]$:

$$\overline{\mathcal{W}_s} = \frac{\mathcal{W}_s - \min(\mathcal{W}_s)}{\max(\mathcal{W}_s) - \min(\mathcal{W}_s)}. \tag{3.46}$$

where the $^-$ represents the normalising operation.

Finally, in order to further suppress the noise, the Kalman filter with configuration discussed in the previous subsection will be applied on $\overline{\mathcal{W}_s}$ by letting $Z_s = \overline{\mathcal{W}_s}, \forall s$. The resulting estimated set of *clean* states, denoted as $\overline{\mathcal{W}_s^-}$, is the measure of signal weights to be used.

3.5.3 The Orientation Consistency Measure

The strong inter-scale correlation of linear features suggests that the estimated orientations of the features from different scales should be very close. This is close to the idea that Calway [18] used for checking scale consistency, but in [18] the centroid positions were not considered for the sake of computational simplicity. Also, it is very likely that two neighbouring blocks also have similar orientations if they overlap. Therefore, checking the consistency of a feature orientation with its parent and neighbours is useful in identifying it as a linear feature.

For a certain feature parameter pair $\zeta_{\eta,s}$, with a slight abuse of notation, we denote its 4-neighbour feature set of parameters as

$$\mathcal{N}_{\eta,s} = \{\zeta_{\eta-\Omega_1,s}, \zeta_{\eta-\Omega_2,s}, \zeta_{\eta+\Omega_1,s}, \zeta_{\eta+\Omega_2,s}\}. \tag{3.47}$$

Its parent is denoted as $\zeta_{\lfloor \eta/2 \rfloor, s-1}$. The orientation in the parameter set $\zeta_{\eta,s}$ is going to be denoted as $\theta_{\eta,s}$.

The consistency is defined as

$$\mathcal{K}_s = \left(|\theta_{\eta,s} - \theta_{\lfloor \eta/2 \rfloor, s}| + \frac{1}{8} \sum_{\theta_n \in \mathcal{N}} (|\theta_{\eta,s} - \theta_n|) \right)^2. \tag{3.48}$$

and the normalised version $\overline{\mathcal{K}_s}$ is taken as the orientation consistency measure.

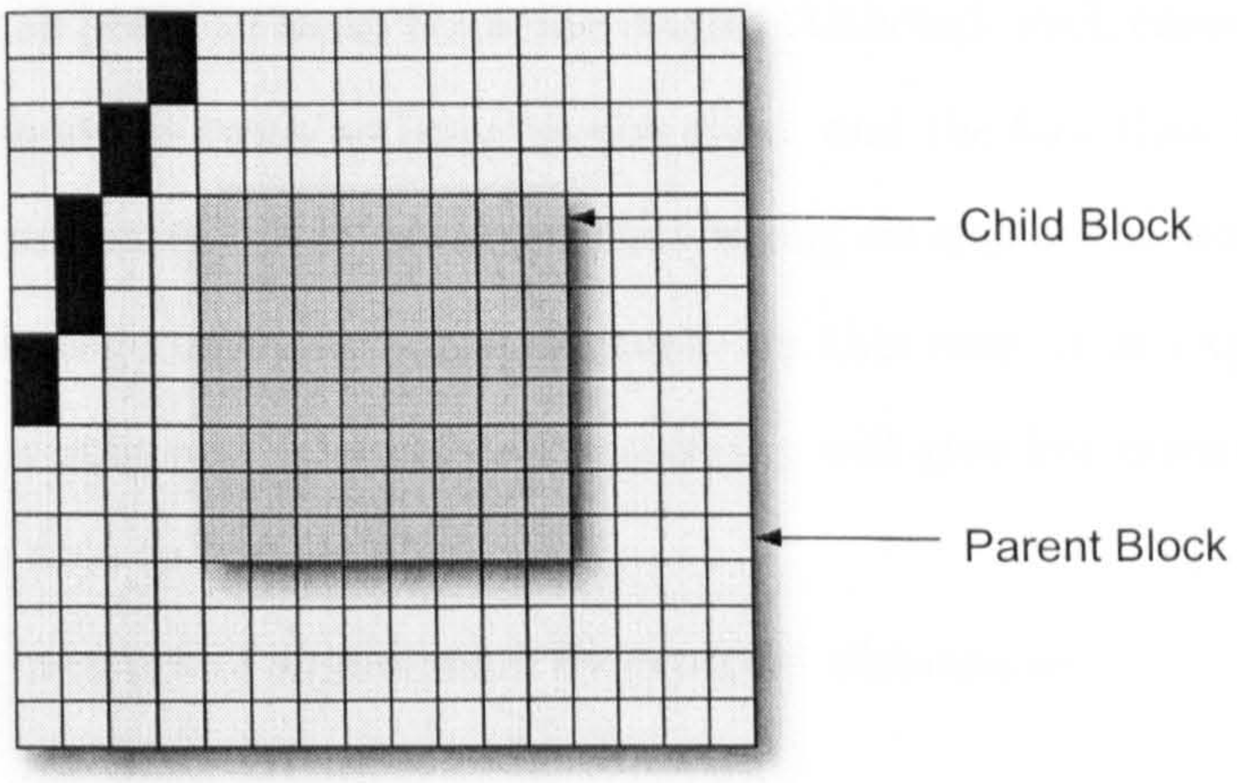


Figure 3.8: A feature passing through the parent window but not the child

3.5.4 The Centroid Distance

Given a centroid position $\boldsymbol{\rho}$, the centroid distance is the length of the vector,

$$\rho = \|\boldsymbol{\rho}\|. \tag{3.49}$$

With the use of overlapping windows, a single feature can appear in several windows. However in each of the windows, its significance can vary. Clearly, if the feature is situated near the origin, then it is more significant than when it resides on the corner of the window. However as previously mentioned, the proposed " Signal+Noise" model is likely to pick up *"phantom features"* in smooth regions due to noise. The centroid position parameter of a phantom feature is also likely to be around the origin, since an empty region's Fourier spectrum does not have a linear phase. It was found that applying the Kalman filter can help to distinguish phantom features.

Figure 3.8 shows a scenario when a line goes through the parent window but does not intersect with the child. In a noisy case, although the child window's estimation is incorrect, the Kalman filter will use the parent's refined estimate to *"correct"* the child's

estimate, pushing the position away from the origin. Although such cases seem to be rare, since the parent window is twice as large as the child, and the fact that the parent's state is refined by the grandparent, it is expected that wrong estimates can only occur when all the blocks in the lineage cover no linear feature. In this case, it is expected that other criteria like signal weight and orientation consistency will give low measurements and the final voting will be able to correct this.

We denote the measure of significance by centroid distance as:

$$\mathcal{D}_s = \{-\rho^{-}_{\eta,s}\},\ \forall \eta. \tag{3.50}$$

Using negative ρ makes it consistent, so that lower measurement corresponds to less significance. We also normalise the measure to $\overline{\mathcal{D}_s}$.

3.5.5 Classification by Voting

In order to determine the classification of a parameter estimate $\zeta_{\eta,s}$, the significance measures need to be combined. For classification, the answer is either "yes" or "no". Precursors like Calway [18], Davies [45] and Meulemans [101] all used a thresholding approach on a single criterion. With multiple measures, we use a voting strategy instead:

- For each scale s, the set

$$\mathcal{C}_s = \{\mathcal{A}_s, \overline{\mathcal{W}}_s, \overline{\mathcal{K}}_s, \overline{\mathcal{D}}_s\} \tag{3.51}$$

  is the measurement set for the image.

- For each $\mathcal{M}_{\eta,s} \in \mathcal{M}_s, \mathcal{M}_s \in \mathcal{C}_s$, the label $\mathcal{M}'_s$ is determined as:

$$\mathcal{M}'_{\eta,s} = \begin{cases} 1 & \text{if } \mathcal{M}_{\eta,s} > MEAN(\mathcal{M}_s); \\ 0 & \text{if otherwise.} \end{cases} \tag{3.52}$$

and correspondingly, C_s' is the set which contains all $\{\mathcal{M}_s'\}$.

- Each of the $\mathcal{M}_s'$ can be seen as a class configuration for a single criterion. Then

$$\mathcal{M}_s^+ = \sum_{\mathcal{M}_s' \in C_s'} \mathcal{M}_s'. \tag{3.53}$$

The final class label for a feature $\zeta_{\eta,s}$ is determined by

$$\mathcal{L}_{\eta,s} = \begin{cases} 1 & \text{if } \mathcal{M}_{\eta,s}^+ \geq \mathcal{T}; \\ 0 & \text{if otherwise.} \end{cases} \tag{3.54}$$

where $\mathcal{T}$ is an integer voting threshold from 0 to 4, and $\mathcal{L}_{\eta,s} = 1$ indicates a significant feature presence, while $\mathcal{L}_{\eta,s} = 0$ means the position is a non-feature block.

Different choices of $\mathcal{T}$ can lead to different voting results. For example, when $\mathcal{T} = 0$, the final classification will contain all features from every measurement, while the case $\mathcal{T} = 4$ is most strict, a feature can appear only if all measurements, vote for its significance and a veto will exclude it.

3.6 The Estimator Implementation

The MFT provides local spectrum estimates over the full range of scales. For a given discrete image $x(\eta)$, the MFT of this image is given by

$$\hat{x}_{\eta,s}(\omega) = \frac{1}{s} \sum_{\xi} x(\xi) w\left(\frac{\xi - \eta}{s}\right) \exp[-j\xi \cdot \eta]. \tag{3.55}$$

The discrete definition of the transform is still less specific in terms of implementation, for instance the variable s is continuous and the window function is not defined. In this

section, these implementation issues for the MFT used in linear feature extraction are considered (cf. [18]).

3.6.1 The Level Operator

The MFT at a fixed scale is referred to as a *level* of the MFT. The sampling of scales in the discrete case is usually dyadic, which means that two adjacent levels differ by a factor of 2. Therefore, a discrete-scale MFT can be seen as a stack of windowed Fourier transforms at different scales. The digital implementation of the MFT here uses the Laplacian Pyramid (LP) [17] as the *level operator*. For a particular level x_s, it is computed as

$$x_s(\eta) = (\mathbf{I} - \mathbf{G}_{s,s+1}\mathbf{G}_{s+1,s})x_s'(\eta). \tag{3.56}$$

where $\mathbf{I}$ is the identity operator and x_s' is the Gaussian lowpass pyramid representation of $x(\eta)$

$$x_s'(\eta) = \sum_{l=0}^{s-1} \mathbf{G}_{l+1,l}x(\eta). \tag{3.57}$$

and $\mathbf{G}_{s,s+1}$, $\mathbf{G}_{s+1,s}$ are the raising and lowering operators associated with transitions between levels in the Gaussian pyramid [17].

3.6.2 Choice of the Window Function

The purpose of the window function $w(\xi)$ is to limit the analysis on a locally confined region as well as to suppress the boundary effect of FFT on a rectangular region. The cosine window, shown in Figure 3.9, has both good spatial localisation and frequency localisation, as described in [112]. Another useful property of this window function is that with 50% overlapping windows, when squared, they add up to 1. This is particularly useful for the spatial signal synthesis.

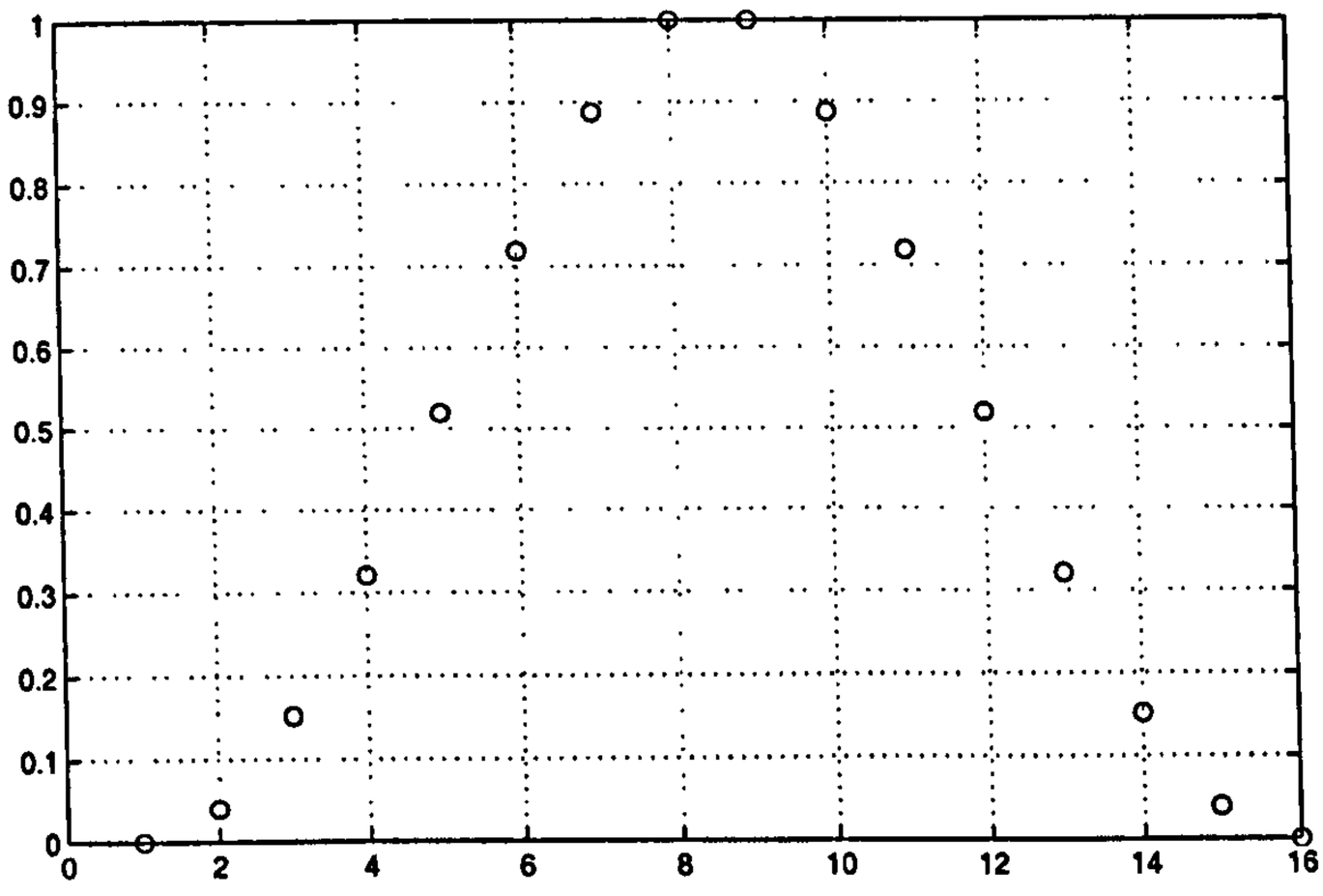


Figure 3.9: A 16-point cosine squared window function

The cosine squared window function in 2D is defined as :

$$w^M(\boldsymbol{\xi}) = \cos^2\left[\frac{\pi}{4M}\xi_1\right]\cos^2\left[\frac{\pi}{4M}\xi_2\right]. \tag{3.58}$$

where M is the windowing size. In the actual implementation, $M = 16$ and each window is 50% overlapped.

The complete process of the feature extraction algorithm is presented in Table 3.1 in the form of pseudo-code.

3.7 Experimental Results

3.7.1 Test Data

Four images were used in this experiment for linear feature extraction. The `barcelona` image, which contains an object with well defined contours and a smooth homogenous

Algorithm 3.6.1: NOISYLINEEXTRACTION(X, J)

comment: X is the noisy image and J is the total levels of decomposition

$pyr \leftarrow$ LAPLACIANPYRAMID(X, J)
for $s \leftarrow 1$ **to** $J - 1$
  do $[\theta_s, \rho_s, \mathcal{W}_s, \mathcal{A}_s] \leftarrow$ BLOCKPARAMETERESTIMATE(pyr_s)

$\mathcal{K} \leftarrow$ SCALECONSISTENCY(θ, J)
$\overline{\mathcal{K}} \leftarrow$ KALMANFILTER$(\mathcal{K}, J)$
$\overline{\theta} \leftarrow$ KALMANFILTER(θ, J)
$\overline{\rho} \leftarrow$ KALMANFILTER(ρ, J)
$\overline{\mathcal{W}} \leftarrow$ KALMANFILTER$(\mathcal{W}, J)$

$\mathcal{L} \leftarrow$ CLASSIFICATIONVOTE$(\mathcal{A}, \overline{\mathcal{K}}, \overline{\mathcal{W}}, -\overline{\rho})$
for $s \leftarrow 1$ **to** $J - 1$
  do $map_s \leftarrow$ DRAWFEATUREMAP$(\mathcal{L}_s, \overline{\theta_s}, \overline{\rho_s}, s)$

Table 3.1: The pseudo-code for the noisy linear feature extraction algorithm

background; the `lena` image, which has sophisticated linear features as well as curved contours; the `pepper` image is a composition of several similar objects with highlights and shades; and finally `cameraman`, which includes contours with both sharp and vague contrast. All the images are at 512×512 pixels and can be seen in Figure 3.10.

3.7.2 Candidates for Comparison

In the experiments, the proposed method is compared with three other candidates.

1. **The Geometric Hidden Markov Tree (GHMT) Model**

   The Geometric Hidden Markov Tree (GHMT) model [135] for linear feature extraction is built upon the complex wavelet transform. The GHMT method estimates the linear feature parameters by matching edge responses from pre-calculated prototype edge images with different orientations and the offset distance by the phase response in the complex wavelet domain. The classification of features is done by a multiscale

Viterbi algorithm [131].

2. **The Custom-Built Moments**

The second candidate [127] uses custom-built moments (CBM) to parameterise step edges in terms of a line: $ax + by = c$. Such an approach does not require a linear transform and is naturally noise-resistant. Like the proposed method, the CBM algorithm is block-based and configured with the same 16×16 local windows with 50% overlap.

3. **The Canny Edge Detector**

The classical Canny edge detector results are also included in the presentation. Although parametric linear feature extraction shares certain properties with the traditional edge detection, fundamentally these two approaches are completely different. While the Canny detector shows good resistance against noise, this can be attributed to the effects of preprocessing like smoothing and the gradient computation. Also, since it is usually followed by some parameter estimator, such as the method presented here, the whole process of edge detection can be seen as a preprocessing stage for linear feature extraction. Despite its disadvantages, as mentioned in section 3.1, such **"edge detection + parameter estimation"** is less suitable in some applications, where the feature extraction and other tasks are performed simultaneously in some transform domain, such as feature-based image compression or denoising (which will be discussed in Chapter 5). Therefore the Canny detector does not represent a suitable comparison, but is included for reader's interest.

3.7.3 A Step-by-Step Example

The process of the proposed algorithm is better illustrated by a presentation where the results are shown in a step-by-step manner. The example uses the 10dB `barcelona` image

(Figure 3.12). For a comparison, the inertia-tensor based feature extraction results in very poor estimates and the feature selection does not function at all in the presence of noise, as can be seen in Figure 3.13.

Figure 3.14 is the feature map generated by the proposed "Signal+Noise" model for the Fourier magnitude as in equation (3.16). In order to improve the computation for speed, the parameters for the model are computed by a fixed 7 times of iteration of the EM algorithm instead of checking the convergence condition at the end of each iteration. It is also found that the algorithm converges or is close to convergence within 7 iterations. We can see that, on the smooth regions, the extracted features are lines with random orientation due to the noise, but the structural contours are identifiable in the feature map.

Next, the estimates are refined by the Kalman filter performed across scales in a top-down fashion. The refined feature map $\hat{\zeta}_{\eta,4}$ in Figure 3.15 is greatly improved, while the contour features are a better fit, many of the smooth region features disappear due to the refined centroids being taken out of the scope of the local windows.

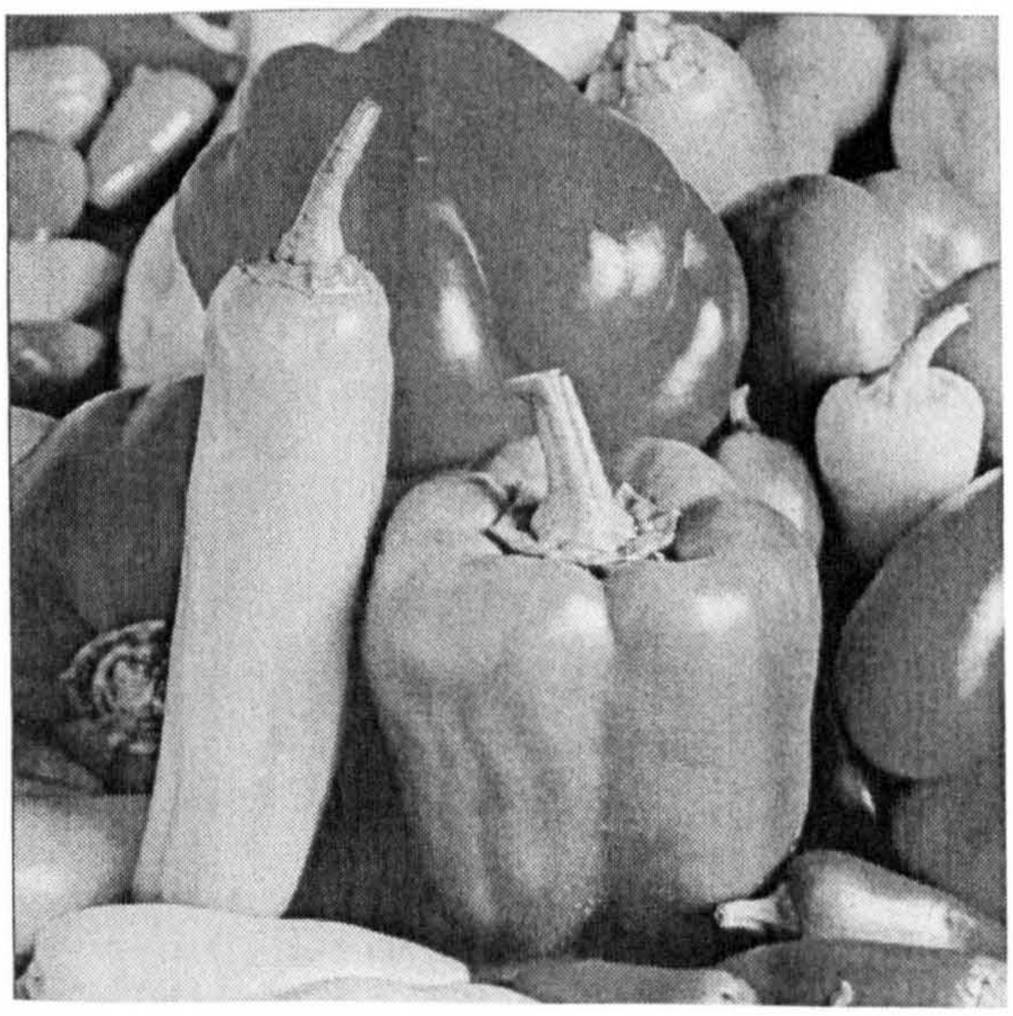
The refined linear features are classified by a 3-out-of-4 voting from the measures described in Section 3.5. This high majority vote proved to be reasonable in our experiments. In Figure 3.16, we show the significant linear features from three different resolutions, which stands as the final output of the algorithm

3.7.4 Visual Comparison

The test images were corrupted by two different levels of noise, 10dB and 15dB in SNR respectively. The head-to-head comparative results are presented in a series of figures (Figure 3.17-3.24). From the images, the resulting feature maps from GHMT exhibit heavy broken-contour artifact at 10dB. Although the algorithm does take advantage of multiscale geometric coherence, the Viterbi algorithm does not refine the estimated parameters as the

barcelona

lena



pepper

cameraman

Figure 3.10: Four test images for linear feature extraction

Figure 3.11: Features extracted from clean images by the MFT inertia tensor method

Figure 3.12: The noisy `barcelona` image, SNR=10dB

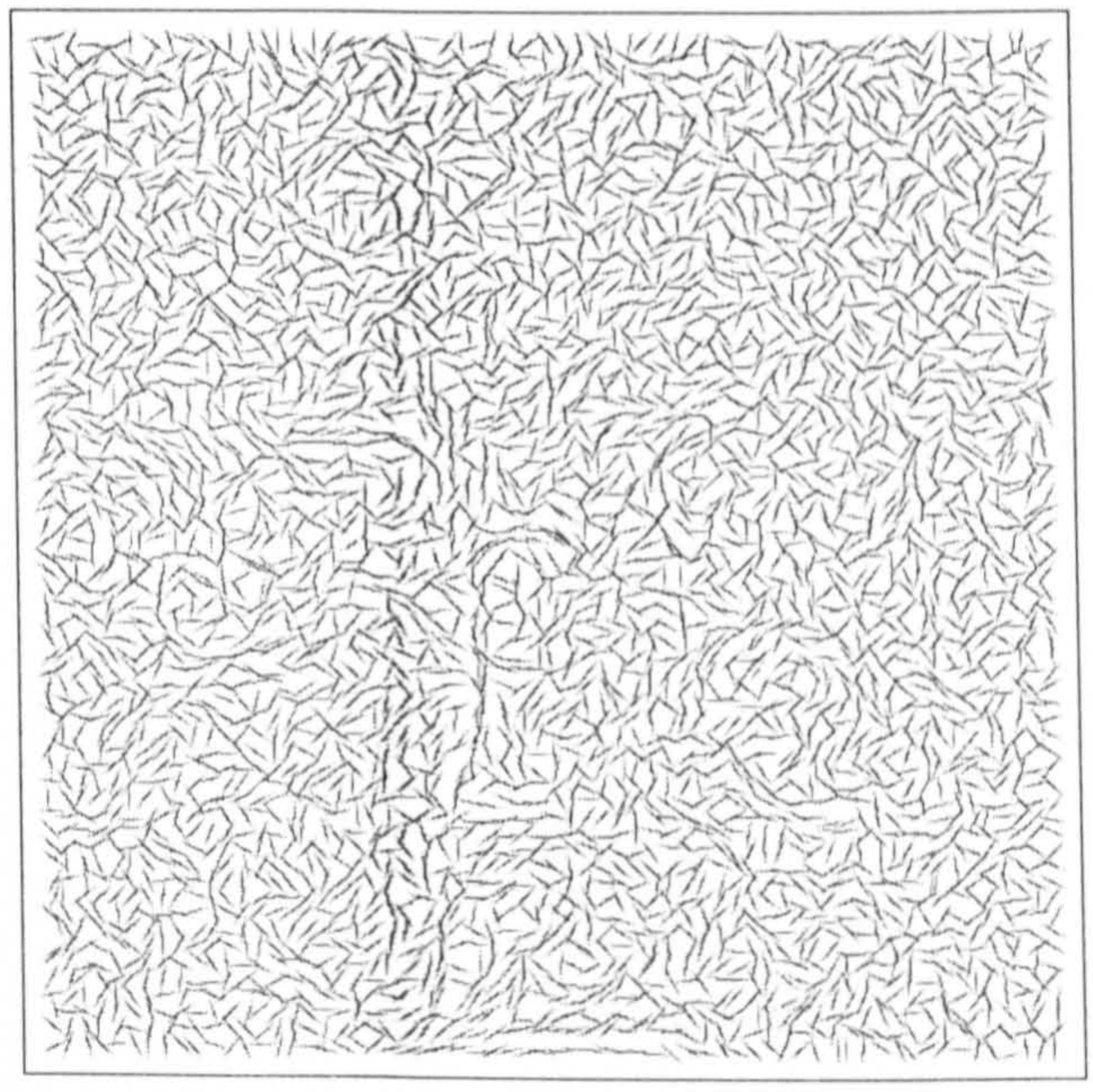


Figure 3.13: The $\zeta_{\boldsymbol{\eta},4}$ estimate using inertia tensor

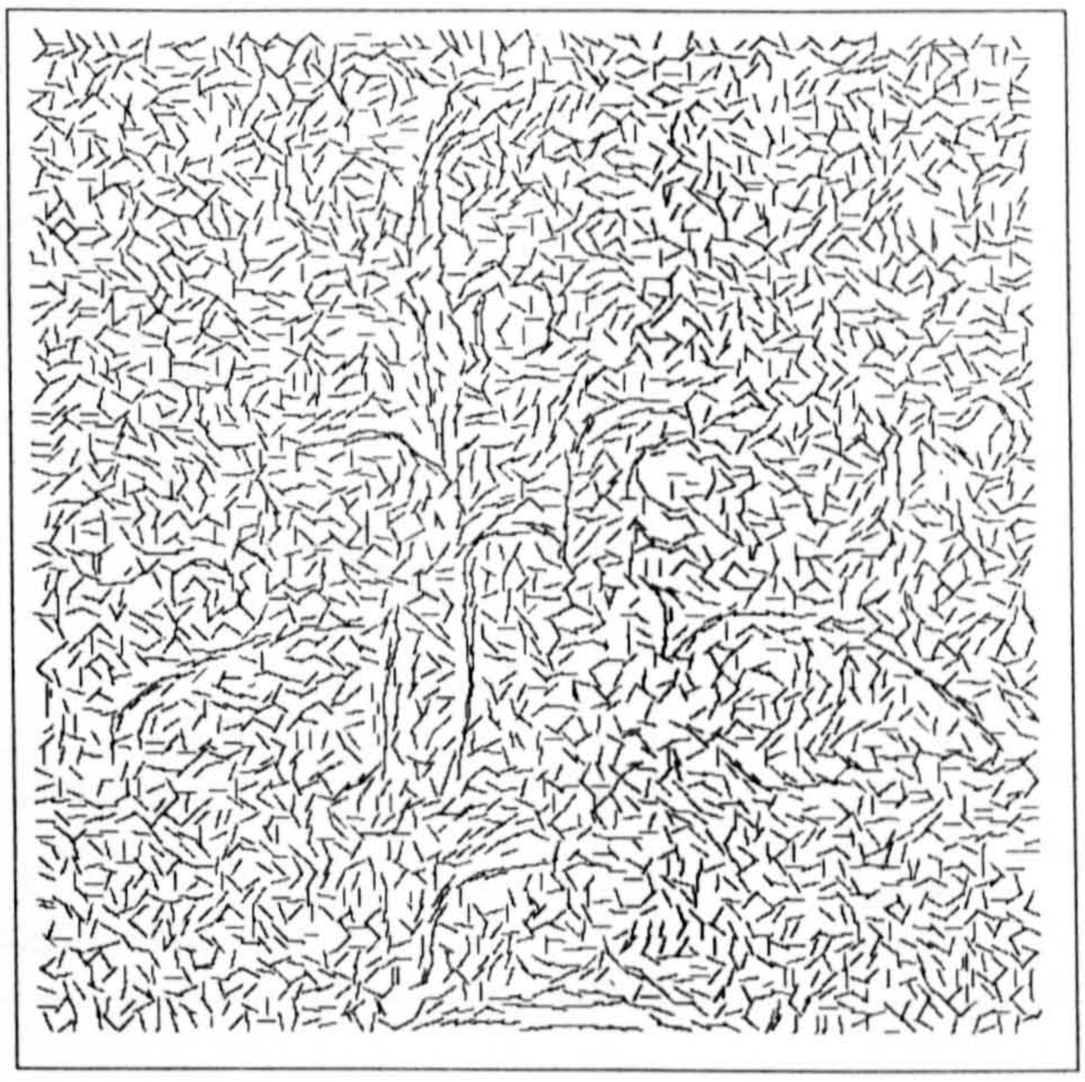


Figure 3.14: The $\zeta_{\eta,4}$ estimate using the proposed signal-noise model

Figure 3.15: The refined $\hat{\zeta}_{\eta,4}$ estimate using Kalman filter

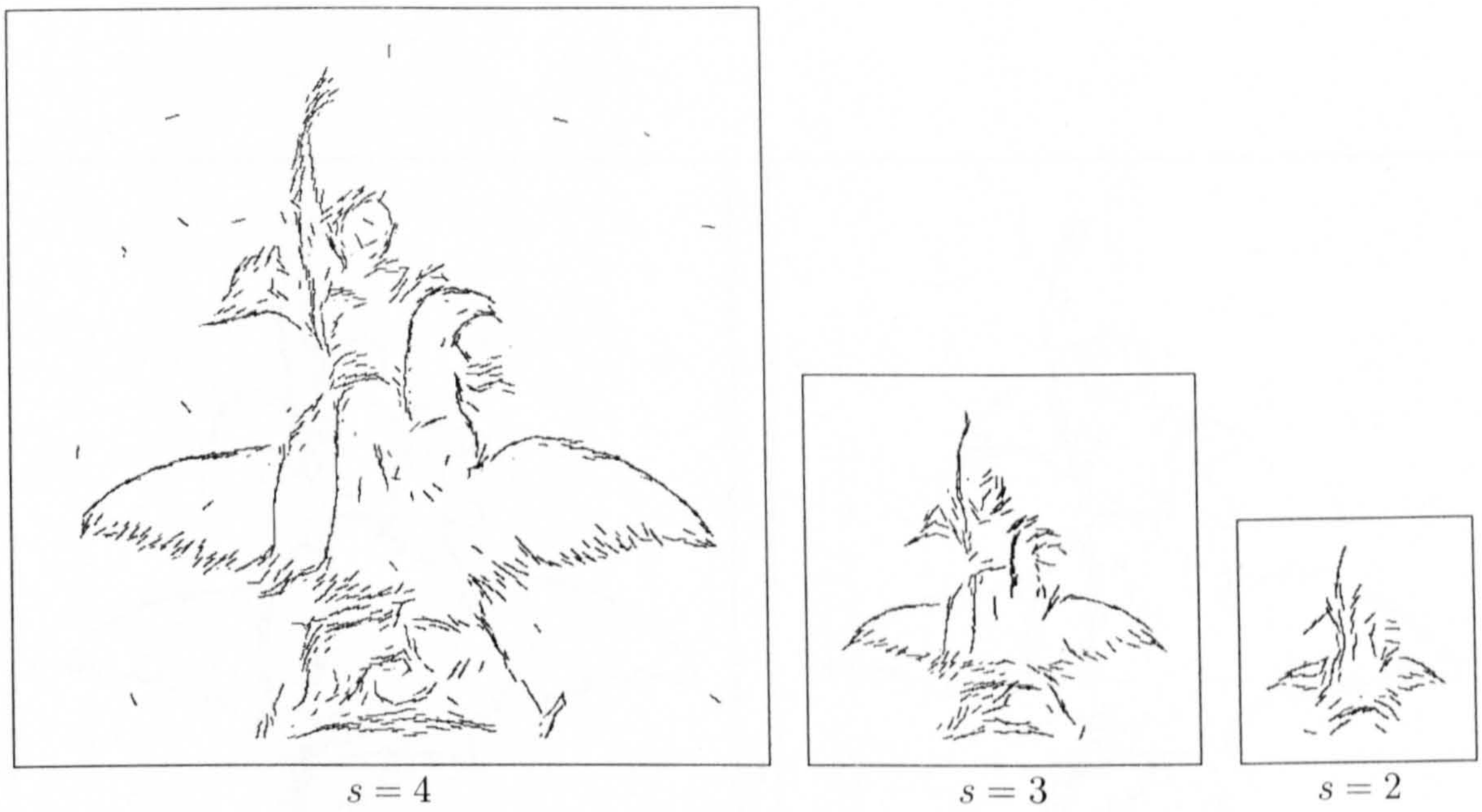
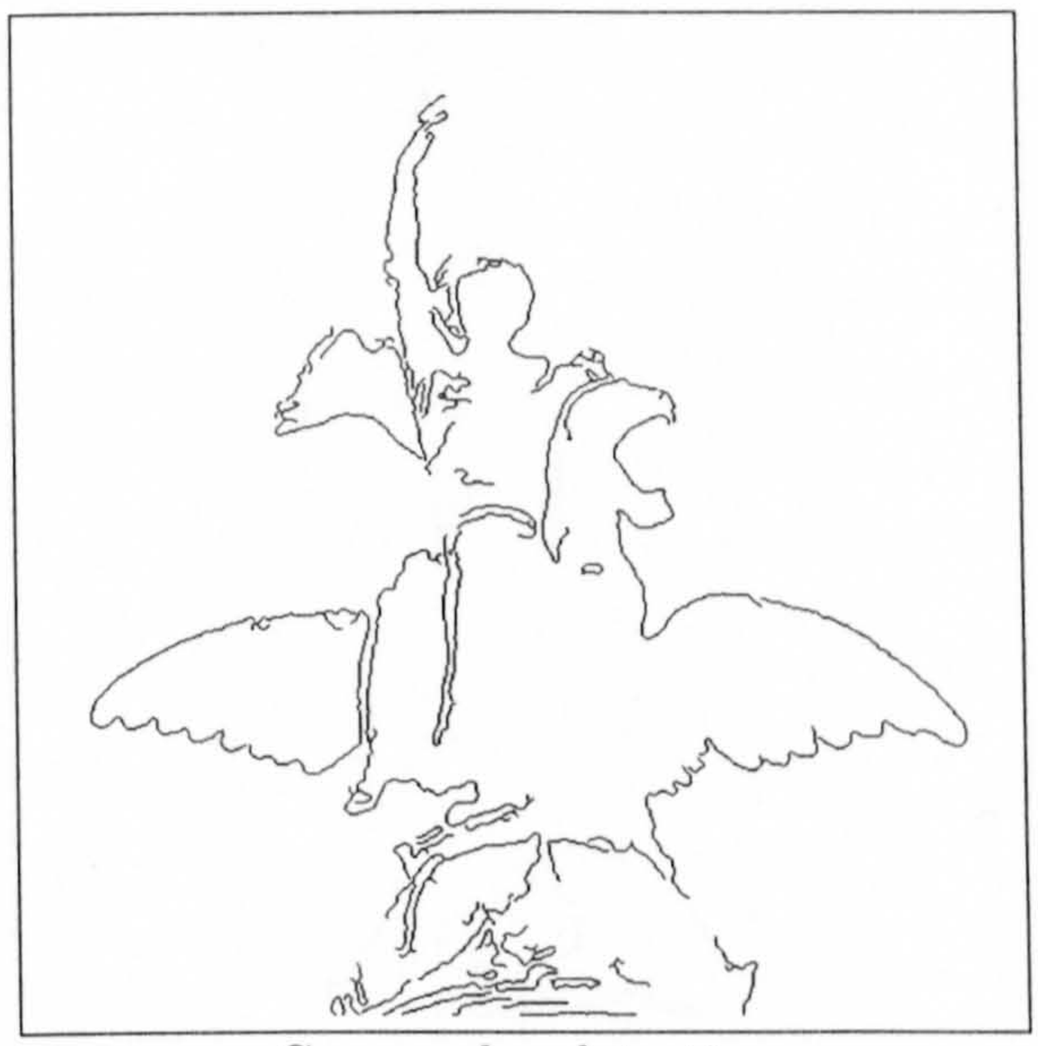


Figure 3.16: The refined $\hat{\zeta}_{\eta,s}$ for scales 4-2 after removing features classified as non-contour

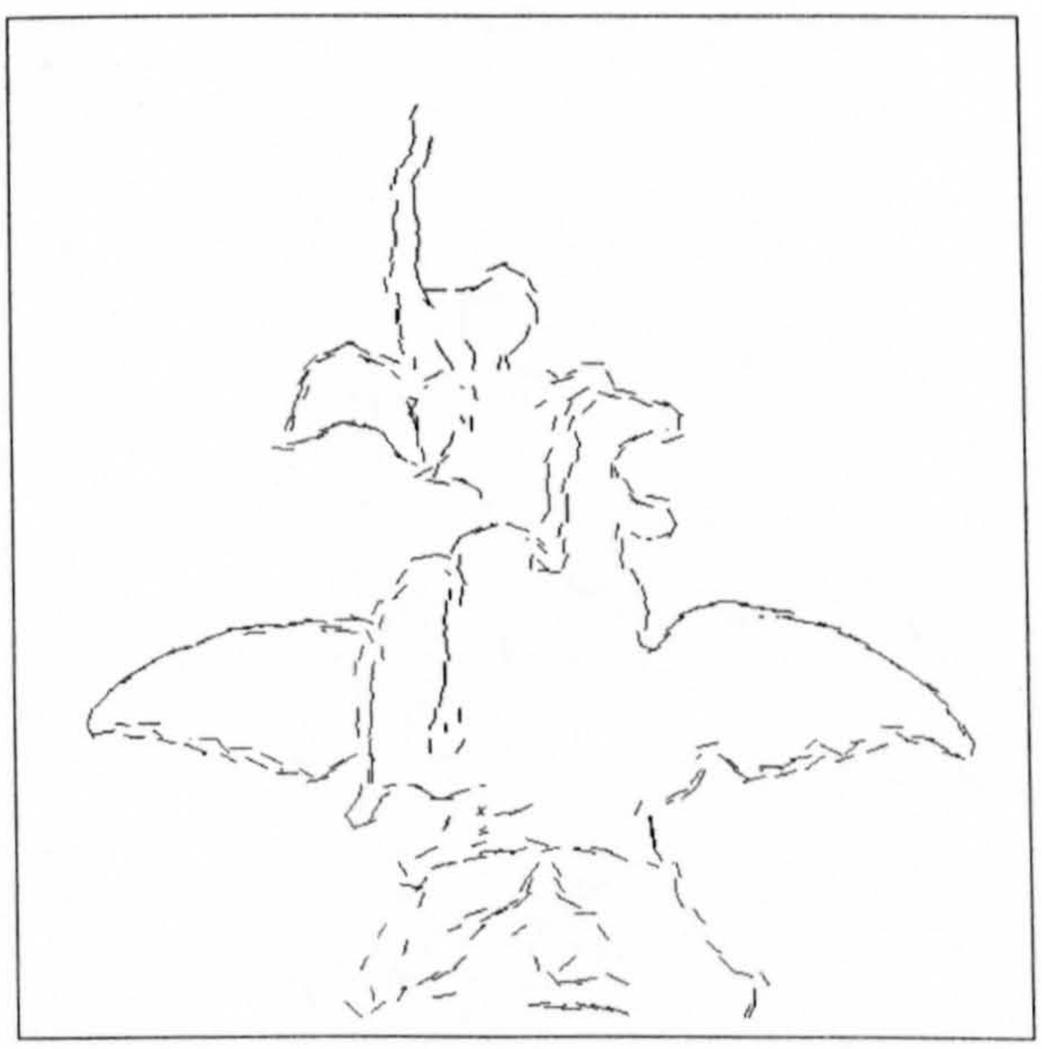
Kalman filter does. The estimated linear features, therefore, show the influence of the noise and do not fit the original signal well. In this particular implementation, the distortion tolerance threshold in the Viterbi algorithm must be given manually instead of estimated, and an appropriate threshold seems to be image-dependent from our experiments. To obtain a decent feature map, one possibly needs to try several values for this threshold.

The claim of noise-resistance of the CBM method seems to hold true in the experiments. However, the method extracts fewer features than other candidates, although prominent edges are sketched well and there are no *"noisy features"* included in the feature maps. It is noted that there can be a mismatch where the curvature is high such as on the eagle's head in `barcelona` (Figure 3.17). It is observed that the CBM is better at extracting some blurred linear edges on `lena`'s background than others. This is due to the fact that both GHMT and the proposed method are built upon subband transforms, where the blurred feature does not appear in the high frequency subband. The proposed method overcomes



Canny edge detection

The GHMT estimate



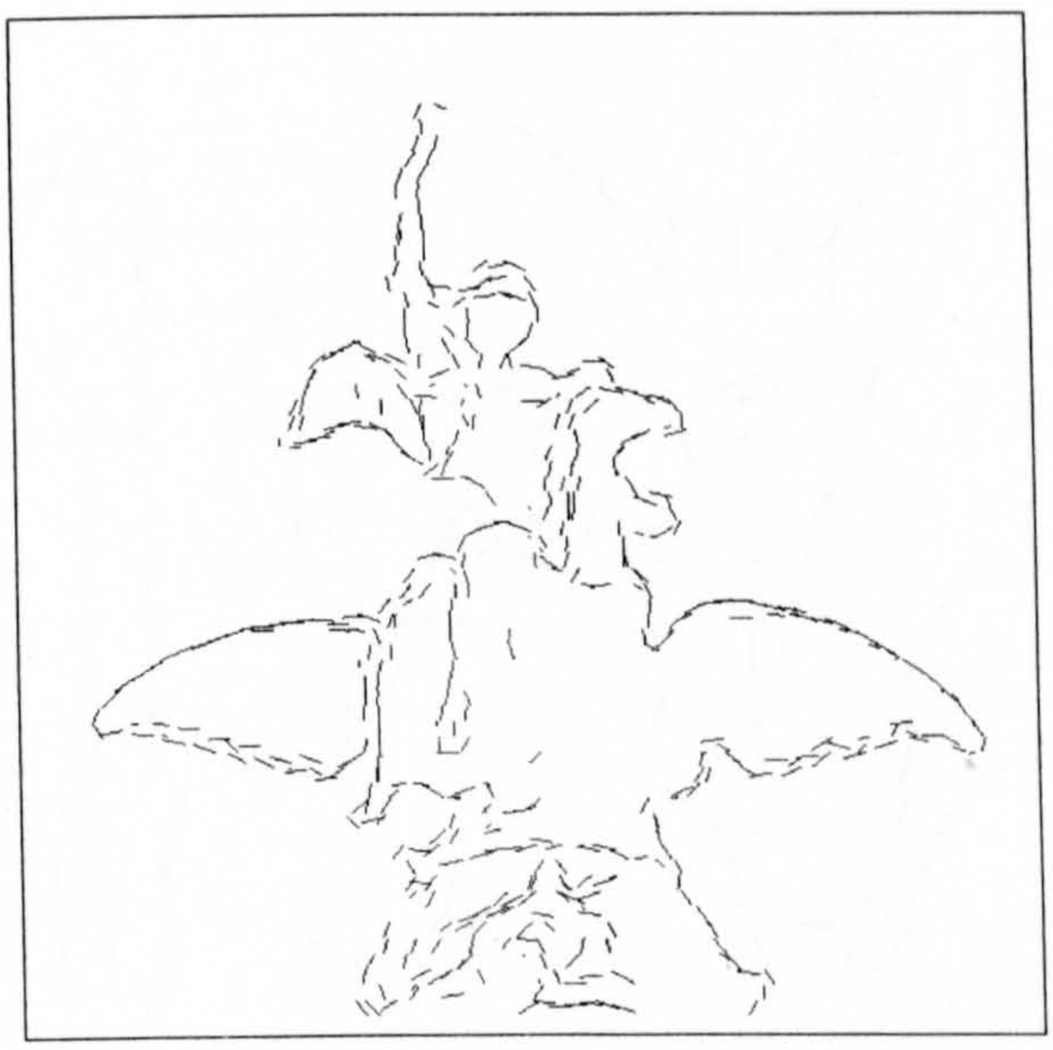
The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.17: Comparative results for 10dB noisy `barcelona`

Canny edge detection

The GHMT estimate

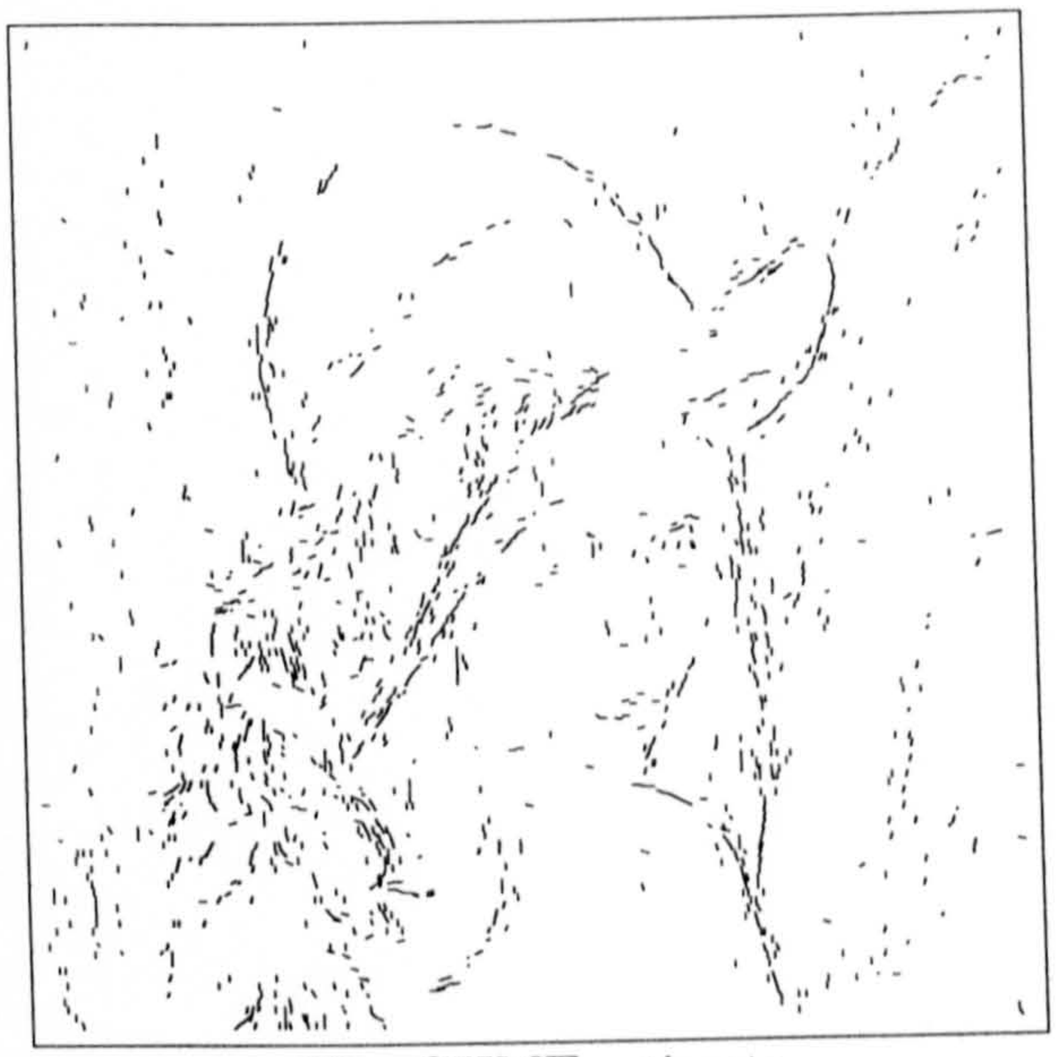


The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.18: Comparative results for 15dB noisy `barcelona`

Canny edge detection



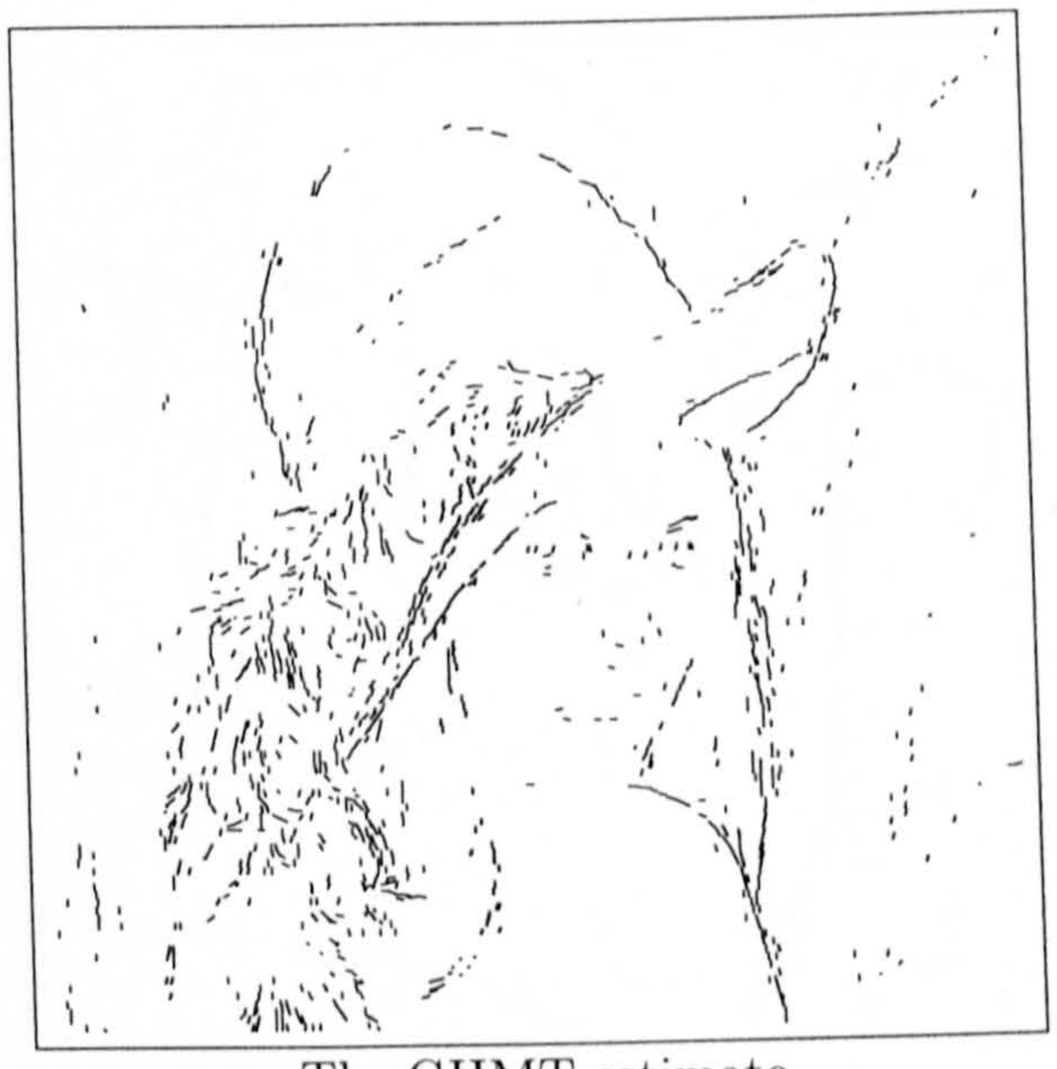
The GHMT estimate

The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.19: Comparative results for 10dB noisy `lena`

Canny edge detection



The GHMT estimate

The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.20: Comparative results for 15dB noisy `lena`

Canny edge detection

The GHMT estimate

The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.21: Comparative results for 10dB noisy pepper

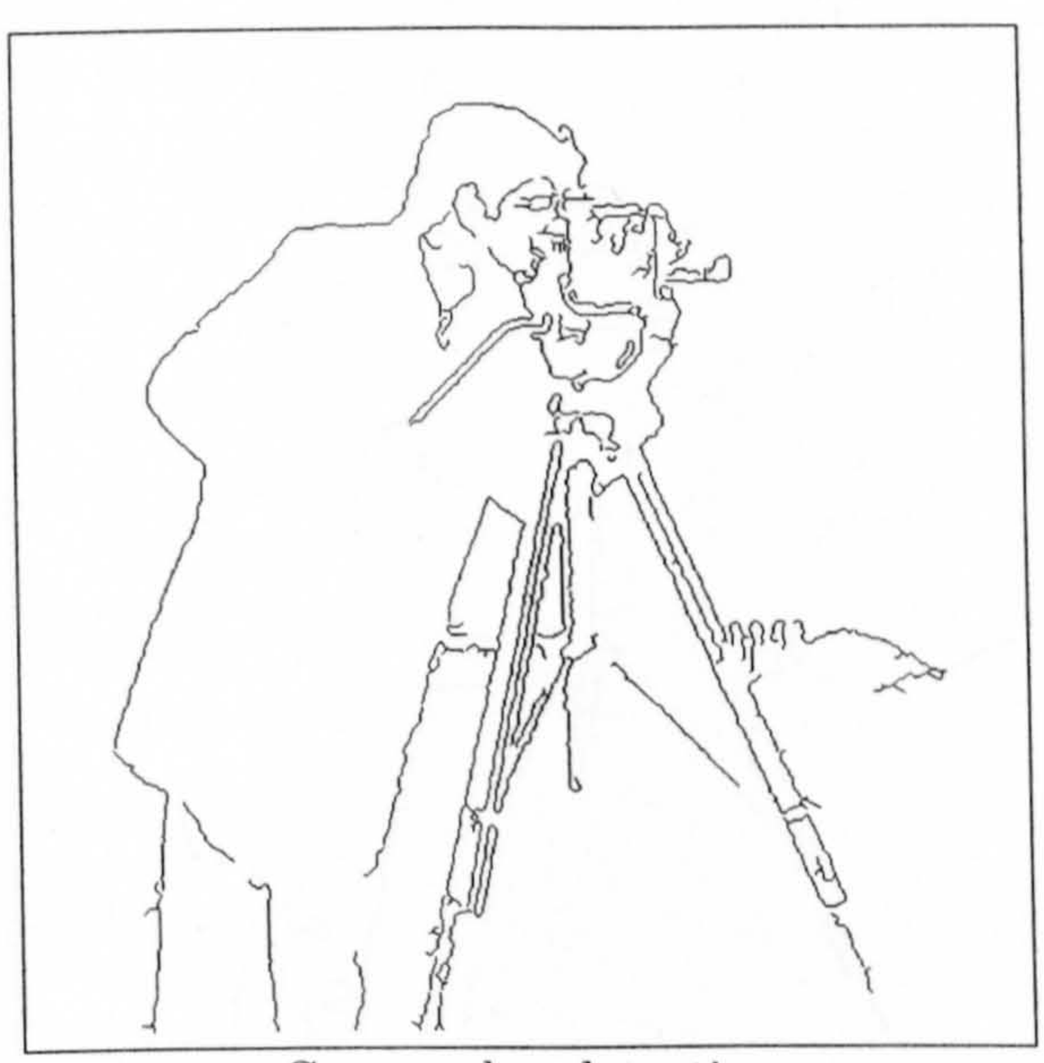
Canny edge detection

The GHMT estimate

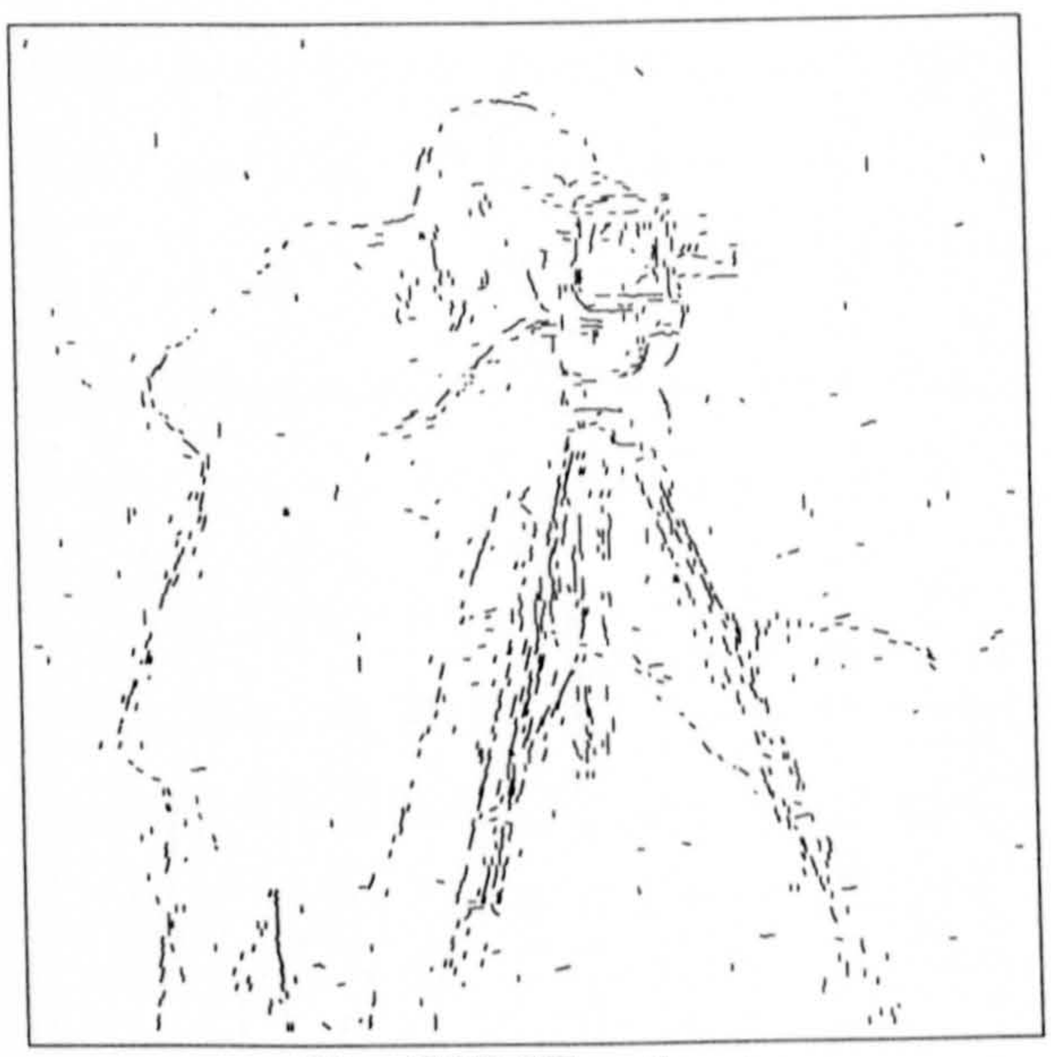
The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.22: Comparative results for 15dB noisy pepper



Canny edge detection

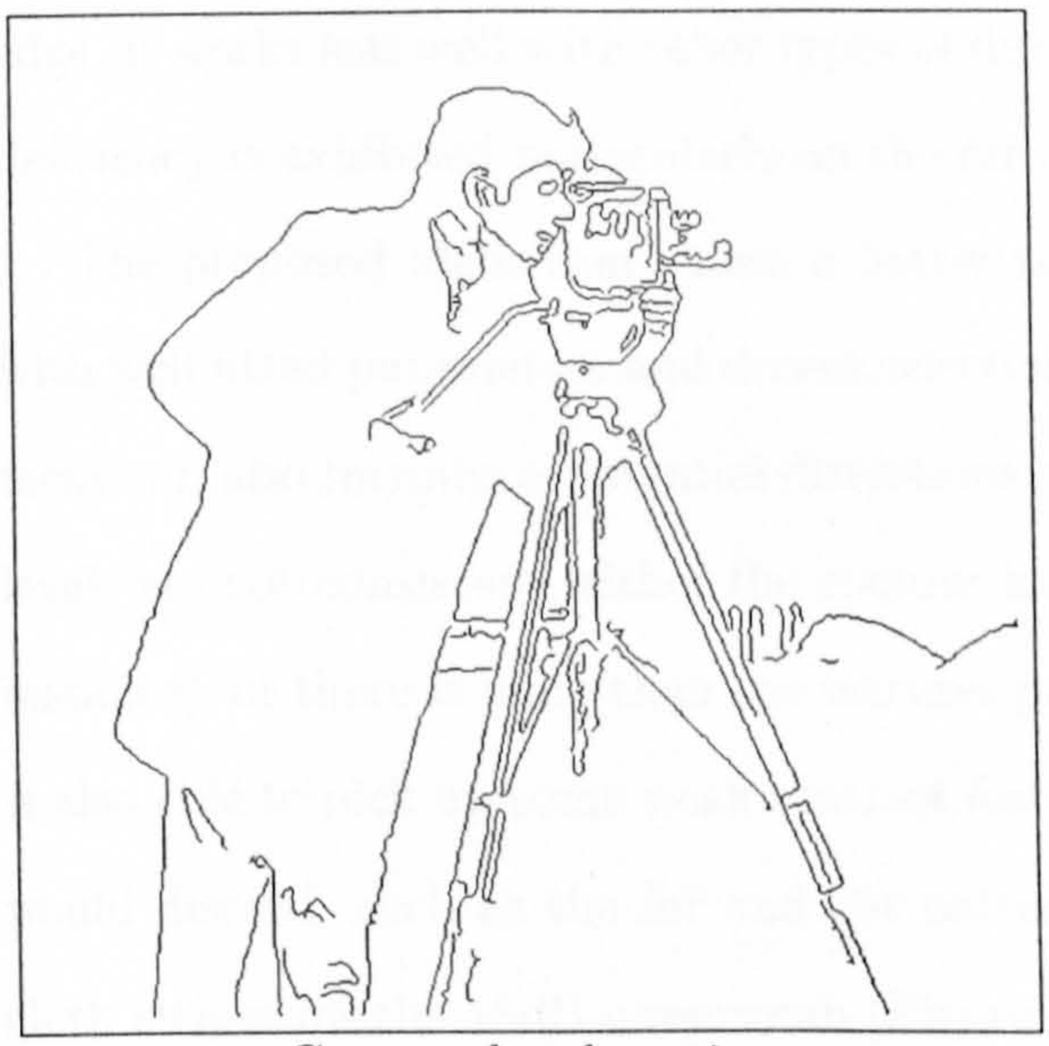


The GHMT estimate

The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.23: Comparative results for 10dB noisy `cameraman`



Canny edge detection

The GHMT estimate

The CBM estimate

Proposed $\hat{\zeta}_{\eta,4}$

Figure 3.24: Comparative results for 15dB noisy `cameraman`

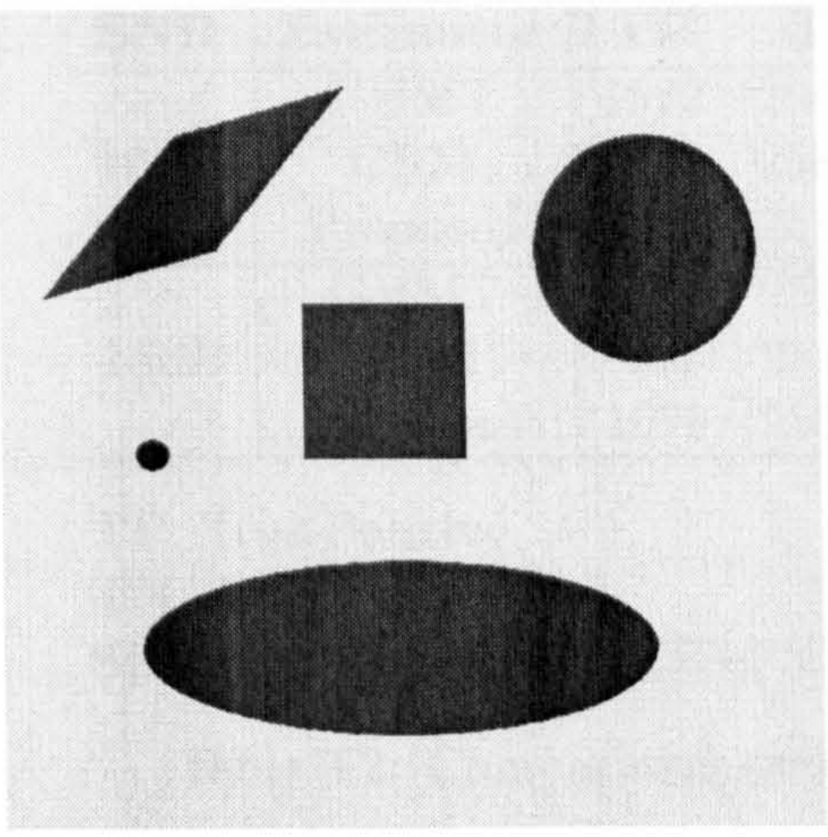
this limitation by multiscale Kalman filtering. CBM, on the other hand, operates on the original signal directly. However, since the moment model is based on a perfect step linear edge, it works less well with other types of directional patterns such as a simple line. This deficiency is exhibited particularly on the fur on the `lena`'s hat (Figure 3.19).

The proposed algorithm shows a better result than both the GHMT and the CBM, with well fitted parameters and decent selection of the linear features. However, the refinement can also introduce potential distortions on certain regions where estimates from two levels are not consistent, either the contour has very high curvature (the eyes of `lena`, for instance), or there is more than one features present in the window. The proposed method is also able to pick up some weak contrast features which other methods, including Canny, would discard, such as the fur and hat pattern on `lena` (Figure 3.19) and the hand and cloth stripes on the 15dB `cameraman` (Figure 3.24).

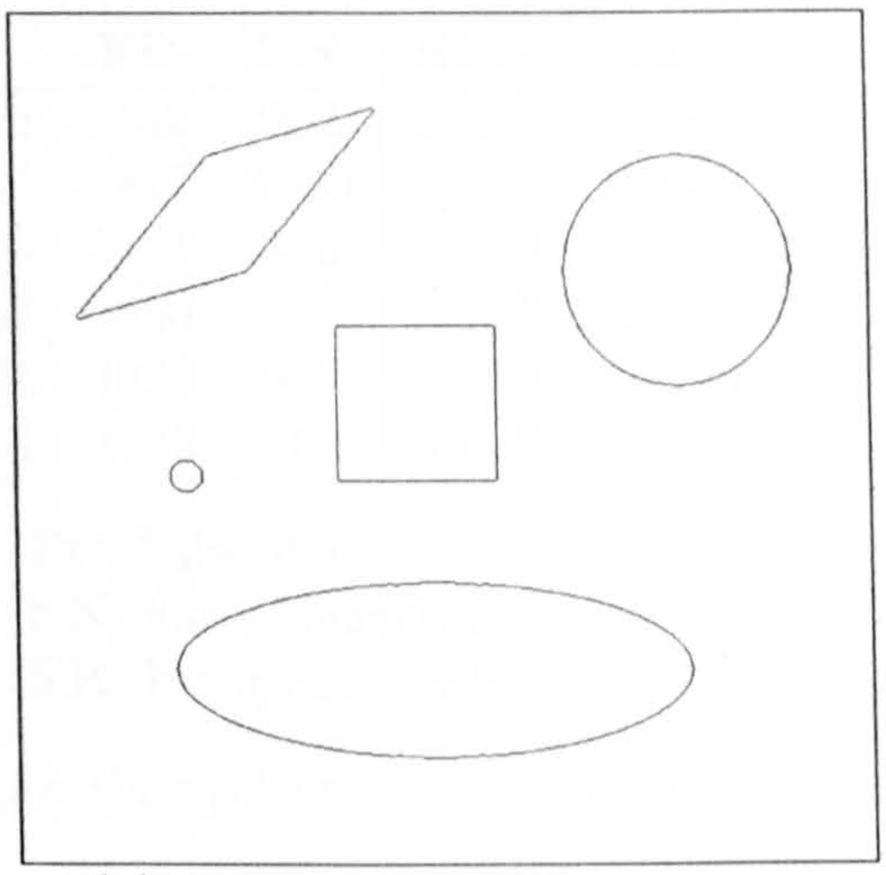
3.7.5 Error Comparison

In order to quantify the performance of different estimators, we also conducted experiments on a simple 512×512 synthetic image with well-defined geometric shapes (Figure 3.25(a)). The image consists of binary gray levels and the geometric edges are detected pixel-wise in Figure 3.25(b) as the ground truth edge map. For a 10dB noisy version of the synthetic image (Figure 3.25(c)), the results generated by GHMT, CBM and the proposed method are shown respectively in Figure 3.25(d-f).

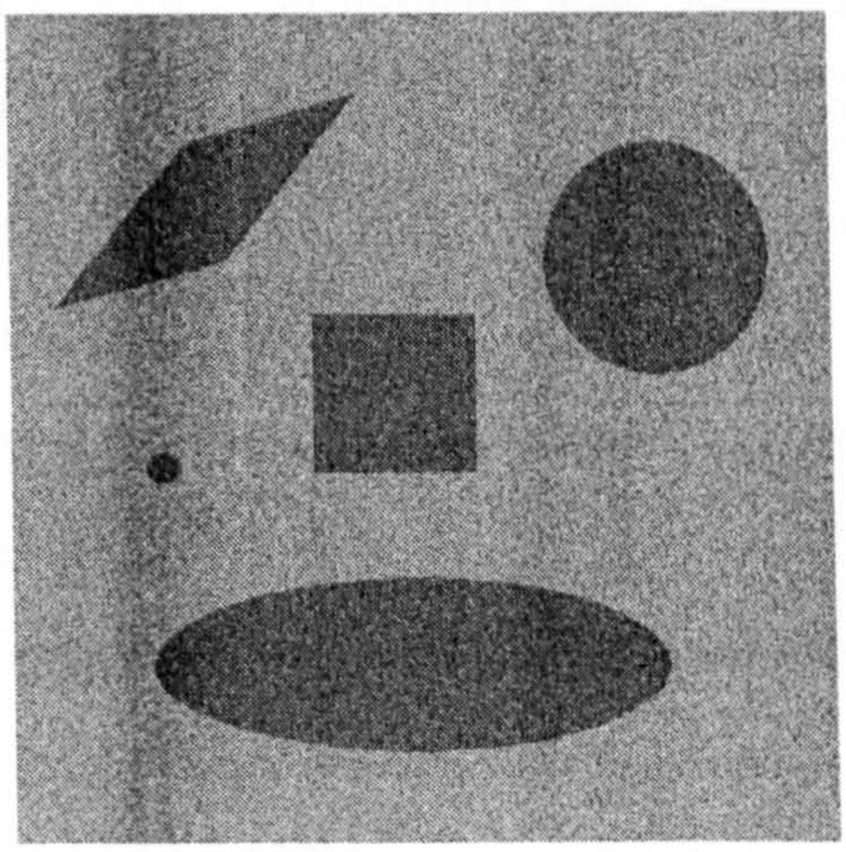
The estimated parametric features are rasterised into the original image's size before comparing with the ground truth data. Several statistics were used to measure the performance: *true positive*, which is the number of pixels that are labeled as edges by both the estimator and the ground truth; *true negative*, which is the number of pixels that are labeled as non-edges by both the estimators and the ground truth; *false positive* is the number of pixels where the ground truth image marked as non-edges but rejected by the



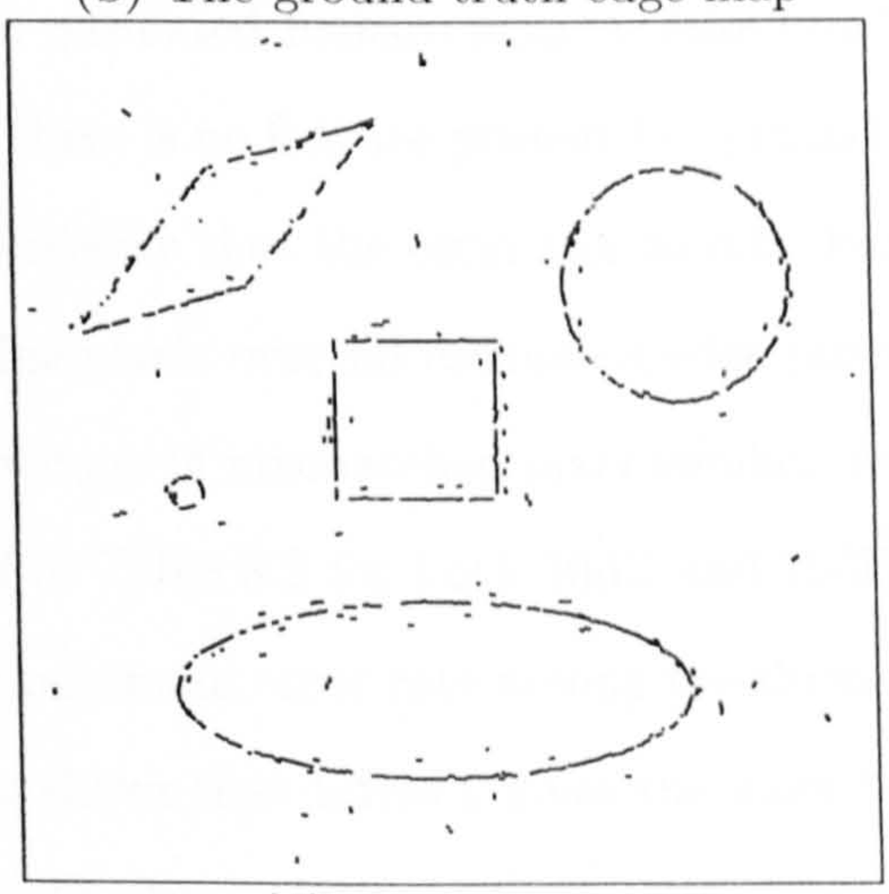
(a) The original image



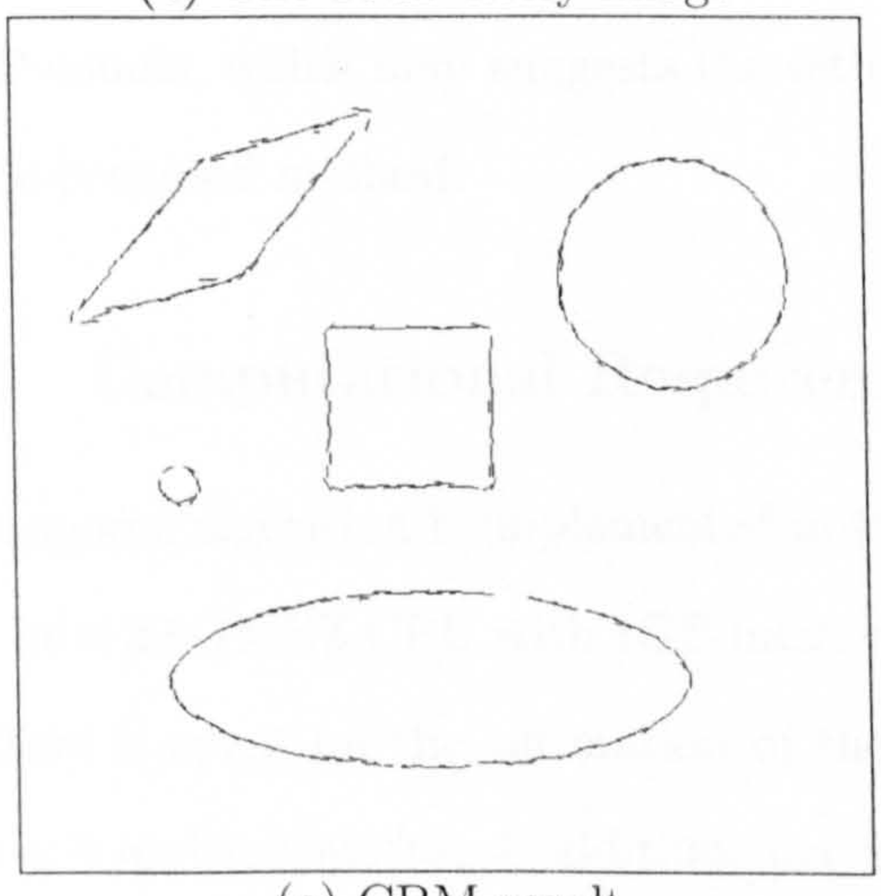
(b) The ground truth edge map



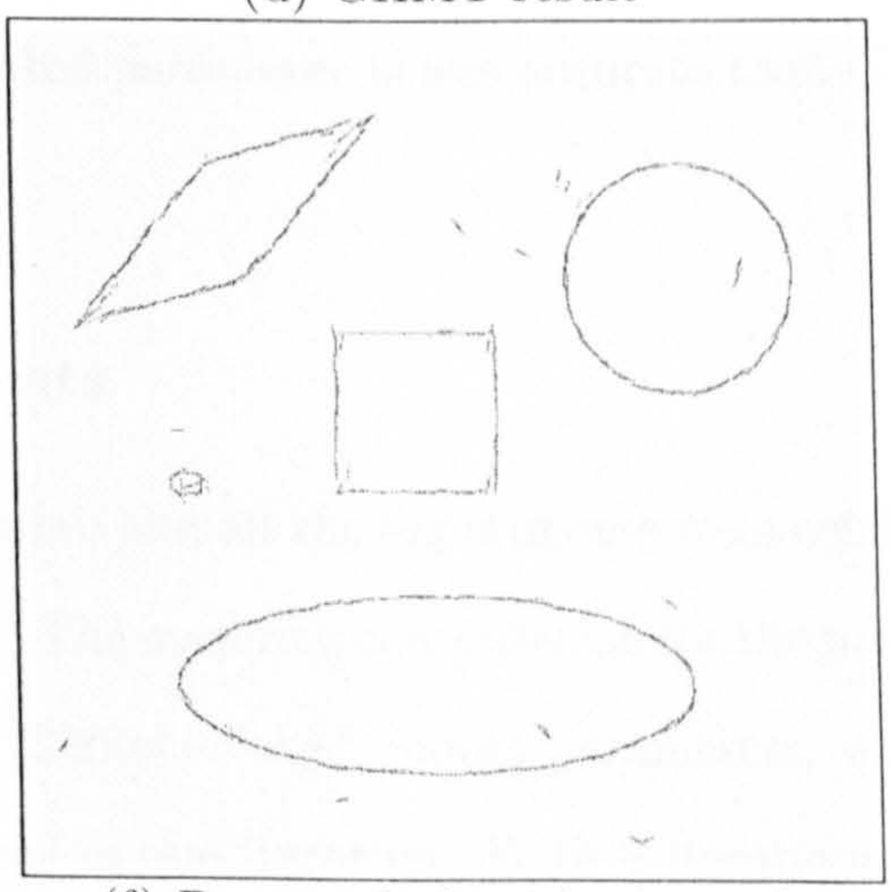
(c) The 10dB noisy image



(d) GHMT result



(e) CBM result



(f) Proposed algorithm's result

Figure 3.25: Extracted features on the synthetic shape image

SNR	Estimator	TP	TN	FP	FN	SEN	ER
10dB	GHMT	1572	254466	5152	952	23.3%	1.40%
	CBM	792	258736	1477	1139	34.9%	0.95%
	Proposed	1605	258205	1481	816	52.0%	0.88%
15dB	GHMT	1780	256057	3890	417	31.3%	0.98%
	CBM	940	258891	1322	991	41.6%	0.75%
	Proposed	1679	258451	1365	552	55.2%	0.73%

TP: True Positive. **FP**: False Positive.
TN: True Negative. **FN**: False Negative.
SEN: Sensitivity, TP/(TP+FN). **ER**: Error rate, (FP+FN)/512^2

Table 3.2: Comparative errors for the synthetic shape image

estimator, which can indicate how "noisy" the estimated feature map is; *false negative* is the number of pixels where the estimator says there is no feature present but ground truth image says otherwise, which indicates the probability that the estimator to miss features; *sensitivity* is the ratio of correct estimated edge pixels over all estimated edge pixels and the error rate, which is calculated as the percentage of mismatched pixel number over the total pixel number. The results are presented in Table 3.2 for both 10dB and 15dB. It is shown that the proposed algorithm yields the minimum error rate among the three candidates and the best sensitivity. The table also shows that GHMT gives the most "noisy" estimates, while CBM, on the other hand, usually gives relatively clean estimates, but has low TP counts, which may suggests the estimated parameter is less accurate than GHMT and the proposed method.

3.7.6 Computational Requirements

The proposed algorithm is implemented in Matlab and all the experiments were conducted on an Intel 2.0GMHZ CPU with 1GB memory. The majority computation for the proposed algorithm is spent on the estimation of the "Signal+Noise" model parameters, where it requires 6 multiplications, 4 additions per pixel in one iteration. With 7 iterations, there

is a total of 42 multiplications and 28 additions. For the Kalman filter refinement stage, under the simplification on the model, there are 5 multiplications, 3 additions for one block, plus an extra matrix inversion for one scale. As shown in Table 3.1, the Kalman filter is performed on 4 different parameters across 4 resolutions. It typically requires 40 seconds to run on a 512×512 image.

The CBM implementation is written in C and compiled with a GNU C compiler. With a pre-calculated lookup table, the algorithm only requires 4 multiplications and 1 addition per pixel. The running time for this monoscale algorithm is around 0.06 seconds.

For the GHMT algorithm, the complexity of the actual implementation is unclear from the conceptual description in [135]. Nevertheless, since the package is written in Matlab, it can be compared with the proposed method in terms of running time. Typically, some 50 seconds is need for GHMT to complete its calculation for one image.

It is clear that the CBM is most efficient in terms of computation while the GHMT has the highest computational burden. The proposed algorithm has a modest complexity while yielding the best estimates.

3.8 Summary and Discussion

The problem of linear feature extraction from noisy data was discussed in this chapter. With a *local* Gaussian mixture spectrum model handling the separation of noise and signal, it was shown that better estimates can be obtained for the linear feature parameters using the EM algorithm. A *global* multiscale image model which takes advantage of the multiscale geometrical edge coherence is also proposed so that the parameters estimated from the noisy image can be refined from coarser scales to finer scales using a Kalman filter. Finally, the significance of the features can be determined by a voting scheme using several significance measures. The proposed algorithm compares favourably with other parametric estimators

in terms of visual quality and accuracy, with a modest computational overhead. The use of the extracted parameters will be discussed later in Chapter 5 to improve the MFT-based image denoising scheme.

Chapter 4

The Multiscale Polar Cosine Transform

4.1 Introduction

As we have shown, not only edges, but also periodic textures can exist at all possible locations, orientations and scales in an image. The ability to efficiently analyse and describe directional patterns is thus of fundamental importance for image analysis. It is widely recognised that in natural images, there is usually a presence of strongly oriented local harmonics (textures) separated by curvilinear edges. While models for curved edges are well-established in various ways, such as some new directional transforms, and some wavelet coefficient statistical models, for texture representation, there are certain difficulties in defining complicated image patterns mathematically, making quantitative analysis difficult. There are, however, certain assumptions that can be made for most natural textures:

- *Oriented.*

  Many spatial textures have a prominent direction, for example wood patterns and seismograms, or a combination of several such components.

- *Periodic.*

  The texture tends to repeat certain basic elements at a given frequency and tends to repeat itself with a degree of affine invariance.

- *Localised.*

  In natural images, the textures are usually confined within a particular region. In other words, they are spatially localised.

Over the years, several kinds of wave packet bases such as local cosines [4, 2], wavelet packets [41, 134] and brushlets [106] have been proposed as suitable representations for oscillatory textures. However, they either lack orientation selectivity, or their spatial-frequency localisation is not satisfactory. Moreover, due to the Uncertainty Principle, a sparse representation for both singularities and periodic oscillations in a fixed basis is a fundamental dilemma [159]. A resort is to find an image-dependent *"adaptive basis"* which can accommodate both directional linear features as well as directional textures in an unified manner according to the *"image = edge + texture"* model.

The benefit of such sparse representations can be reflected in many applications. In particular, a simple thresholding of noisy coefficients in such expansions should be effective in removing noise while preserving image features of interest. The main objectives of this chapter are to construct a transform called the *"Multiscale Polar Cosine Transform"* and to establish its connections with other transforms. The denoising results are presented with comparison to other state-of-the-art transforms.

4.2 The Polar Cosine Transform

4.2.1 The Continuous Transform

A prototypical oriented texture pattern can be seen as a higher dimensional function which has a waveform constant along a fixed direction. Such functions are usually referred to as *"planar waves"* or *"ridge functions"* in related literature on approximation [123]. Formally, a ridge function is a multivariate function of the form $\psi(\boldsymbol{\xi} \cdot \boldsymbol{\theta})$ where $\psi(\cdot)$ is a univariate function usually referred to as the *ridge profile*, with spatial coordinate vector $\boldsymbol{\xi} \in \mathbb{R}^d$ and $\boldsymbol{\theta} \in \mathbb{S}^{d-1}$. This means $\boldsymbol{\theta}$ is on the unit sphere $\mathbb{S}^{d-1}$ in dimension $d > 1$ and indicates the orientation.

In order to model oriented texture, a real-to-complex ridge profile $\psi : \mathbb{R} \rightarrow \mathbb{C}$ can be defined as

$$\psi_\omega(\xi) = e^{i\omega\xi}. \tag{4.1}$$

which is basically the 1-D Fourier basis function, and the corresponding multivariate ridge function by definition is :

$$\psi_\omega(\boldsymbol{\xi} \cdot \boldsymbol{\theta}) = e^{i\omega(\boldsymbol{\xi} \cdot \boldsymbol{\theta})}. \tag{4.2}$$

The ridge function gives a complex trigonometric oscillatory constant along the direction $\boldsymbol{\theta}$. Taking the inner product of it with a function $f(\boldsymbol{\xi})$ gives:

$$\hat{f}(\omega, \boldsymbol{\theta}) = \langle f(\boldsymbol{\xi}), \psi_\omega(\boldsymbol{\xi} \cdot \boldsymbol{\theta}) \rangle = \int f(\boldsymbol{\xi}) e^{-i\omega(\boldsymbol{\xi} \cdot \boldsymbol{\theta})} d\boldsymbol{\xi}. \tag{4.3}$$

This is exactly the d-dimensional Fourier transform expressed in polar coordinate form, or just simply *polar Fourier transform.* The conversion to Cartesian coordinates can be done

by separating the radial frequency into a frequency vector:

$$\boldsymbol{\omega} = \omega\boldsymbol{\theta}. \tag{4.4}$$

With a slight rearrangement of equation (4.3), the equivalence becomes obvious:

$$\hat{f}(\omega, \boldsymbol{\theta}) = \int f(\boldsymbol{\xi}) e^{-i\boldsymbol{\xi}\cdot(\omega\boldsymbol{\theta})} d\boldsymbol{\xi} = \int f(\boldsymbol{\xi}) e^{-i\boldsymbol{\xi}\cdot\boldsymbol{\omega}} d\boldsymbol{\xi} = \hat{f}(\boldsymbol{\omega}). \tag{4.5}$$

Meanwhile, the definition for the continuous Radon transform in d-dimensions is

$$\mathcal{R}f(t, \boldsymbol{\theta}) = \int f(\boldsymbol{\xi}) \delta(\boldsymbol{\xi}\cdot\boldsymbol{\theta} - t) d\boldsymbol{\xi}. \tag{4.6}$$

The relationship between the Radon transform and the polar Fourier transform is stated as the *Fourier-Slice Theorem* [16]:

Theorem 1 (Fourier Slice Theorem). *The 1D Fourier transform with respect to t of the projection $\mathcal{R}f(t, \boldsymbol{\theta})$ is equal to a central slice, at a given orientation $\boldsymbol{\theta}$, of the higher dimensional Fourier transform of the function $f(\boldsymbol{\xi})$, that is,*

$$\hat{\mathcal{R}}f(t, \boldsymbol{\theta}) = \hat{f}(\omega, \boldsymbol{\theta}). \tag{4.7}$$

If some boundary condition is to be posed at each polar orientation, instead of being defined on $\mathbb{R}$, the Radon slice $\mathcal{R}_{\boldsymbol{\theta}}f(t)$ is assumed to be an even function $\mathcal{R}_{\boldsymbol{\theta}}f(t) = \mathcal{R}_{\boldsymbol{\theta}}f(-t)$, then its Fourier transform can be written as follows:

$$\hat{\mathcal{R}}f(t, \boldsymbol{\theta}) = \sqrt{\frac{2}{\pi}} \int_0^{+\infty} \mathcal{R}f(t, \boldsymbol{\theta}) \cos(\omega t)\, dt. \tag{4.8}$$

It is noted that only the real cosine part remains in the above equation due to the even

boundary assumption on the slices. This is equivalent to a cosine tranform with ridge-type basis functions defined as :

$$\psi_{\omega}^{\mathcal{C}}(\boldsymbol{\xi}\cdot\boldsymbol{\theta}) = \sqrt{\frac{2}{\pi}}\cos(\omega\boldsymbol{\xi}\cdot\boldsymbol{\theta}). \tag{4.9}$$

In the same way by assuming $\mathcal{R}_{\theta}f(t) = -\mathcal{R}_{\theta}f(-t)$, the sine counterpart is

$$\psi_{\omega}^{\mathcal{S}}(\boldsymbol{\xi}\cdot\boldsymbol{\theta}) = \sqrt{\frac{2}{\pi}}\sin(\omega\boldsymbol{\xi}\cdot\boldsymbol{\theta}). \tag{4.10}$$

Correspondingly, the continuous polar cosine and sine transform are denoted respectively as:

$$\mathcal{C}f(\omega,\boldsymbol{\theta}) = \langle f(\boldsymbol{\xi}), \psi_{\omega}^{\mathcal{C}}(\boldsymbol{\xi}\cdot\boldsymbol{\theta})\rangle = \sqrt{\frac{2}{\pi}}\int_{0}^{+\infty} f(\boldsymbol{\xi})\cos(\omega\boldsymbol{\xi}\cdot\boldsymbol{\theta})d\boldsymbol{\xi}. \tag{4.11}$$

$$\mathcal{S}f(\omega,\boldsymbol{\theta}) = \langle f(\boldsymbol{\xi}), \psi_{\omega}^{\mathcal{S}}(\boldsymbol{\xi}\cdot\boldsymbol{\theta})\rangle = \sqrt{\frac{2}{\pi}}\int_{0}^{+\infty} f(\boldsymbol{\xi})\sin(\omega\boldsymbol{\xi}\cdot\boldsymbol{\theta})d\boldsymbol{\xi}. \tag{4.12}$$

These two transforms, along with the polar Fourier transform, can be regarded as a family of *Polar Trigonometric Transforms*. In the context of image processing, the polar cosine transform is the transform of interest in this work, due to its better approximation convergence properties attributed to the even boundary extension [3]. Since it is essentially the Fourier transform, the completeness of the transform makes the transform operator $\mathcal{C}$ unitary which means $\mathcal{C}^{-1} = \mathcal{C}^{*}$. Since the transform is real, the operator is self-conjugate, making the inverse transform exactly the same as the forward transform:

$$\mathcal{C}^{-1} = \mathcal{C}. \tag{4.13}$$

The same also holds for the continuous polar sine transform and its inverse.

DCT-I	$\frac{1}{2}(x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n \cos\left[\frac{\pi}{N-1} nk\right]$
DCT-II	$\sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]$
DCT-III	$\frac{1}{2}x_0 + \sum_{n=1}^{N-1} x_n \cos\left[\frac{\pi}{N} n\left(k+\frac{1}{2}\right)\right]$
DCT-IV	$\sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]$
DST-I	$\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N+1}(n+1)(k+1)\right]$
DST-II	$\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)(k+1)\right]$
DST-III	$\sum_{n=0}^{N-2} x_n \sin\left[\frac{\pi}{N}(n+1)\left(k+\frac{1}{2}\right)\right]$
DST-IV	$\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]$

Table 4.1: List of DCTs and DSTs

4.2.2 The Discrete Transform

The continuous Polar Cosine Transform essentially replaces the polar Fourier transform's complex basis with a real cosine sinusoid as the ridge function, by assuming the projected Radon slice is even. In a discrete case, the Fourier-related transforms that operate on a function over a finite domain can be thought of as implicitly defining an extension of that function outside the domain. The Discrete Fourier Transform (DFT) implies a periodic extension of the original function. A Discrete Cosine Transform (DCT), like the continuous cosine transform, implies an even extension of the original function.

However, when the transform is to operate on finite, discrete sequences, two issues arise that are not relevant in case of the continuous cosine transform. First, one has to specify whether the function is even or odd at both the left and right boundaries of the domain. Secondly, one has to specify around what point the function is even or odd. These issues result in several different versions of DCTs, a full list of which along with corresponding DSTs is given in Table 4.1. For a discrete polar cosine transform, any of the DCTs can be adopted to form the basis ridge function. In particular, the so-called DCT-II is widely used in many applications such as the famed JPEG compression [154] due to its even boundary extensions on both ends, which gives better approximation convergence. The

1-dimensional DCT-II function is defined by the ridge profile:

$$\psi_k[n] = \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]. \tag{4.14}$$

where N is the possible number of frequencies as well as the length of the Radon projection slice and $k = 0, 1, ..., N-1$ is the frequency index. In the context of image processing, the discussion of discretisation of the polar cosine transform will be further restricted to the 2-dimensional case, which means the unit orientation vector becomes

$$\theta = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}. \tag{4.15}$$

With $\boldsymbol{\xi} = [\xi_1, \xi_2]$, the discrete cosine ridge function is

$$\psi_k[\xi_1\cos\theta + \xi_2\sin\theta] = \cos\left[\frac{\pi}{N}\left(\xi_1\cos\theta + \xi_2\sin\theta + \frac{1}{2}\right)k\right]. \tag{4.16}$$

Then the discrete polar cosine transform on a $M \times M$ 2-D image $f[\boldsymbol{\xi}]$ can be defined as :

$$\begin{aligned} \mathcal{C}f[k,\theta] &= \langle f[\boldsymbol{\xi}], \psi_k[\xi_1\cos\theta + \xi_2\sin\theta]\rangle \\ &= \sum_{\xi_1=0}^{M-1}\sum_{\xi_2=0}^{M-1} f[\xi_1,\xi_2]\cos\left[\frac{\pi}{N}\left(\xi_1\cos\theta + \xi_2\sin\theta + \frac{1}{2}\right)k\right]. \end{aligned} \tag{4.17}$$

The inverse transform for DCT-II is the DCT-III transform. The corresponding cosine ridge function for the inverse transform is

$$\psi_k^{-1}[\xi_1\cos\theta + \xi_2\sin\theta] = \lambda_k\cos\left[\frac{\pi}{N}\left(\xi_1\cos\theta + \xi_2\sin\theta + \frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right], \tag{4.18}$$

where

$$\lambda_k = \begin{cases} 1/2 & \text{if } k = 0 \\ 1 & \text{if } k \neq 0, \end{cases}$$

and the discrete inverse polar cosine transform operator $\mathcal{C}^{-1}$ is given by:

$$\begin{aligned} \mathcal{C}^{-1} f[k,\theta] &= \langle f[\xi], \psi_k[\xi_1 \cos\theta + \xi_2 \sin\theta] \rangle \\ &= \sum_{\xi_1=0}^{M-1} \sum_{\xi_2=0}^{M-1} f[\xi_1,\xi_2] \lambda_k \cos\left[\frac{\pi}{N}\left(\xi_1 \cos\theta + \xi_2 \sin\theta + \frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]. \end{aligned} \tag{4.19}$$

4.2.3 Radon-based Digital Implementation

Equations (4.7) and (4.8) suggest that the discrete Polar Cosine Transform (PCT) can be implemented by a Radon transform, that is, taking the 1-D discrete cosine transform on Radon slices:

$$\mathcal{C}f[k,\theta] = \sum_{t=0}^{N-1} \mathcal{R}f[t,\theta] \cos\left[\frac{\pi}{N}\left(t+\frac{1}{2}\right)k\right]. \tag{4.20}$$

As the Fourier Slice Theorem shows, the Radon transform can be implemented by taking the central slice in the Fourier spectrum and then performing a 1-D inverse Fourier transform on it. Applying DCT on the Radon slices, we can obtain the discrete PCT.

This gives a means for a digital implementation of the discrete polar cosine transform. For the Radon transform, various ways to implement it in a discrete fashion have been attempted, amongst which the Fast Slant Stack [6] was chosen, which is based on a pseudo-polar FFT [5]. According to [6], the transform is computationally efficient, algebraically exact, geometrically faithful and its inversion is numerically stable.

With the implementation described above, the polar cosine transform basis vectors for an 8×8 image block can be seen in Figure 4.1. The basis vectors are indexed by frequency k and orientation θ. As a result of the Cartesian-to-Polar conversion in the frequency domain, the low-frequency basis vectors are clearly being over-sampled, resulting in a redundant

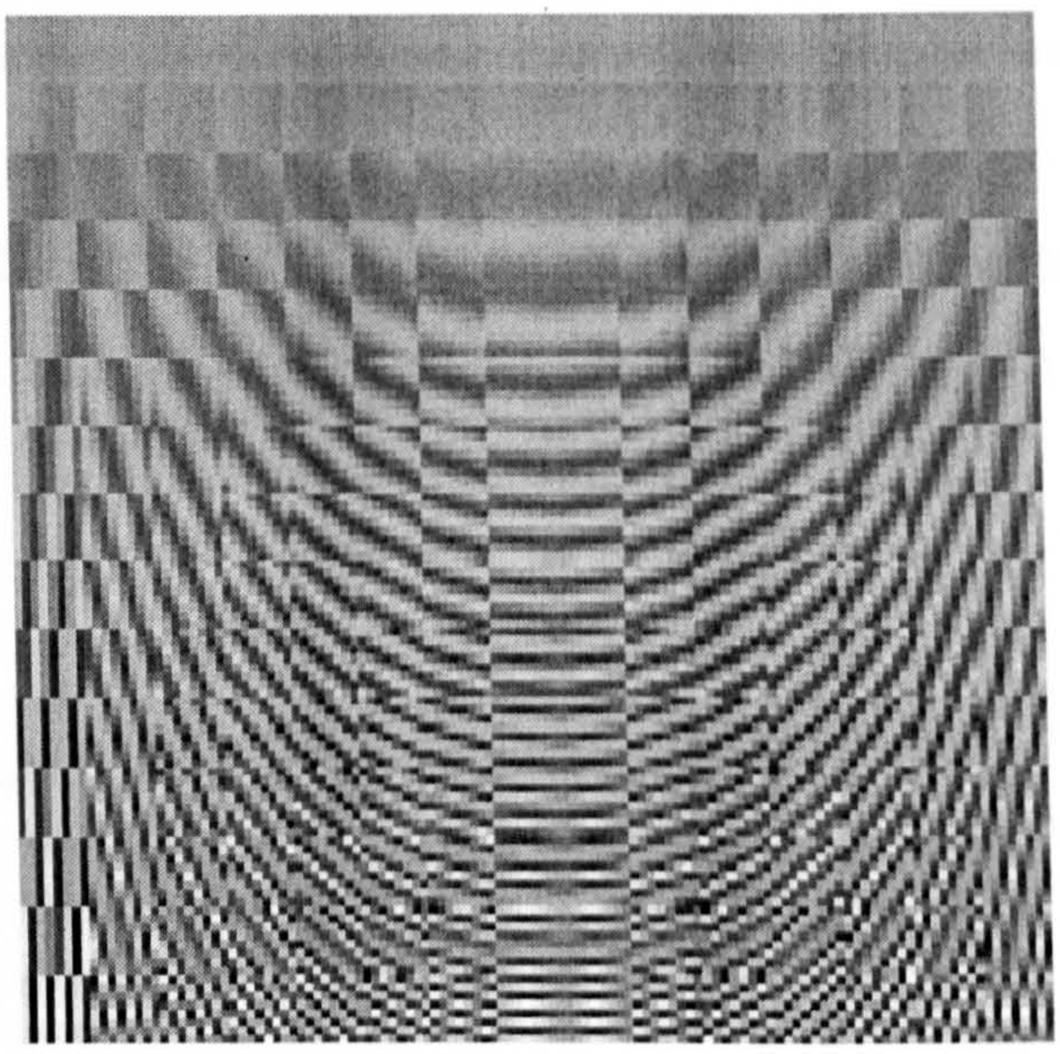


Figure 4.1: The 8×8 discrete PCT-II basis vectors

frame where the number of coefficients is four times the original data size.

4.2.4 Butterfly-based Digital Implementation

The above digital implementation of the PCT based on the Radon transform is akin to the digital ridgelet transform [27]. Such a Radon-based approach is flexible in constructing various directional ridge-type transforms, but it suffers from several drawbacks of the underlying Radon transform. First is the high computational requirement both for the forward and inverse transforms. The forward transform includes a Pseudo Polar Fourier transform, 1-D inverse Fourier transforms on polar slices and forward cosine transforms. The inverse Fast Slant Stack uses an iterative algorithm in the reconstruction to minimise numerical instability. Secondly, the redundancy factor could be an issue in some applications. While it seems implausible to construct a digital ridgelet transform without invoking the Radon transform, it does not necessarily hold true for the PCT. A possible construction of such directional real trigonometric transform in the context of modulated lapped

transforms was discussed in [1]. The same construction is adopted here in implementing a digital polar cosine transform.

It can be observed from Table 4.1 that the DCT-IV and DST-IV share the same function parameters. In 2D, Cartesian separable basis functions can be formed by the tensor product of 1D bases:

$$\psi^{\mathcal{C}}_{k_1,k_2}[\xi_1,\xi_2] = \cos\left[\frac{\pi}{N}\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)\right]\cos\left[\frac{\pi}{N}\left(\xi_2+\frac{1}{2}\right)\left(k_2+\frac{1}{2}\right)\right]. \tag{4.21}$$

$$\psi^{\mathcal{S}}_{k_1,k_2}[\xi_1,\xi_2] = \sin\left[\frac{\pi}{N}\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)\right]\sin\left[\frac{\pi}{N}\left(\xi_2+\frac{1}{2}\right)\left(k_2+\frac{1}{2}\right)\right]. \tag{4.22}$$

By setting $A = \frac{\pi}{N}\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)$ and $B = \frac{\pi}{N}\left(\xi_2+\frac{1}{2}\right)\left(k_2+\frac{1}{2}\right)$, the expression is simplified as follows:

$$\begin{aligned}\psi^{\mathcal{C}}_{k_1,k_2}[\xi_1,\xi_2] - \psi^{\mathcal{S}}_{k_1,k_2}[\xi_1,\xi_2] &= \cos A\cos B - \sin A\sin B \\ &= \cos[A+B] \\ &= \cos\left[\frac{\pi}{N}\left(\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)+\left(\xi_2+\frac{1}{2}\right)\left(k_2+\frac{1}{2}\right)\right)\right] \\ &= \cos\left[\frac{\pi}{N}\left(\xi_1k_1+\xi_2k_2+\frac{1}{2}(\xi_1+\xi_2+k_1+k_2+1)\right)\right].\end{aligned} \tag{4.23}$$

Therefore the difference between the 2D basis functions of DCT-IV and DST-IV at given frequencies can be reduced into a cosine component. Furthermore, since $k_1 = k\cos\theta$, $k_2 = k\sin\theta$, where $k = \sqrt{k_1^2+k_2^2}$ is the polar frequency index and $\theta = \arctan(k_2/k_1)$, the

above equation (4.23) can be rewritten in the polar cosine form:

$$\begin{aligned}\psi^{\mathcal{X}}_{k,\theta}[\xi_1,\xi_2] &= \psi^{\mathcal{C}}_{k\cos\theta,k\sin\theta}[\xi_1,\xi_2] - \psi^{\mathcal{S}}_{k\cos\theta,k\sin\theta}[\xi_1,\xi_2] \\ &= \cos\left[\frac{\pi}{N}\left((\xi_1\cos\theta+\xi_2\sin\theta)\,k+\frac{1}{2}\left(\xi_1+\xi_2+k\cos\theta+k\sin\theta+1\right)\right)\right].\end{aligned} \tag{4.24}$$

However, it should be noted that compared with the PCT-II defined in equation (4.17), although the basis vectors do not represent a strict form of ridge functions, they are directional and can be arranged in polar form. The above is set for the case when $k_1 = 0,...,N-1$ and $k_2 = 0,...,N-1$. For negative frequencies, it is clear that

$$\begin{aligned}\psi^{\mathcal{X}}_{k_1,-k_2-1}[\xi_1,\xi_2] &= \psi^{\mathcal{C}}_{k_1,-k_2-1}[\xi_1,\xi_2] - \psi^{\mathcal{S}}_{k_1,-k_2-1}[\xi_1,\xi_2] \\ &= \cos A\cos[-B] - \sin A\sin[-B] \\ &= \cos A\cos B + \sin A\sin B \\ &= \psi^{\mathcal{C}}_{k_1,k_2}[\xi_1,\xi_2] + \psi^{\mathcal{S}}_{k_1,k_2}[\xi_1,\xi_2].\end{aligned} \tag{4.25}$$

This suggests that for $-k_2-1 = -1,....,-N$, the basis function is just the sum of DCT-IV and DST-IV. In the same way, it is not difficult to see that when $k_1 = -1,...,-N$ and $k_2 = -N,...,N-1$, which is the basis functions for the lower half plane of the frequency spectrum:

$$\psi^{\mathcal{X}}_{k_1,k_2}[\xi_1,\xi_2] = \psi^{\mathcal{X}}_{-k_1-1,-k_2-1}[\xi_1,\xi_2]. \tag{4.26}$$

Thus the corresponding transform domain exhibits the same Hermite symmetry as the Fourier transform, with basis functions very close to the FFT's basis functions. As a result, the transform

$$\mathcal{X}f[k_1,k_2] = \langle f[\xi_1,\xi_2], \psi^{\mathcal{X}}_{k_1,k_2}[\xi_1,\xi_2]\rangle \tag{4.27}$$

is only twice redundant, by removing a half-plane of the spectrum and can be efficiently

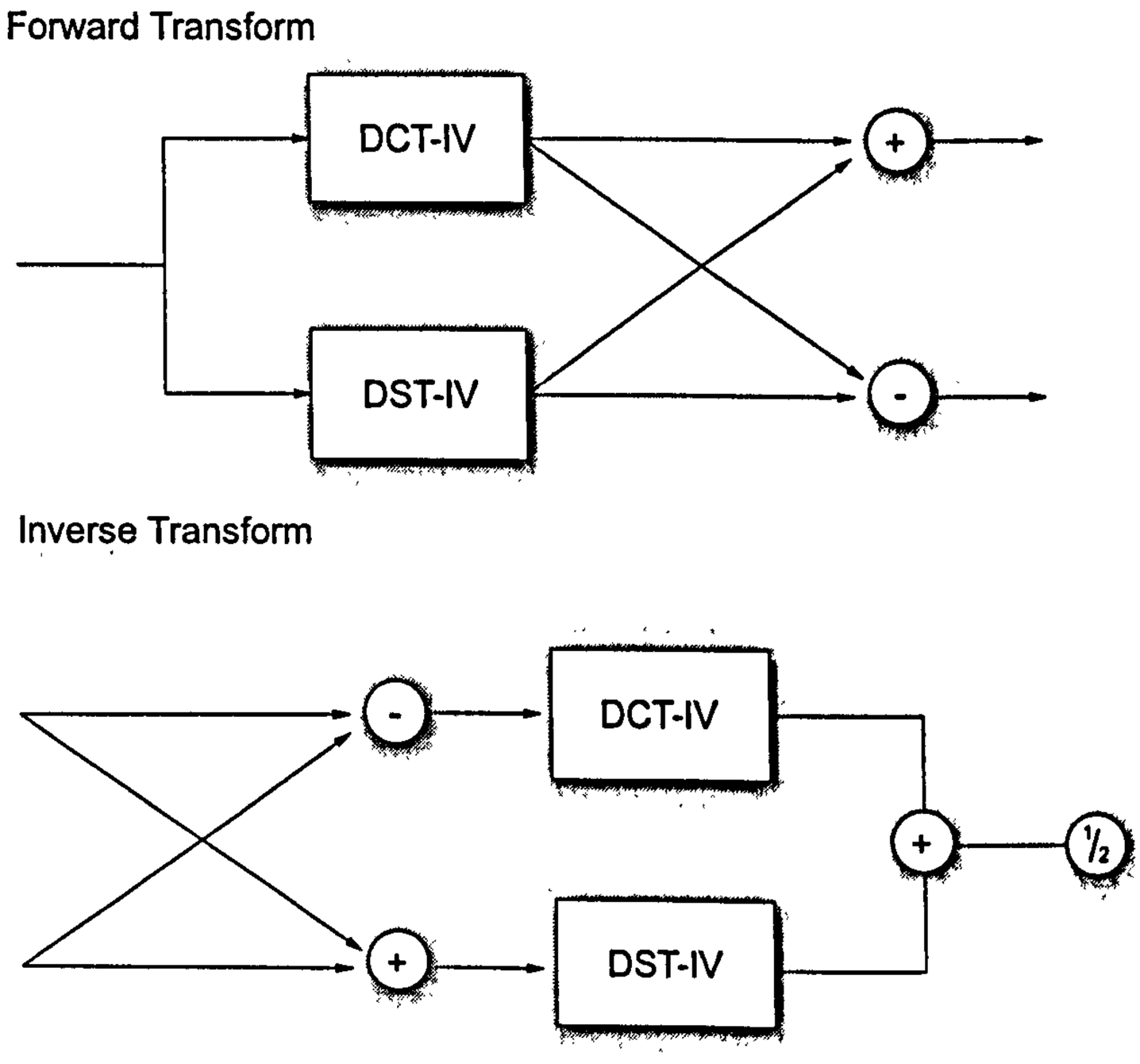


Figure 4.2: The forward PCT-X transform and the inverse transform

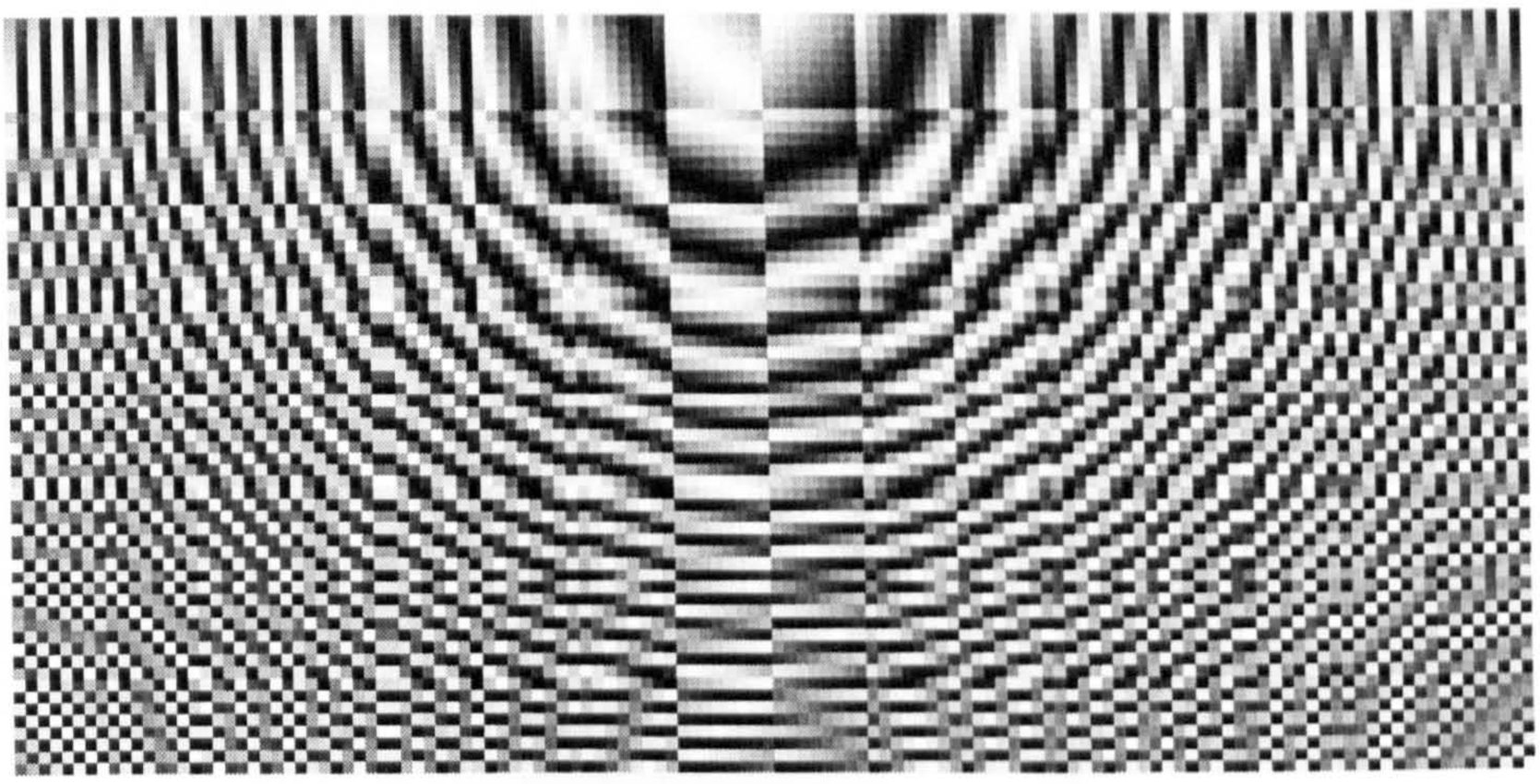


Figure 4.3: The 8×8 discrete PCT-X basis vectors

implemented by a butterfly computation. A schematic illustration of the transform can be seen in Figure 4.2. For the sake of convenience, this particular digital implementation will be referred to as PCT-X. A complete basis for size 8×8 is shown in Figure 4.3.

It is observed that the PCT-X is closely related to the Fourier transform itself, despite the new transform being real. The expansion of PCT-X, therefore, can be expected to have similarities with the Fourier magnitude spectrum.

In Figure 4.4, a comparison between the PCT-X spectrum and the Fourier spectrum is made. A reptile texture patch is chosen as the test data and the two transforms yield very similar magnitude responses. However, the histograms suggest that the coefficients of the PCT-X expansion are sparser than those of the Fourier transform, attributed to the symmetric boundary extension of the underlying DCT-IV and DST-IV.

4.2.5 Nonlinear Approximation

As previously mentioned in Section 4.1, with the difficulty in mathematically formulating texture, it is hard to approach the nonlinear approximation problem quantitatively. The

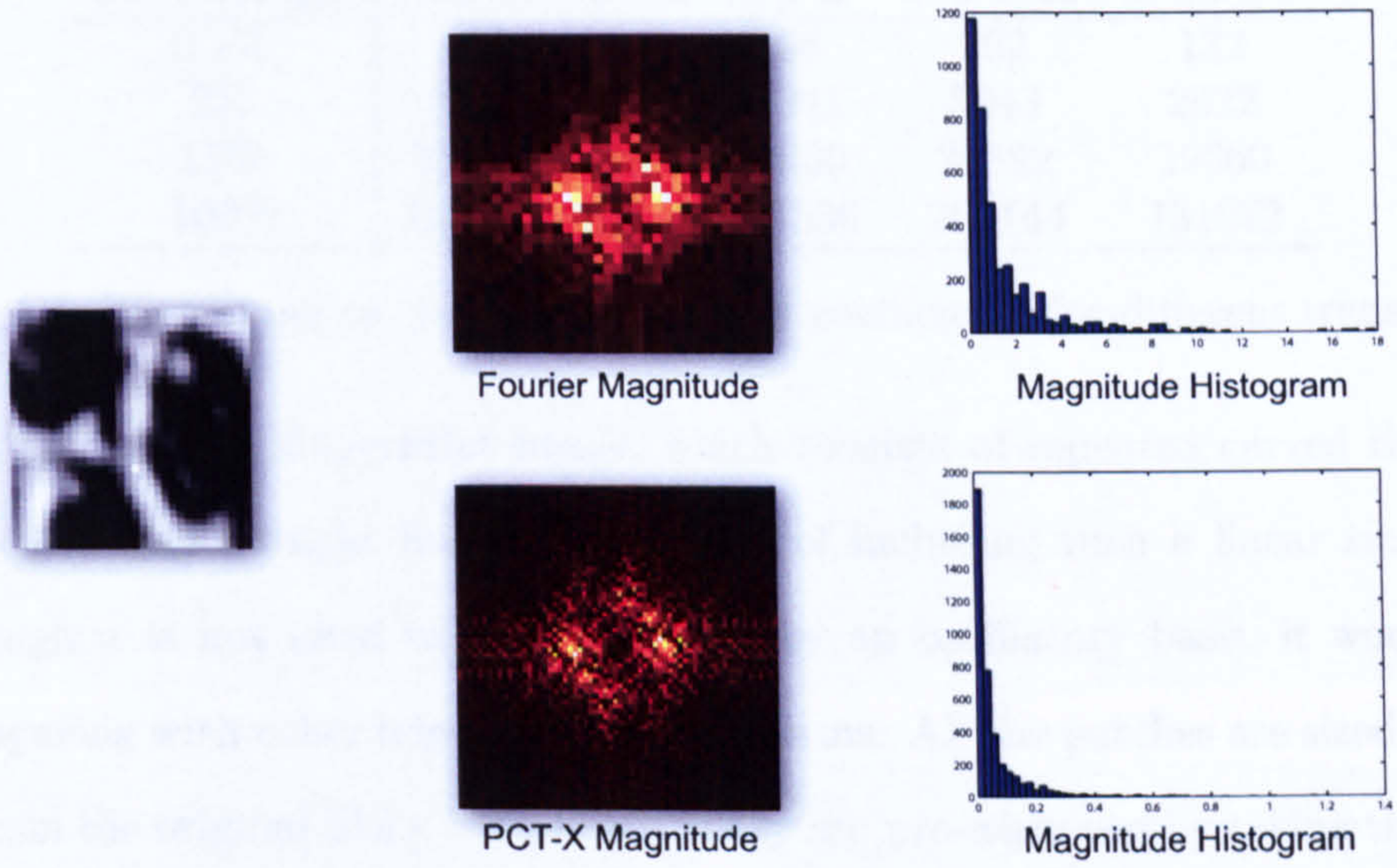


Figure 4.4: Spectrum comparison between the PCT-X and the Fourier Transform

theoretical study of the transform in the context of computational harmonic analysis is beyond the scope of this thesis, where the main concern is on examining its effectiveness in providing sparsity in the transformed expansion. The approach taken here is more empirical, where the nonlinear approximation results are presented in the form of outputs from numerical experiments.

Two digital PCT implementations were selected for nonlinear approximation experiments: the Radon-based PCT with DCT-II as ridge profile (PCT-II); and the butterfly PCT-X implementation as shown in Figure 4.2. They are compared with three other transforms: the Fast Fourier Transform (FFT); the DCT-II, which is a Cartesian separable real-to-real implementation of FFT; and the *Fast Curvelet Transform* (FCT) which is implemented by wrapping of specially selected subbands of Fourier samples [22].

Three different image patches are tested for nonlinear approximation. The first one is a rotated version of the wood grain texture from USC-SIPI Image Database numbered `1.1.09`. The texture is well-oriented but with certain discontinuities. The second one

Percentage	FCT	DCT	FFT	PCT-II	PCT-X
0.1%	185	66	66	262	132
2%	3700	1311	1311	5243	2622
15%	27748	9830	9830	39322	19660
100%	184985	65536	65536	262144	131072

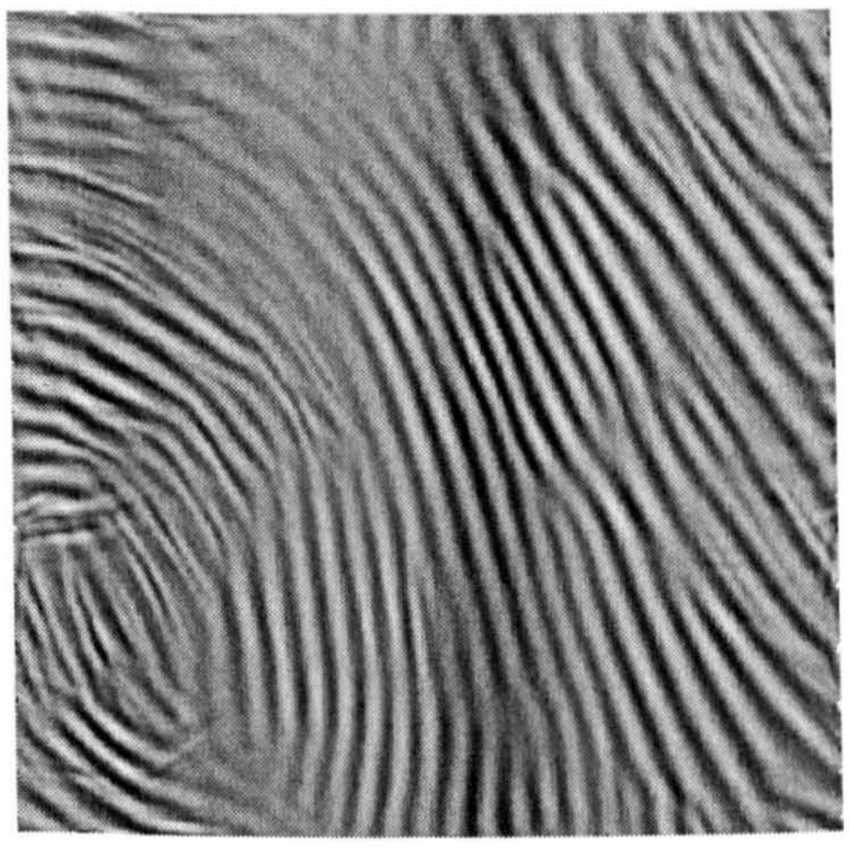
Table 4.2: Percentage vs. number of retained coefficients for different transforms

contains a portion of a fingerprint image, which consists of repeated curved ridges. The last one is a simple straight line. The purpose of including such a linear singularity is that although it is not ideal to be represented by an oscillatory basis, it would be still worth comparing with other trigonometric transforms. All the patches are sized 256×256, cropped from the original 512×512 images. They are pre-whitened by taking the highpass subband Laplacian pyramid.

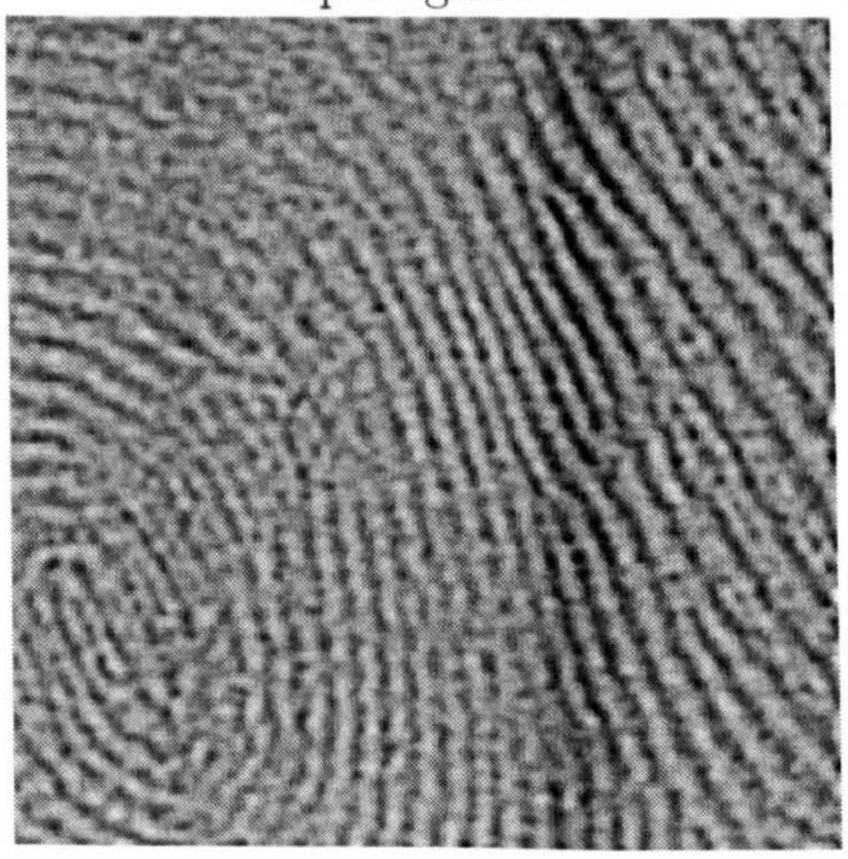
The nonlinear approximation results are represented in Figure 4.8-4.10 as PSNR curves. The PSNR values are plotted against the top *percentage of coefficients*, from 2% to 50%. The percentage instead of the number of coefficients is used due to the fact that different transforms yield different number of coefficients. For the sake of convenience, a table of percentages and the corresponding retained numbers of coefficients for different transforms are listed in Table 4.2. It is observed that for two textured images, `wood` and `fp`, the PCTs outperform the other candidates consistently. With fewer coefficients, the PCT-X's PSNR is usually close to the PCT-II, occasionally outperforms it (on `fp`). This is due to the fact that PCT-II's reconstruction is less stable with fewer coefficients while PCT-X's inverse transform is exact. However, the goodness of DCT-II's even symmetric boundaries eventually yields better PSNR with relatively more coefficients involved. While low PSNR performances are to be expected from the trigonometric transforms, it is also reassuring that the two PCTs give better PSNR than the DCT and FFT.

Figure 4.5 shows the reconstructions from the top 2% coefficients of these four transforms on `fp`. The reconstruction from FCT looks artificial, due to its inability to capture

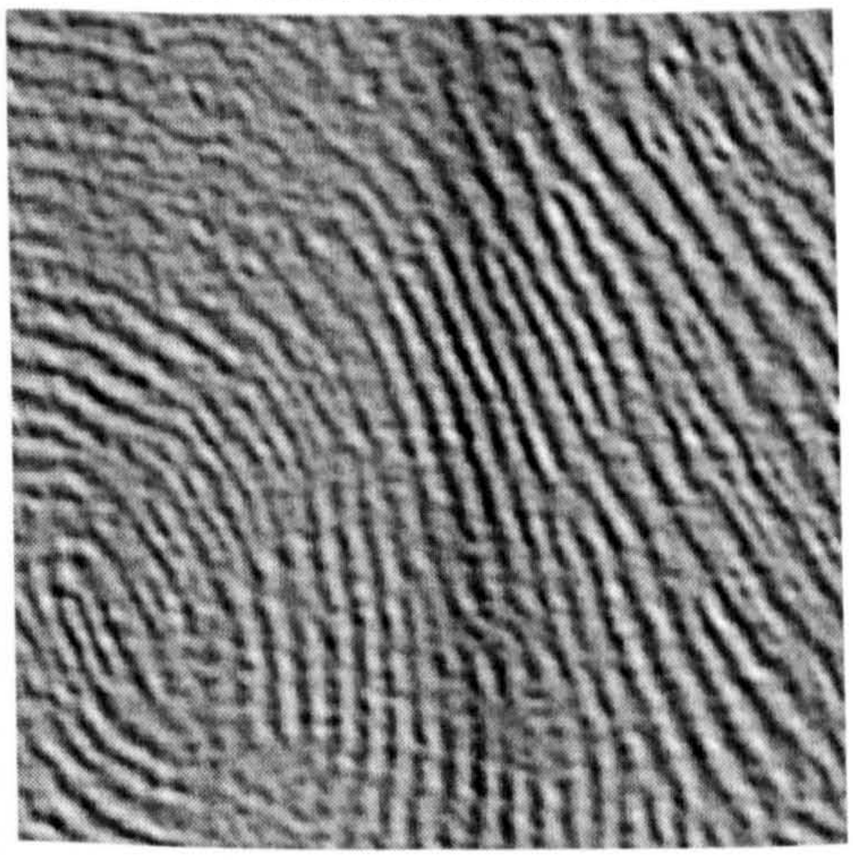
fp original



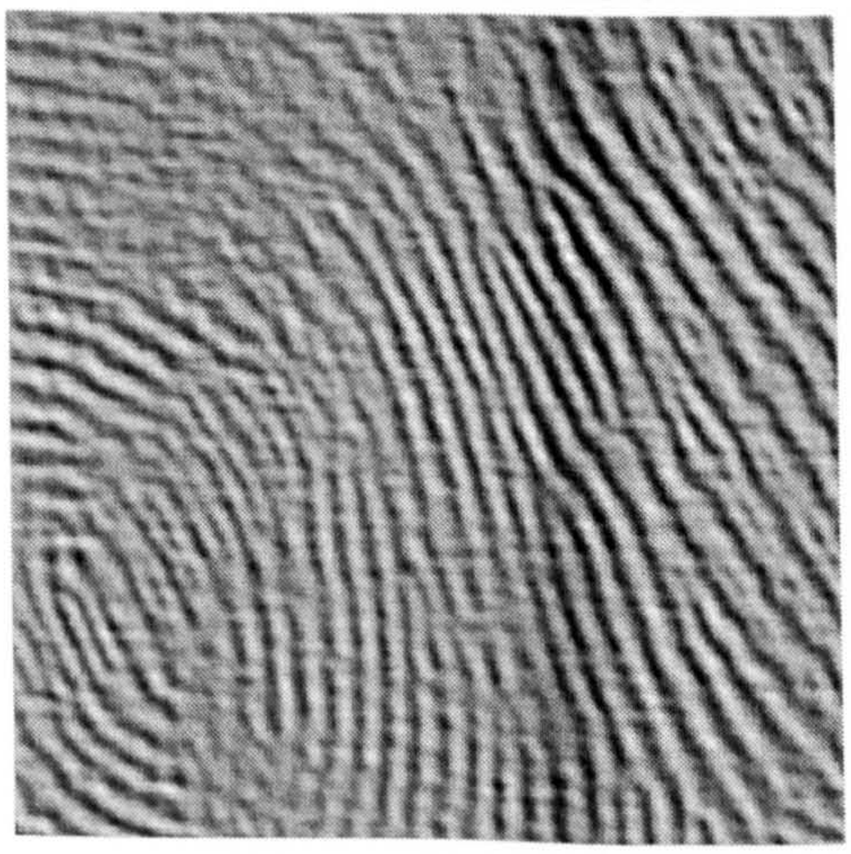
Fast Curvelet Transform



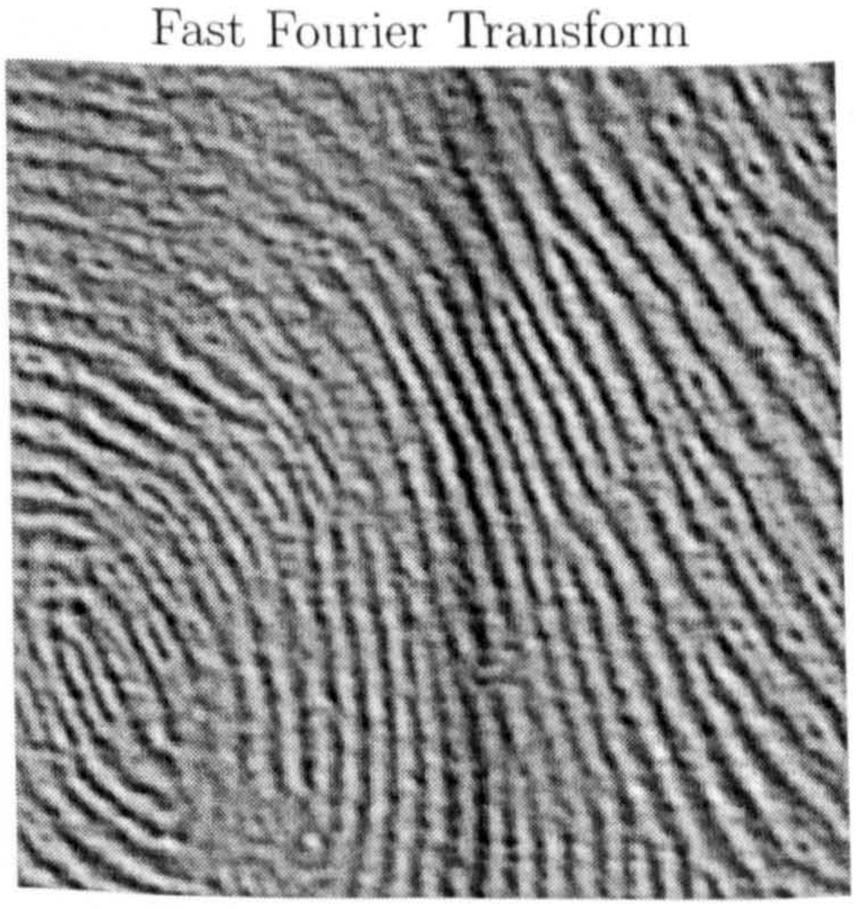
Discrete Cosine Transform



Fast Fourier Transform

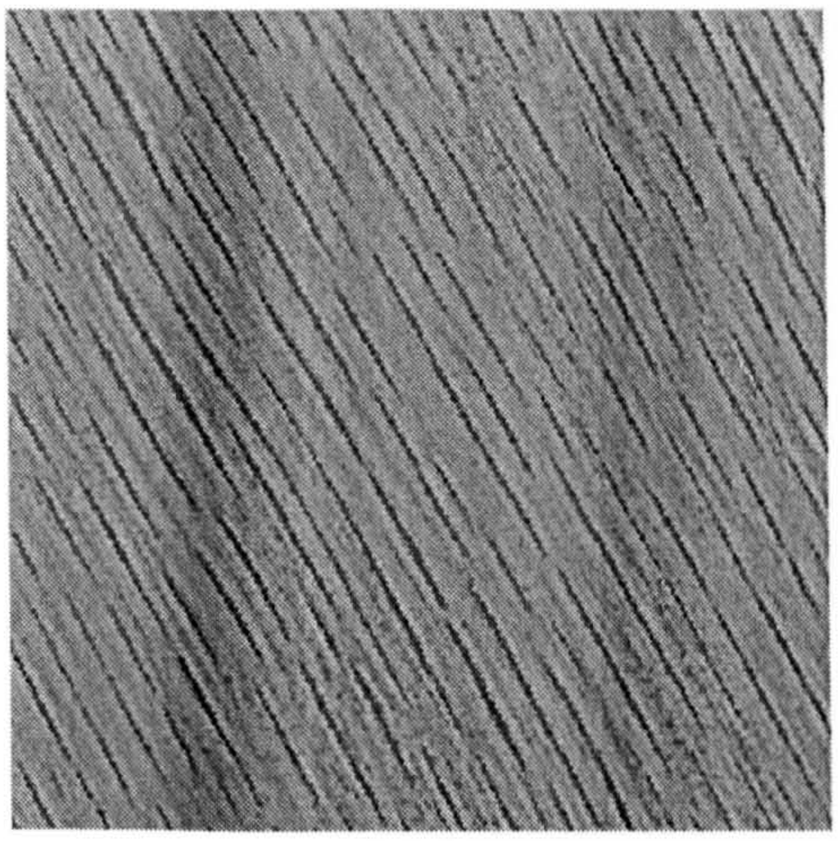


PCT-II

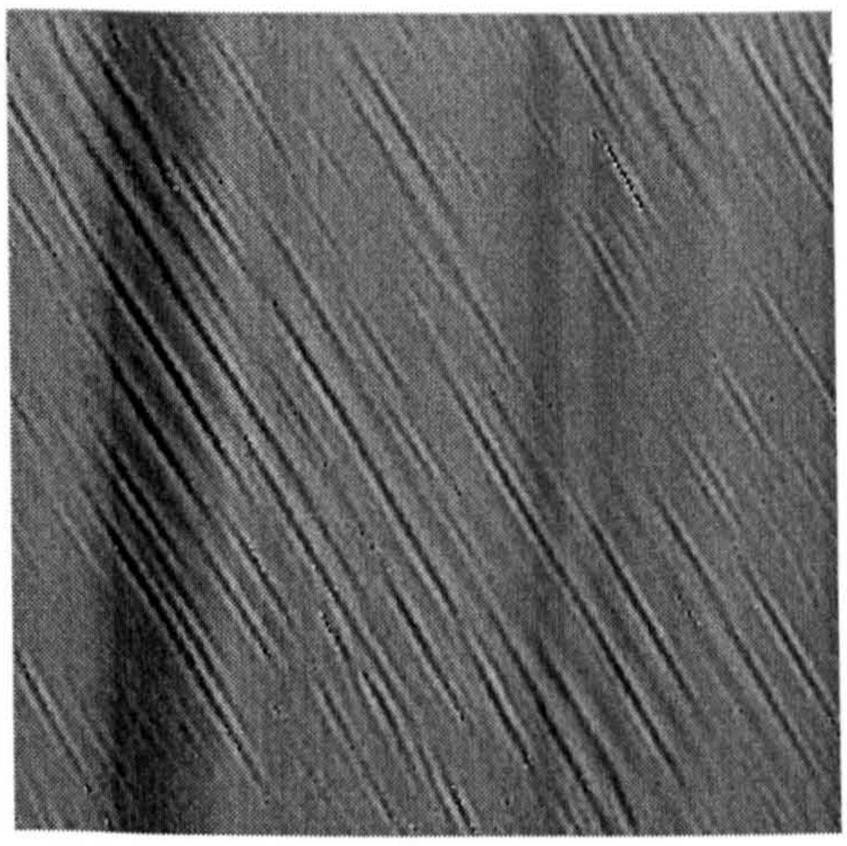


PCT-X

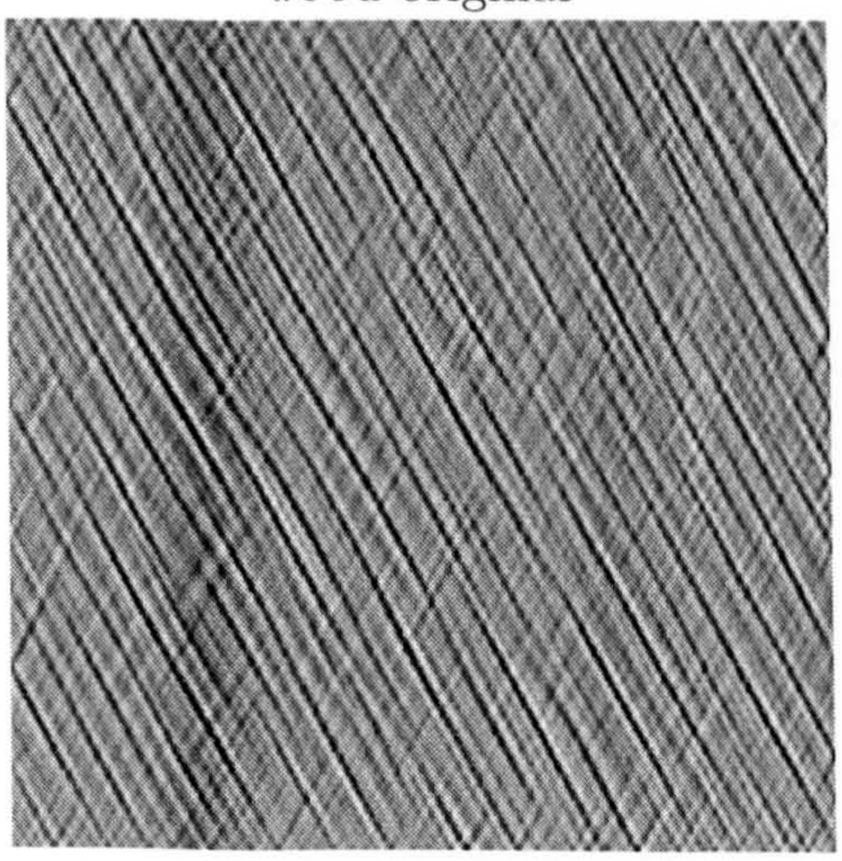
Figure 4.5: Illustrative results for nonlinear approximation on fp



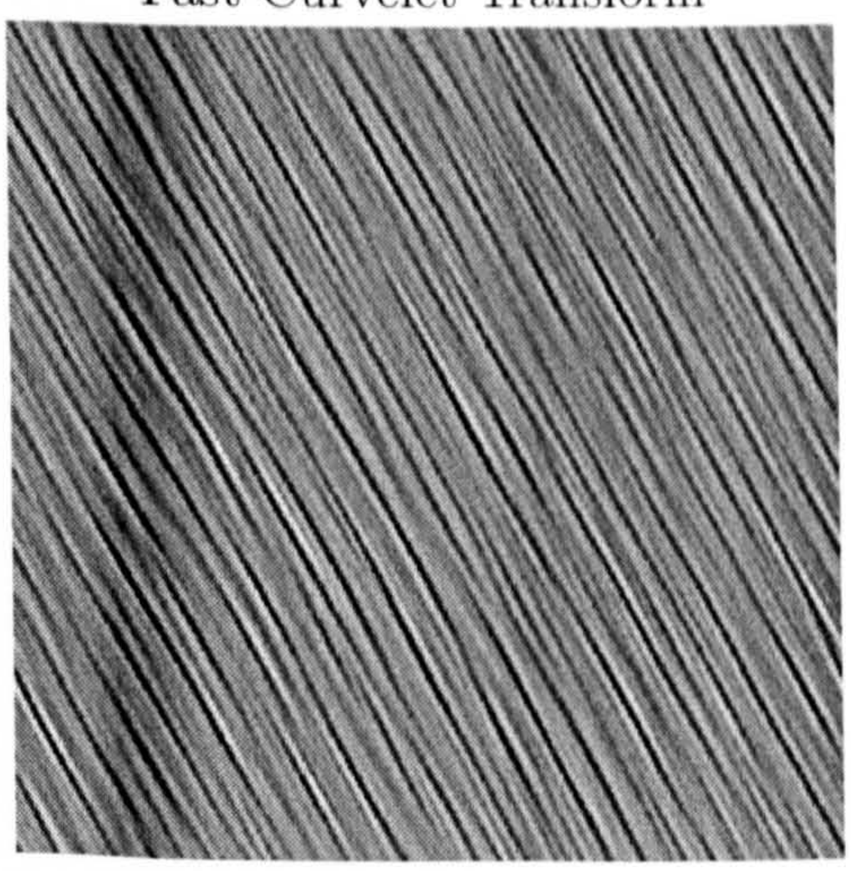
wood original



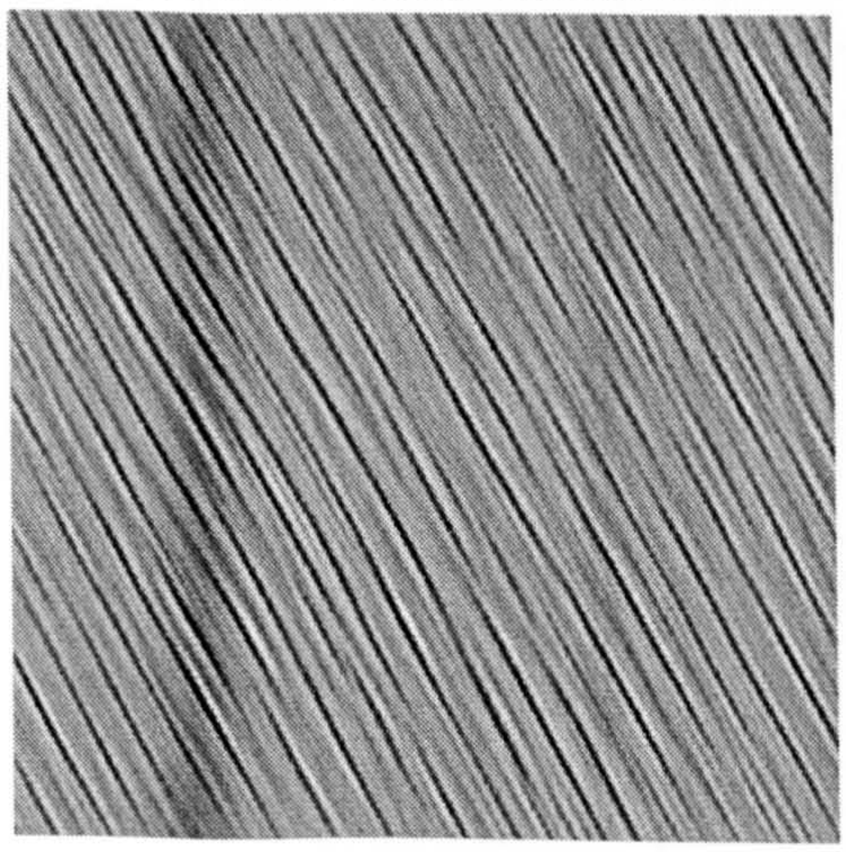
Fast Curvelet Transform



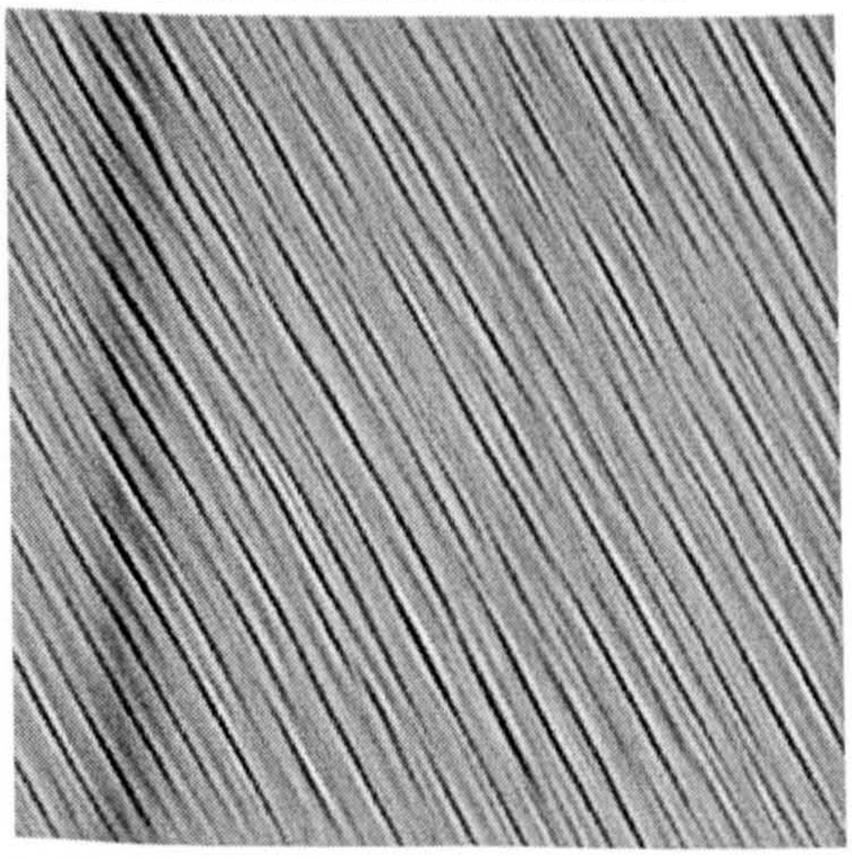
Discrete Cosine Transform



Fast Fourier Transform

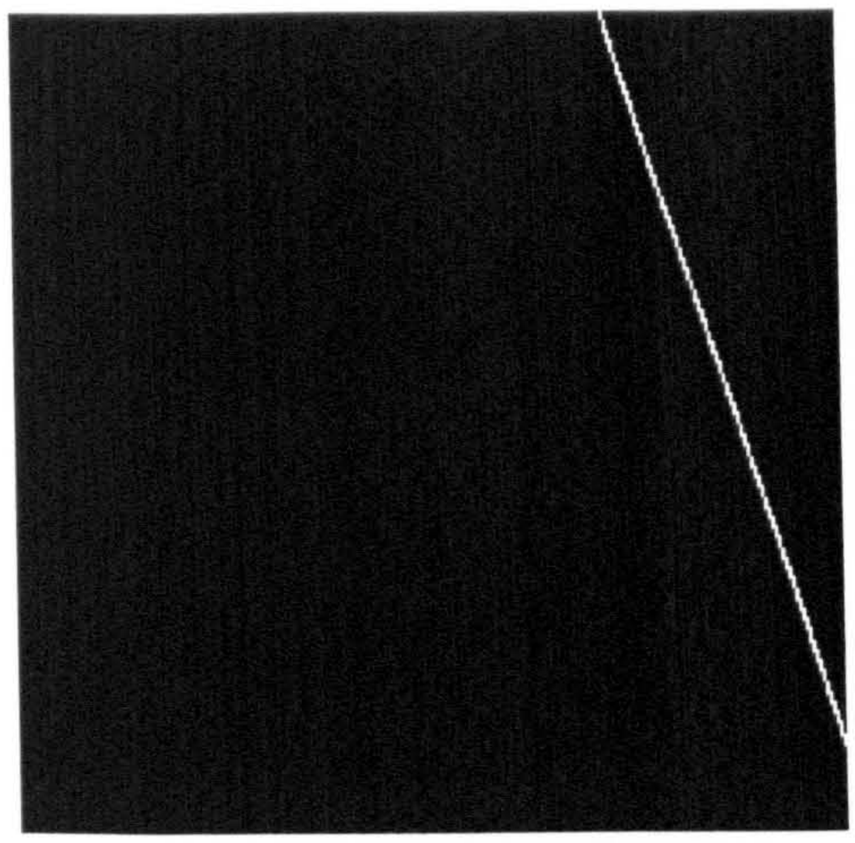


PCT-II

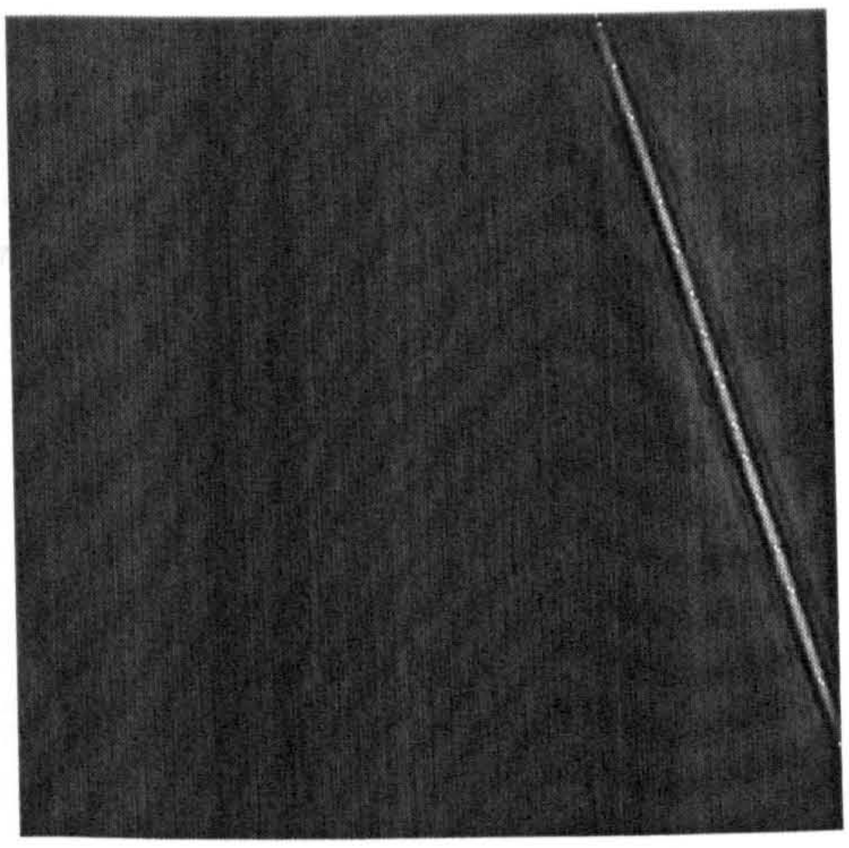


PCT-X

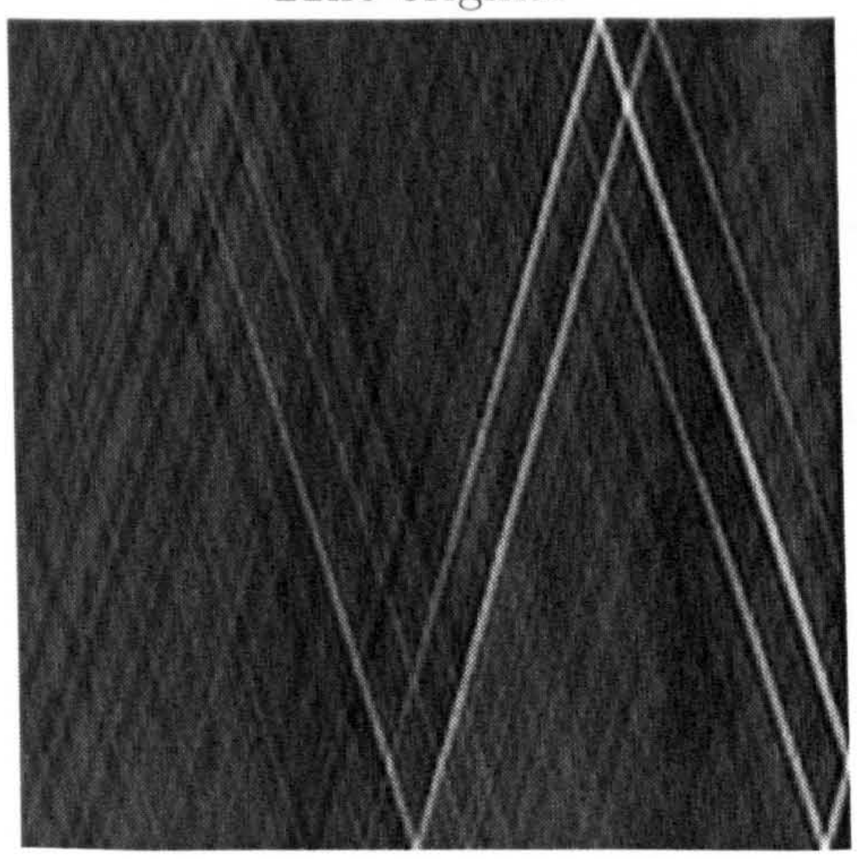
Figure 4.6: Illustrative results for nonlinear approximation on wood



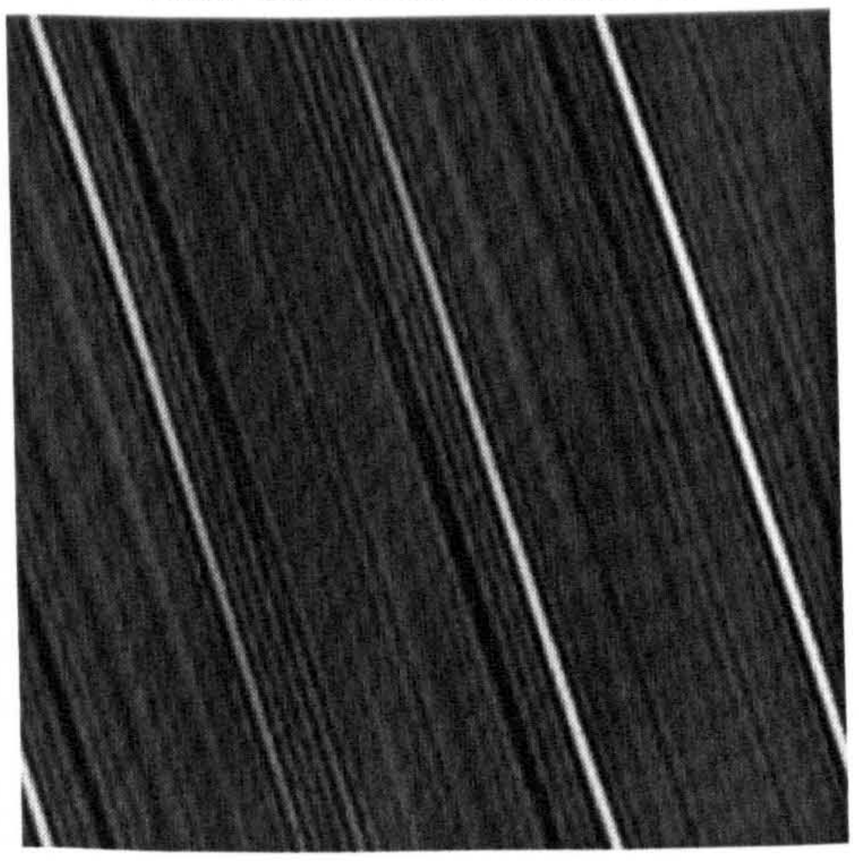
`line` original



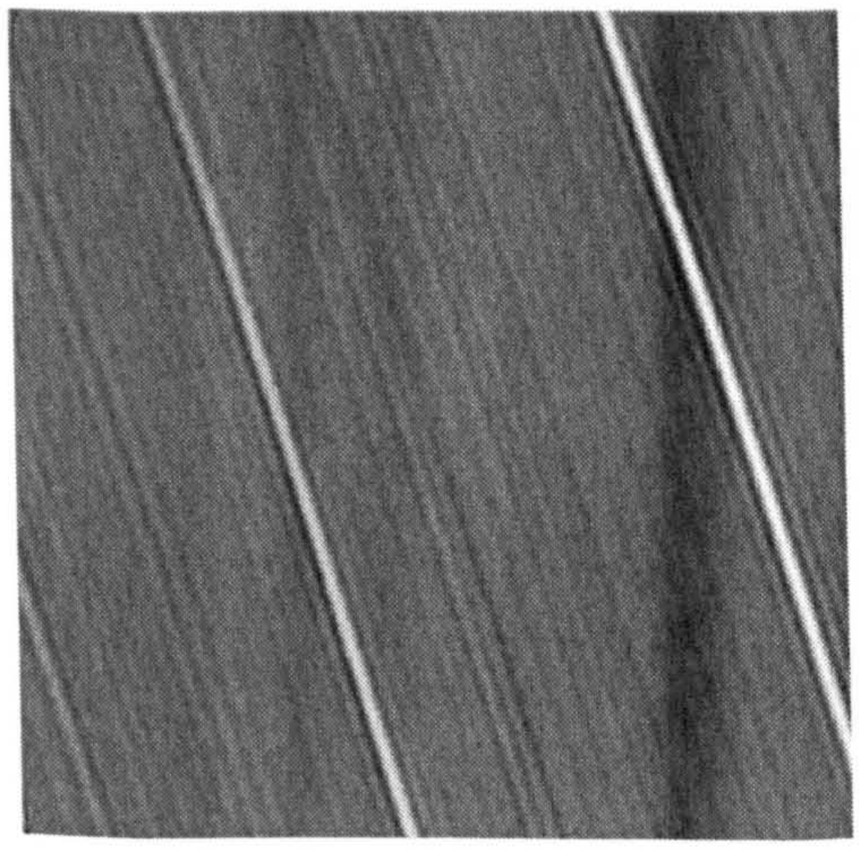
Fast Curvelet Transform



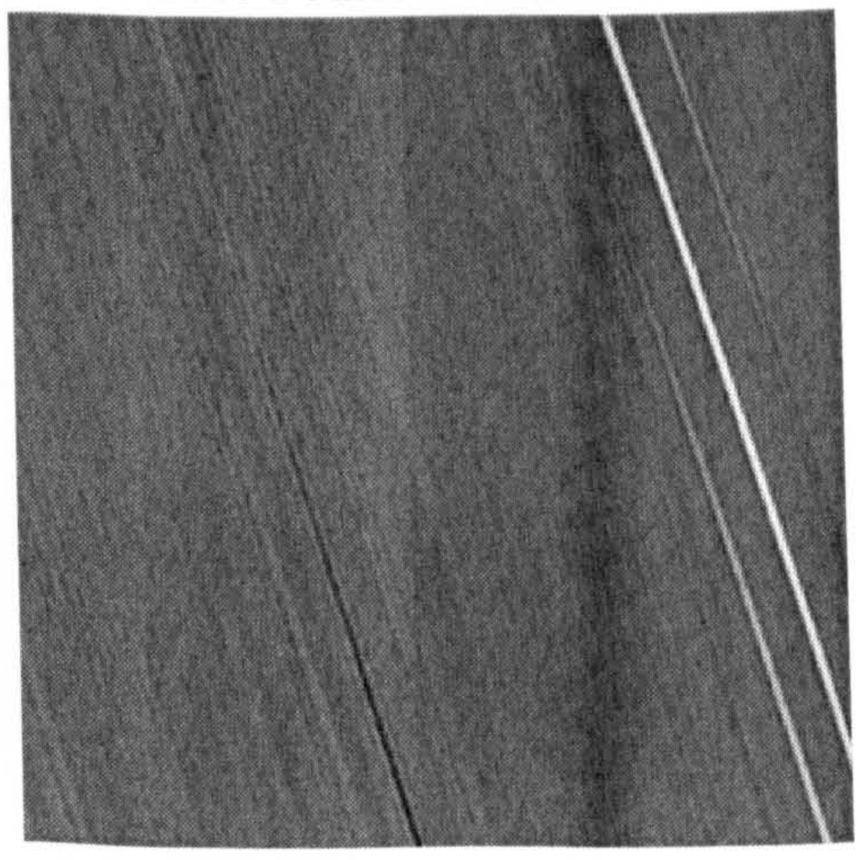
Discrete Cosine Transform



Fast Fourier Transform



PCT-II



PCT-X

Figure 4.7: Illustrative results for nonlinear approximation on `line`

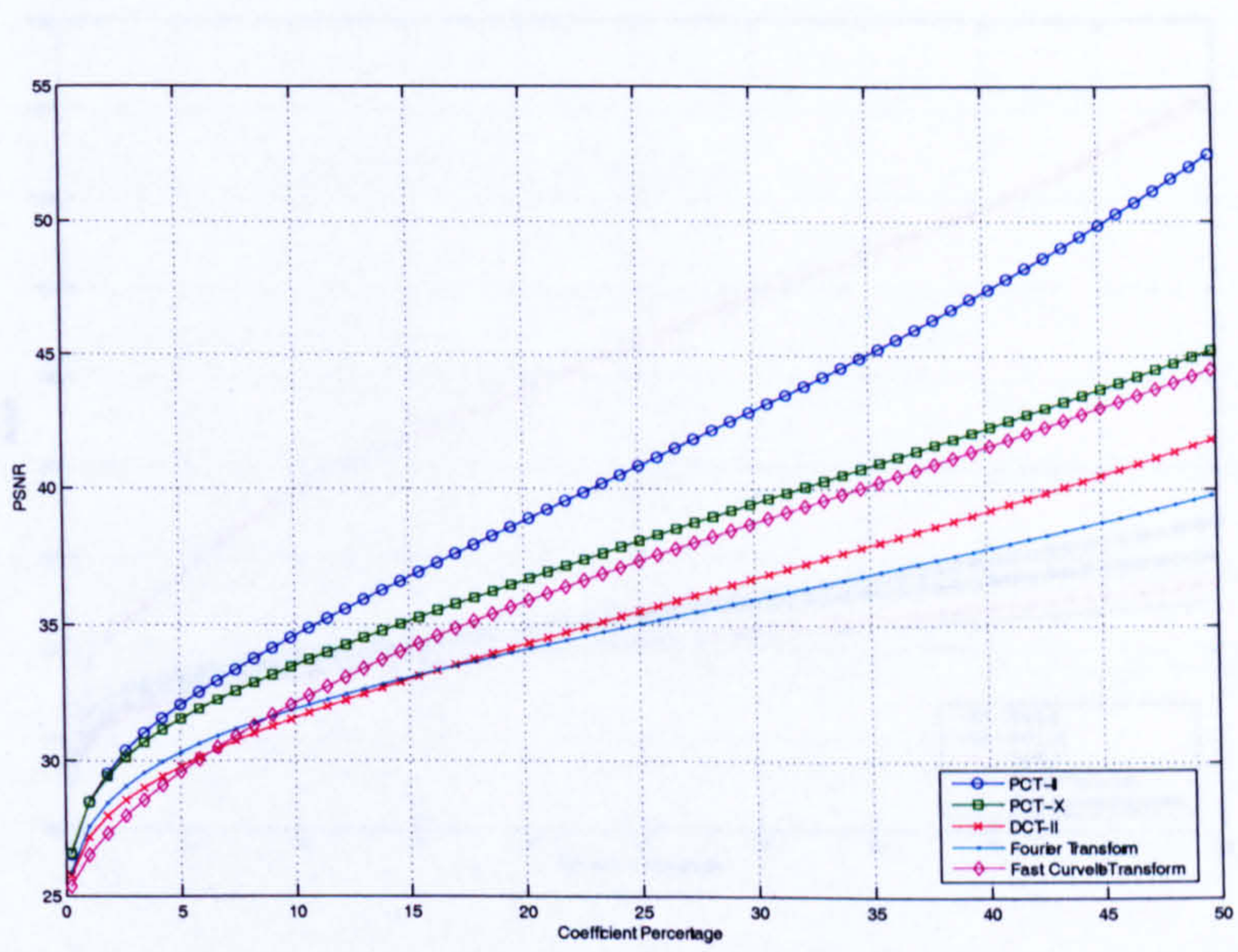


Figure 4.8: Nonlinear approximation PSNR curves for `wood`

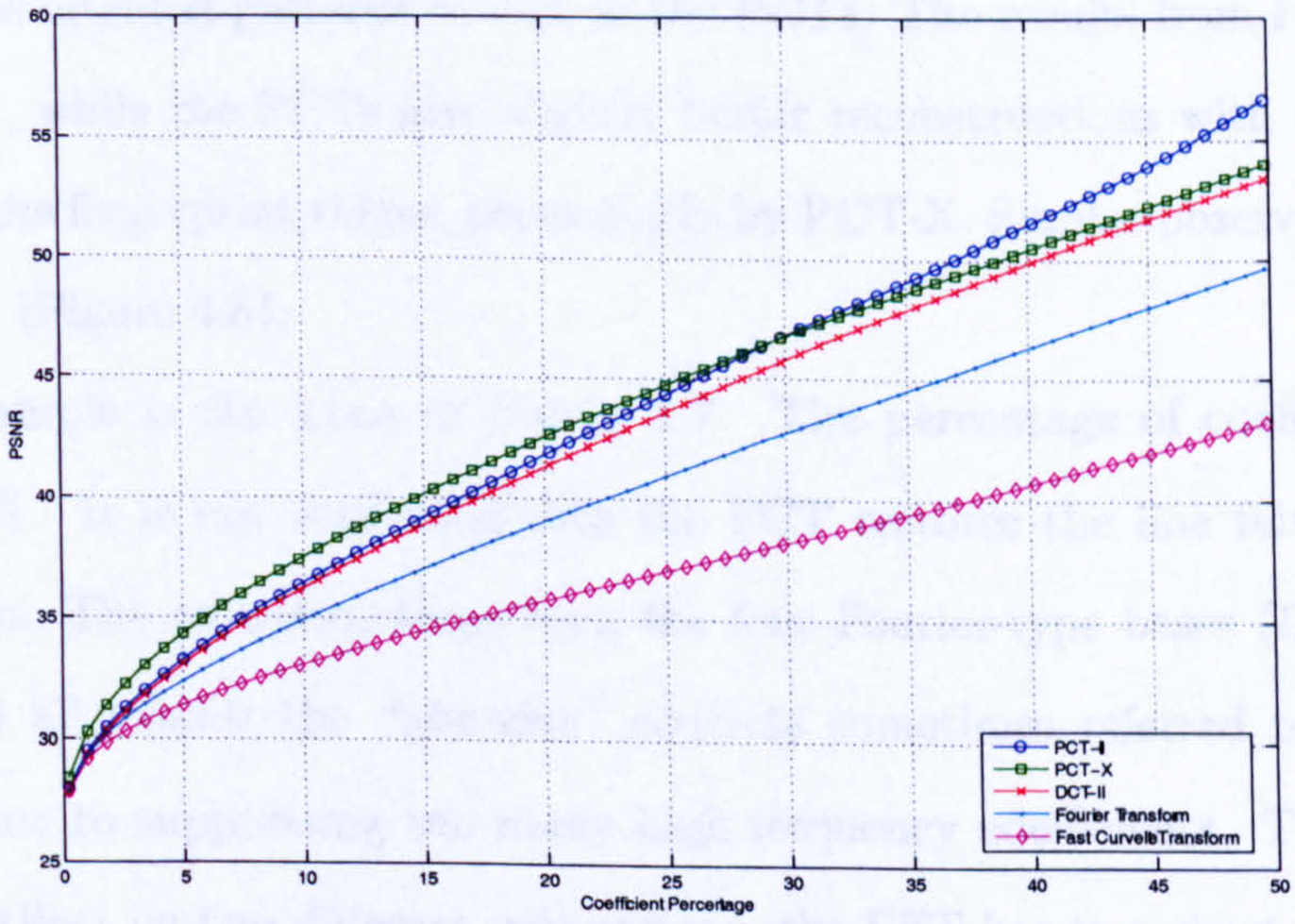


Figure 4.9: Nonlinear approximation PSNR curves for `fp`

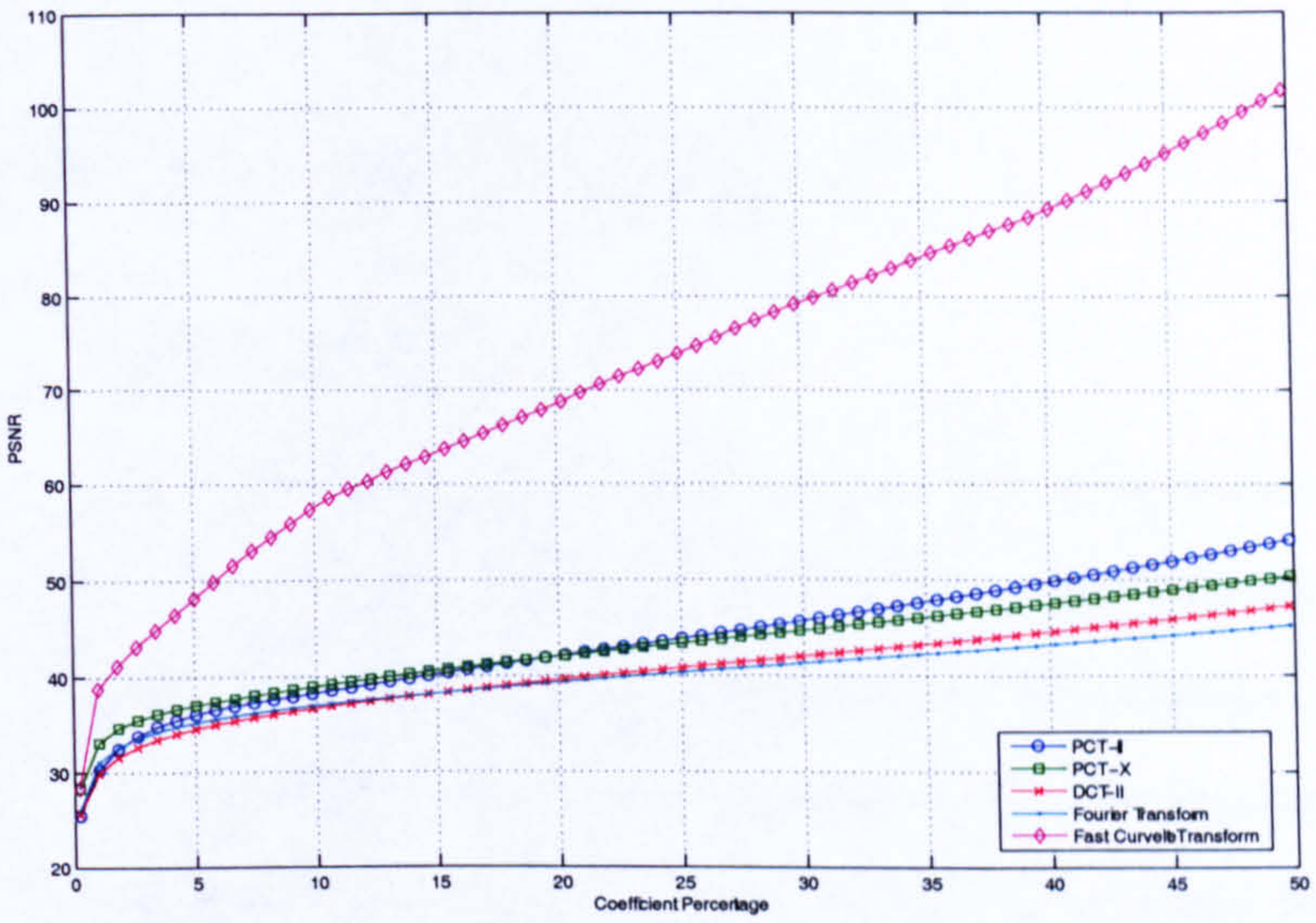


Figure 4.10: Nonlinear approximation PSNR curves for `line`

directional harmonics of the image, an oscillatory pattern has to be described as several directional singularities. The DCT reconstruction, while being closer to the original, does not highlight the oriented patterns as well as the PCTs. The results from FFT and PCTs are very similar, while the PCTs give slightly better reconstructions with the directional regularities on the fingerprint ridges, particularly by PCT-X. Similar observations are seen on `wood` as well (Figure 4.6).

Another example is the `line` in Figure 4.7. The percentage of coefficients is kept as little as 0.1%. It is not surprising that the FCT restores the line faithfully even at such a low ratio. The reconstructions from the four Fourier-type bases (DCT, FFT and the two PCTs) all exhibit the *"ghosting"* artifacts sometimes referred to as the *Gibbs phenomenon*, due to suppressing too many high frequency coefficients. The DCT shows the ghosting artifact on two different orientations, the FFT has two ghost lines appearing on the reconstruction. The PCT-II, on the other hand, due to the boundary extension being assumed on the Radon slices, shows only one significant ghost line. The ghosting

artifact is further reduced in PCT-X's reconstruction.

4.3 The Polar Cosine Packets

4.3.1 Local Cosine Bases

As previously discussed, the Radon transform can reduce the polar cosine transform into a matter of performing the cosine transform on Radon slices. This also makes it possible to analyse directional features other than sinusoidal ridges. This is achieved by taking the transform in separate spatial windows. Local trigonometric bases proposed by Coifman and Meyer [39] and by Malvar [97] use smooth window functions to split the signal and to fold overlapping parts back into the original pieces so that the orthogonality is preserved. This treatment can avoid the discontinuity artifacts caused by a rectangular window, while introducing no redundancy.

The local cosine basis is composed of basis functions of the form

$$\varphi_{p,k}(t) = w_p(t)\sqrt{\frac{2}{|I_p|}}\cos\left[\pi\left(k+\frac{1}{2}\right)\frac{t-c_p}{|I_p|}\right]. \tag{4.28}$$

The Cosine-VI basis is used and modulated by a window function $w_p(t)$ which lies on an interval $[a_{p-1}, a_p]$ with $c_p = (a_p + a_{p-1})/2$, and $|I_p| = c_{p+1} - c_p$ being the length of the window, with the overlapping part included. With a careful choice of the window (see [4]), the set $\{a_p\} \subset \mathbb{R}$ forms a partition of unity and the local cosine basis associated to such partition is said to form a *library of orthonormal bases* usually referred to as the *cosine packets.*

The Polar Cosine Packets can be implemented by placing the 1-D cosine packets on the

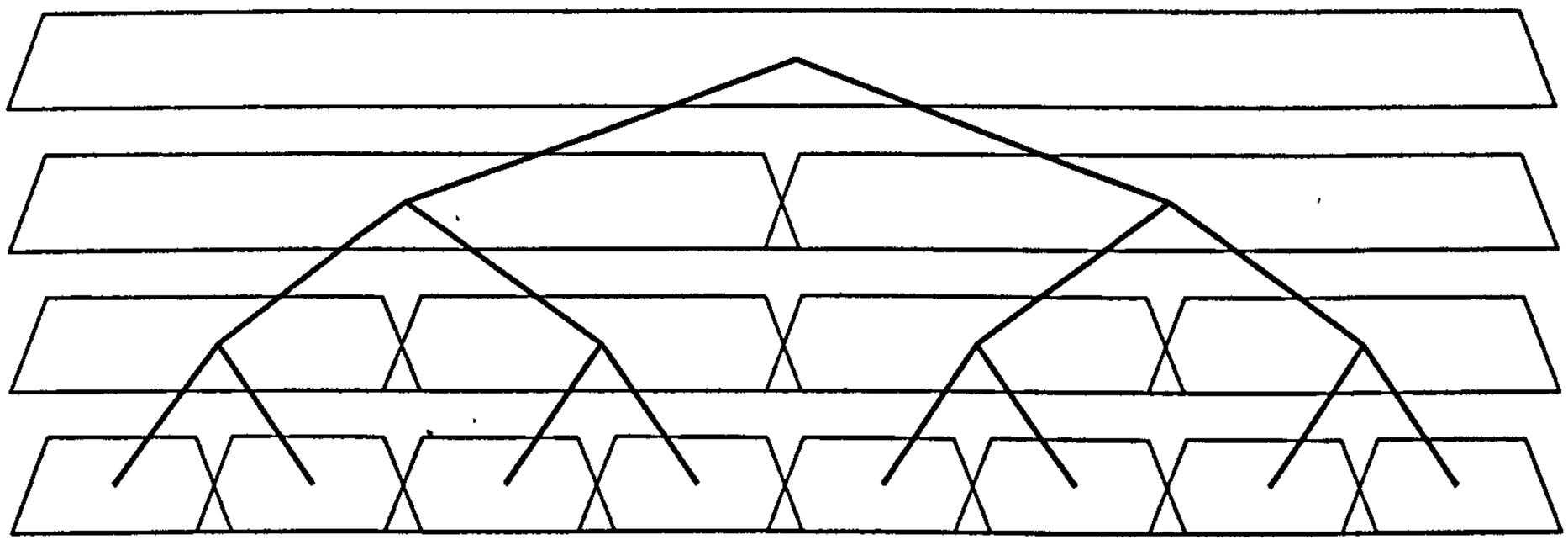


Figure 4.11: The binary tree of the localisation of partitioned intervals

Radon slices; the operator is denoted as

$$\mathcal{P}f[p,k,\theta] = \langle \mathcal{R}f[t,\theta], \varphi_{p,k}[t] \rangle. \tag{4.29}$$

Unfortunately, due to its implementation, it is not possible to construct similar Polar Cosine Packets with the PCT-X.

4.3.2 The Best Basis Selection

The arbitrary choice over the library of local trigonometric bases over a compact interval U is of an extremely large range. To seek a feasible *"best basis"*, the library of cosine packets is reduced to only taking dyadically partitioned decomposition of U only. This organisation is depicted schematically in Figure 4.11. Then I_{00} is a cosine basis on the entire U and $I_p s$ will correspond to the local cosine basis over interval p of the 2^s intervals at level s of the tree. The best basis can be found by induction on s.

Let $B_{p,s}$ denote the cosine basis vectors corresponding to interval $I_{p,s}$, and $A_{p,s}$ be the

best basis. For $s = 0$, the best basis $A_{p,0} = B_{p,0}$, otherwise

$$A_{p,s+1} = \begin{cases} B_{p,s+1} & \text{if } \mathcal{M}(B_{p,s+1}x) < \mathcal{M}(A_{2p,s}x) + \mathcal{M}(A_{2p+1,s}x), \\ A_{2p,s} \oplus A_{2p+1,s} & \text{otherwise.} \end{cases} \tag{4.30}$$

where $\oplus$ denotes a concatenation operation, and $\mathcal{M}(\cdot)$ is a certain cost function.

The resulting best basis is optimal relative to the cost function $\mathcal{M}$, which means different choices of the cost function reflecting certain criterion can result in different selections of the best basis. Such an approach is called *"entropy-based best basis selection"*. For a complete treatment on the subject, see [42].

The best basis can be sought with the local cosine basis on the Radon slices. The resulting transform is called the *Polar Cosine Packet Transform* (PCPT), while the choice of the cost function $\mathcal{M}$ depends on the application.

4.4 The Multiscale Polar Cosine Transform

It was recognised, as early as in 1978 [75] that, for an image representation to be useful, the transform should be spatially well localised. More importantly, the transform should be multiscale, in order to capture patterns of interest at different scales. While the PCT is able to provide a sparse expansion for directional patterns with good frequency resolution, putting it into a multiresolution, spatially localised manner is required. The result is a new wavelet-type transform called the *Multiscale Polar Cosine Transform* (MPCT), which is discussed in the following sections.

4.4.1 The Construction of MPCT

A prototypical multiscale polar cosine function has the form

$$\Psi_{k,\theta,s,\eta}(\xi) = w\left(\frac{\xi-\eta}{s}\right)\psi_k\left(\frac{(\xi-\eta)\cdot\theta}{s}\right). \tag{4.31}$$

where k, θ, η and s denote the frequency, orientation, location and scale parameters of the function respectively and $w(\cdot)$ is the smooth window function chosen along with the sampling interval to ensure invertibility of the discrete form of the transform.

Effectively the transform can be viewed as a stack of windowed polar cosine transforms at a range of scales, with different block sizes for windowing. The coarser the scale s is, the larger the window becomes. The digital implementation of such a *multiscale lapped transform* requires two operators: a *level operator* which decomposes the signal into different scales, and a *local operator* which handles the decomposition locally within a block - in this case could be either the PCT operator $\mathcal{C}$ or the Polar Cosine Packet operator $\mathcal{P}$, which gives us the *Multiscale Polar Cosine Packet Transform* (MPCPT).

A desirable level operator should have the following two properties:

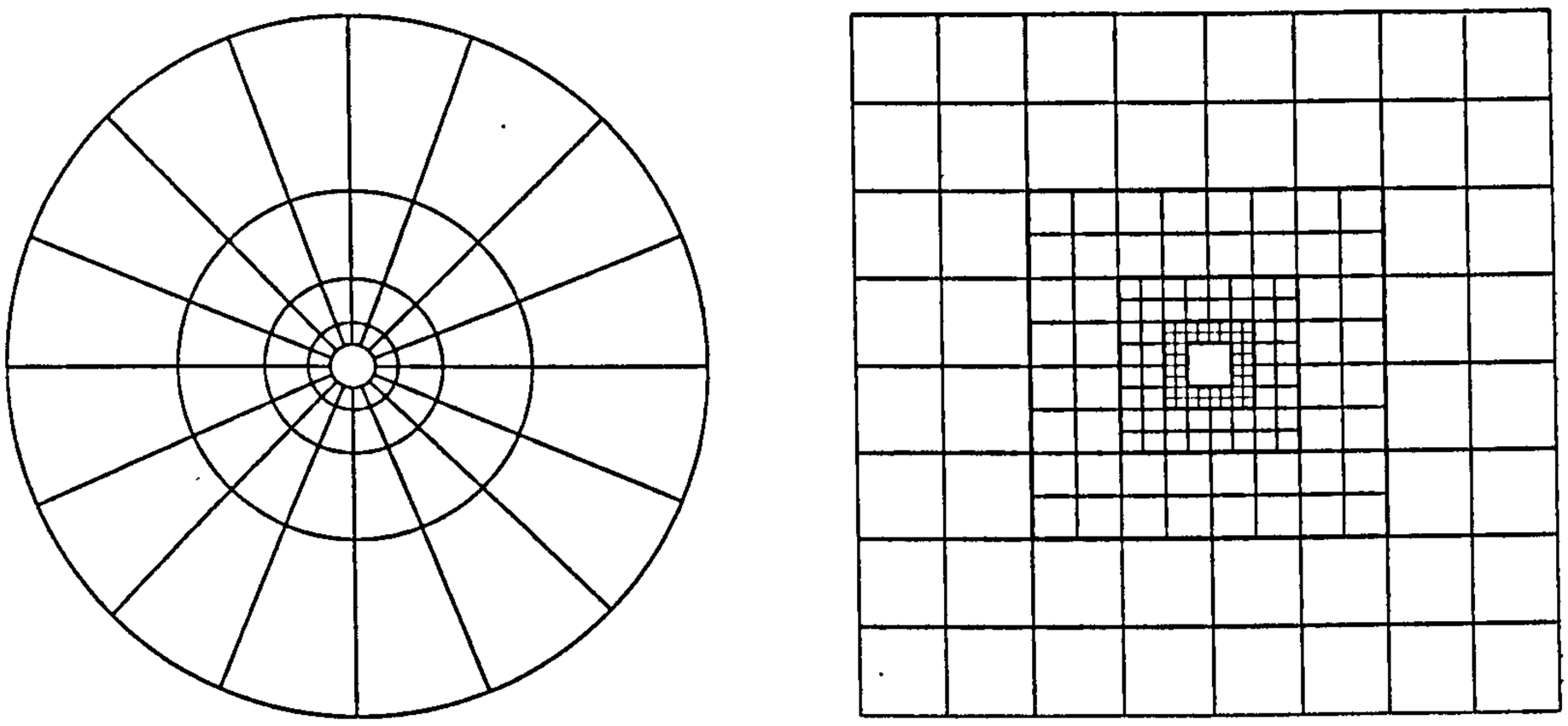
1. The operator should be able to separate the signal into different frequency subbands.

2. The decomposed subbands should be isotropic, which then can be exploited well by the high frequency resolution and the directional selectivity of the PCT.

A reasonable choice of the level operator is the Laplacian pyramid [17]. As in Chapter 3, for a particular level x_s of subband, it is computed as :

$$x_s(\eta) = (\mathbf{I} - \mathbf{G}_{s,s+1}\mathbf{G}_{s+1,s})x'_s. \tag{4.32}$$

where $\mathbf{I}$ is the identity operator and x'_s is the Gaussian lowpass pyramid representation of



Continuous case Discrete case

Figure 4.12: Frequency tilings of the MPCT

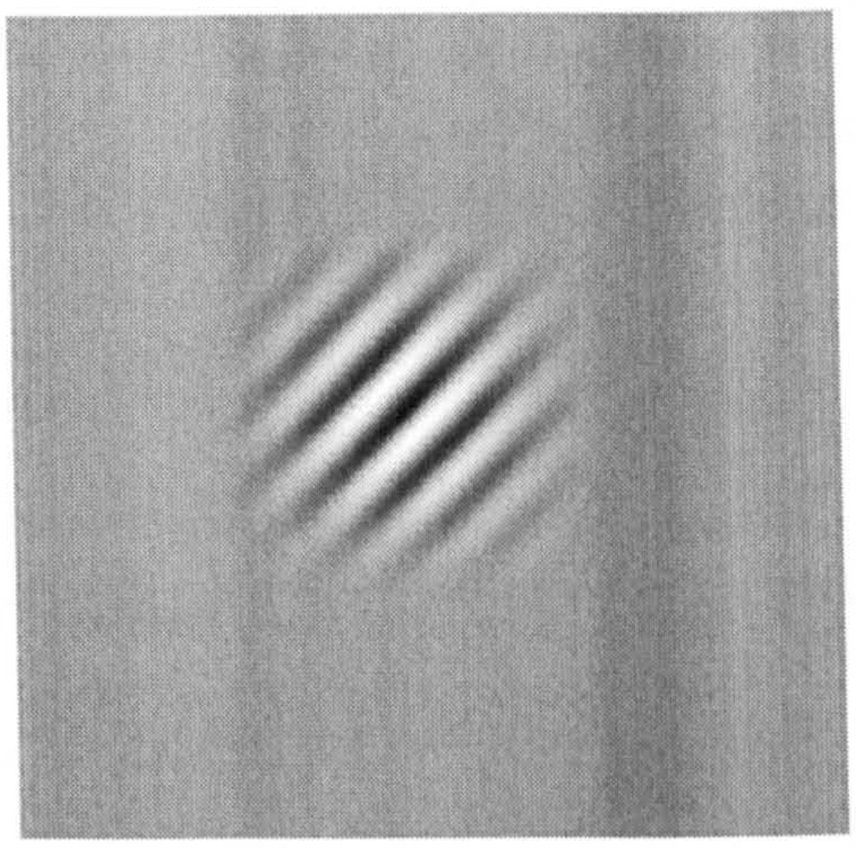
$x(\eta)$

$$x'_s(\eta) = \sum_{l=0}^{s-1} G_{l+1,l} x(\eta). \tag{4.33}$$

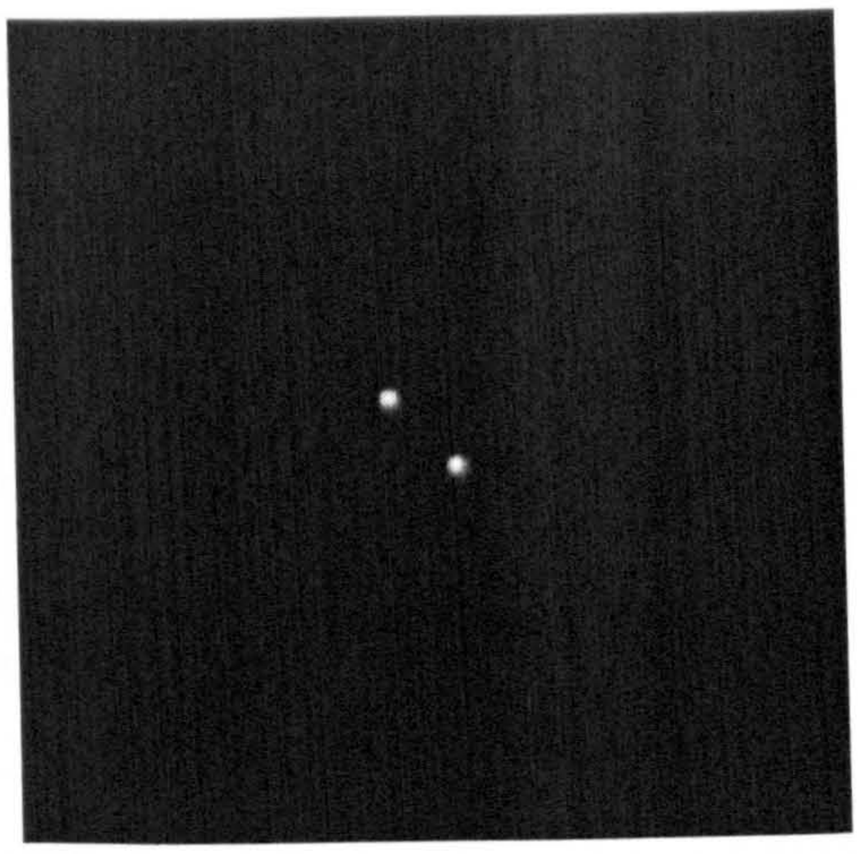
and $G_{s,s+1}$, $G_{s+1,s}$ are the raising and lowering operators associated with transitions between levels in the Gaussian pyramid, as defined previously in Chapter 3. Due to the dyadic decimation, the pyramid is known to have some 33% extra redundancy in 2D, which is acceptable for general use in many image processing tasks. This also allows us to use a constant window size over different scales, which is equivalent to a dyadic increment of the window size.

4.4.2 Basis Functions and Frequency Tiling

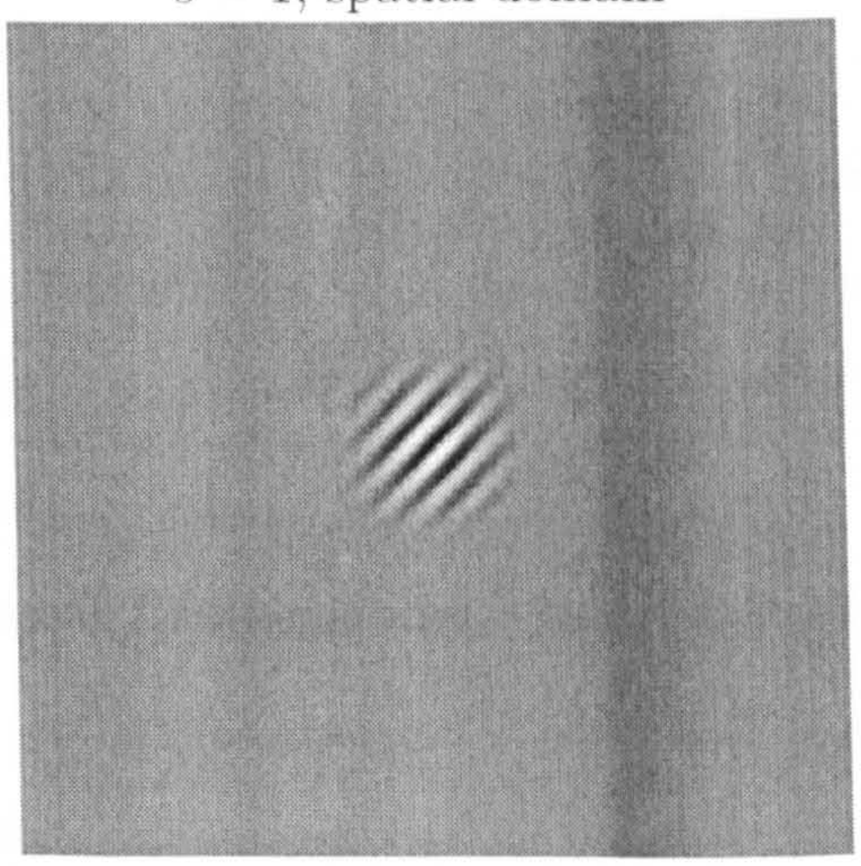
The basis functions of the MPCT are 2D cosine ridges with certain frequency, orientation confined to certain spatial location and scale. Several MPCT basis functions are shown in both spatial and frequency domains in Figure 4.13. These are three members of the MPCT basis at increasing scales on a 256×256 grid. Every function oscillates coherently in a preferred direction and frequency and confined by a cosine-squared window. In the fre-



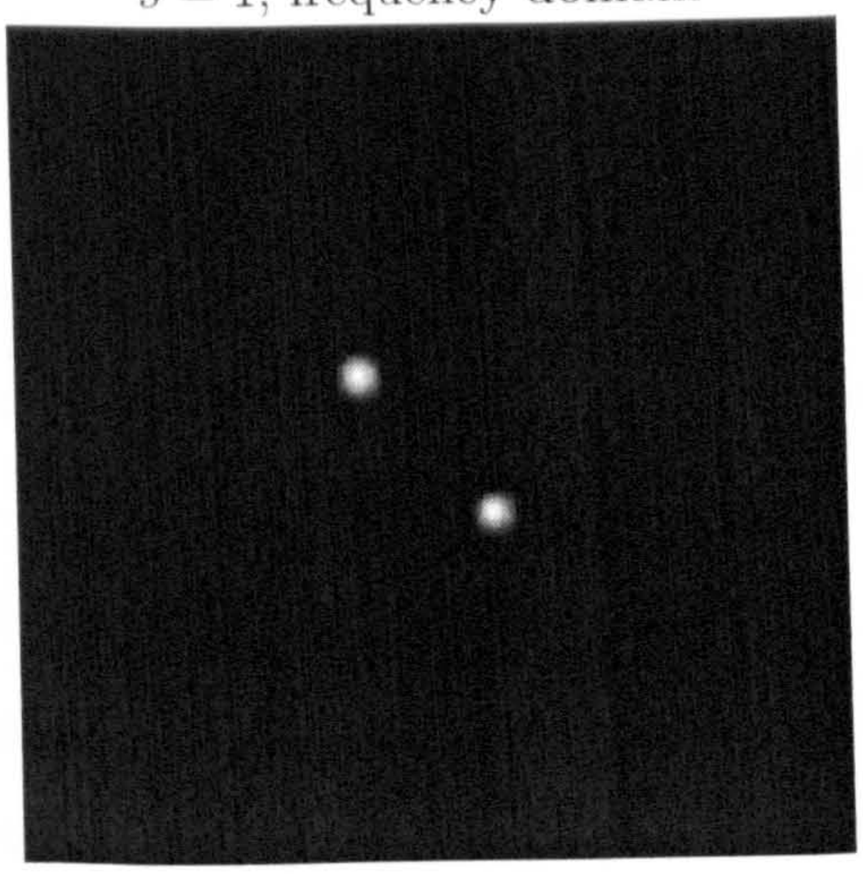
$s = 1$, spatial domain



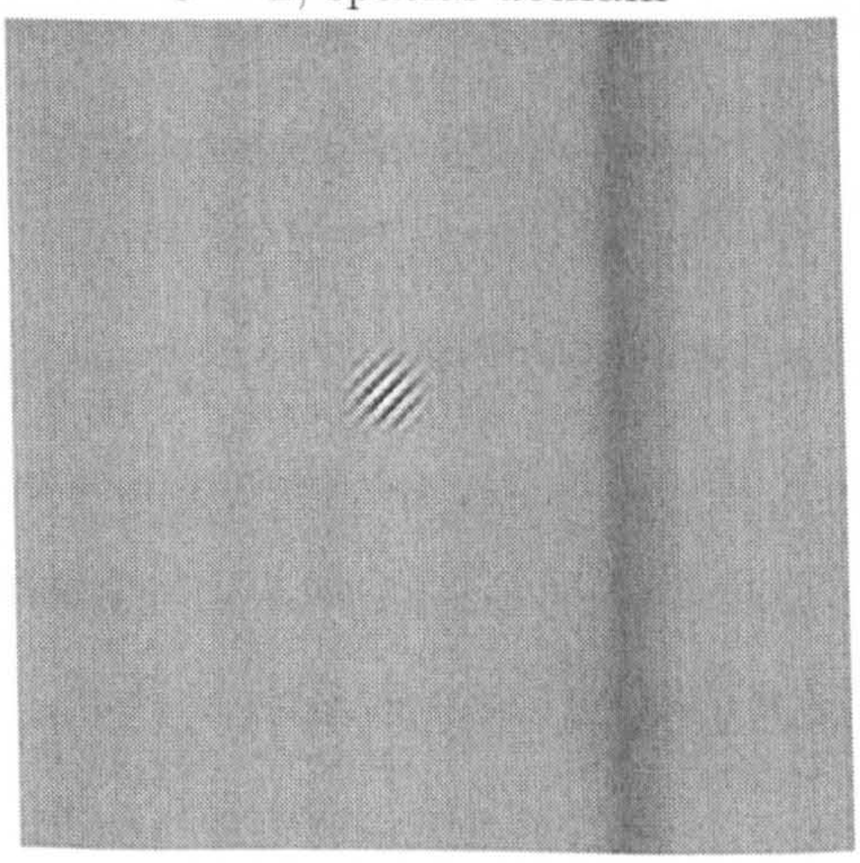
$s = 1$, frequency domain



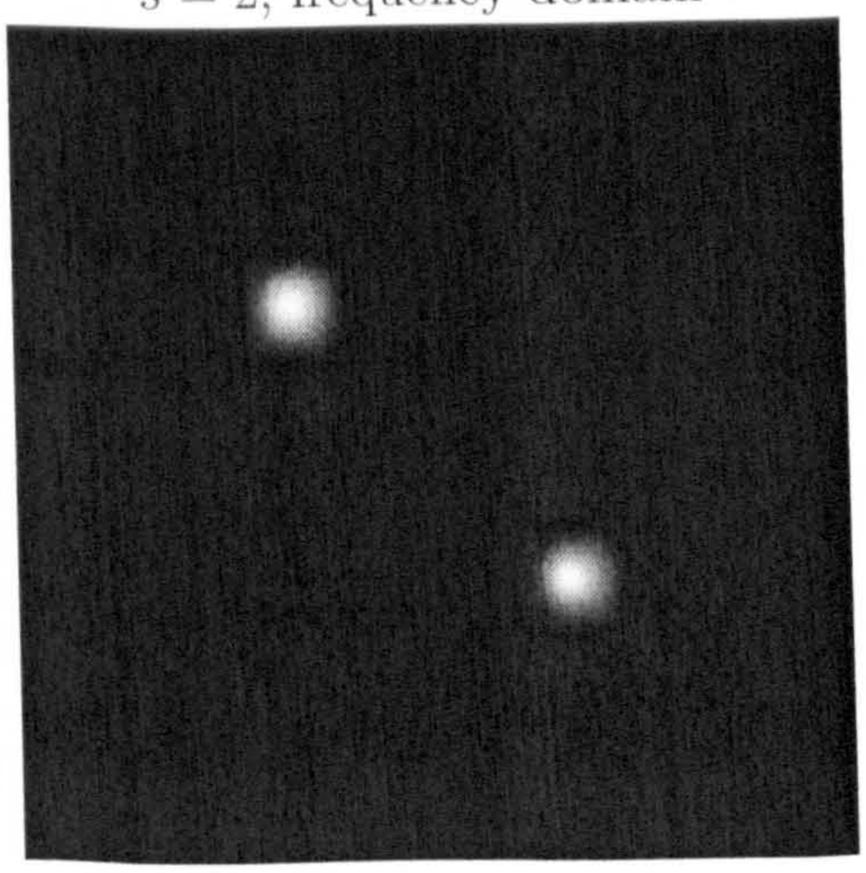
$s = 2$, spatial domain



$s = 2$, frequency domain



$s = 3$, spatial domain



$s = 3$, frequency domain

Figure 4.13: Basis functions at different scales and their frequency responses

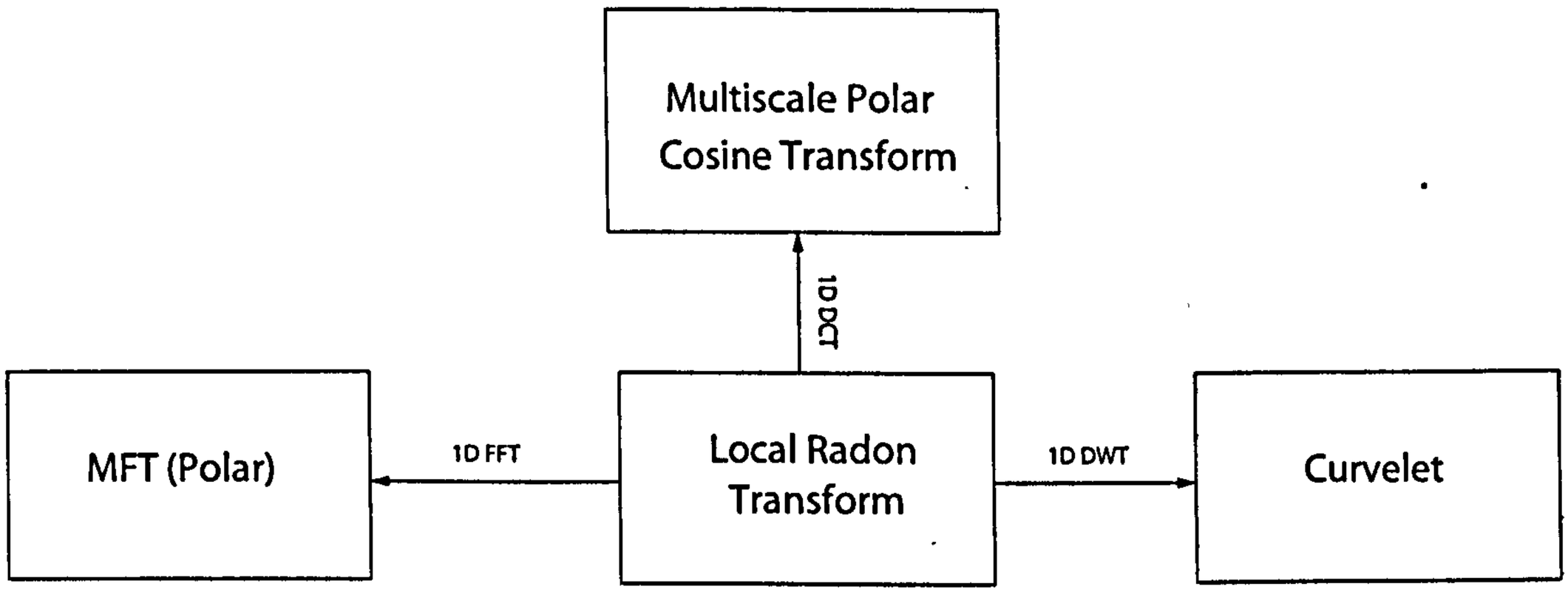


Figure 4.14: The relationships between MFT, Local Radon, Curvelet and MPCT

quency domain, the Fourier transform of these functions is well-localised as well, exhibiting two symmetric blobs at an orientation perpendicular to that of the spatial functions. The size of the frequency blob increases with the scale, while the spatial window size decreases dyadically.

Figure 4.12 schematically demonstrates how the frequency plane is partitioned by the MPCT decomposition, both in continuous and discrete cases. It is clear that the polar orientation resolutions at different scales are the same and each of the subbands from the level operator is partitioned into equal number of boxes.

4.4.3 Relationship with Other Transforms

The construction of the MPCT presented here follows a long legacy in both harmonic analysis and image processing, and is directly connected with many other transforms proposed in the literature.

MFT

The *Multiresolution Fourier Transform* is a windowed Fourier transform at different resolutions and the digital implementation is usually done by a similar lapped block transform like the MPCT with a local FFT operator. Due to the close relationship between PCT and FFT, the MPCT can also be seen as a real version of the MFT, and the assumption of polar symmetric boundary extension increases the sparsity of the transformed coefficients. Unlike the conventional 2D DCT, which is constructed by tensor product of the 1D transform, the local operator PCT of MPCT is a true polar transform, combining the merits of both FFT and DCT into one setting.

Ridgelets and Curvelets

The first digital curvelet transform, the so-called *curvelet-99* implementation proposed in [143] also adopts a multiscale lapped block transform approach. The level operator used in that specific transform was an undecimated *á trous* wavelet transform. While the redundancy is useful in the denoising task, substituting it with a less redundant operator like the Laplacian pyramid would still retain the curvelet notion. The local operator is the digital ridgelet transform which is a 1-D wavelet transform on Fast Slant Stack Radon slices. Therefore, the essential difference between the curvelet-99 transform and the MPCT transform is just the 1D transform performed on Radon slices: one being the wavelet and another being the cosine transform.

In [70], implementations of possible digital *ridgelet packets* were discussed. One of these uses a basis on the Radon domain from a wavelet packet or cosine packet dictionary. This coincides with the polar cosine packets discussed before, which can be used to deal with features like both ridgelet and polar cosine basis functions.

It is clear that the Fourier Slice Theorem is the fundamental tool in implementing the Radon transform, and it links the curvelets, polar cosine and Fourier transforms. The

relationship between those multiscale lapped transforms can be illustrated schematically as in Figure 4.14.

Brushlets and Wave Atoms

The brushlets, proposed by Meyer and Coifman [106], partition the Fourier frequency plane by local cosine windowing, achieving basis functions of directional oscillating patterns localised in orientation, frequency, location and scale. However, due to the nature of frequency implementation and the choice of the window, the basis functions are not well localised in space, having significant spread. The recently introduced 2D wave atoms frame [49] as implemented by using the Villemoes wavelet packets [152] in the frequency domain. The resulting basis functions are much like brushlets, but with only two bumps in the spatial domain.

The MPCT's basis functions are in essence same as the brushlets and Wave Atoms, but the transform is implemented by spatial windowing instead of frequency windowing. The MPCT and these two transforms are spatial-frequency duals of the same idea, like wavelet packets and the local trigonometric bases.

4.5 Image Denoising Experiments

In order to demonstrate the effectiveness of the MPCT transform, the transform is applied to the task of noise removal in 2D by thresholding the transformed coefficients (cf. Chapter 2).

The MPCT implementation used in the denoising has the following configuration:

1. The level of decomposition, or the total number of scales is set to $J = 5$.

2. The local window is 16×16, modulated with a squared cosine.

3. The windows are overlapped by 50%.

4. The threshold for each scale is computed as :

$$\mathcal{T}_s = a\frac{\sqrt{2\log M^2}\sigma}{1.23^{J-s}}. \tag{4.34}$$

with a set to $a = 0.08$ for the MPCT and $a = 0.062$ for the MPCPT. In the above equation, σ is the standard deviation of the noise. The best basis is computed according to the cost function [89]:

$$\mathcal{M}(f, \mathcal{T}_s) = \sum_{i=1}^{N} \Phi\left(|\langle f, \psi_k[i]\rangle|^2\right). \tag{4.35}$$

where

$$\Phi(u) = \begin{cases} u - \sigma^2 & \text{if } u \leq \mathcal{T}_s^2 \\ \sigma^2 & \text{if } u > \mathcal{T}_s^2. \end{cases} \tag{4.36}$$

4.5.1 Candidates for Comparison

In the experiment, the results are compared with four other candidates:

- **Wavelet Packets**

  An orthonormal adaptive wavelet packet (WP) transform can capture point singularities and oscillatory patterns. It is a long-standing choice for noise removal and research on the threshold selection is well established. The threshold for certain subband H in the WP transform used here is known as the S-Bayes [133], calculated as:

$$\mathcal{T}_H = \sqrt{\log\frac{N}{J}}\left(\frac{\sigma^2}{\sqrt{\max(\sigma_H^2 - \sigma^2, 0)}}\right). \tag{4.37}$$

  where N is the total number of data points in H subband and J is the total level of decomposition, and σ_H^2 is the variance of the H subband.

In order to suppress the pseudo-Gibbs Phenomena in the wavelet-baed denoising, the *cycle spining* [40] treatment or *translation invariant denoising* is used, with the number of spins being equal to the filter length, in this case, the 8-point Symmlet. This is equivalent to a wavelet shrinkage with an undecimated basis.

- **Curvelets**

  The curvelet-99 transform was proposed in [143] for image denoising. The curvelet transform used here is a variant of the original curvelet-99, with the undecimated wavelet transform replaced by the Laplacian pyramid, which is also used in the MPCT. Locally, the same 16×16 cosine squared window with 50% overlap is used, so the redundancy factor of the two transforms is the same, making their results more comparable. The threshold for different levels are the same as used in the MPCT denoising.

- **Wave Atoms**

  Wave atoms frame has similar basis functions to the MPCT. As reported in [49], the denoising is done by simply setting the threshold to

$$\mathcal{T} = 3\sigma. \tag{4.38}$$

The software implementation of these schemes are all in Matlab. The Wave Atoms implementation and denoising routine is from the `WaveAtom` package[1].

4.5.2 Test Data

The dataset used in the denoising experiments consists of four images: `barbara` contains some directional and non-directional periodic textures; `lena`, which can be regarded as

[1]The `WaveAtom` package can be downloaded from `http://www.waveatom.org`.

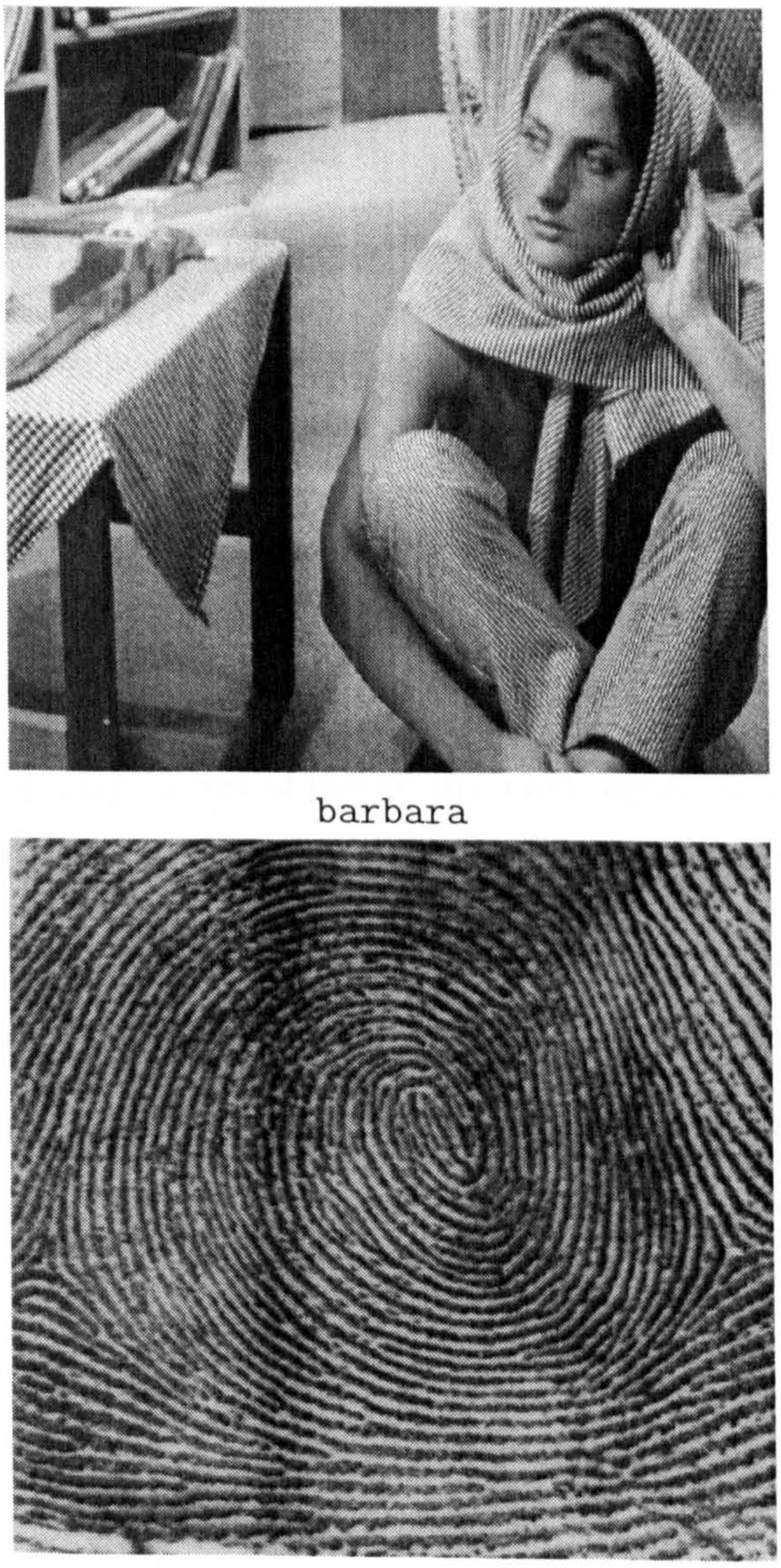


Figure 4.15: A set of four images for denoising experiments

one of the "curvelet-friendly" images, since it mainly consists of linear singularities at different scales; `fp` is a fingerprint image and finally `grain` image is a texture image which was considered to be very difficult to compress since it contains some oriented oscillation mixed with random linear singularities. These four images are given in Figure 4.15.

4.5.3 Denoising Results

The experimental results on noisy images from 0dB to 20dB measured in SNR are presented in Table 4.3, and Figures 4.19-4.21 contain denoised results by different methods which can help to establish a qualitative comparison. It is clear from the detailed images in Figures 4.20 and 4.21 that although the translation-invariant WP approach outputs SNR measures comparable to other directional transforms, the visual sharpness of its output is not as pleasing as rest of the group, have significantly blurred edges. Artifact problem exists with the wave atoms denoised images. Although it performs well with textural images such as `fp` and `grain` which are supposed to be its forte, on natural images its visual quality is far from satisfactory, mainly due to the fact that its basis functions are not well localised in space and due to its inability to model edges well.

The curvelet transform, MPCT and MPCPT are all constructed by block-based local transforms on top of the Laplacian pyramid. It is surprising to see that the MPCT gives consistently better numerical results than curvelets, as well as well-matched visual results. Although the underlying local operator has an oscillatory pattern instead of anisotropic ridge forms, the overlapping blocks effectively compensate for the drawback in edge areas by confining the oscillations locally. In textural regions, the MPCT shows significant gains over the curvelets. The MPCT packet denoising, as expected, combines the merits of both curvelet transform and the MPCTs into one setting. In order to further illustrate their effectiveness, subband reconstructions of these transforms are given in Figures 4.16-4.18, which are obtained from thresholding the 10dB `barbara` image in the respective transform

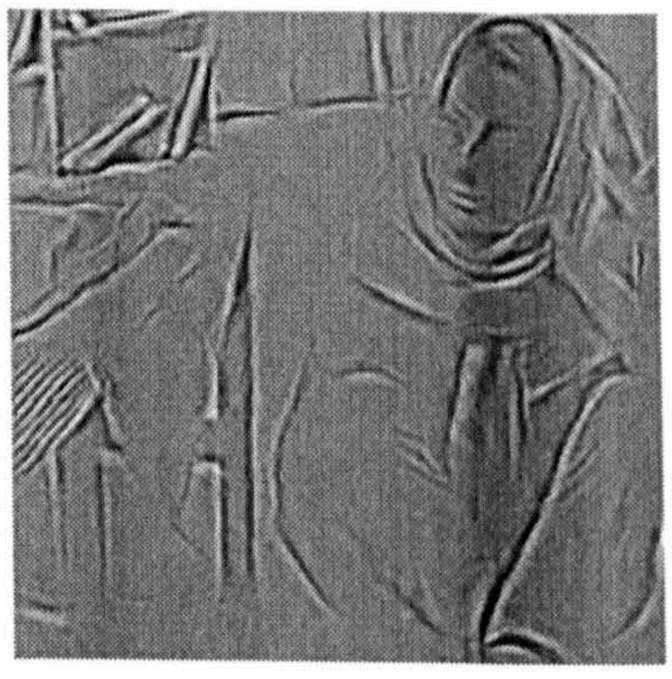
Table 4.3: The comparative image denoising results in SNR

Image	Noise (dB)	TI S-Bayes	Curvelet	MPCT-II	MPCT-X	MPCPT	Wave Atom
barbara	0	14.75	14.59	14.89	14.63	**15.19**	13.38
	5	16.08	15.93	16.42	**17.17**	16.85	16.40
	10	18.00	17.64	18.44	18.75	18.85	**19.22**
	15	21.12	19.64	20.34	20.46	20.71	**22.01**
	20	24.92	21.41	21.88	21.04	22.10	**24.89**
lena	0	17.06	17.20	17.24	16.59	**17.75**	14.47
	5	18.96	18.82	18.99	18.93	**18.99**	17.60
	10	21.08	20.93	21.05	21.03	**21.89**	20.36
	15	23.49	23.30	23.35	22.88	**24.26**	23.21
	20	26.15	25.45	25.60	25.17	**26.31**	26.08
fp	0	12.20	12.03	12.83	12.82	**13.00**	12.20
	5	15.18	13.48	14.39	**15.17**	14.60	14.64
	10	**18.29**	15.40	16.63	17.19	17.15	17.44
	15	**21.44**	18.45	19.59	19.27	20.24	20.52
	20	24.74	21.66	22.56	22.44	23.23	**26.08**
grain	0	12.93	13.06	13.10	12.77	**13.22**	11.93
	5	13.81	13.66	13.84	14.00	**14.29**	13.96
	10	15.87	15.01	15.51	16.09	16.18	**16.35**
	15	18.72	17.28	18.00	18.78	18.84	**19.02**
	20	22.02	20.05	20.56	20.43	21.38	**22.02**

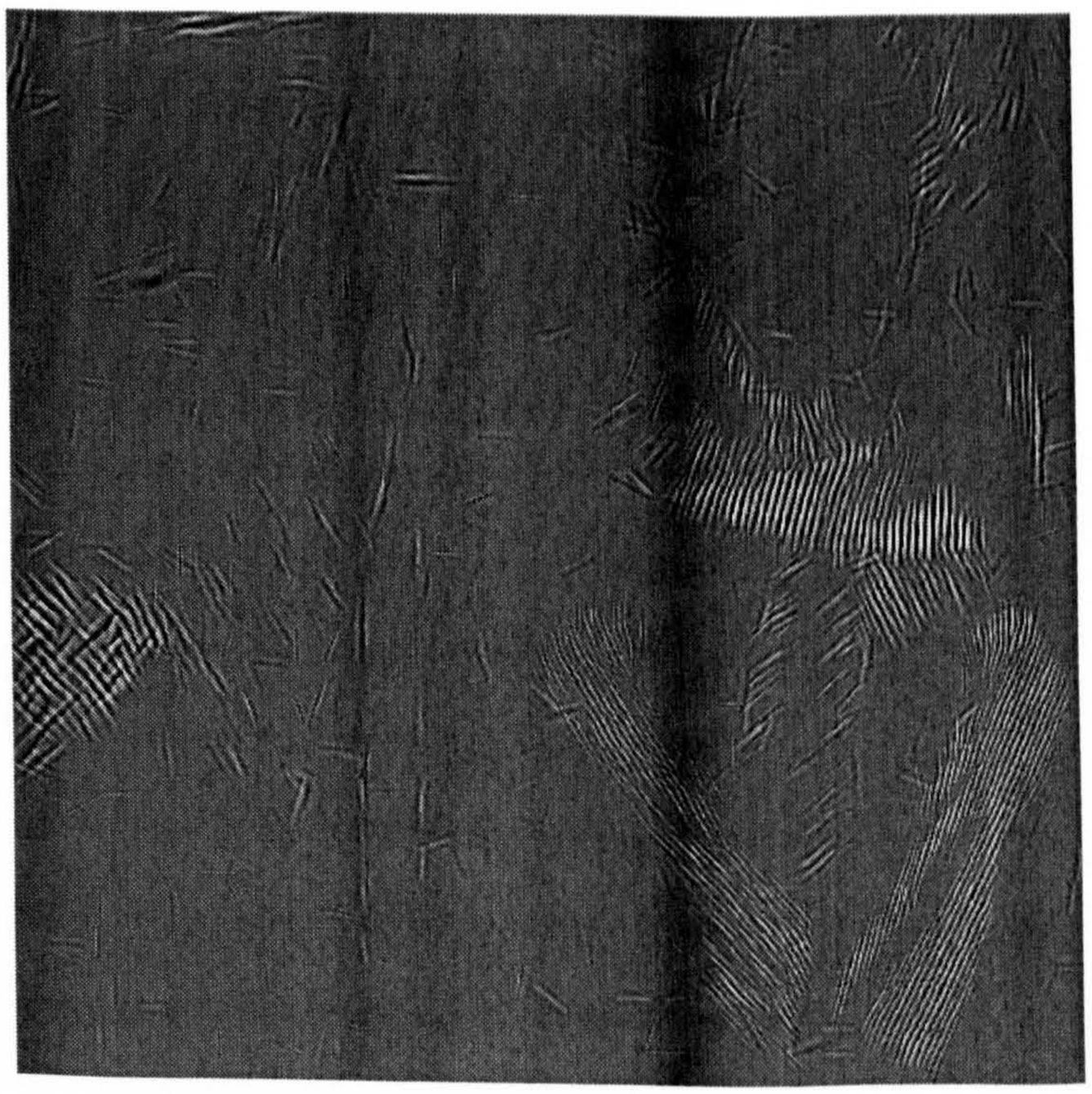
domain. The reader is encouraged to compare the sharpness of recovered edges and textures in these results.

The difference in the numerical results between the MPCT-II and MPCT-X is compatible with the nonlinear approximation results in section 4.2.5. The MPCT-X generally performs better with heavier noise, and MPCT-II can outperform MPCT-X by a significant margin when there is less noise. However, visually, the two implementation do not differ significantly. Overall, the MPCT Packet can be regarded as the winner for its better visual quality as well as numerical error measures. The wave atoms based thresholding gives the worst denoising result visually, although its reconstruction gives very high SNR measure.

$s = 2$



$s = 3$

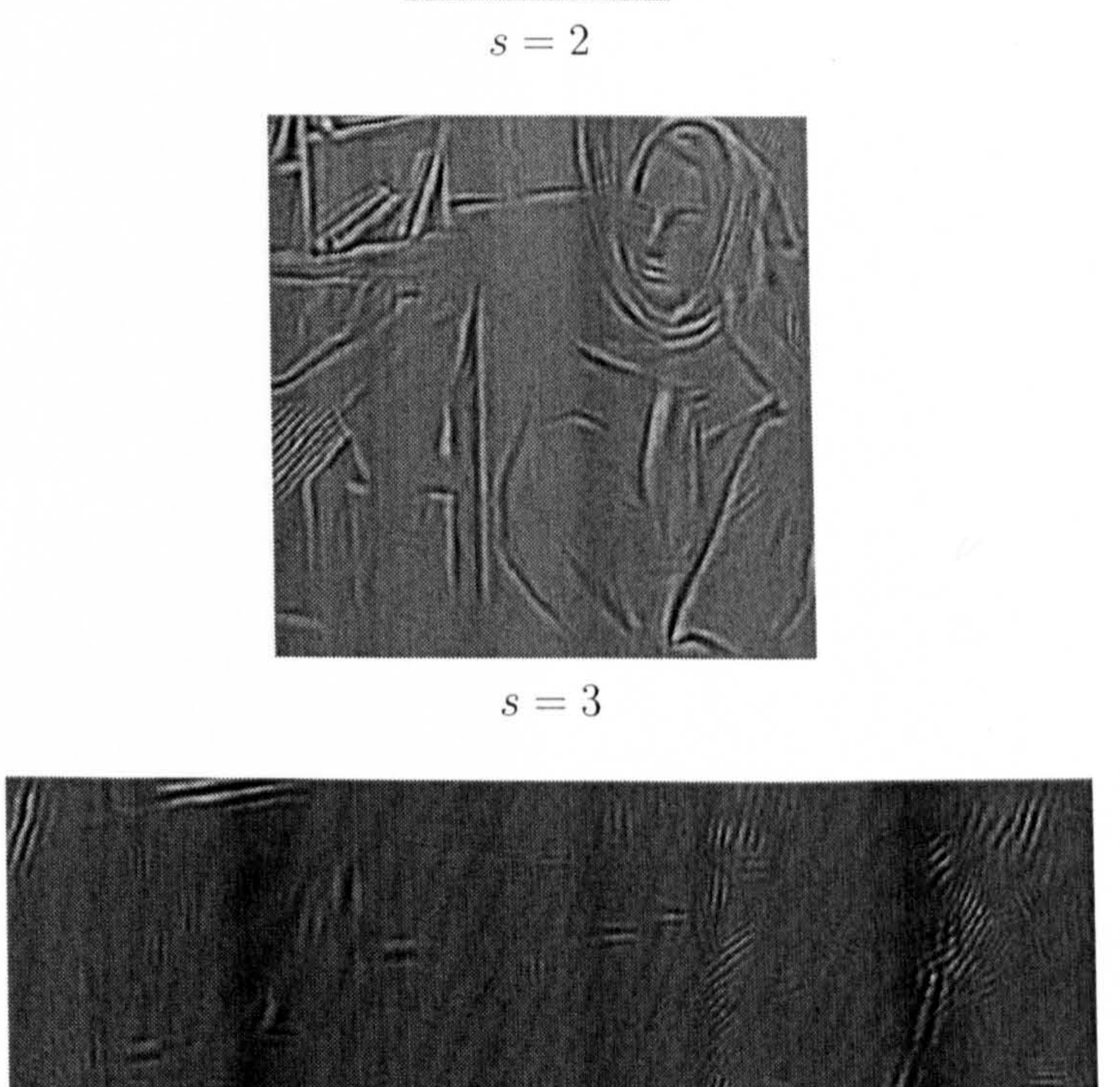


$s = 4$

Figure 4.16: Resulting pyramid by curvelet denoising on `barbara`

$s = 2$

$s = 3$

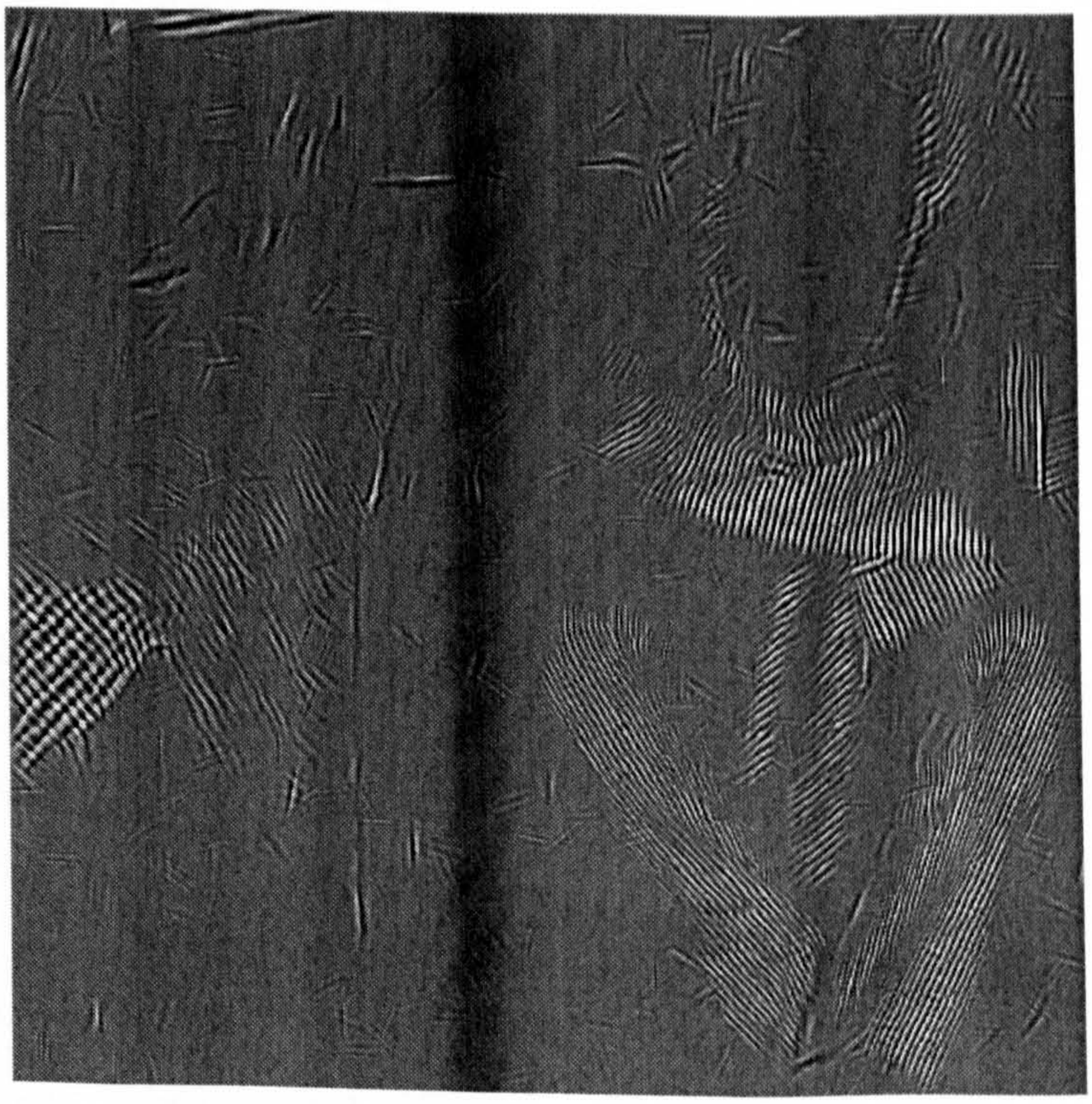


$s = 4$

Figure 4.17: Resulting pyramid by MPCT-II denoising on barbara

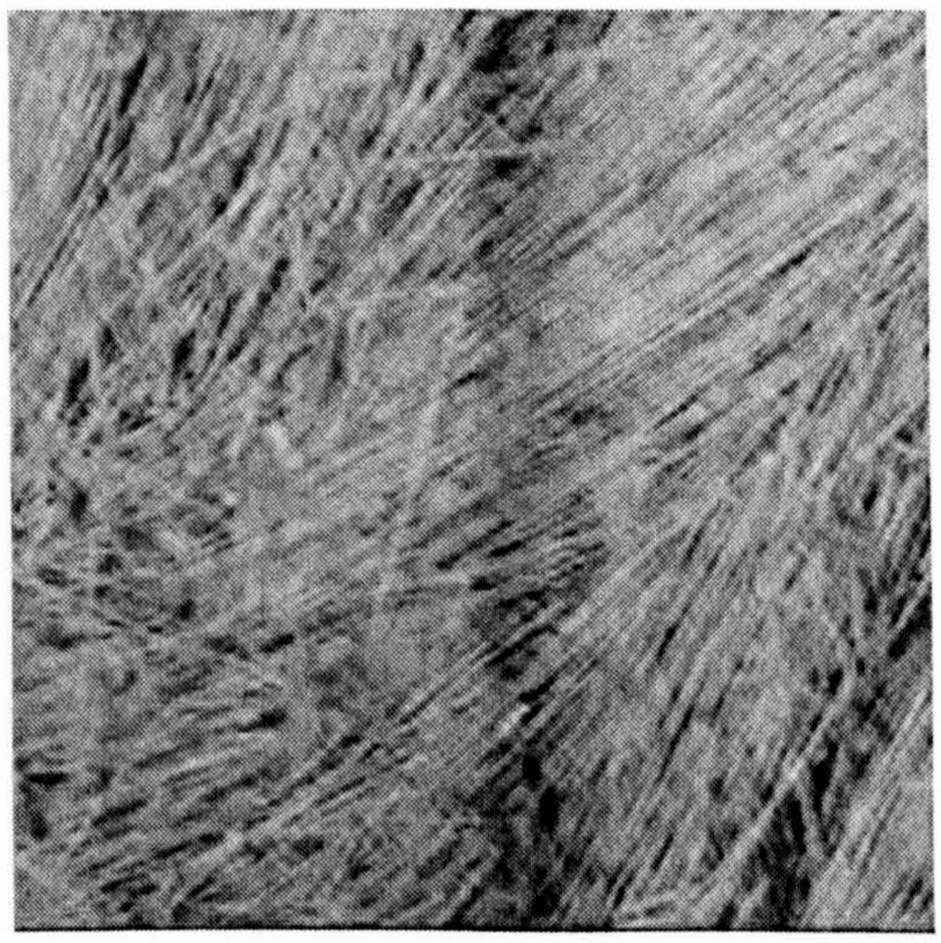
$s = 2$

$s = 3$

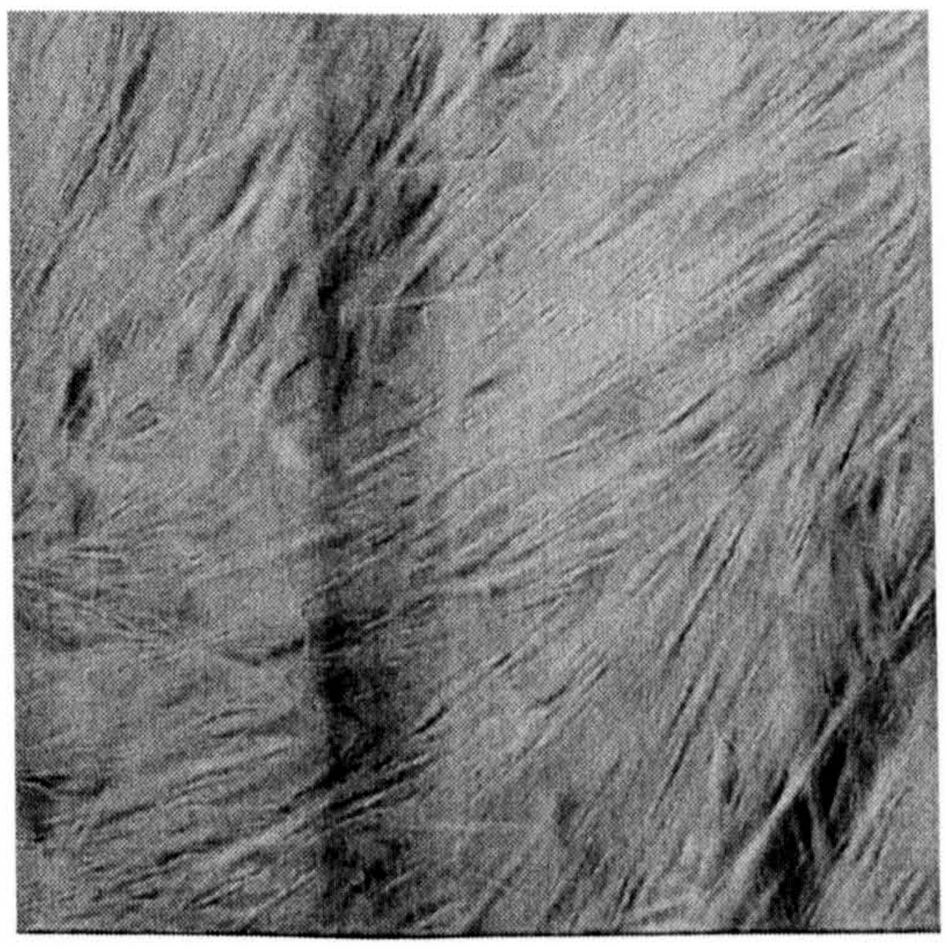


$s = 4$

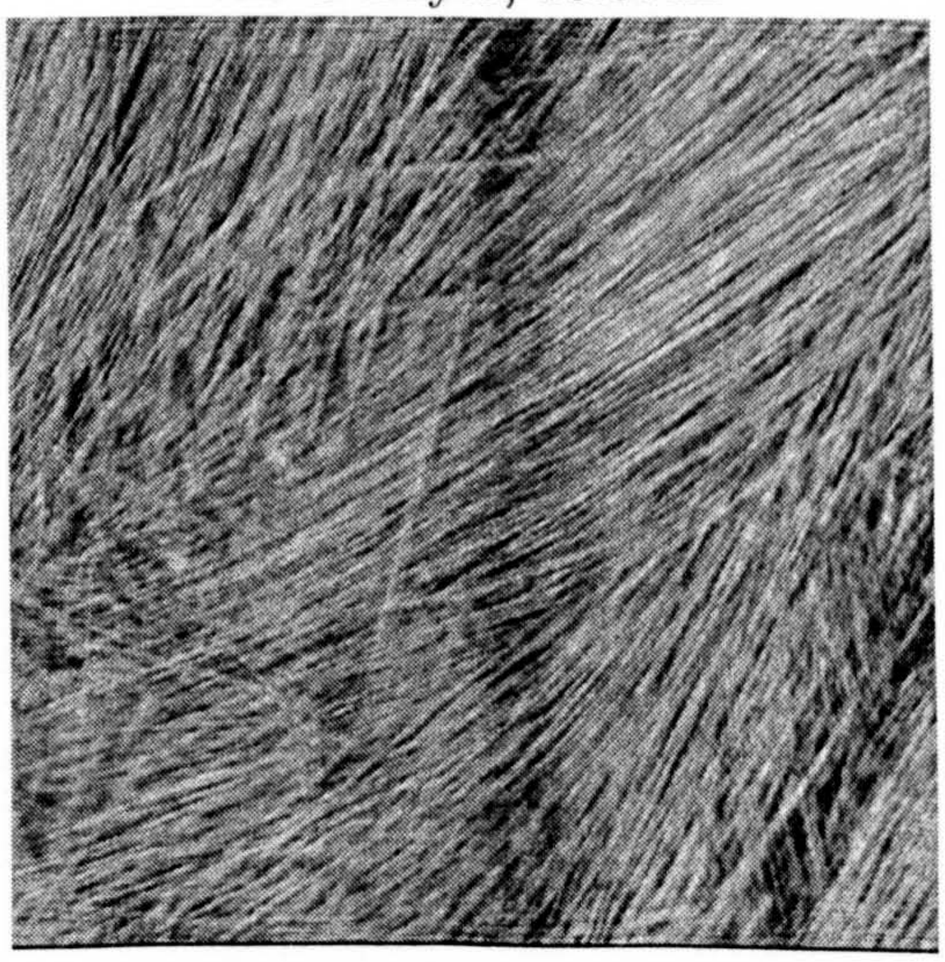
Figure 4.18: Resulting pyramid by MPCPT denoising on `barbara`



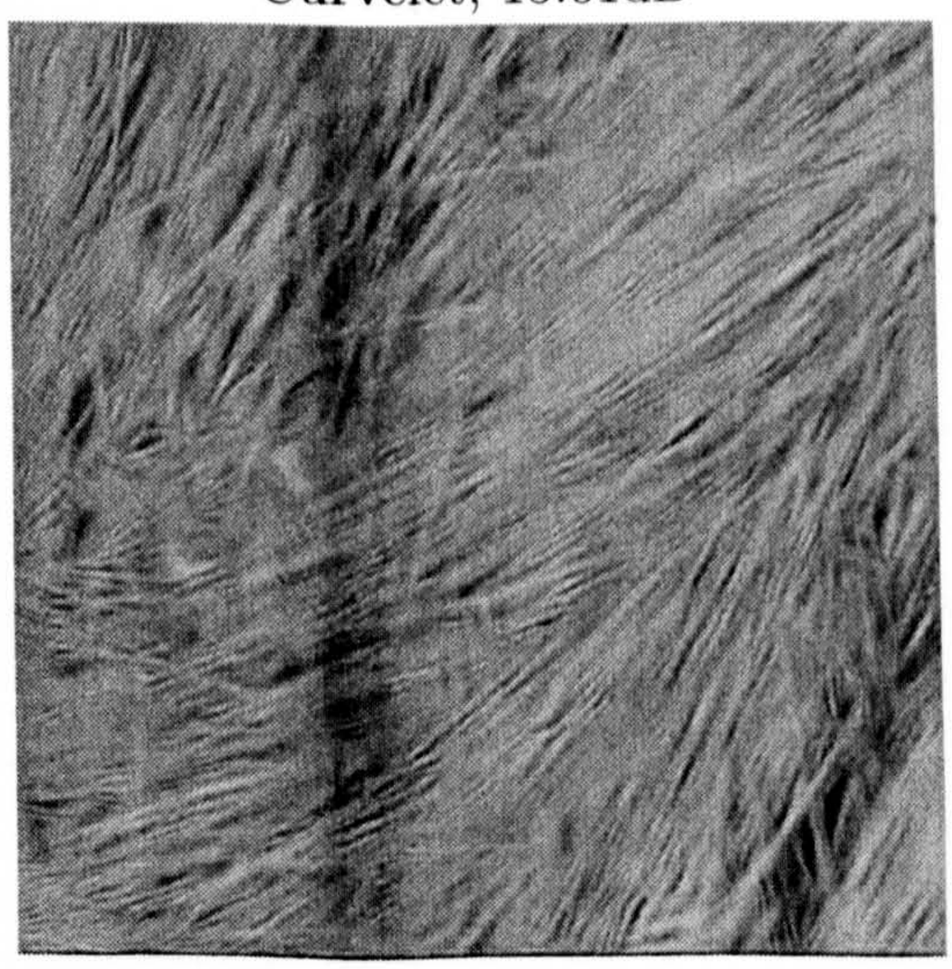
WP $\mathcal{S}$-Bayes, 15.87dB



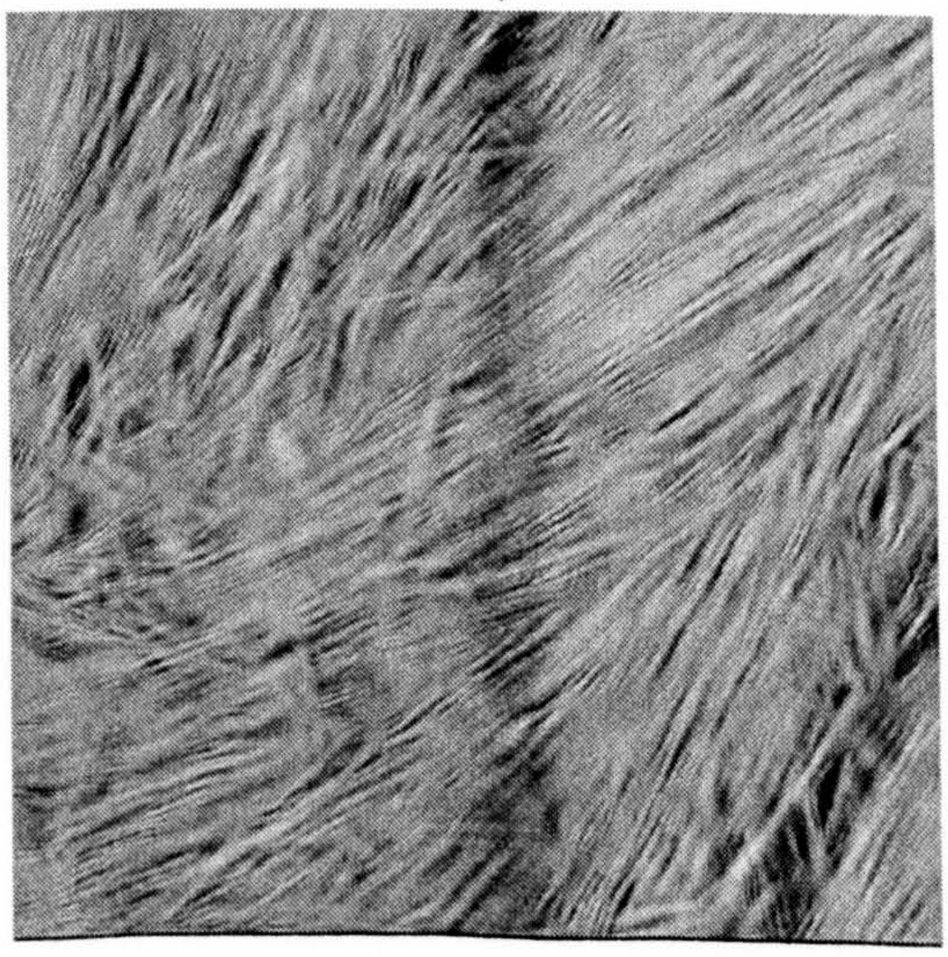
Curvelet, 15.01dB



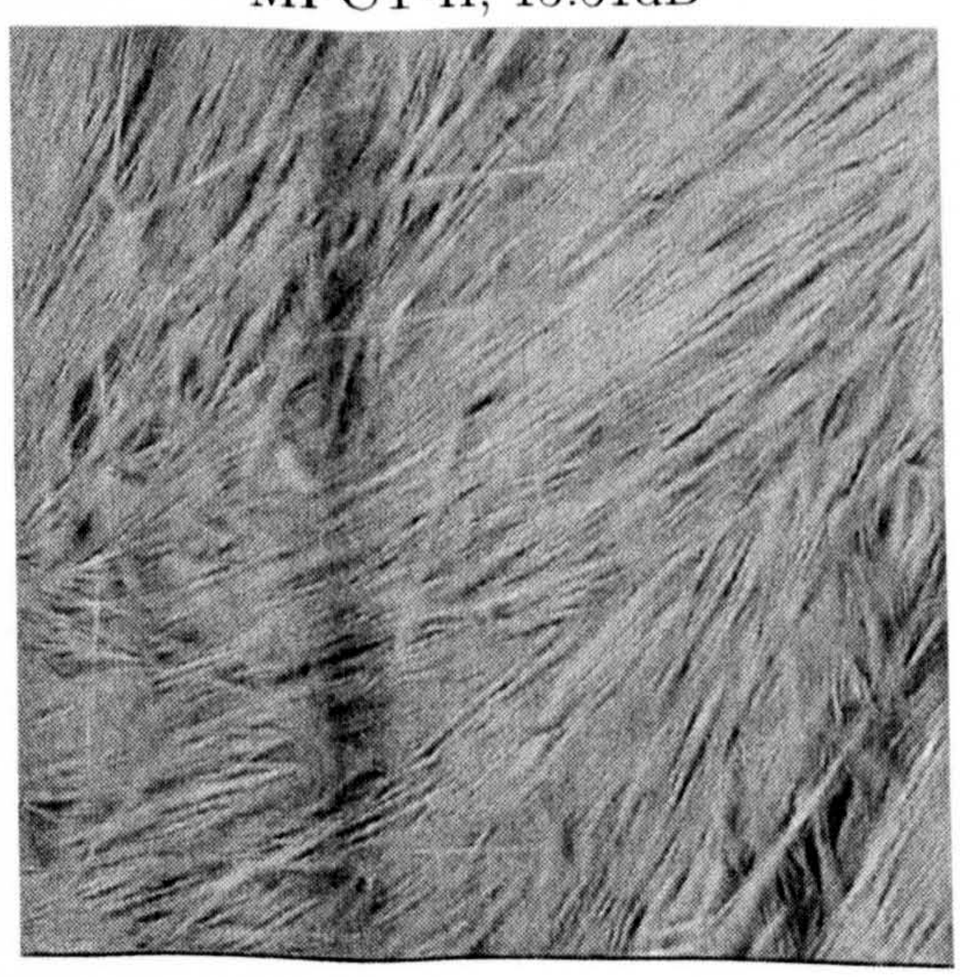
Wave atoms, 16.35dB



MPCT-II, 15.51dB



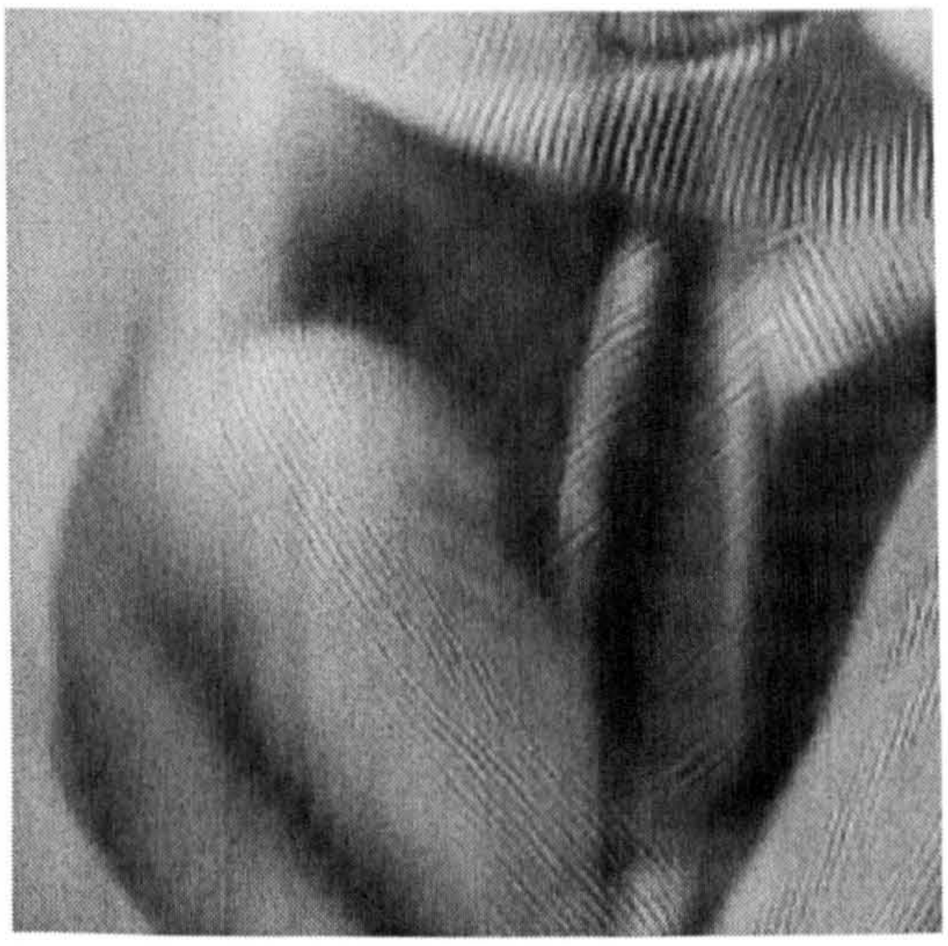
MPCT-X, 16.09dB



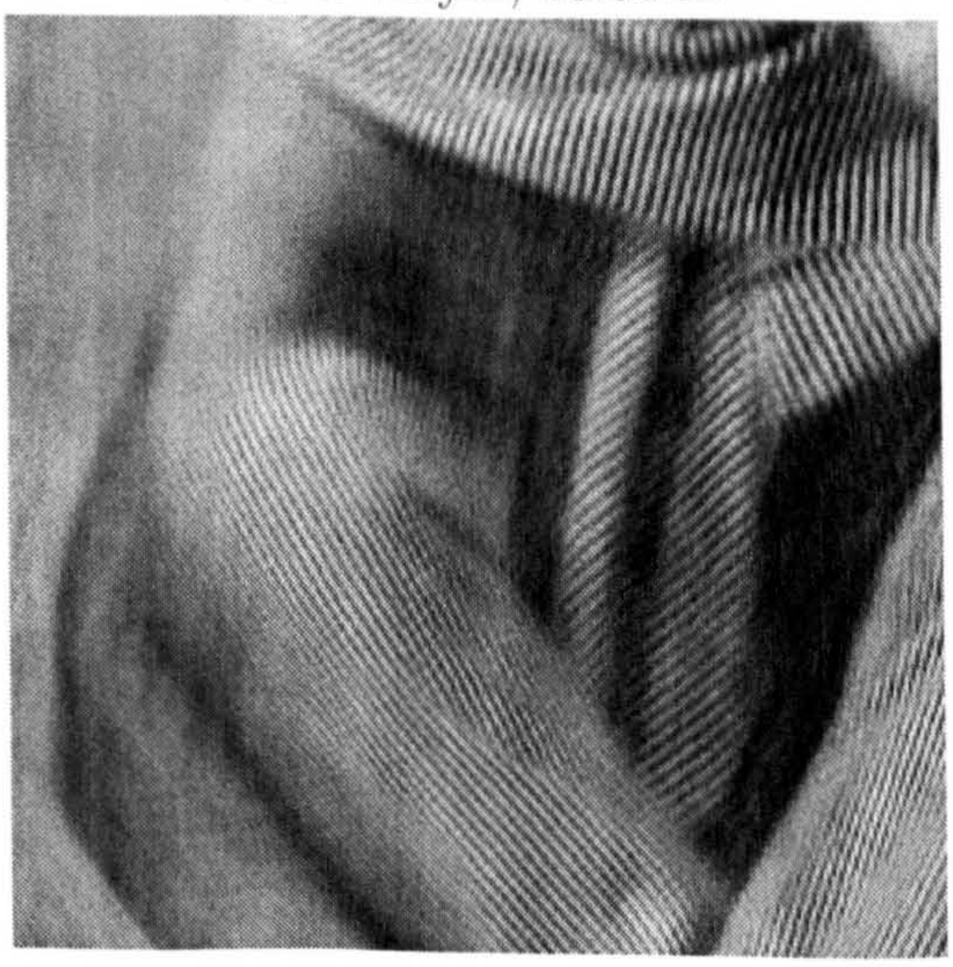
MPCPT, 16.18dB

Figure 4.19: Comparative denoising results on `grain`, denoised from 10dB

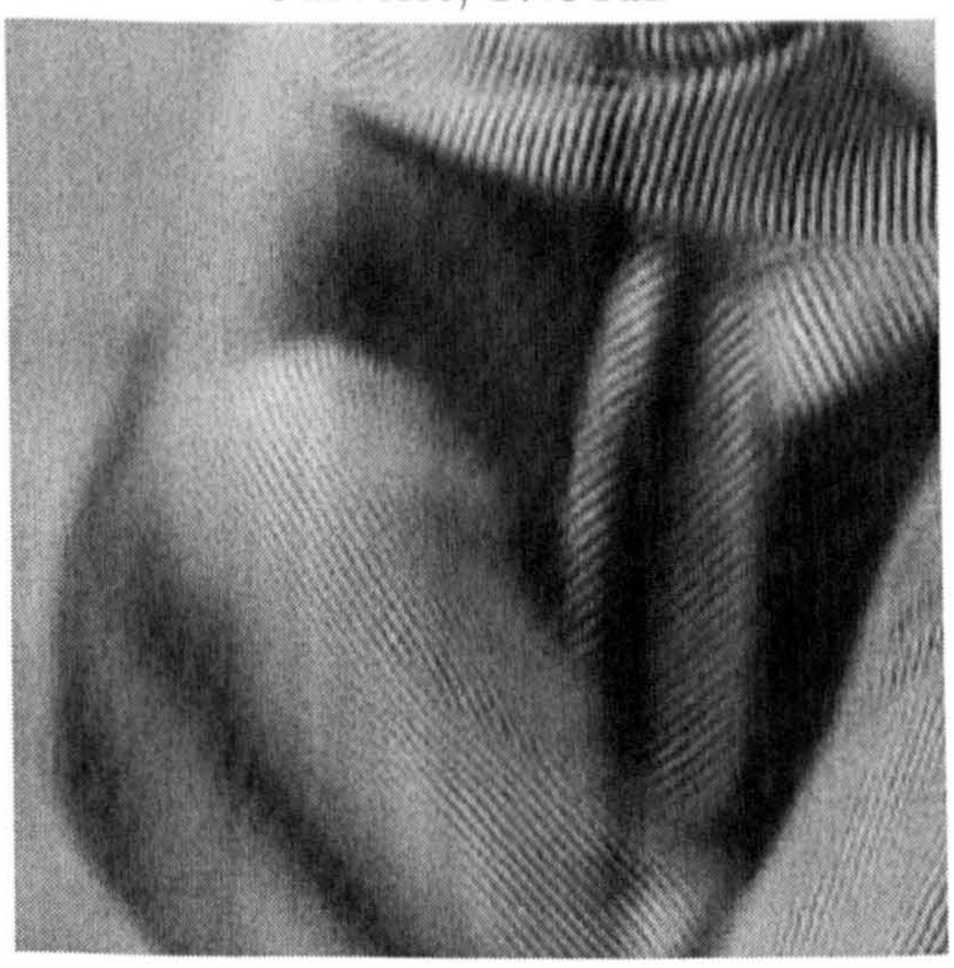
WP $\mathcal{S}$-Bayes, 18.00dB



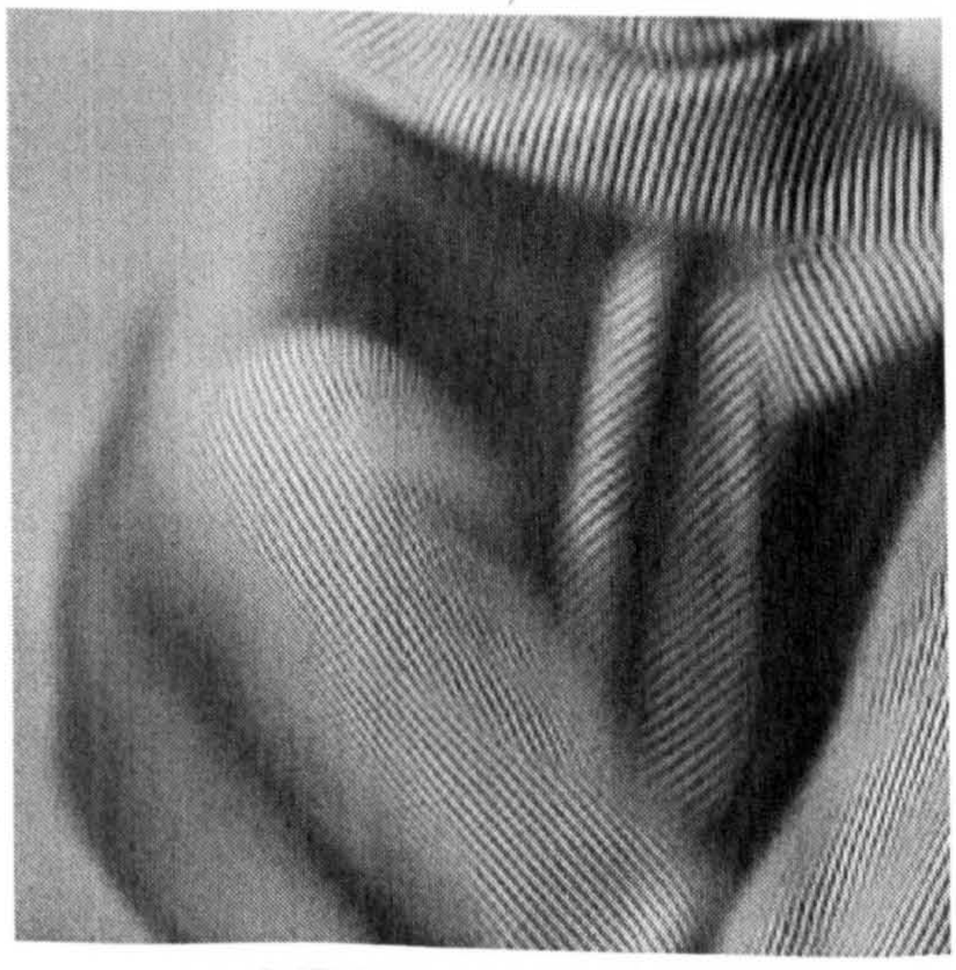
Curvelet, 17.64dB



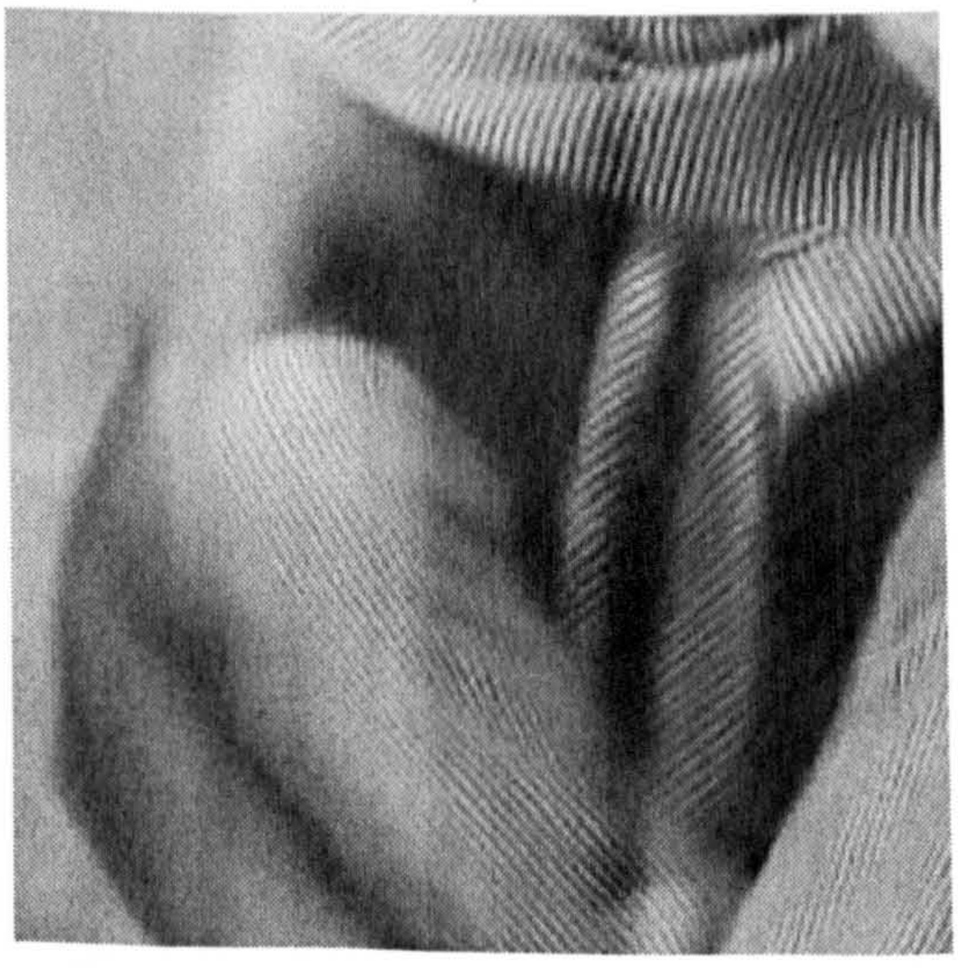
Wave atoms, 19.22dB



MPCT-II, 18.44dB

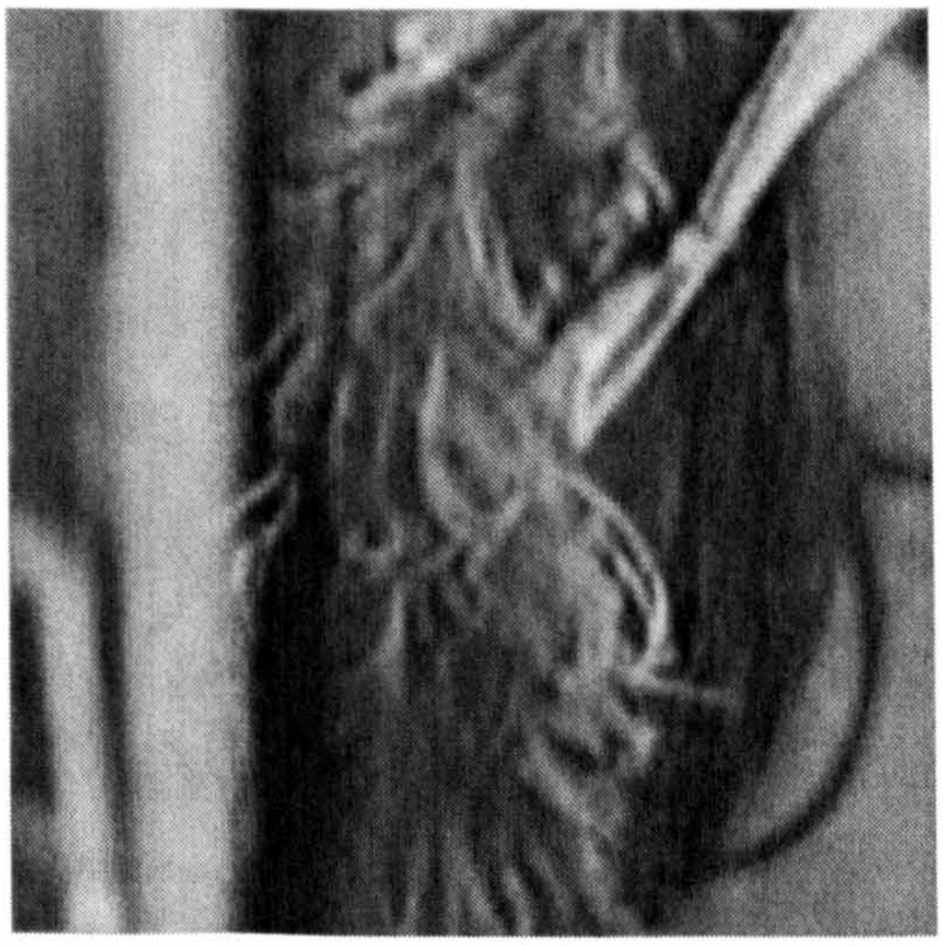


MPCT-X, 18.75dB



MPCPT, 18.85dB

Figure 4.20: Comparative denoising results on detailed 10dB `barbara`



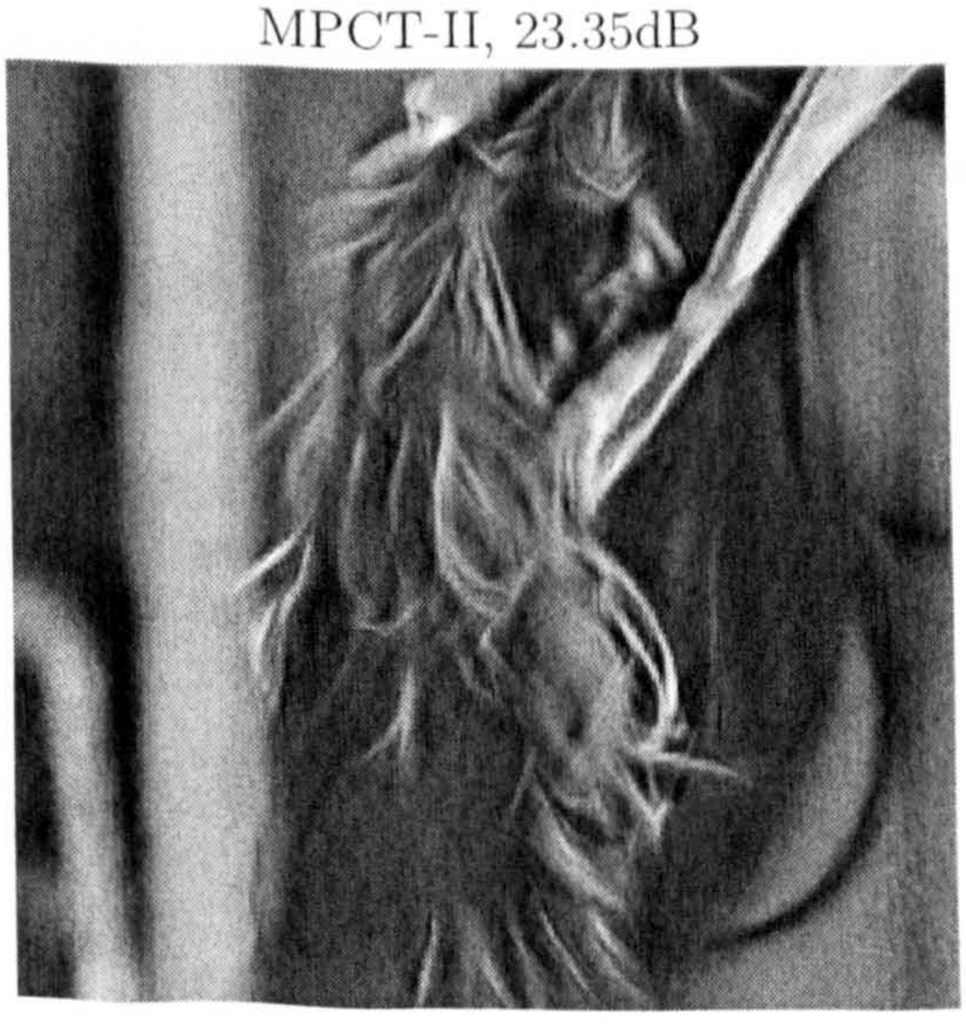
WP $\mathcal{S}$-Bayes, 23.49dB

Curvelet, 23.10dB

Wave atom, 23.21dB

MPCT-II, 23.35dB

MPCT-X, 22.88dB



MPCPT, 24.26dB

Figure 4.21: Comparative denoising results on detailed 15dB `lena`

Transform	Running Time in seconds
TIWP	398
Curvelet	224
Wave atoms	57
MPCT-II	199
MPCT-X	25
MPCPT	1067

Table 4.4: Denoising running time comparison

4.5.4 Comparison of the Computational Complexity

Since all the transforms are implemented in Matlab, a comparison of their running time can effectively reflect their computational requirements. A list of their running time counted in seconds is given in Table 4.4.

The translation-invariant wavelet packet implementation is based on the `Wavelab` package[2], which contains a few natively compiled `.mex` routines. The curvelet and MPCT-II use the Fast Slant Stack implementation in the `Beamlab` package[3].

Since curvelets and MPCT-II only differ in the 1D transform on Radon slices, their complexities are similar. However, in our experiment, the MPCT-II usually requires tens of seconds less than the curvelet transform. This is due to the fact that underlying DCT is an optimised routine from Matlab's own toolbox. The computational cost of the two adaptive transforms, MPCPT and TIWP, topped the rest of the group in the comparison. The MPCPT, in particular, requires about 17 minutes. The MPCT-X, on the other hand, is the fastest transform in the group, while the wave atoms requires slightly longer to denoise an image.

[2]The `Wavelab` package can be found on `http://www.wavelab.org`.
[3]The `Beamlab` package can be found on `http://www.beamlab.org`.

4.6 Summary and Discussion

A new family of higher dimensional trigonometric transforms were introduced in this chapter. By generalising the Fourier transform in a polar fashion, the polar cosine transform offers good convergence properties as the separable cosine transform as well as oriented basis functions and orientation selectivity like the Fourier transform. Two digital implementations were discussed for the discrete PCT. One implementation combines the Radon transform with a 1D DCT and the other is computed from DCT-IV and DST-IV in a butterfly fashion. The Radon-based implementation offers more flexibility such more choices of the ridge profile by applying different types of DCT on the Radon slices and allows a best basis search for cosine packets, while the butterfly version, termed as the PCT-X, is much faster and less redundant.

The construction of the multiscale polar cosine transform, which offers localisation in space and frequency, as well as orientation selectivity, and employs a multiresolution pyramidal framework allows analysing the image at different scales, was also presented. In a sense, the MPCT qualifies as directional wavelets and share a lot of similarities with other directional wavelet frames proposed previously. The MPCPT, constructed by seeking best cosine bases on Radon slices, combines the representation powers of curvelets and MPCT, which is a good representation for directional textures as well as edges.

The effectiveness of the MPCT bases was tested against the state-of-the-art translation-invariant wavelet packet based shrinkage method and directional transforms such as curvelets and wave atoms. The new bases demonstrated a strong potential in the experiments, producing more visually pleasing reconstructions than the opponents, with competitive quantitative results.

Chapter 5

Directional Image Denoising

5.1 Introduction

In the previous chapter, it has been shown that the MPCPT, which applies the cosine packet analysis on local Radon slices in a multiscale fashion can be an efficient representation for natural images, due to its ability to accommodate both directional linear edges and textures. However, such a packet basis has two major drawbacks. The first problem is its high computational cost, since it requires the best basis search to be conducted on every Radon projection slice. The second problem is associated with the underlying Radon transform, whose inversion usually introduces more numerical errors than other linear transforms, which is undesirable in many image processing tasks.

When the basis for a transform is inadequate to representing desired patterns or features, one solution is to "*upgrade the transform* ", such as seeking an adaptive basis (e.g. the wavelet packets [104]) or using a new transform like curvelets [28]. On the other hand, another approach is to derive a model in the old transform domain for the desired feature, in order to "*upgrade the analysis*". For example, in order to analyse linear features, some researchers proposed new transforms such as curvelets and contourlets, while others built

models for linear edges in the orthonormal wavelet domain like the Multiscale Edge Model by Mallat et al [96].

As mentioned previously, while seeking an adaptive basis for representing both edges and textures may be too expensive for some applications, another approach is to model the linear edges in a fixed transform domain, such as the frequency linear feature model presented in Chapter 3. It is, therefore, natural to combine a directional fixed transform which is suitable for representing textures and a linear feature model in the corresponding domain to solve the problem.

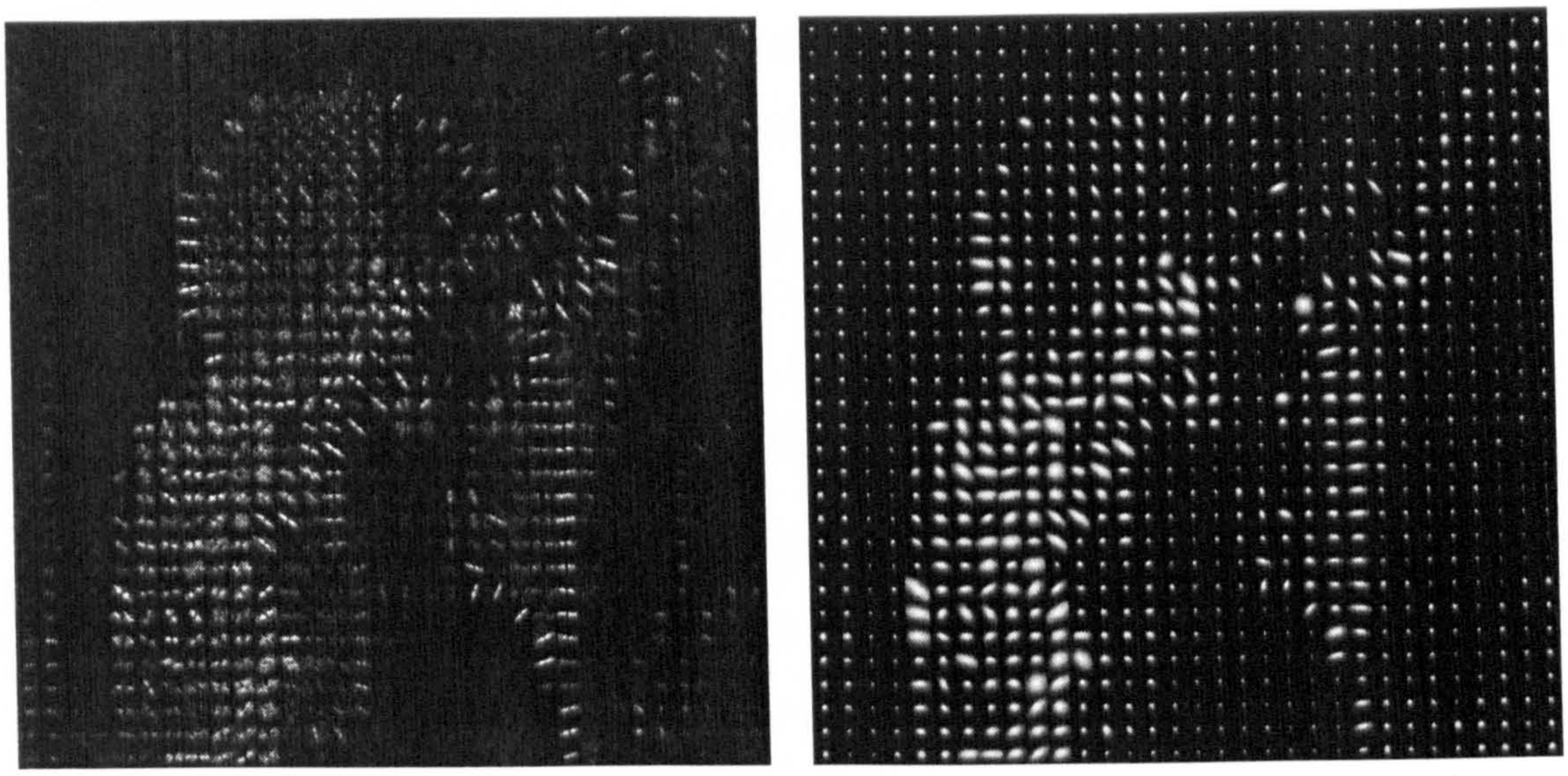
Such a combination is realised in this chapter, in the form of fast directional image denoising algorithms in the MFT domain, by applying orientation-adaptive frequency filters. The proposed methods give similar, sometimes superior, results to the MPCPT in denoising, while the computational requirement is much lower.

5.2 Spectrum Model and Directional Filtering

5.2.1 The Spectrum Model

The estimation of Fourier spectrum magnitudes is a well-studied problem for the restoration of signals from noise [91, 150]. As discussed in Chapter 3, it is known that in 2D, the Fourier transform of a directional linear feature results in a concentration of energy along the line perpendicular to the spatial orientation and passing through the origin. It is, therefore, natural to apply less attenuation to coefficients along the direction of interest than others. This suggests an anisotropic model is useful for the magnitude spectrum. Moreover, the model should be computationally inexpensive and general enough to allow different types of features with all possible degrees of anisotropy.

An appropiate choice for modelling the significant spectral energy is a 2D zero-mean Gaussian function, with its centre fixed at origin, similar to the mixture model presented



The Local Magnitude Spectrum The Gaussian Spectrum Estimation

Figure 5.1: The local magnitude spectrum and its estimate on `lena`

in Chapter 3:

$$g(\boldsymbol{\omega}) = \exp\left(-\frac{\boldsymbol{\omega}^T \Gamma^{-1} \boldsymbol{\omega}}{2}\right). \tag{5.1}$$

The covariance matrix Γ in the above equation can be estimated from the inertia tensor of the spectrum by equation (3.10).

An illustration of local Fourier spectrum estimation can be seen in Figure 5.1, where a non-overlapping Windowed Fourier Transform (WFT) with 16×16 window is performed over the pre-whitened `lena` image. The left hand side of the figure shows the local Fourier magnitudes, and the right hand side the estimated Gaussian functions.

5.2.2 The Gaussian Frequency Filter

The estimated 2D Gaussian function can be seen as an approximation of the spectrum energy, reflecting the spectrum orientation and dispersion. But it does not consider the range of the coefficients' intensity values (as the Gaussian function lies between 0 to 1). However, this does not pose a serious disadvantage, when the estimate is being used as a

frequency filter, whose purpose is to separate significant coefficients from insignificant ones. The reason for choosing the Gaussian function as the model for the filter is three-fold:

1. The Gaussian function has optimal spread in time and space, and it is smooth in both domains.

2. The parameters for the function are easy to compute.

3. The dispersion of the Gaussian function can be easily controlled by altering the covariance matrix Γ. When the filter is anisotropic, which indicates the presence of oriented pattern, the filter allows only a narrow oriented band to pass. When the filter tends to be isotropic, it functions as conventional lowpass smoothing.

5.2.3 Related Methods

An anisotropic filter in the frequency domain in its essence resembles the effect of thresholding in the ridgelet domain. The selection of an oriented frequency band can be regarded as extracting a Fourier slice, which in turn can be converted into a Radon slice, given the assumption that the feature is linear and that the significant spectrum energy is concentrated in the given orientation. Although, compared to the ridgelet transform, anisotropic filtering makes the assumption of the presence of a single feature, such simplification does not dramatically degrade its effectiveness. With an appropriate choice of windowing size and overlap factor, the assumption is sufficient for most circumstances, with the exception of corners and crossings. In exchange, the computational burden is significantly reduced. Without invoking the interpolation for the Radon transform, the numerical stability for the reconstruction is also improved.

The use of directional filtering is not new in the field of image representation. The steerable filter [141] is an example of separating the higher dimensional signals into different oriented subbands and also anisotropic smoothing filters such as [88]. The contourlet

transform, sometimes considered as an alternative implementation of the curvelet transform, uses a *double filter bank* approach. It employs first a Laplacian Pyramid decomposition to capture the multiscale point singularities, then each of the subbands is directed into a directional filter bank, which in essence groups the contour segments like a local Radon transform. The MFT with Gaussian frequency filter is then directly related to the contourlet, which also uses the Laplacian pyramid to decompose the image, but the directional filtering is done locally and adaptively within the window.

5.3 Gaussian Frequency Filtering for Denoising

5.3.1 The Denoising Process

As discussed in the previous section, the effectiveness of the Gaussian frequency filter is similar to the thresholding in the ridgelet/contourlet domain when there is a directional feature present. It is, therefore, natural to apply the filter on the images as a form anisotropic smoothing and subsequently can be used to suppress the noise.

Since the estimation of the Gaussian filter parameters uses the inertia tensor, which does not work well on noisy images, the filter's estimation for denoising is not straightforward. A decent approach is to use the "Signal+Noise" spectrum model presented in Chapter 3. However, due to the complexity of the EM algorithm, this approach is rejected, since it conflicts with the main desiderata: low computational cost. A simple sub-optimal solution is to estimate the inertia tensor after thresholding the target noisy spectrum. Since the purpose here is to give an illustrative demonstration, the threshold is selected as

$$\mathcal{T} = 0.6 \max(\hat{f}(\omega)). \tag{5.2}$$

The whole process is illustrated in Figure 5.2, taking the straight line image as the

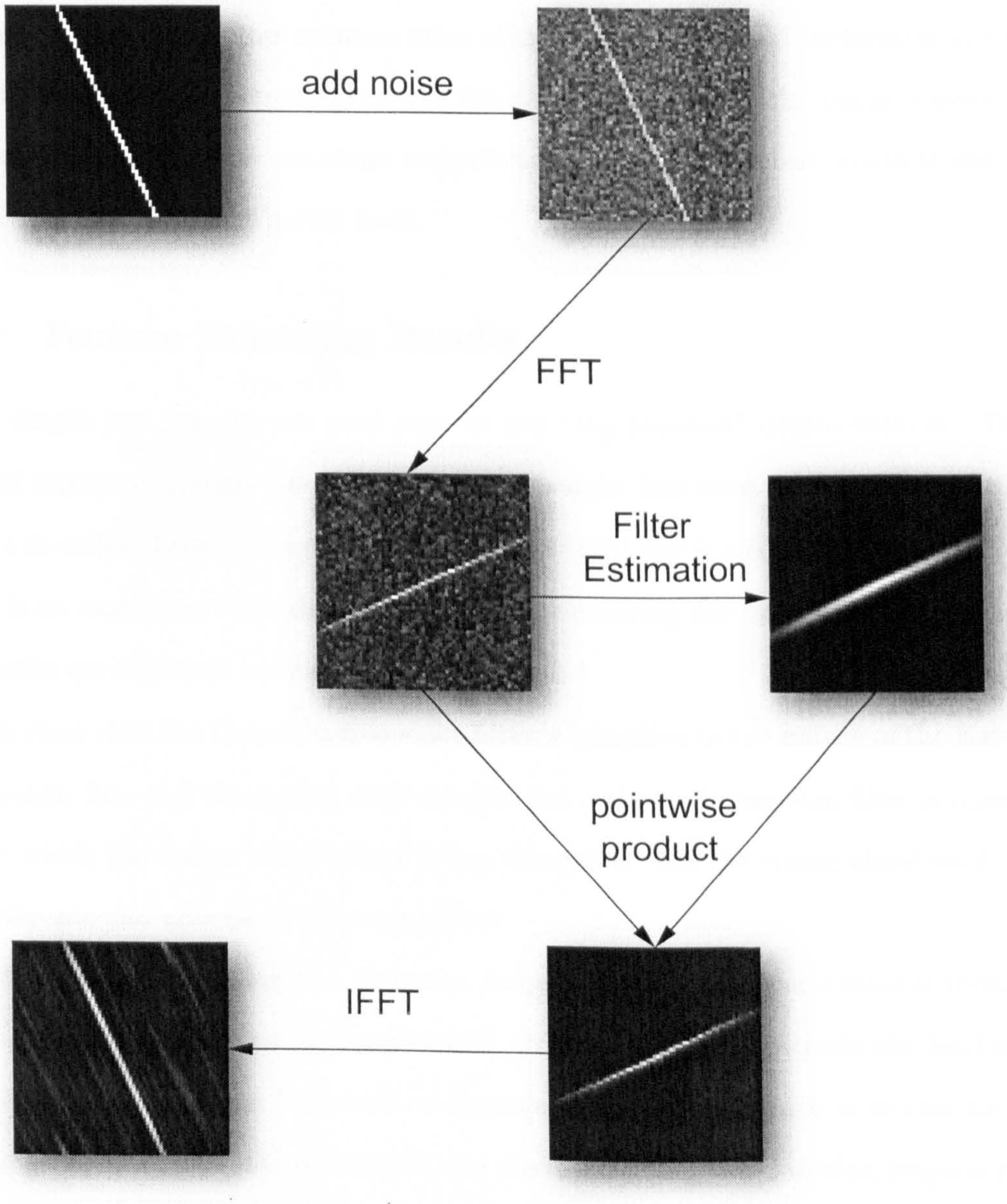


Figure 5.2: A simple straight line denoising with Gaussian filtering

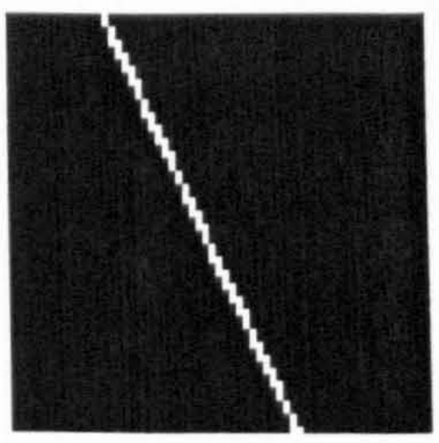
example. First, the image is corrupted by 0dB Gaussian white noise. The noisy image then undergoes the forward FFT. The resulting spectrum contains a perpendicular straight line from which the corresponding Gaussian filter can be estimated. Taking the product of the spectrum with the filter removes most of the noise energy and performing the inverse FFT on the filtered spectrum gives the denoised image where the linear feature is well preserved, although there are some insignificant "directional" noisy artifacts due to the nature of the underlying Fourier basis.

5.3.2 Feature Denoising Results

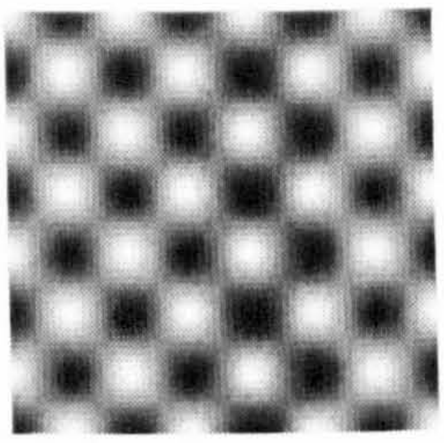
Three simple test images are used here as the "*toy problem*" demonstration. They are selected representatively: the first one is the straight line used in Figure 5.2, the second one is a directional cosine ridge texture, while the third one is a cosine "chessboard" feature which is an example of non-directional texture. Following the procedure described before, the results are depicted in Figure 5.3.

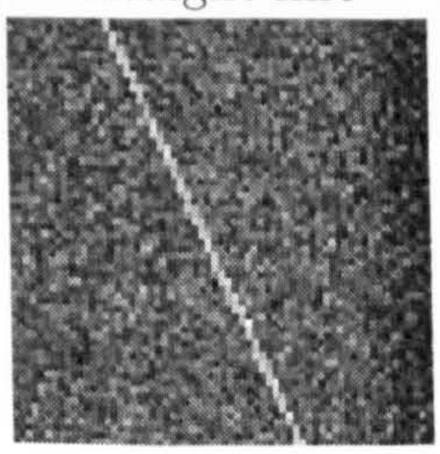
It is clear that the Gaussian frequency filter is adaptive to the nature of the feature: for the straight line and the cosine ridge images, the estimated Gaussian filter is anisotropic, among which the cosine ridge's filter is less elongated. For the cosine chessboard texture, it is isotropic and effectively a lowpass filter.

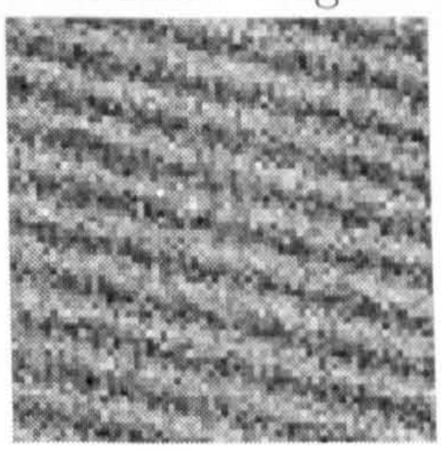
The advantage of using the Gaussian frequency filter over conventional thresholding denoising is clear. The simple "keep-or-kill" strategy on the coefficients can lead to severe pseudo-Gibbs phenomena [40], and an appropriate threshold which is crucial for the denoising outcome is difficult to estimate. On the other hand, the Gaussian frequency filter is smooth, and has the ability to single out a narrow, oriented band. This is particularly useful in dealing with linear singularities, which are less well represented by the Fourier basis. In Figure 5.4, the denoising outcomes for the straight line images by Gaussian frequency filtering and hard thresholding are compared. The threshold is set as in [40], known as the

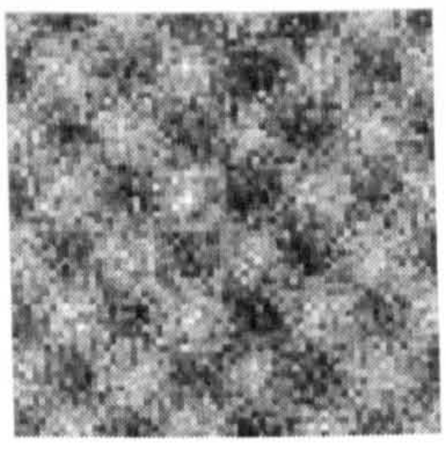

Straight line

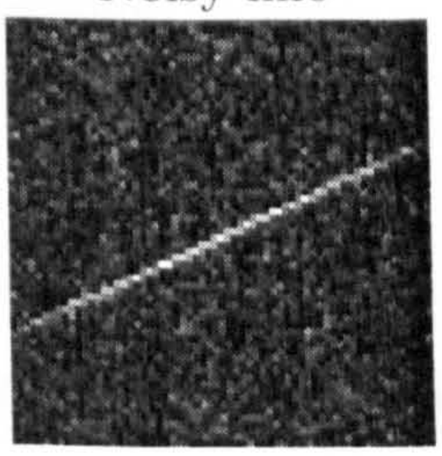

Cosine ridge

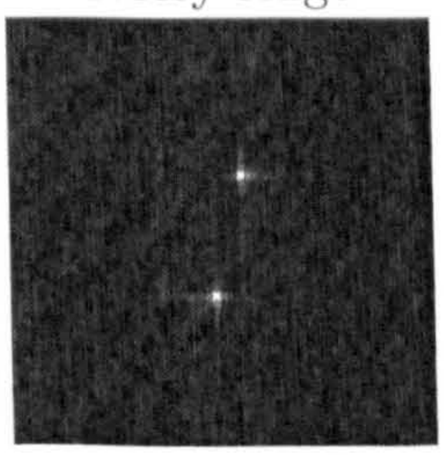

Cosine chessboard

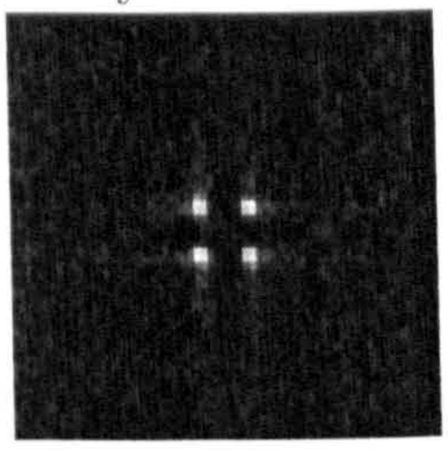

Noisy line

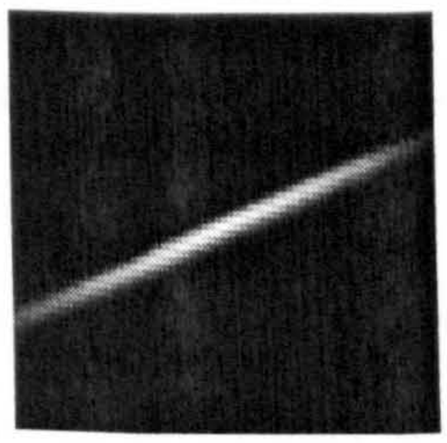

Noisy ridge

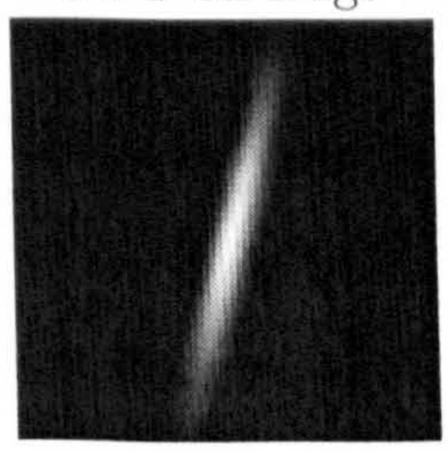

Noisy chessboard

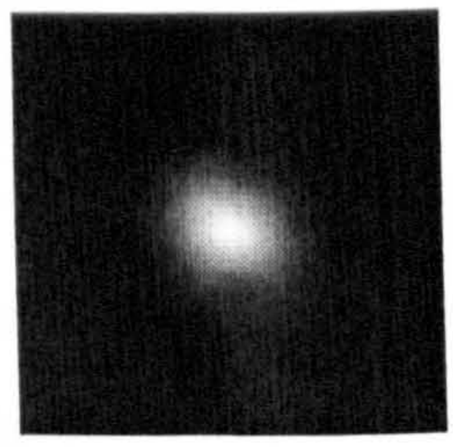

FFT on line

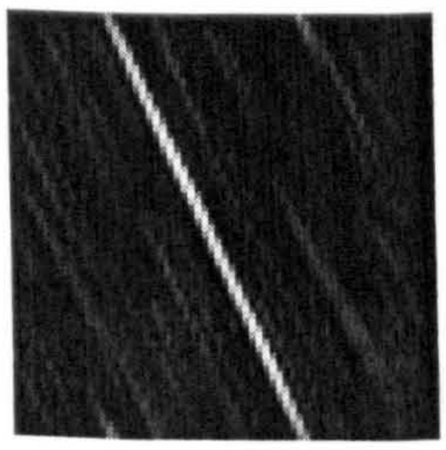

FFT on ridge

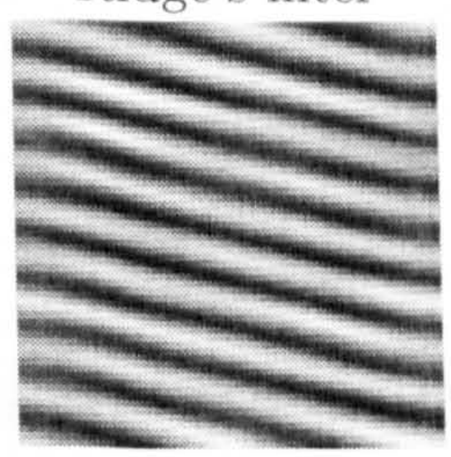

FFT on chessboard


Line's filter


Ridge's filter


Chessboard's filter


Denoised line
PSNR 22.04dB


Denoised ridge
PSNR 18.28dB

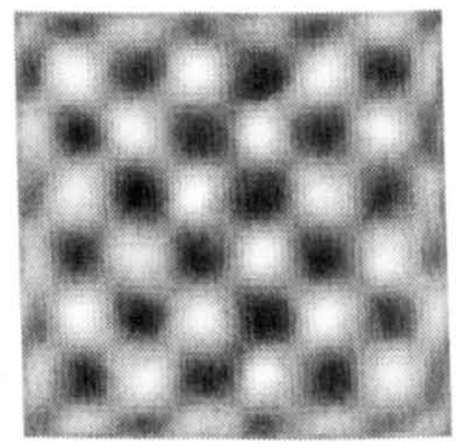
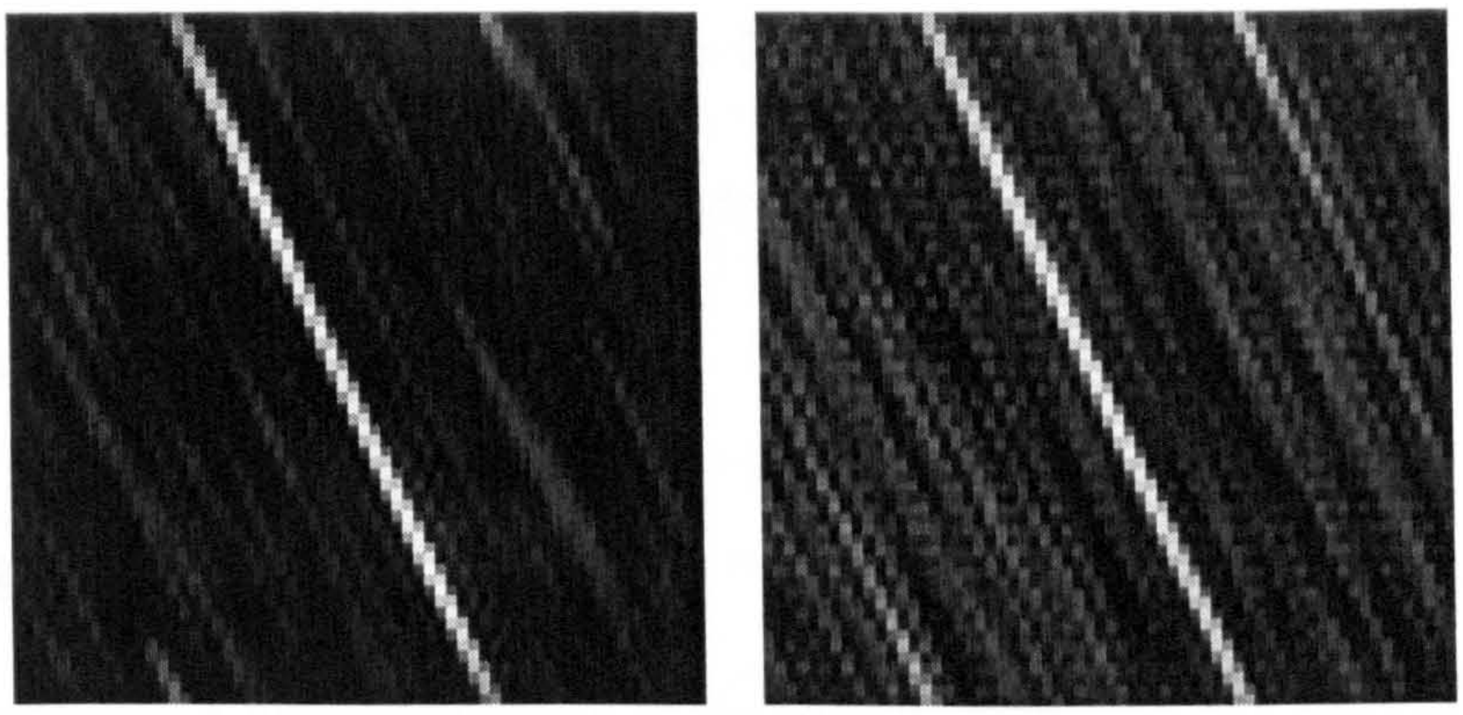

Denoised chessboard
PSNR 18.83dB

Figure 5.3: Simple feature denoising results



Gaussian Frequency Filtered
PSNR: 22.04dB

Hard Thresholded
PSNR 20.43dB

Figure 5.4: Gaussian frequency filtering compared with thresholding

'universal' threshold:

$$\mathcal{T} = \sqrt{2\log(n)}\sigma. \tag{5.3}$$

where the value of σ is informed.

The Gaussian filter clearly yields better results. On patterns such as the cosine ridge and chessboard, which the Fourier basis is able to represent efficiently on its own, simple thresholding has its advantage since the spectrum energy is not concentrated on a line, and the nature of the Gaussian filter is likely to allow more *"noisy"* coefficients to pass. However, visually the results are satisfactory, since the filter tends to keep the lowpass and lowpass noise is visually less significant than highpass. Also, in a real application, the option to choose between these two strategies is always possible, by measuring the anisotropy of the Gaussian filter.

5.4 Image Denoising Algorithms

For both variations, the decomposition setting of the MFT is exactly the same as used in Chapter 3 and 4:

- The Laplacian Pyramid is used at $J = 5$ levels of subbands.

- The window size is $M^2 = 16 \times 16$, modulated with squared cosine.

- The windows are overlapped by 50%.

5.4.1 Algorithm I

The first algorithm is a direct extension of the idea described in the previous section. The algorithm estimates the Gaussian filter from a thresholded version of the spectrum, and uses the anisotropy measure in equation (3.43) to determine whether the block should be denoised with thresholding or the Gaussian frequency filter. The whole process can be described as follows:

1. Decompose the image with the forward MFT.

2. For a given block $x_{\eta,s}(\xi)$, its spectrum $\hat{x}_{\eta,s}(\omega)$ is thresholded by

$$\mathcal{T}_s = 8\frac{\sqrt{2\log M^2}\sigma}{1.23^{J-s}}. \tag{5.4}$$

   The threshold is set on a crude level in order to minimise the effect of noise in order to obtain a good estimate from the inertia tensor, which was empirically found to be adequate. Then the filter $H_{\eta,s}(\omega)$ is estimated from the thresholded spectrum $\hat{x}'_{\eta,s}(\omega)$ according to equation (5.1).

3. The anisotropic measure $a_{\eta,s}$ is computed as in equation (3.43) on the Gaussian filter $H_{\eta,s}(\omega)$.

4. If $a_{\eta,s}$ is larger than a constant (here it is set to 0.43), which indicates there is a presence of strong directional feature, the spectrum is filtered by applying the

frequency filter :

$$\hat{x}^R_{\eta,s}(\omega) = a_{\eta,s}\left(\hat{x}_{\eta,s}(\omega)H_{\eta,s}(\omega)\right) + (1 - a_{\eta,s})\hat{x}'_{\eta,s}(\omega). \tag{5.5}$$

This is a linear combination of the thresholded spectrum and filtered spectrum, with anisotropy measure $a_{\eta,s}$ controlling the proportion. When the filter's elongation is greater, the combination tends to assign more weight to the filtered version. When the filter tends to be more isotropic, which means the thresholding strategy could be more effective, larger weight is given to the thresholded part.

5. Otherwise, since there are no significant linear features, the $\hat{x}^R_{\eta,s}(\omega)$ is simply obtained by thresholding the spectrum with a less crude threshold value:

$$T'_s = 7.4\frac{\sqrt{2\log M^2\sigma}}{1.23^{J-s}}. \tag{5.6}$$

6. Perform the inverse transform on $\hat{x}^R_{\eta,s}(\omega)$ to obtain the denoised block.

The listing of the pseudo-code of the algorithm is presented in Table 5.1.

5.4.2 Algorithm II

The second variation takes a completely different approach. It is possible that sometimes linear feature extraction (discussed in Chapter 3) and denoising sometimes are to be performed simutanously. One example could be a database for images captured by a noisy remote sensor. It is, therefore, required to extract semantic feature vectors for content-based retrieval, while the images also need to be enhanced which is a typical requirement for medical images. It is then sensible to use the information obtained by the linear feature extraction stage to aid the denoising process. Certainly, there are many possibilities how

Algorithm 5.4.1: MFTG-I(X, σ, J)

comment: X is the noisy image; σ is the standard deviation;
comment: J is the total level of decomposition; .* denotes pointwise product

$pyr \leftarrow$ LAPLACIANPYRAMID(X, J)
for $s \leftarrow 1$ **to** $J-1$
do {
  $\sigma_s = \sigma 1.23^{J-s}$
  for each $block \in pyr_s$
  do {
    $\hat{f} \leftarrow$ FFT($block$)
    $\hat{f}' \leftarrow$ HARDTHRESHOLD($\hat{f}, 8\sqrt{2 \log M^2}\sigma_s$)
    $\Gamma \leftarrow$ INERTIATENSOR($\hat{f}'$)
    $anisotropy \leftarrow$ ESTIMATEDISPERSION(Γ)
    if $anisotropy > 0.43$
      then { $H \leftarrow$ DRAWGAUSSIANFILTER(Γ)
                 $\hat{f}^R \leftarrow anisotropy * (\hat{f} .* H) + (1 - anisotropy)\hat{f}'$
      else $\hat{f}^R \leftarrow$ HARDTHRESHOLD($\hat{f}, 7.4\sqrt{2 \log M^2}\sigma_s$)
    $denoised \leftarrow$ IFFT($\hat{f}^R$)
  }
}
$X' \leftarrow$ INVLAPLACIANPYRAMID(pyr)

Table 5.1: The pseudo-code for MFTG-I algorithm

one may incorporate the feature extraction with the denoising. Here a simple version of such a combination is presented.

After extracting the linear features, as discussed in Chapter 3, the following information is available:

- $\theta^{-}_{\eta,s}$: The estimated orientations for each block after the Kalman filtering.

- $\mathcal{L}_{\eta,s}$: The class label for each block which indicates a linear feature's presence.

Instead of using the anisotropic dispersion measure $a_{\eta,s}$, the classification label $\mathcal{L}_{\eta,s}$ can be used to determine if the spectrum needs to be cleaned by a Gaussian frequency filter or thresholding. The orientation information then can be used to construct the filter instead of estimating it from the noisy spectrum.

Following is the detailed description:

1. Decompose the image with the forward MFT.

2. For a given block $x_{\eta,s}(\xi)$, the corresponding classification label $\mathcal{L}_{\eta,s}(\xi)$ can be either 0, which indicates the block contains a non-linear feature; or 1, a linear feature.

3. If $\mathcal{L}_{\eta,s}(\xi) = 1$, the spectrum is denoised by a Gaussian frequency filter. The parameters of the filter can be either the covariance matrix Γ or the eigenvalues λ_1, λ_2 and the orientation of the eignvectors (see Figure 3.6). The shape of the Gaussian function is characterised by λ_1 and λ_2, which is basically the dispersion in two orthogonal directions.

   Given the orientation $\theta^-_{\eta,s}$, the eignvector matrix can be computed as

$$V_{\eta,s} = \begin{pmatrix} v_1 & v_2 \\ v_3 & v_4 \end{pmatrix}, \tag{5.7}$$

where

$$\begin{aligned} v_1 &= \cos(\theta^-_{\eta,s} + \pi/2), \\ v_2 &= \cos(\theta^-_{\eta,s} + \pi), \\ v_3 &= \sin(\theta^-_{\eta,s} + \pi/2), \\ v_4 &= \sin(\theta^-_{\eta,s} + \pi). \end{aligned}$$

The two eigenvalues, which control the Gaussian function's dispersion, are fixed at a given scale:

$$\lambda_1 = r * (J - s)^{J-s-1} \tag{5.8}$$

$$\lambda_2 = 1 \tag{5.9}$$

The r in the above equation is used to control the dispersion of the Gaussian filter and is set to 0.007 in the experiment to make the filter highly anisotropic, but also will become incrementally isotropic when going from the highpass band to the lowpass band.

Then the covariance matrix can be obtained by an inverse eigen-decomposition:

$$\Gamma_{\eta,s} = \begin{pmatrix} \frac{\lambda_1 v_1 v_4 - \lambda_2 v_2 v_3}{v_1 v_4 - v_2 v_3} & \frac{\lambda_1 v_1 v_3 - \lambda_2 v_1 v_3}{v_2 v_3 - v_1 v_4} \\ \frac{\lambda_1 v_2 v_4 - \lambda_2 v_2 v_4}{v_1 v_4 - v_2 v_3} & \frac{\lambda_1 v_2 v_3 - \lambda_2 v_1 v_4}{v_2 v_3 - v_1 v_4} \end{pmatrix}. \tag{5.10}$$

With the covariance matrix, the Gaussian frequency filter $H_{\eta,s}(\omega)$ can then be obtained following equation (5.1). The spectrum is filtered as :

$$\hat{x}^R_{\eta,s}(\omega) = \hat{x}_{\eta,s}(\omega) H_{\eta,s}(\omega). \tag{5.11}$$

4. If $\mathcal{L}_{\eta,s}(\xi) = 0$, the $\hat{x}^R_{\eta,s}(\omega)$ is simply obtained by thresholding the spectrum :

$$\mathcal{T}'_s = 7.4 \frac{\sqrt{2 \log M^2} \sigma}{1.23^{J-s}}. \tag{5.12}$$

5. The inverse transform on $\hat{x}^R_{\eta,s}(\omega)$ to obtain the denoised block.

The listing of the pseudo-code of the algorithm is presented in Table 5.2.

5.5 Experimental Results

Extensive denoising experiments were conducted for the two proposed algorithms. For the sake of clarity, the MFT combined with Algorithm I is denoted as **MFTG-I** and similarly for Algorithm II the notation is **MFTG-II**. They are directly compared with the results of **MPCPT** reported in Chapter 4. All three of them are expected to be able to preserve

Algorithm 5.4.2: MFTG-II$(X, \sigma, J, \overline{\theta}, \mathcal{L})$

comment: X is the noisy image; σ is the standard deviation;
comment: J is the total level of decomposition
comment: $\overline{\theta}$ is the refined feature orientation set
comment: $\mathcal{L}$ is the classification label; .* denotes pointwise product ,

$pyr \leftarrow$ LAPLACIANPYRAMID(X, J)
for $s \leftarrow 1$ **to** $J-1$
do $\begin{cases} \sigma_s = \sigma 1.23^{J-s} \\ \textbf{for each } block \in pyr_s \\ \quad \textbf{do} \begin{cases} \hat{f} \leftarrow \text{FFT}(block) \\ \hat{f}' \leftarrow \text{HARDTHRESHOLD}(block, 8\sqrt{2\log M^2}\sigma_s) \\ \textbf{if } \mathcal{L}_{block} = 1 \\ \quad \textbf{then} \begin{cases} \Gamma \leftarrow \text{ESTIMATECOVARIANCEMATRIX}(\overline{\theta}_{block}) \\ H \leftarrow \text{DRAWGAUSSIANFILTER}(\Gamma) \\ \hat{f}^R \leftarrow (\hat{f} .* H) \end{cases} \\ \quad \textbf{else } \hat{f}^R \leftarrow \text{HARDTHRESHOLD}(\hat{f}, 7.4\sqrt{2\log M^2}\sigma_s) \\ denoised \leftarrow \text{IFFT}(\hat{f}^R) \end{cases} \end{cases}$
$X' \leftarrow$ INVLAPLACIANPYRAMID(pyr)

Table 5.2: The pseudo-code for MFTG-II algorithm

both linear features and textures in the noise removal process. Also, results for MFT denoising by simple thresholding are included, in order to demonstrate the advantages of the Gaussian frequency filtering over thresholding on the block spectrum. These algorithms are all coded in Matlab. The test dataset used here is identical to the one used in Chapter 4 (see Figure 4.15), containing four 512×512 images.

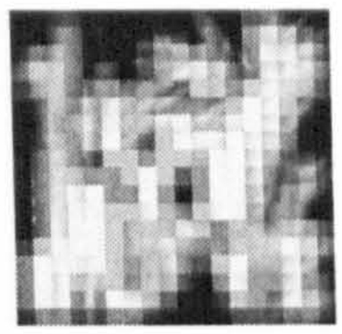
The experimental results are organised into three parts. First, the difference in the filter configuration between MFTG-I and MFTG-II will be illustrated. Then, both the visual and numerical results on the testing dataset will be presented and discussed. Finally, the results are compared with some state-of-the-art directional wavelet bases.

5.5.1 Filter Configuration Comparison

The performance of the proposed Gaussian frequency filtering with MFT is directly affected by the configuration of the filter. More precisely, the configuration includes the filter shape and the selection of blocks where the filter is applied. The MFTG-I and MFTG-II, as discussed, follow two different approaches. The MFTG-I directly estimates the filter from the spectrum and the estimates vary from block to block, while MFTG-II uses prior information given by the linear feature extraction and the filter's dispersion is fixed at a given scale.

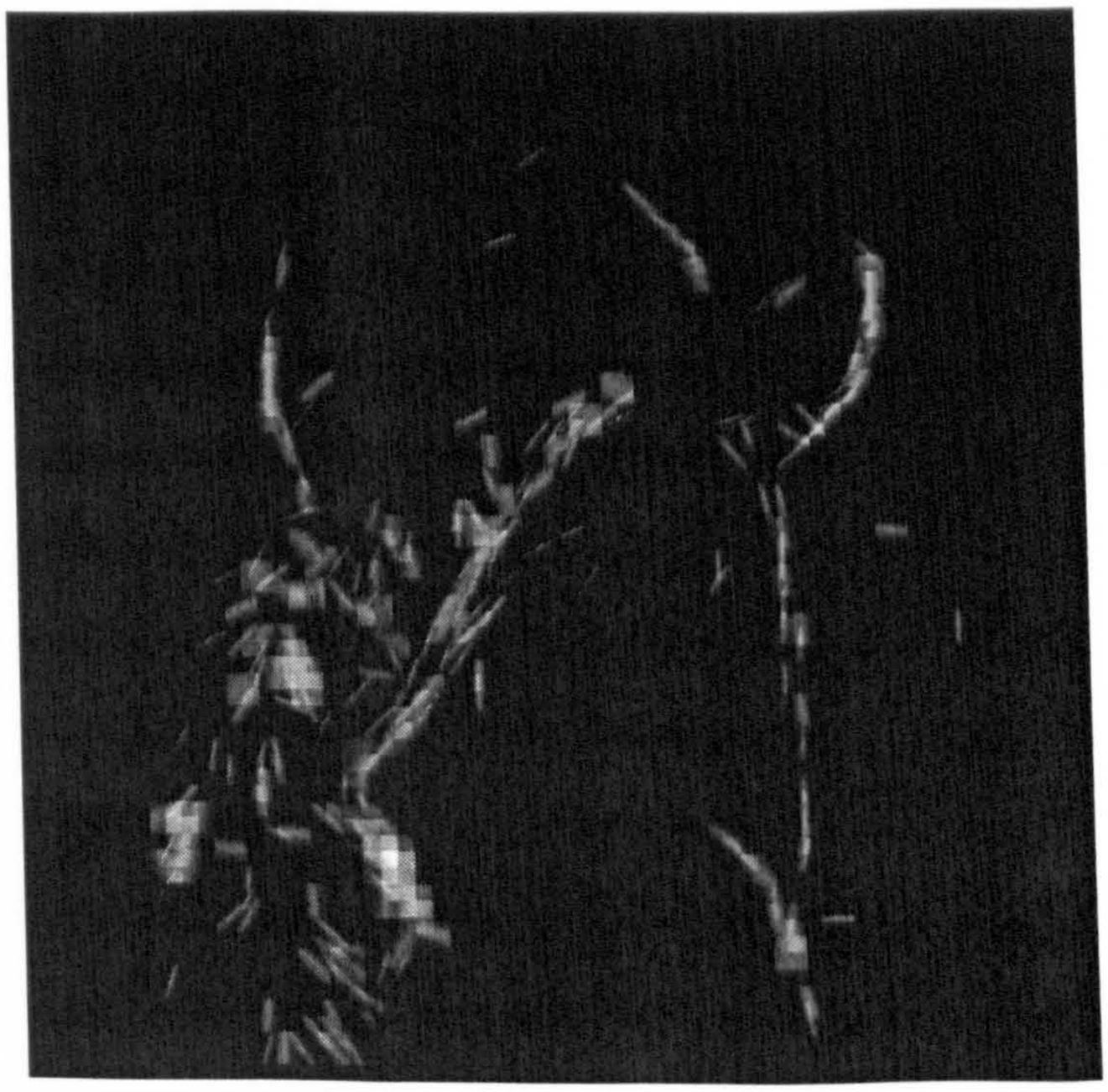
The comparison is performed on the 15dB noisy `lena` image and the 10dB `barbara` image. Figure 5.5 is the estimated frequency filters of MFTG-I plotted according to their spatial location with varying dispersion. For the reader's convenience, the filters are rotated 90 degrees to match their spatial orientations, instead of the actual orientations in the Fourier domain. The output of MFTG-II is depicted in Figure 5.6. With the help of the feature classification label, MFTG-II is able to locate more linear features than MFTG-I. The filtering is conducted exactly on the locations indicated by the linear feature extraction process where linear features are present. Also, the artificially constructed Gaussian filters of MFTG-II match well with the average dispersion of MFTG-I's estimated filters on all scales.

Similar observations can be made from Figure 5.7 and Figure 5.8. However, it is apparent that, due to the severity of the noise, MFTG-I is unable to identify most of the linear edges for filtering, but the directional textural patterns. This is due to the fact that directional strips can result in high peaks in the Fourier domain which could survive from the thresholding before estimating the filter. A linear feature's spectrum however, is less sparse and the magnitude of the useful coefficients cannot stand out if the noise variance is high. On the other hand, the MFTG-II leaves these strip textures as unfiltered. This is simply because they are not linear features. The feature extraction



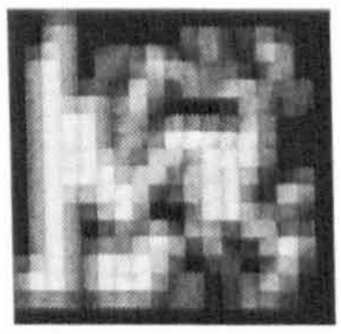
Scale $s = 2$

Scale $s = 3$

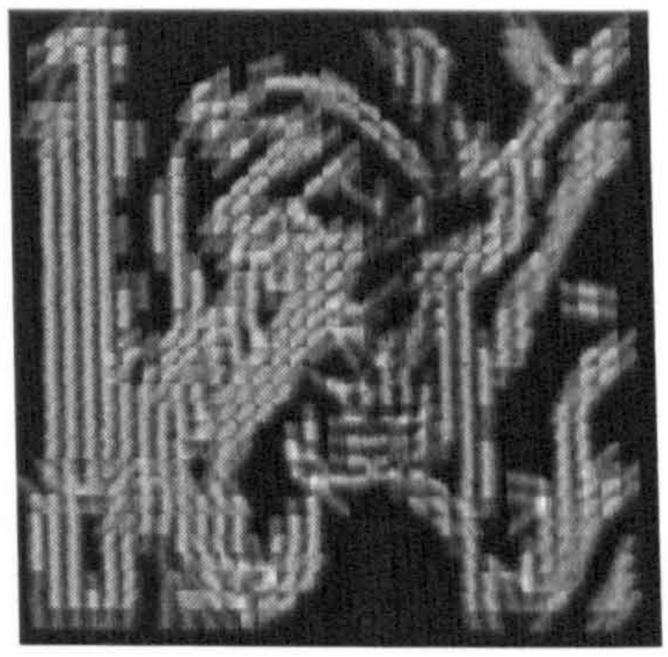


Scale $s = 4$

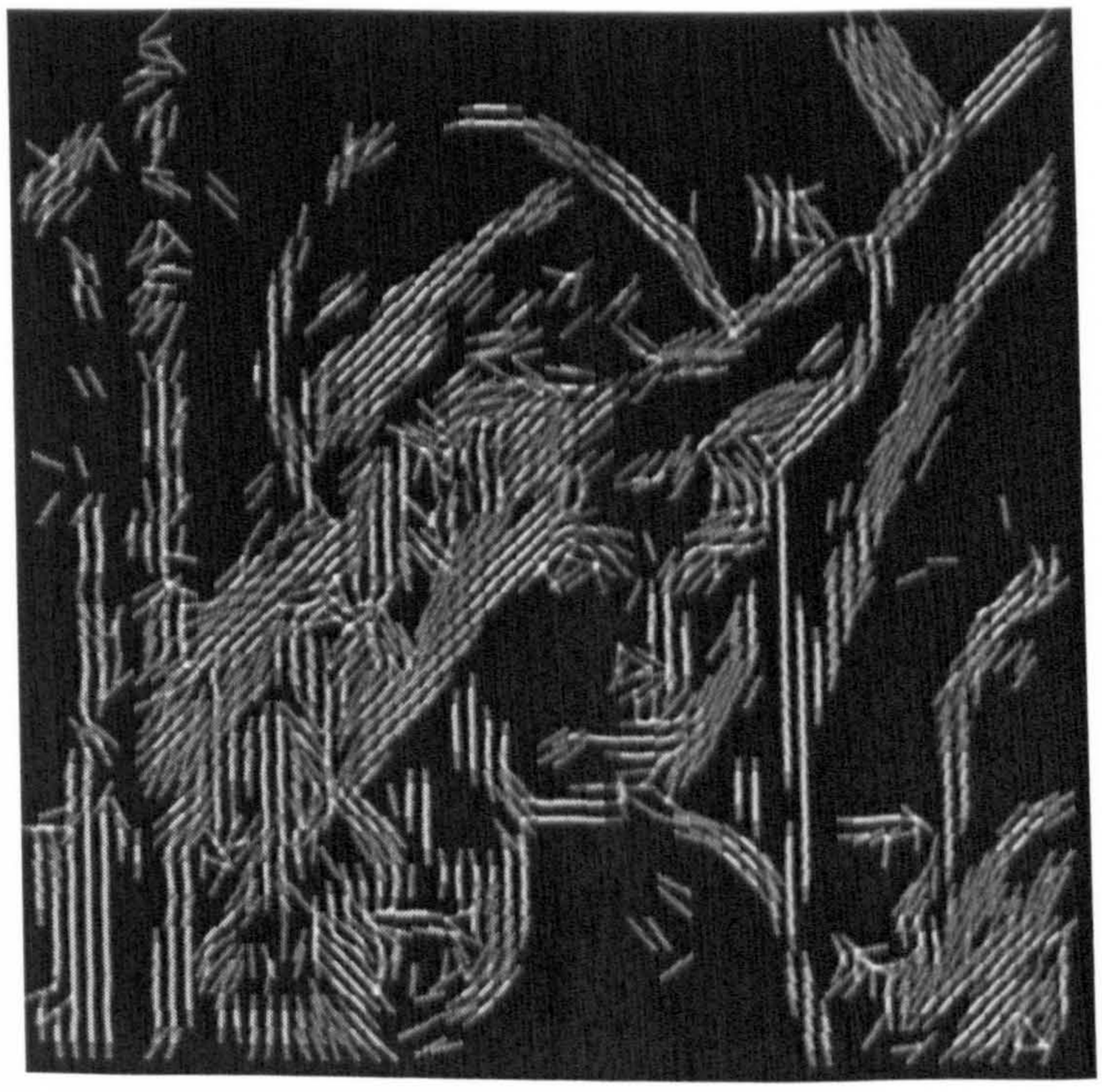
Figure 5.5: Filter configurations for MFTG-I on 15dB `lena`



Scale $s = 2$

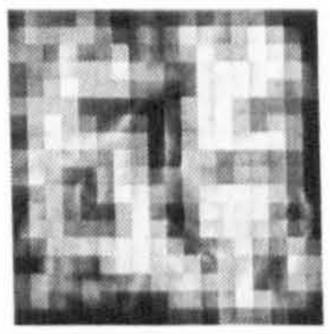


Scale $s = 3$



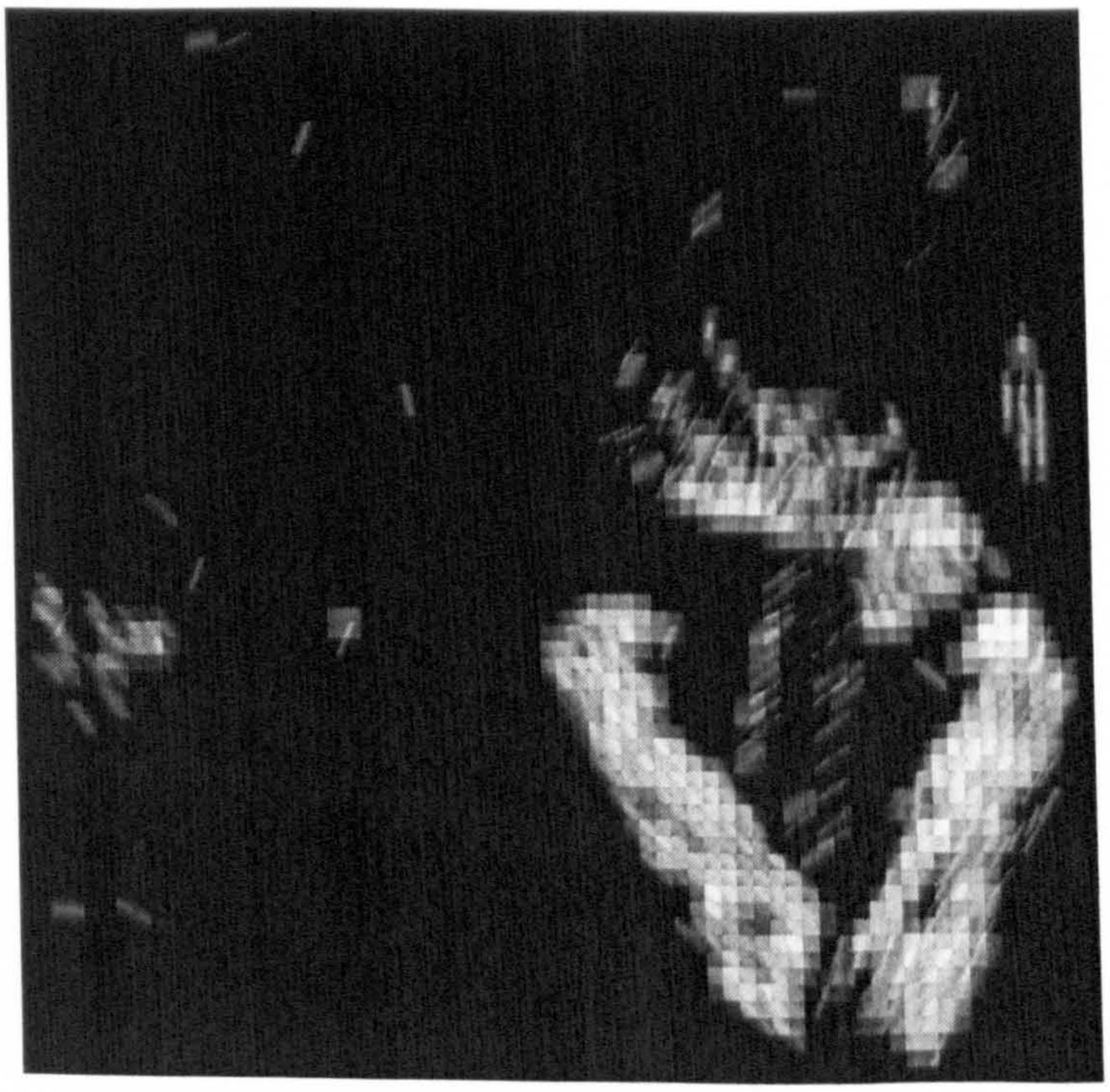
Scale $s = 4$

Figure 5.6: Filter configurations for MFTG-II on 15dB `lena`



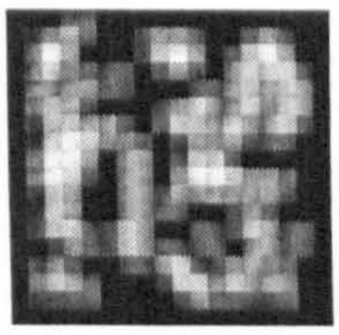
Scale $s = 2$

Scale $s = 3$

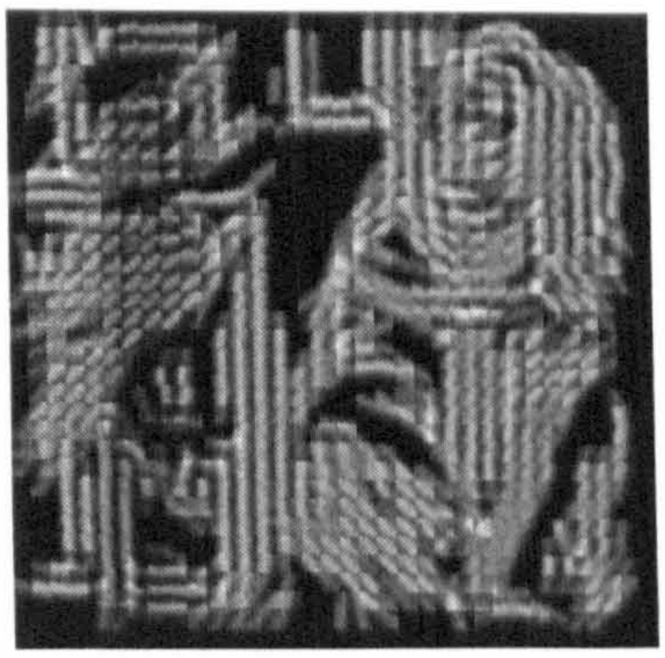


Scale $s = 4$

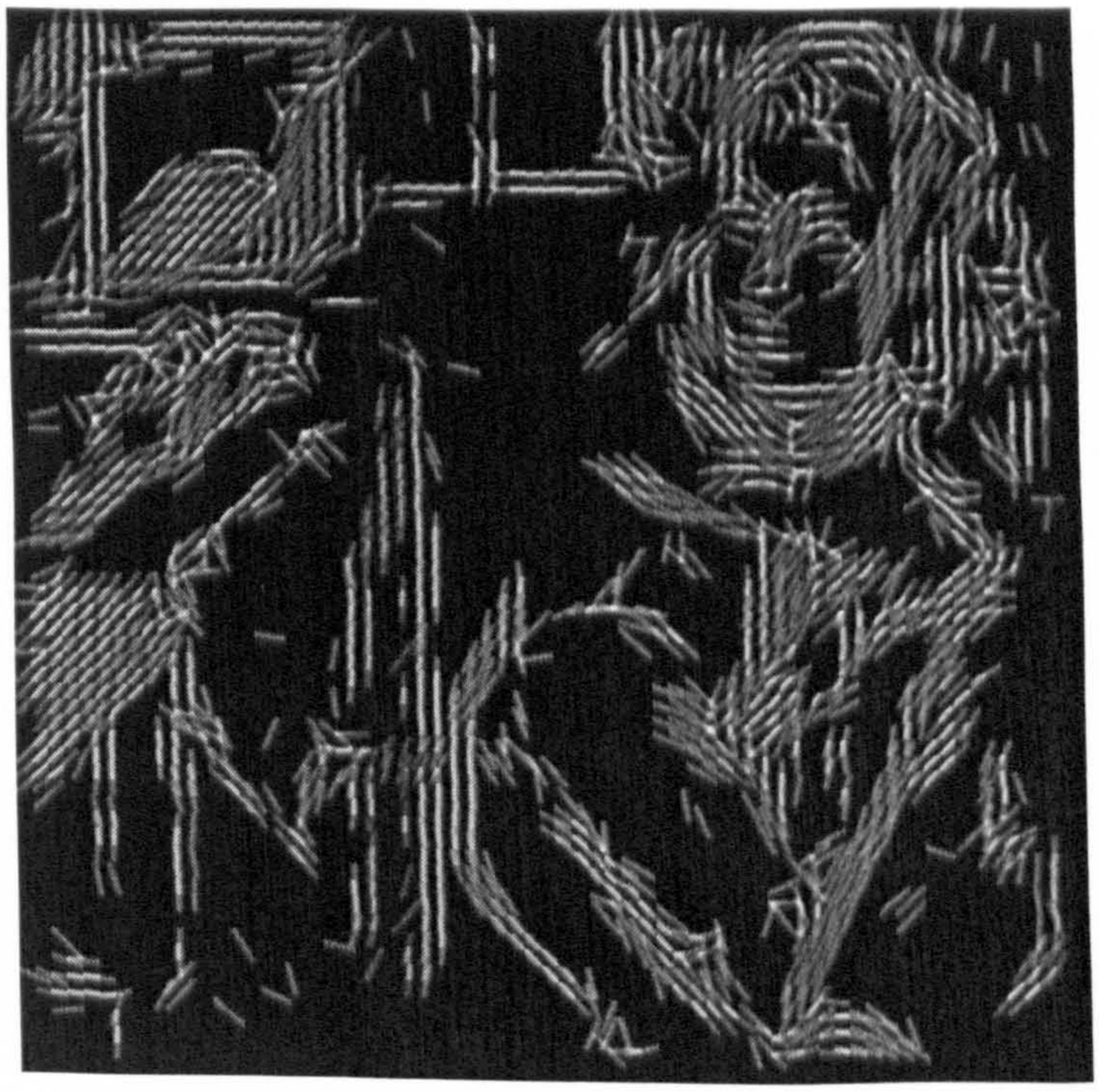
Figure 5.7: Filter configurations for MFTG-I on 15dB `barbara`



Scale $s = 2$



Scale $s = 3$

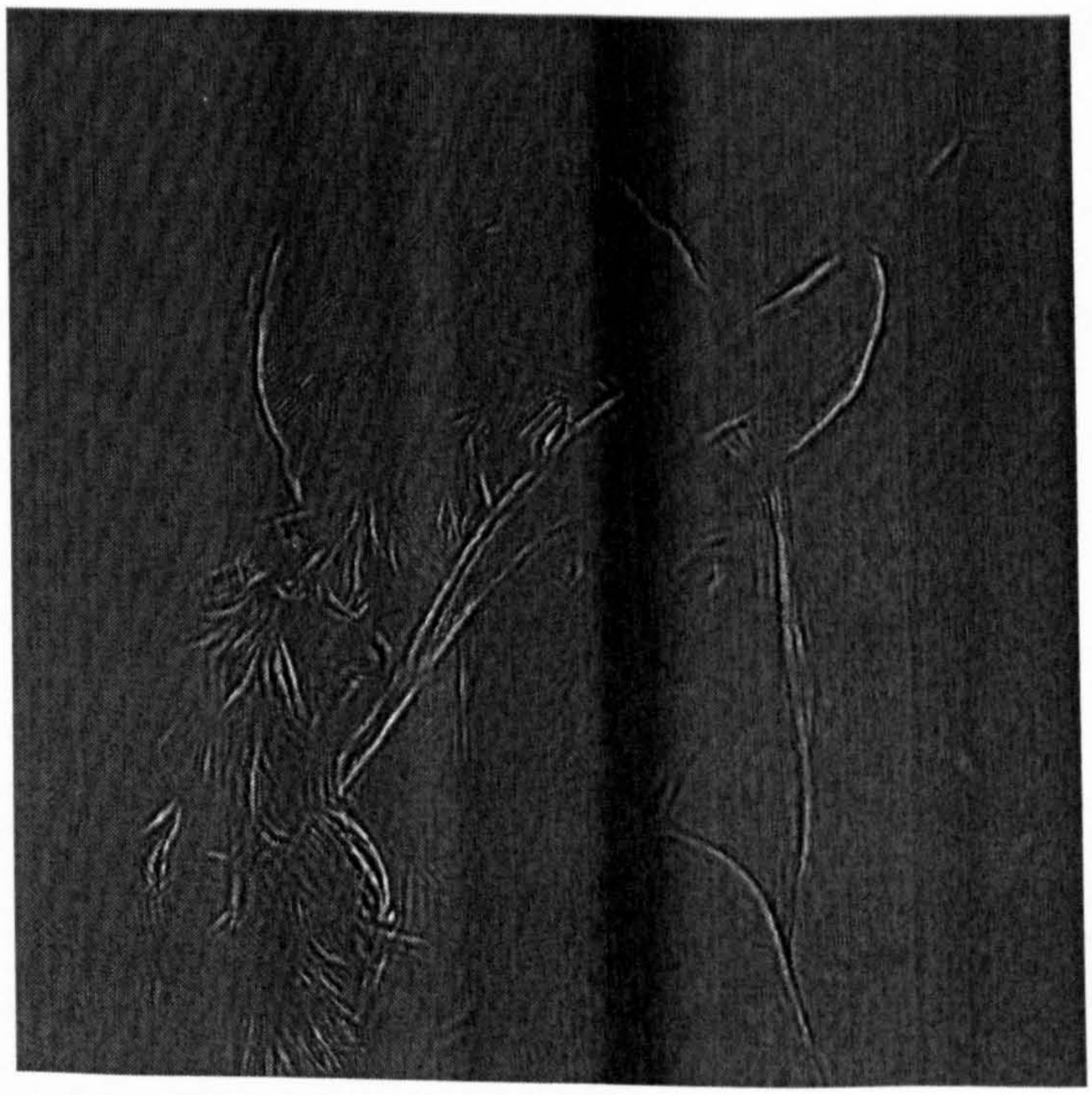


Scale $s = 4$

Figure 5.8: Filter configurations for MFTG-II on 10dB `barbara`

Scale $s = 2$

Scale $s = 3$



Scale $s = 4$

Figure 5.9: Resulting denoised pyramid by MFTG-I on 15dB lena

Scale $s = 2$

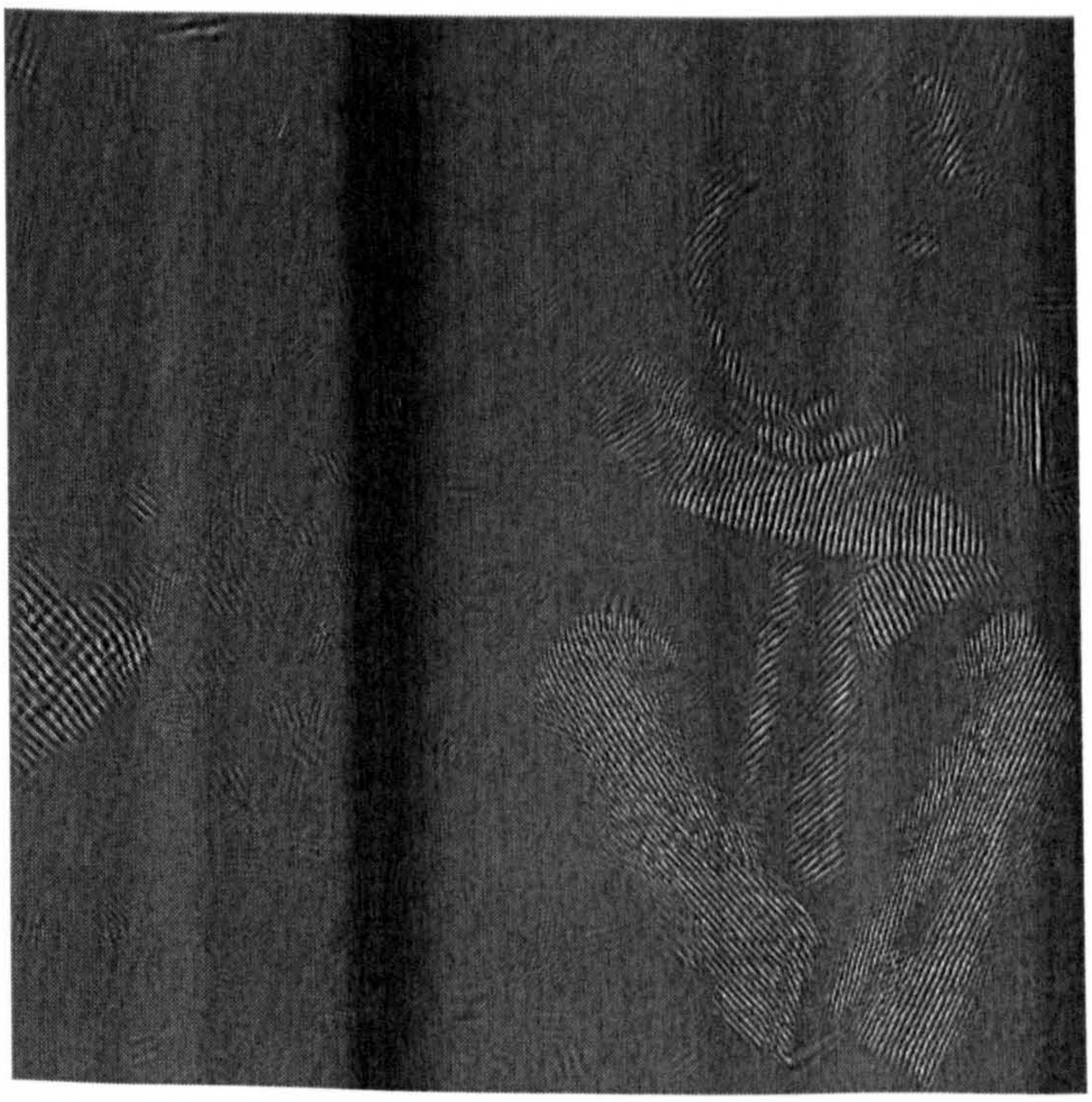
Scale $s = 3$

Scale $s = 4$

Figure 5.10: Resulting denoised pyramid by MFTG-II on 15dB `lena`

Scale $s = 2$

Scale $s = 3$



Scale $s = 4$

Figure 5.11: Resulting denoised pyramid by MFTG-I on 10dB `barbara`

Scale $s = 2$

Scale $s = 3$

Scale $s = 4$

Figure 5.12: Resulting denoised pyramid by MFTG-II on 10dB `barbara`

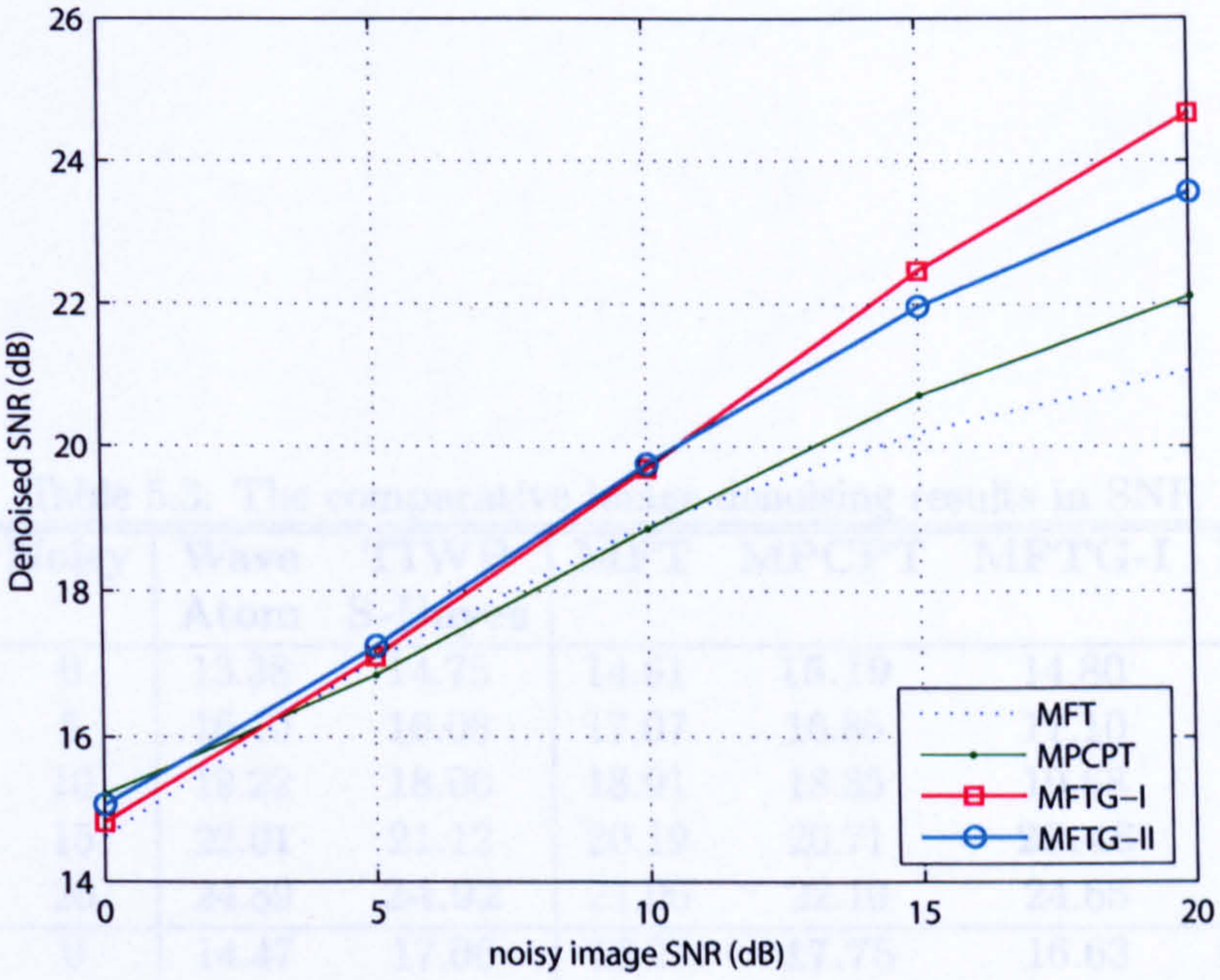


Figure 5.13: Comparative SNR curves for denoising on `barbara`

process recognises linear features mainly due to the multiscale coherence, since a linear edge has response on all frequency bands. The periodic textures, however, are localised within certain frequency subbands, and therefore will be missed by the model. Figure 5.9-5.12 show the corresponding outcome of the pyramid subbands after applying the Gaussian frequency filters shown in Figures 5.5-5.8. It is clear that the MFTG-II can restore a richer set of features, although it also contains more "directional" artifacts than MFTG-I. The readers are encouraged to compare these results with the illustrations presented in Figures 4.16-4.18, which are the outputs from curvelets, MPCT-II and MPCPT on the same noisy images.

Table 5.3: The comparative image denoising results in SNR

Image	Noisy	Wave Atom	TIWP S-Bayes	MFT	MPCPT	MFTG-I	MFTG-II
barbara	0	13.38	14.75	14.61	**15.19**	14.80	15.04
	5	16.40	16.08	17.07	16.85	17.10	**17.24**
	10	19.22	18.00	18.91	18.85	19.68	**19.75**
	15	22.01	21.12	20.19	20.71	**22.46**	21.96
	20	24.89	**24.92**	21.06	22.10	24.65	23.56
lena	0	14.47	17.06	16.27	**17.75**	16.63	17.36
	5	17.60	18.96	18.03	18.99	19.02	**19.69**
	10	20.36	21.08	20.76	21.89	21.60	**22.42**
	15	23.21	23.49	22.58	24.26	24.43	**24.97**
	20	26.08	26.15	23.69	26.31	**26.94**	26.88
fp	0	12.20	12.20	12.48	**13.00**	12.85	12.88
	5	14.64	15.18	15.02	14.60	15.56	**15.57**
	10	17.44	18.29	17.07	17.15	17.51	**18.53**
	15	20.52	**21.44**	18.05	20.24	20.20	21.08
	20	**26.08**	24.74	19.23	23.23	23.29	23.30
grain	0	11.93	12.93	12.60	**13.22**	12.85	12.92
	5	13.96	13.81	13.70	14.29	14.01	**14.41**
	10	16.35	15.87	15.47	16.18	16.30	**17.20**
	15	19.02	18.72	18.51	18.84	19.66	**20.08**
	20	22.02	22.02	20.20	21.38	**22.02**	21.55

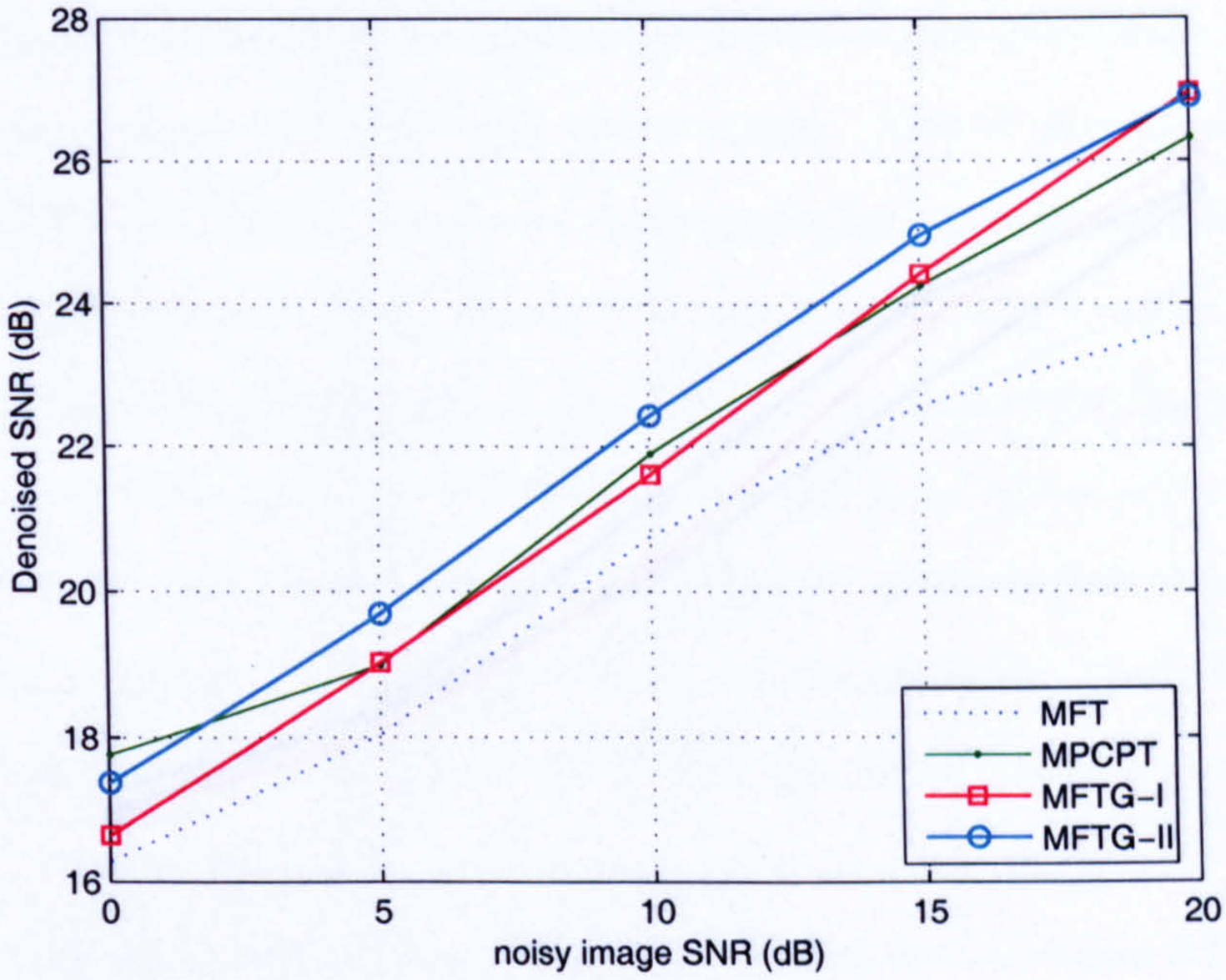


Figure 5.14: Comparative SNR curves for denoising on `lena`

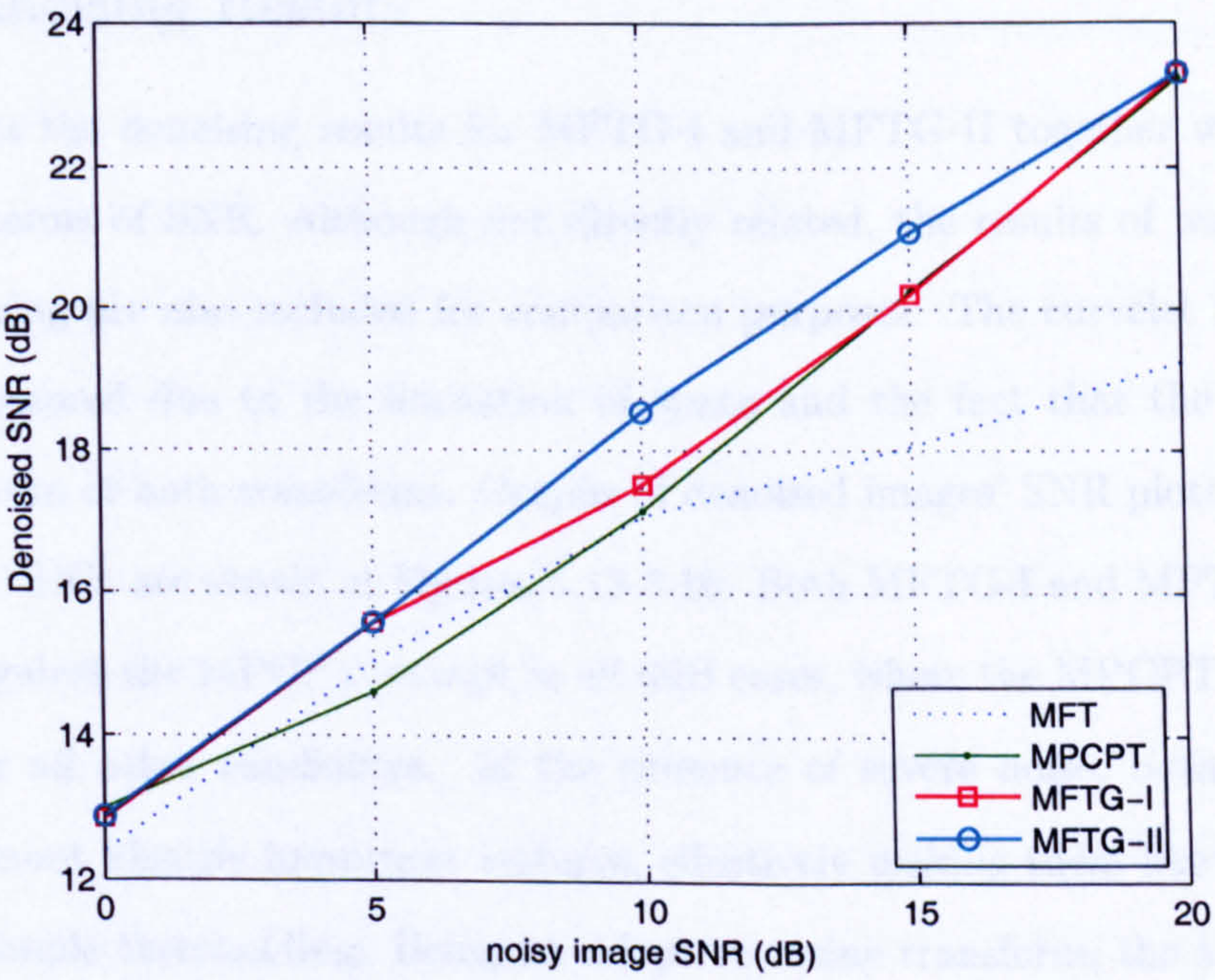


Figure 5.15: Comparative SNR curves for denoising on `fp`

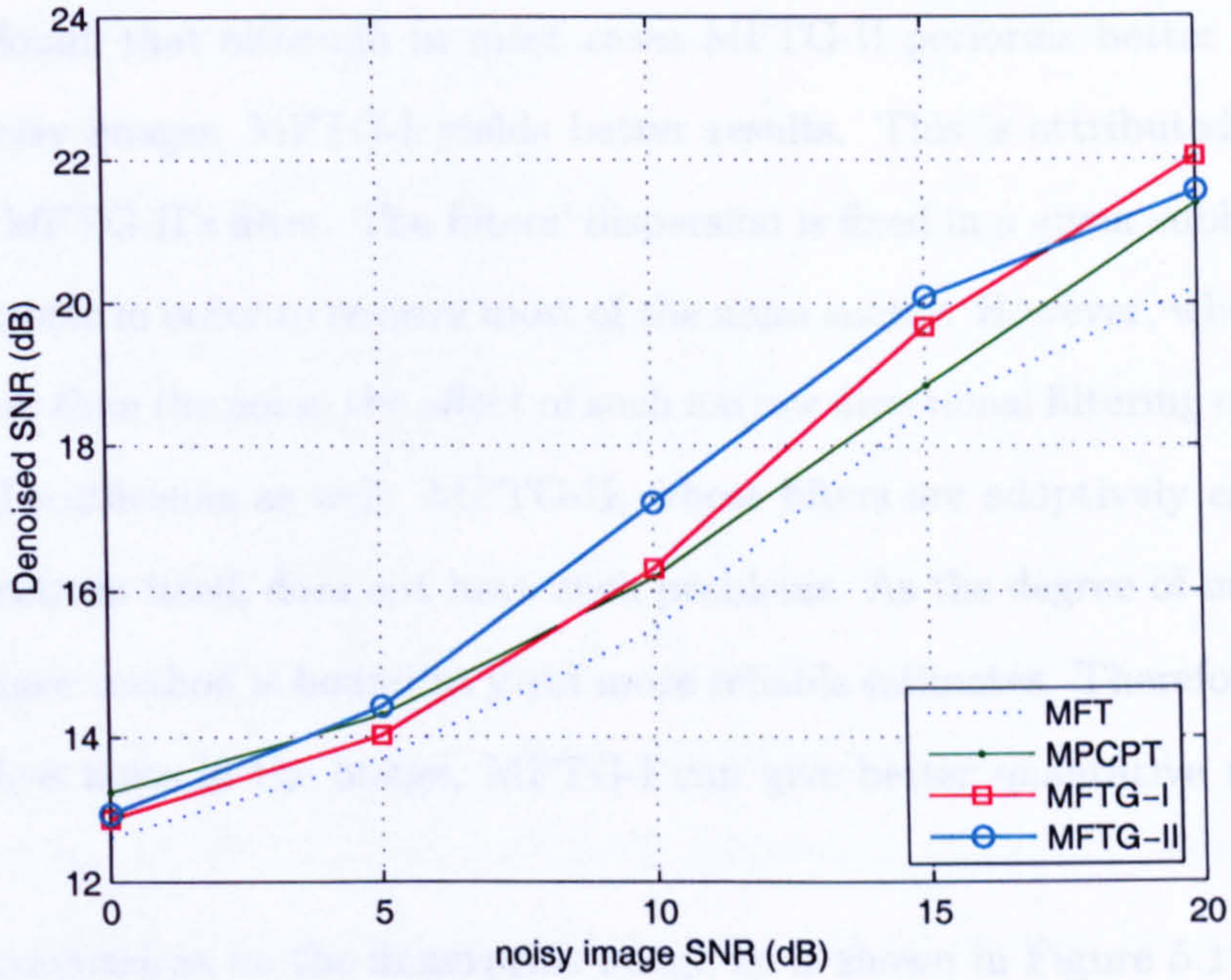


Figure 5.16: Comparative SNR curves for denoising on `grain`

5.5.2 Denoising Results

Table 5.3 lists the denoising results for MFTG-I and MFTG-II together with MFT and MPCPT in terms of SNR. Although not directly related, the results of wave atoms and TIWP denoising are also included for comparison purposes. The curvelet transform and MPCT are ignored due to the limitation of space and the fact that the MPCPT is a generalised form of both transforms. Graphs of denoised images' SNR plotted against the noisy images' SNR are shown in Figures 5.13-5.16. Both MFTG-I and MFTG-II compare favourably against the MPCPT, except in all 0dB cases, where the MPCPT seems to have an edge over all other candidates. In the presence of severe noise, both MFTG-I and MFTG-II cannot identify important features, effectively making them like a conventional MFT with simple thresholding. Being an adaptive cosine transform, the MPCPT clearly is a superior choice in severe noisy situations. However, as the noise level goes down, the

MFTG-I and MFTG-II gain a significant margin over MPCPT.

It is also found that although in most cases MFTG-II performs better than MFTG-I, on 20dB noisy images, MFTG-I yields better results. This is attributed to the static nature of the MFTG-II's filter. The filters' dispersion is fixed in a given subband, which is usually anisotropic in order to remove most of the noise energy. However, when the signal's energy is higher than the noise, the effect of such narrow directional filtering can potentially discard useful coefficients as well. MFTG-II, whose filters are adaptively estimated from the data's spectrum itself, does not have such problems. As the degree of noise decreases, the inertia tensor method is bound to yield more reliable estimates. Therefore, when there is relatively less noise in the image, MFTG-I can give better quantative measures than MFTG-II.

A visual comparison on the fingerprint image `fp` is shown in Figure 5.17 for MPCPT, MFTG-I and MFTG-II. The noisy image's SNR is 5dB, compared with the original. The denoised image of MPCPT has some minor "curvelet"-like directional artifacts. This is caused by the adaptiveness of the MPCPT. The transform seeks a best basis on each of the Radon slices, so that some noisy directional singularities might be kept in less prominent orientations. The reconstructed image using MFTG-I is smoother and almost artifact-free, although a significant portion of the highpass features are discarded. The MFTG-II provides the best result both numerically and visually. The minutiae features are especially visible on the MFTG-II's reconstruction.

Another visual comparison is made on the 10dB `barbara` image in a close-up fashion, showing only a region of interest which contains the stripe patterns on the cloth and certain portion of the woman's arm (Figure 5.18). The illustration also includes the results generated by curvelets and MPCT-II, which were discussed in Chapter 4. While MPCPT's result combines the merits from both the curvelet and the MPCT-II, the output of the MFTG-I recovers more textural details and consequently gains the edge in SNR measure.

Method	Running Time in seconds
Wave atoms	57
TIWP	398
MFT Thresholding	3
MPCPT	1067
MFTG-I	12
Feature Extraction + MFTG-II	45+11 = 56

Table 5.4: Denoising running time comparison

Although compared with MFTG-I, the MFTG-II denoised image does not restore the directional stripe texture as sharply as MFTG-I, the reconstruction around the linear edges (on the arm) is the best among all five denoising algorithms.

Overall, both the MFTG-I and MFTG-II outperform the MPCPT-based denoising, showing their ability to deal with both linear and textural features by using the simple Gaussian frequency filter. The complexity of the denoising process is also dramatically reduced. For a 512×512 grayscale image, the MPCPT requires around 16-17 minutes to perform the denoising on an Intel 2.0GMHZ CPU, as reported in Chapter 4. It took around 12-13 seconds to finish the denoising by the MFTG-I. For the MFTG-II, if the overhead of the linear feature extraction process is included, the whole routine runs for around 50-60 seconds which is comparable to the wave atoms thresholding. If the linear feature information is given, the computational time is around 11 seconds, less than the MFTG-I, since it does not require the inertia tensor estimation, which has $O(M^2)$ complexity (M being the window size) for one block. The table for elapsed time comparison can be found in Table 5.4.

5.5.3 Comparison with the State-of-the-Art Methods

There are various directional transforms reported to date, claiming similar superiority over the conventional orthonormal wavelet transforms. Here, the two MFTG algorithms are

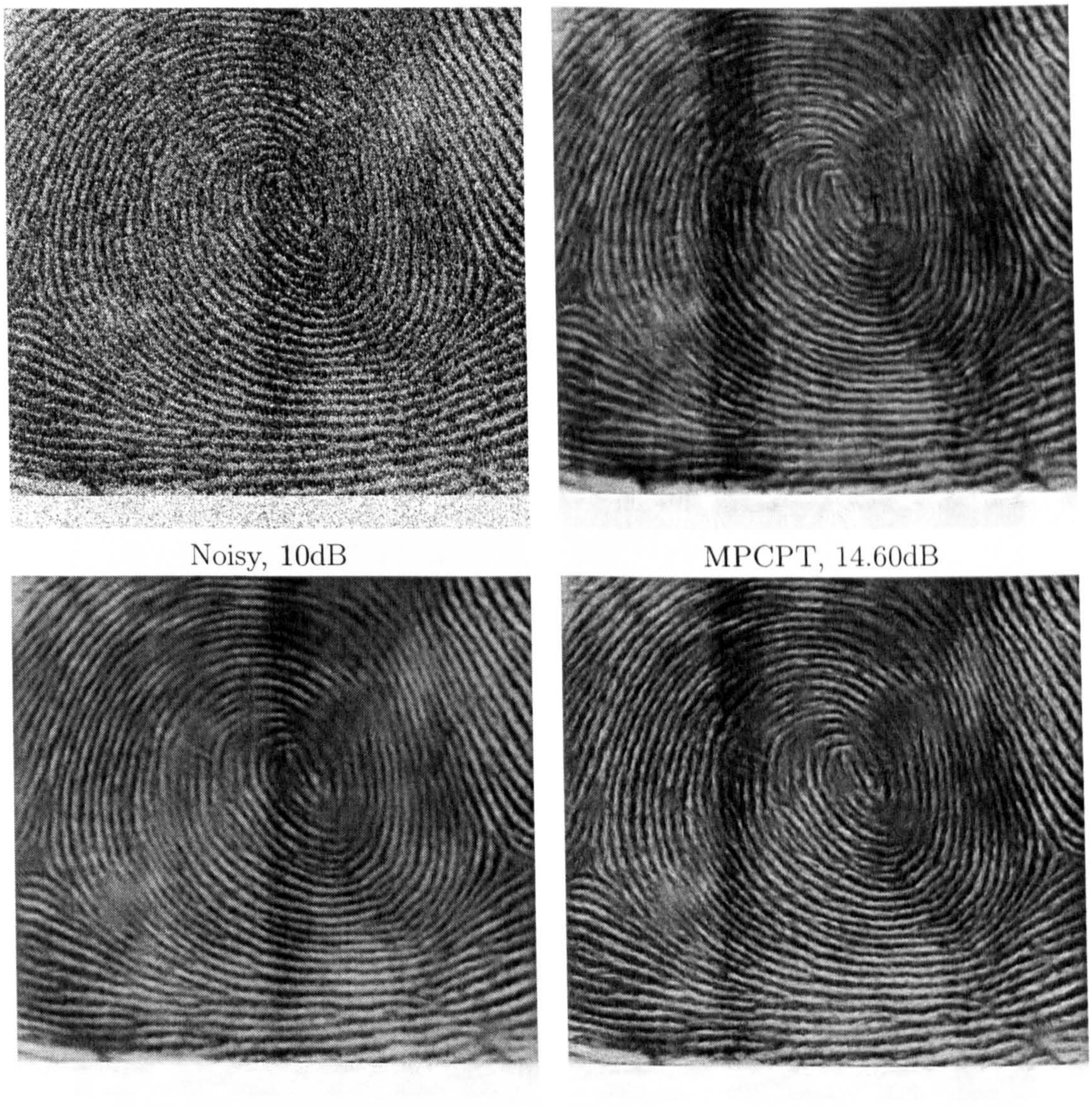
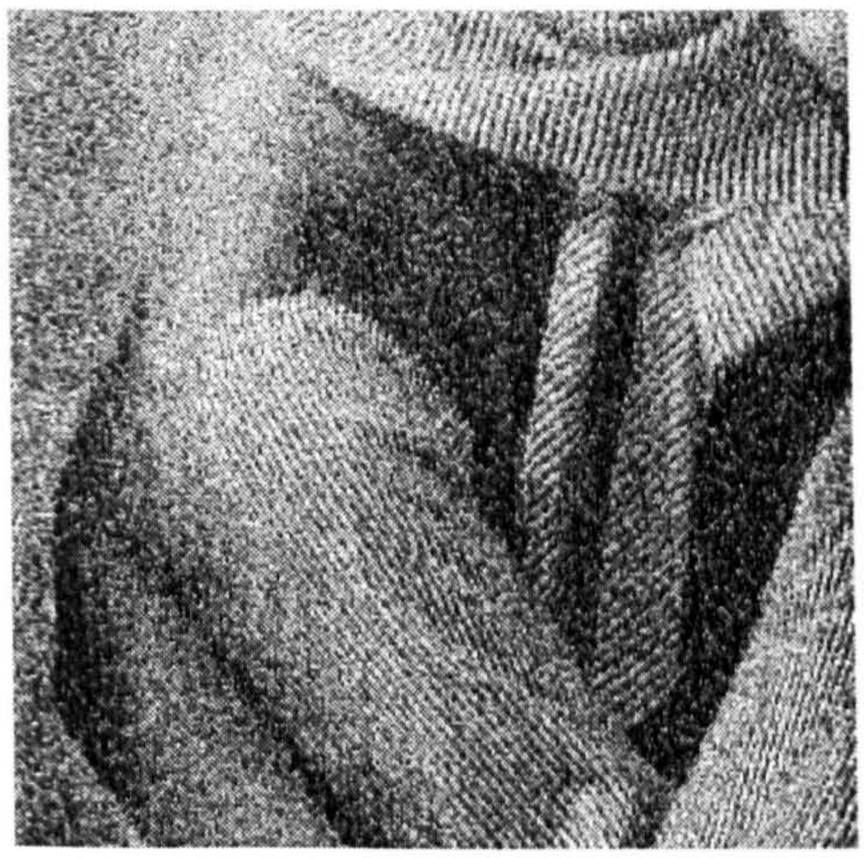
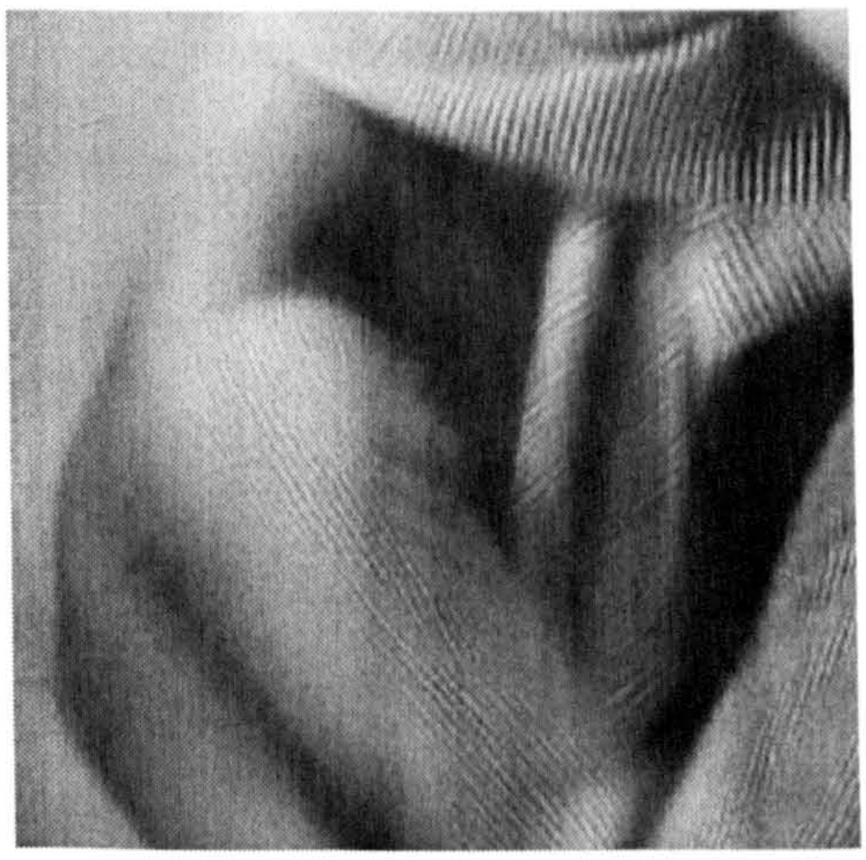


Figure 5.17: Comparative denoising results on 5dB `fp`

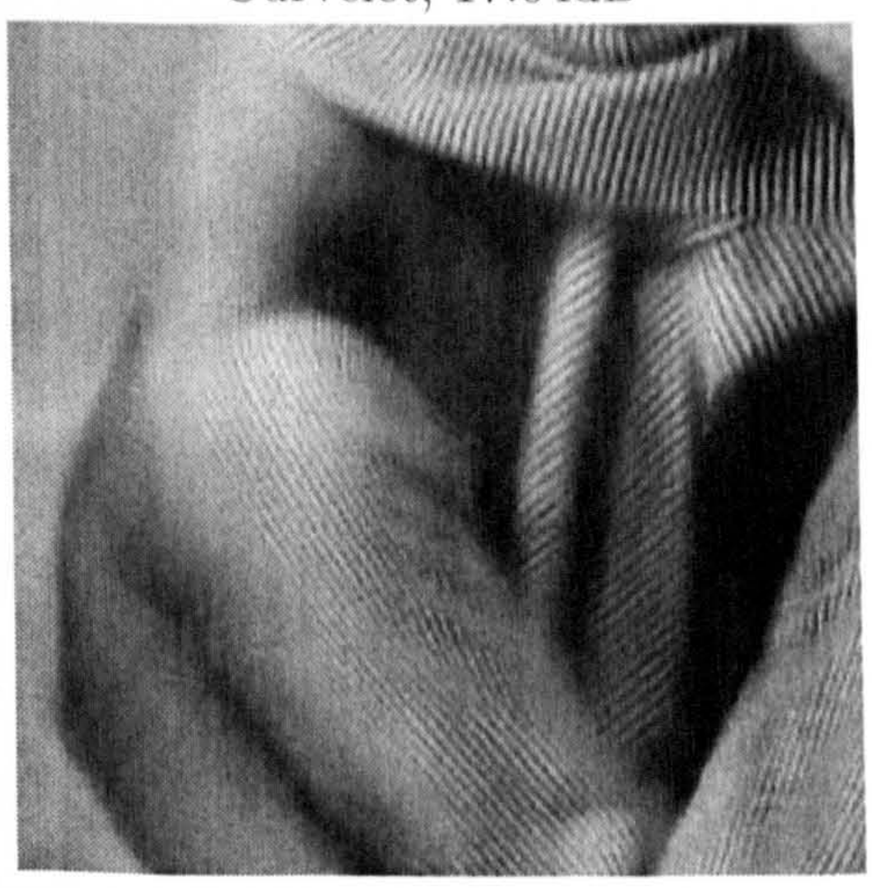


Noisy image, 10dB

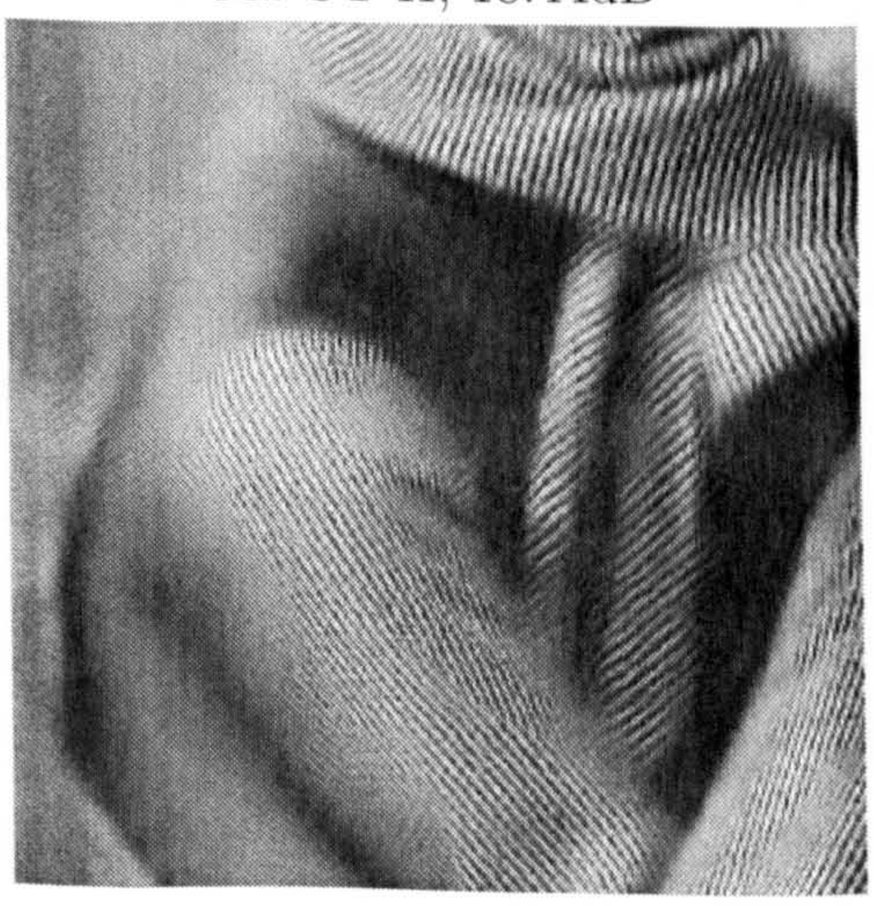


Curvelet, 17.64dB

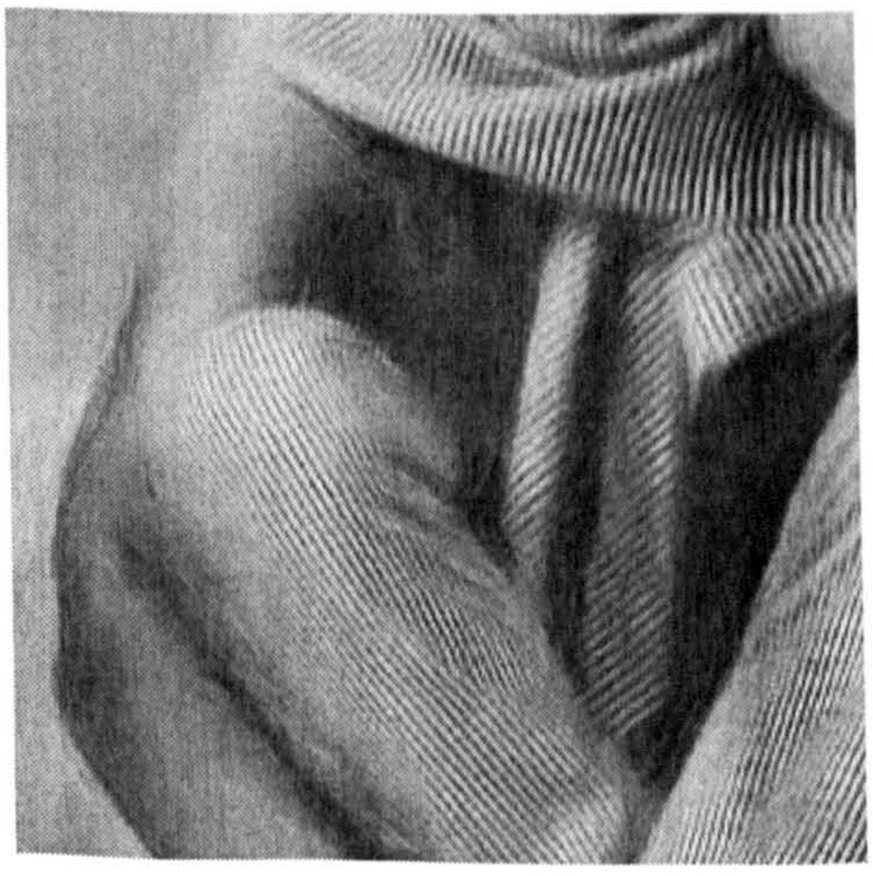
MPCT-II, 18.44dB



MPCPT, 18.85dB



MFTG-I 19.68dB



MFTG-II 19.75dB

Figure 5.18: Closed up comparative denoising results on 10dB `barbara`

compared with a few selected published results. Since most of the published literature uses PSNR as the criteria for image quality measurement, all numbers in comparisons thereafter are measured in PSNR instead of SNR.

In [143], a digital implementation of the *curvelet-99* transform was presented for the image denoising problem. On a noisy `lena` image measured at 22.13dB (PSNR), the authors reported a reconstruction with PSNR of 30.95 dB. With the same image also with noise at 22.13dB, the MFTG-I yields 30.87dB and MFTG-II results in 31.29dB. The underlying curvelet-99 transform is significantly more redundant than the MFT and the computational requirement is also higher, due to the local ridgelet transform which involves the Radon transform.

Contourlet denoising [54] with thresholding on a 24.42dB noisy `lena` gives a reconstruction with 30.47dB PSNR. Both MFTG-I and MFTG-II outperform this by more than a 1dB margin, 32.07dB and 32.19dB, respectively. A non-subsampled version of the contourlet transform (NSCT) was reported in [43]. For a 22.15dB `barbara`, NSCT thresholding gives a 29.41dB result, MFTG-I yields 29.50dB and MFTG-II 28.89dB. An improved Local-Adaptive Shrinkage scheme (NSCT-LAS) gives 30.60dB reconstruction. Another contourlet denoising scheme by employing a multiscale directional statistical hidden markov tree model (CT-HMT) [124] gives a 28.18dB result on the 18.88dB `lena`, while MFTG-I gives 29.03dB and MFTG-II gives 29.80dB.

It is also reported that applying the bandelet transform [116] in denoising a 20dB `barbara` image can improve the filtered image quality to 27dB. The two proposed algorithms outperform this result, producing 28.20dB (MFTG-I) and 27.73dB (MFTG-II) denoised images.

The aforementioned comparisons are assembled in Table 5.5. It is obvious, that even faced with more redundant frames, the combination of MFT and the Gaussian frequency filter perform well, with competitive storage requirement and computational simplicity,

Table 5.5: The comparison between the MFTG-I,II and other directional transforms

Image	Noisy	Transform Name	Transform Result	MFTG-I	MFTG-II
lena	18.88	CT-HMT [124]	28.18	29.03	**29.80**
	20.2	Bandelet [116]	30.3	29.97	**30.58**
	22.13	Curvelet-99 [143]	30.95	30.87	**31.29**
	24.42	Contourlet [54]	30.47	32.07	**32.19**
barbara	18.72	CT-HMT [124]	25.27	**27.28**	27.13
	20.00	Bandelet [116]	27.00	**28.20**	27.73
	22.15	NSCT [43]	29.41	**29.70**	28.89
	22.15	NSCT-LAS [43]	**30.60**	29.70	28.89

outperforming most of the state-of-the-art methods.

5.6 Summary and Discussion

In this chapter, two variations of image denoising algorithms which use a directional Gaussian frequency filter on the MFT local spectrum have been introduced. Such approaches embrace the idea of incorporating a linear feature model in a transform domain which is suitable for representing directional periodic oscillation patterns. As a result, the two algorithms can effectively capture both linear edges and texture patterns. Their results compared favourably, both in terms of quality and complexity, with the MPCPT presented in Chapter 4, which also follows the concept of the *image=edge+texture* philosophy.

Although both algorithms use the same idea of Gaussian frequency filtering, they are fundamentally different in how the filters are constructed. The MFTG-I variation only considers the linear feature in terms of the spectrum magnitude model, and estimates the Gaussian model directly from the data. The MFTG-II version on the other hand, incorporates the information given from the linear feature extraction process of Chapter 3 into the filter configuration, which equips it with a better ability to recover linear edges from the noisy data. Although the MFTG-I is usually outperformed by MFTG-II, its

main merit lies in its simplicity. It is also observed that MFTG-I can sometimes give better results quantitatively when the noise is less severe, and on images with significant textures, such as `barbara`. Both of the algorithms compared well with other published denoising results using state-of-the-art directional transforms.

Chapter 6

Conclusions

The work described in this thesis has concentrated on the problem of representing both edges and textures in natural images with multiscale directional transforms. The contribution of the thesis mainly consists of three parts:

1. The use of MFT in extracting linear features which correspond to object boundaries from noisy images.

2. A new transform called the Multiscale Polar Cosine Transform (MPCT), which is suitable for representing texture and by extending the cosine packet analysis on the Radon slices, the Multiscale Polar Cosine Packet Transform MPCPT is able to accommodate both linear singularities as well as directional oscillating patterns.

3. By applying a signal dependent Gaussian frequency filter on local MFT spectra, fast denoising algorithms which can preserve both edges and textures were obtained.

This chapter gives a recapitulation of the thesis, including these technical contributions, and finally concludes the thesis with a brief discussion on its limitations and future directions for the research.

6.1 Thesis Summary

A general introduction on the motivation of finding an appropriate directional representation for natural images was given in Chapter 1. While it is widely accepted that natural images should be treated as a combination of objects with different regions of texture separated by edges, the traditional separable wavelet basis is not an ideal sparse representation for such images. The attempt to improve over the wavelet basis led to various approaches. Some approaches try to exploit the importance of directional coherence on the images but emphasise one type of features or another, either edges or texture. Some use an overcomplete dictionary to solve the problem (Matching Pursuit [95], Basis Pursuit [34]), while others are focussed on how different image ingredients can be separated from each other, so that they can be represented individually (e.g., [145]). It was also suggested that a transform which is multireslution, directional and well-localised in both frequency and space is highly desirable in order to carry out different image processing tasks.

There are a variety of such transforms proposed in the literature which can improve the approximation rate for directional patterns over wavelets. The bulk of these transforms defines the landscape in the field of applied computational harmonic analysis. The curvelet [28], contourlet [54] and shearlet [63] transforms achieve very similar frequency tiling and are able to approximate piecewise smooth images with $\mathbf{C^2}$ boundaries by the rate $O(M^{-2}(\log M)^3)$ (M being the number of coefficients retained), a significant improvement over the $O(M^{-1})$ by ordinary wavelets. Complementary directional transforms which have oriented sinusoids as basis functions instead of linear singularities were also discussed, such as the MFT [18], brushlets [105] and wave atoms [49]. Notably, bandelets [116] can deal with both directional edges and textures by exploiting the intensity regularity along the *"geometric flow"* but are less efficient with non-directional textures as compared to Fourier type bases.

There are also certain types of parametric representations for edge structures which require the parameters of the underlying geometric model. A common choice of model is the straight line, whereby the parameter set for one segment of the edge consists of the line's scale, position and direction in much the same way as curvelet coefficients are indexed (see curvelet-99 [143]). It was also noted that the MFT, while being a natural basis to represent multiscale harmonic patterns, is also able to extract linear feature parameters.

However, the MFT-based frequency linear feature model overlooked the problem of noisy images and does not work well when heavy noise is mixed with the signal. Chapter 3 dealt with this problem by a combination of different approaches, by introducing a suitable "signal+noise" model, a refinement stage and a robust classification method to distinguish significant linear features. To start with, the local MFT magnitude spectrum is modelled as a mixture of two Gaussian functions, one representing the noise energy with fixed dispersion and an adaptive one whose parameters are estimated via the EM algorithm. Then a vector-based Kalman filtering refinement is applied to the estimated feature parameters by regarding each scale as one state. Finally, it used a simple voting scheme from several feature significance measures in order to determine if a particular block contains a linear feature or not. The results compared favourably with other parametric linear feature estimation algorithms.

It is well known that the Fourier transform assumes periodic extension on signal boundaries, which causes a poor approximation rate and leads to pseudo-Gibbs phenomenon on the boundaries. Although the DCT can remedy this shortcoming by using an even symmetric extension, the resulting 2-D DCT basis does not have the orientation selectivity. In Chapter 4, a new construction of a cosine-type transform was presented and adopted into a multiscale representation. The new transform, called the polar cosine transform, has basis functions of oriented sinusoids similar to the Fourier transform. Two possible digital implementations were presented. The first is a Radon transform based version,

whose construction is similar in its philosophy to that of ridgelets. Instead of applying the 1-D wavelet transform on the Radon slices, the polar cosine transform uses the cosine transform. The second digital implementation uses the Cartesian separable 2-D DCT-IV and DST-IV with a butterfly operation. The resulting tranform, the PCT-X is a real-valued version of the Fourier transform in which the coefficients can be organised in both Cartesian and polar coordinates. Compared with the Radon-based implementation, the PCT-X is less redundant, introducing twice the number of coefficients than the original signal while the Radon based scheme has a redundancy factor of 4 due to the underlying Fast Slant Stack [6]. However, the Radon based scheme has the flexibility of using different trigonometric basis as the ridge profile and with the DCT-II, which uses even boundary extensions on both sides of the block boundaries, it is more suitable for image representation than DCT-IV and DST-IV, whose boundary extensions are respectively even and odd. Also, by performing a best basis search with the local cosine basis on the Radon slices, one can successfully represent both linear and directional harmonic structures (MPCPT). Experiments suggest that the MPCPT consistently outperforms the MPCT by some 0.5-1dB in SNR in denoising and the reconstruction is visually superior to results obtained from wavelet packets and wave atoms.

Although the "cosine packet on Radon" approach can provide a good representation for natural images, its computational cost is extremely high, which would severely limit its use in many applications. Chapter 5 discussed an alternative approach, to represent both edges and textures which is directly related to the frequency domain linear feature model presented in Chapter 3. The idea behind the new approach is that, instead of a having a large library of basis functions, the joint *"edge + texture"* representation can be achieved by incorporating the edge model in a basis which is suitable for textures. This concept is realised in modelling the Fourier spectrum as a bivariate Gaussian function, which can adaptively "cover" the significant coefficients in the transform domain. Efficient denoising

results were obtained by applying the estimated Gaussian functions as the frequency filter on linear feature, directional and non-directional textures. The Gaussian frequency filter was then introduced into the MFT framework for the purpose of image denoising. Two different algorithms were proposed which are differentiated by their filter configurations. The first algorithm (MFTG-I) obtains the filters' parameters directly from the data itself, so the estimated filters are of various dispersions according to the magnitude spectrum, but are prone to error in heavy noise. The second algorithm (MFTG-II), on the other hand, uses a fixed dispersion of high anisotropy for every filter at a given scale, which allows a narrow, oriented band to pass through. The MFTG-II also uses the information provided by the linear feature extraction process in order to decide where and which orientation the filters should be applied on. As a result, the MFTG-II can recover more linear features than the MFTG-I, so performs better with heavy noise. Since the algorithms do not require the Radon transform, they are numerically more stable than MPCPT when reconstructing the signal from a few coefficients. They were compared with other denoising results obtained by thresholding in different directional representations. Most of the other representations are more redundant than the MFT. Still, the proposed algorithms have shown some improvements over most of the other candidates, while the computational requirement is significantly reduced comparing to MPCPT.

6.2 Limitations and Future Work

Although the work in this thesis has achieved success, it is also limited in several aspects that can potentially be improved. These limitations are summarised as follows:

- **The single feature hypothesis**

  In the linear feature analysis of Chapter 3, it was assumed that within a small spatial window of size 16×16, there is at most one significant linear feature. While in most

cases such an assumption is true, the simplification of the model makes it unable to deal with regions of corners or crosses. Subsequently, in Chapter 5, since the same single feature model was adopted, a spectrum with several directional features is modelled as one isotropic Gaussian (the chessboard pattern, for example, which can be seen as two perpendicular oriented sinusoid ridges intersecting with each other). This may potentially include more undesired noise energy in the filtered spectrum which may degrade the reconstruction. Therefore, the linear feature model can be further improved to accommodate multiple features with a trade-off in computational efficiency.

The extension of the MFT frequency model into multiple-feature case has been considered in other works. Davies [45] used a K-means clustering based approach to separate different linear features in order to estimate their parameters. However, the number of clusters is unknown in practical situations, so the model has to test the degree of match between the data and the model, which can be inconvenient in some applications. Wang et al. [155, 156] also used similar technique for modelling multiple features in retinal vascular structures with a Hermite model in the spatial domain. In a more recent work [113], Park et al. proposed a Radon-on-Fourier treatment to obtain a frequency contour signature by performing the Radon transform on the Fourier magnitude spectrum. The resulting contour defines a multi-directional frequency window which is flexible enough to cover scattered high amplitude coefficients. A smooth version of the window was used in denoising with 32×32 block size and obtained similar results to the MFTG-II, with notable improvement at low noise levels.

- **Feature parameter refinement**

  In Chapter 3, in order to take the adavantage of the multiscale coherence of edges, a

Kalman filter is applied on the linear feature parameters so that they can be refined from the more reliable estimates at the coarser scales. In effect, the filter combines the estimates from both scales in a linear fashion. However, such a linear vector-based multiscale refinement cannot fully cope with the complexity of the images, especially when the child's estimate differs from the parent's. The filter will then rely on the parent's estimate and discard the possible innovation on the finer scale. Therefore, in many cases, a good estimate from the child could be "refined" wrongly due to its innovation. In this case, the vector-based Kalman filter is less ideal for parameter refinement, despite its simplicity.

For improvement, one may incorporate an extra step to check the estimated parameter's consistency with the original data. If the model fits the data well, then a refinement is not necessary. This will allow "children" containing newly appeared features at the finer scale to keep their innovation. Of course, the consistency between the feature parameter and the data is central in this, especially when the data are noisy.

Another alternative is to use a more sophisticated nonlinear stochastic model such as a *Markov Random Field* (MRF) [74]. In this case, a slightly modified MRF model, called the Multiresolution MRF (MMRF) [160], should be used by taking the parent of a *site* (block) as one of its spatial neighbours. In this way, the parameter value can be regarded as the *state* of the site and by defining an appropriate *potential function*, the parameter can be refined iteratively according to the interaction with its neighbours. Consequently, the adjustment of the parameters can also take the intra-scale coherence (the consistency of orientation between the neighbours) as well. The major difficulty in this approach is that the intra- and inter-scale coherences of the linear edge have different characteristics, so it is less obvious how they can be combined in a potential function.

- **The phase spectrum model**

  As mentioned in Chapter 3, a line-type feature has a linear phase response in the Fourier transform, where the gradient of the phase is proportional to its centroid position. It is also suggested that the phase defines most of the characteristics of the signal [112]. In Chapter 5, although a model of the magnitude spectrum was presented as a zero-mean Gaussian function and linear features' orientations were used, no attempt has been made to use the phase spectrum in denoising. As a result, the noisy phase gives directional noise artifacts surrounding the edges in the reconstruction. One way to overcome this is to use the linear feature parameters to syntheses the feature's Fourier spectrum. This idea has been reported in [18] with noise-free images. An example of the `lena` image was presented with the highpass subband entirely reconstructed from the feature parameters. While the reconstruction is visually pleasing and can be useful , it is also known that such synthetic images do not give very high numerical measures of fidelity to the original image in terms of SNR.

- **Affine self-similarity in textures**

  It is known that natural images exhibit self-similarity, a property which was exploited by the *"fractal"* methods described by Barnsley and Jacquin [83]. On textures, such self-similarity means there is a strong coherence between the texture patches, so that they can be approximated by a prototypical *texton* [85] undergoing an affine transformations. Such an affine texture representation was studied in depth in a series of works [78, 19, 11] in the context of MFT. While this thesis has concentrated on developing directional representation and analysis for images, the inter-block correlation in the textural regions was overlooked, for which the affine texture model can provide a good complement. A combination of these two approaches can provide a powerful image representation. Also, it was also shown that directional analysis can be used to aid the affine parameter estimation. In [11], a Gaussian mixture model

similar to the one used in this for the magnitude spectrum was employed to estimate the affine transformation.

Besides the limitations which can be further improved, there are also several directions for extension or application of the methods discussed in this work.

- **Applications of MPCT-X**

  In this thesis, the PCT-X was discussed as an alternative real-valued transform to the conventional Fourier transform with a redundancy factor of 2. The magnitudes of the PCT-X's coefficients are very similar to the magnitude spectrum of the Fourier transform. It is, therefore, possible to apply the same linear feature analysis with the MPCT-X analogue to the MFT. Although the MFT-based linear feature extraction process relies on the phase spectrum to estimate the centroid position of the line, while the real-valued MPCT-X does not have the phase information, Popovici and Withers [128] suggested that the moment-based linear feature estimation approach can be extended to the DCT and other linear transform domains such as PCT-X. If such a linear feature model can be established in the PCT-X domain, it is also expected that the Gaussian frequency filter idea of Chapter 5 can be used with the MPCT-X in denoising. Since the PCT-X has shown better results in the nonlinear approximation tests in Chapter 4, the MPCT-X denoising should also have an edge over the MFT based denoising. Also, since with the PCT-X it is possible to construct a local adaptive basis, it is also likely that a compression scheme can be derived by seeking a local best PCT-X basis with the Gaussian frequency window.

- **3-Dimensional extension**

  Although the main focus of this thesis has been limited to directional representations for 2D images, there is also an increasing interest in extending the 2D directional bases into 3D case in order to analyse video sequences or medical volume data.

The number 20 and 68 frames in `missa` sequence

The extracted planelet features from `missa`.

Figure 6.1: 3D planar feature visualisation for an image sequence

In [166], Ying et al. extended the digital curvelet transform into 3D. The basis functions of the 3D curvelet representation are surface-like singularities instead of a "curve" in the literal sense. However, since the digital implementation is based on the FFT frequency windowing, it requires access to the entire volume. Therefore the storage requirement is prohibitively large, requiring some 320MB of memory for a $128 \times 128 \times 128$ volume.

Therefore, in order to analyse a 3D signal, a block-based approach is more appropriate, since it requires less memory and for video this means less delay in frames. The extension of MFT to 3D was presented in [10] which can represent both 3D planar and linear feature in medical volume images. Later this type of representation was studied under the name of *"planelets"* [132], due to its ability to represent a planar feature by a line in the magnitude spectrum analogue to the linear feature in 2D. As a demonstration of some preliminary work, an example of using the planelets in extracting planar structures from a $128 \times 128 \times 128$ image sequence (the *Miss America*, aka `missa`) is shown in Figure 6.1.

Moreover, it would also be useful to have the MPCT extended into 3D as well. For the Radon based scheme, there has been an implementation of the 3D Radon transform [139], so one possible extension is just simply applying the 1D DCT onto the 3D Radon. For the PCT-X variation, however, the higher dimensional extension is less than obvious. Appendix A gives a generalised form of the PCT-X transform in arbitrary dimensions and a detailed account of its 3D digital implementation.

6.3 Concluding Remarks

The thesis has proposed a linear feature estimation method for extracting edge structures from noisy images with MFT and the multiscale polar cosine transform, which is suitable to

represent textures in terms of oriented harmonics. Motivated by the linear feature model, the Gaussian frequency filter was applied to the denoising problem with MFT which can preserve both edges and textures. The effect of the MFT combined with a Gaussian frequency filter can be regarded as a form of adaptive wave packets implemented in a fixed transform and can be an effective tool for directional image analysis.

Appendix A

3D PCT-X Extension

A.1 Generalisation

In Chapter 4, we have discussed the 2D digital construction of the $N \times N$ PCT-X basis in the form of

$$\psi^{\mathcal{X}}_{k_1,k_2}[\xi_1,\xi_2] = \cos\left[\frac{\pi}{N}\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)+\left(\xi_1+\frac{1}{2}\right)\left(k_1+\frac{1}{2}\right)\right]. \tag{A.1}$$

To generalise this into an arbitrary dimension n, let $\boldsymbol{\xi} = [\xi_1,...,\xi_n]^T$ be the spatial coordination index and $\boldsymbol{k} = [k_1,...,k_n]^T$ be the frequency index, the n-dimensional PCT-X basis of size N^n is defined as:

$$\psi^{\mathcal{X}}_{\boldsymbol{k}}[\boldsymbol{\xi}] = \cos\left[\frac{\pi}{N}\sum_{i=1}^{n}\left(\xi_i+\frac{1}{2}\right)\left(k_i+\frac{1}{2}\right)\right]. \tag{A.2}$$

It is interesting to see that in 1D, the PCT-X reduces to

$$\psi^{\mathcal{X}}_{k}[\xi] = \cos\left[\frac{\pi}{N}\left(\xi+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]. \tag{A.3}$$

which is exactly the DCT-IV basis in 1D. Therefore the PCT-X can be regarded as an alternative non-separable extension of DCT-IV in higher dimensions. For comparison, the n-dimensional DCT-IV formed by the tensor product has the basis functions :

$$\psi_{\boldsymbol{k}}^{\mathcal{C}}[\boldsymbol{\xi}] = \prod_{i=1}^{n} \cos\left[\frac{\pi}{N}\left(\xi_i + \frac{1}{2}\right)\left(k_i + \frac{1}{2}\right)\right]. \tag{A.4}$$

which does not have selectivity of orientation.

A.2 3D Implementation

The PCT-X in 2D can be implemented by the *butterfly* operation between the 2D separable DCT-IV and DST-IV as presented in Chapter 4. More specifically, it requires one addition for $k_1, k_2 \leq 0$, and a substraction for $k_1 \leq 0, k_2 < 0$ (note that the DCT-IV and DST-IV only have non-negative frequencies). However, in 3D the situation is more complicated.

To start with, the 3D PCT-X basis can be derived from equation (A.2):

$$\psi_{k_1,k_2,k_3}^{\mathcal{X}}[\xi_1,\xi_2,\xi_3] = \cos\left[\frac{\pi}{N}\left(\xi_1 + \frac{1}{2}\right)\left(k_1 + \frac{1}{2}\right) + \left(\xi_2 + \frac{1}{2}\right)\left(k_2 + \frac{1}{2}\right) + \left(\xi_3 + \frac{1}{2}\right)\left(k_3 + \frac{1}{2}\right)\right]. \tag{A.5}$$

For the sake of convenience, let $A = \frac{\pi}{N}(\xi_1 + 1/2)(k_1 + 1/2)$, $B = \frac{\pi}{N}(\xi_2 + 1/2)(k_2 + 1/2)$ and $C = \frac{\pi}{N}(\xi_3 + 1/2)(k_3 + 1/2)$, so that the above equation can be expressed as

$$\psi_{k_1,k_2,k_3}^{\mathcal{X}}[\xi_1,\xi_2,\xi_3] = \cos\left[A + B + C\right]. \tag{A.6}$$

Because the DCT-IV and DST-IV's frequency indices are non-negative, the $A+B+C$ case applies when k_1, k_2, k_3 are all non-negative as well, where there are 7 different cases. Due to Hermite symmetry, only half plane is required. The quadrants in the half plane are

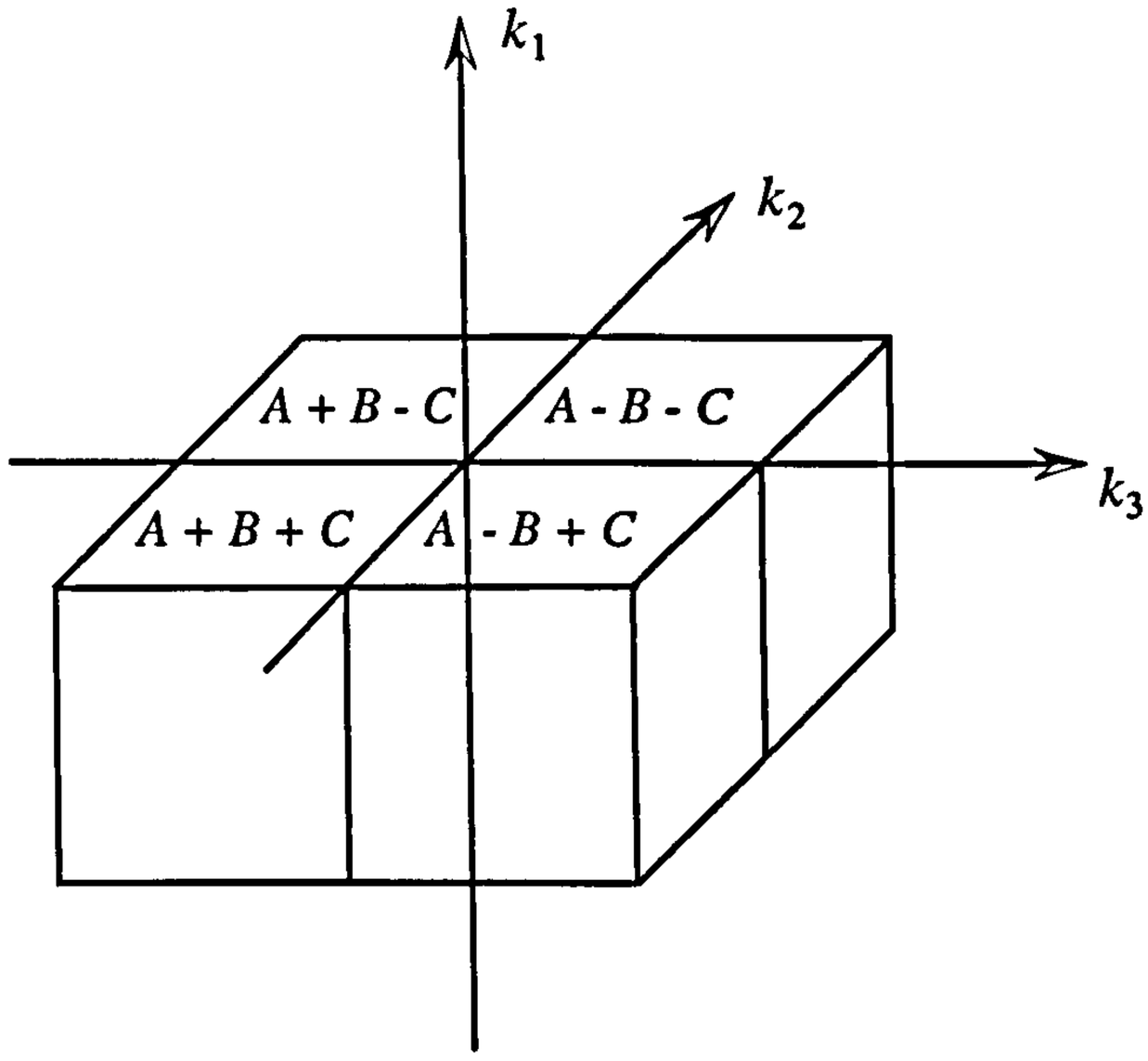


Figure A.1: The quadrants in the frequency half-plane of 3D PCT-X

depicted in Figure A.1 with their corresponding basis definition in terms of A, B, C.

To see how these quadrants can be obtained by Cartesian separable transforms, we need to expand the following cases:

$$\begin{aligned}\cos[A+B+C] &= \cos[A+B]\cos C - \sin[A+B]\sin C \\ &= (\cos A\cos B\cos C - \sin A\sin B\cos C) + (\sin A\cos B\sin C + \cos A\cos B\sin C).\end{aligned} \tag{A.7}$$

$$\begin{aligned}\cos[A+B-C] &= \cos[A+B]\cos C + \sin[A+B]\sin C \\ &= (\cos A\cos B\cos C - \sin A\sin B\cos C) - (\sin A\cos B\sin C + \cos A\cos B\sin C).\end{aligned} \tag{A.8}$$

$$\begin{aligned}\cos[A-B+C] &= \cos[A-B]\cos C - \sin[A+B]\sin C \\ &= (\cos A\cos B\cos C + \sin A\sin B\cos C) - (\sin A\cos B\sin C - \cos A\cos B\sin C).\end{aligned} \tag{A.9}$$

$$\begin{aligned}\cos[A-B-C] &= \cos[A-B]\cos C + \sin[A+B]\sin C \\ &= (\cos A\cos B\cos C + \sin A\sin B\cos C) + (\sin A\cos B\sin C - \cos A\cos B\sin C).\end{aligned} \tag{A.10}$$

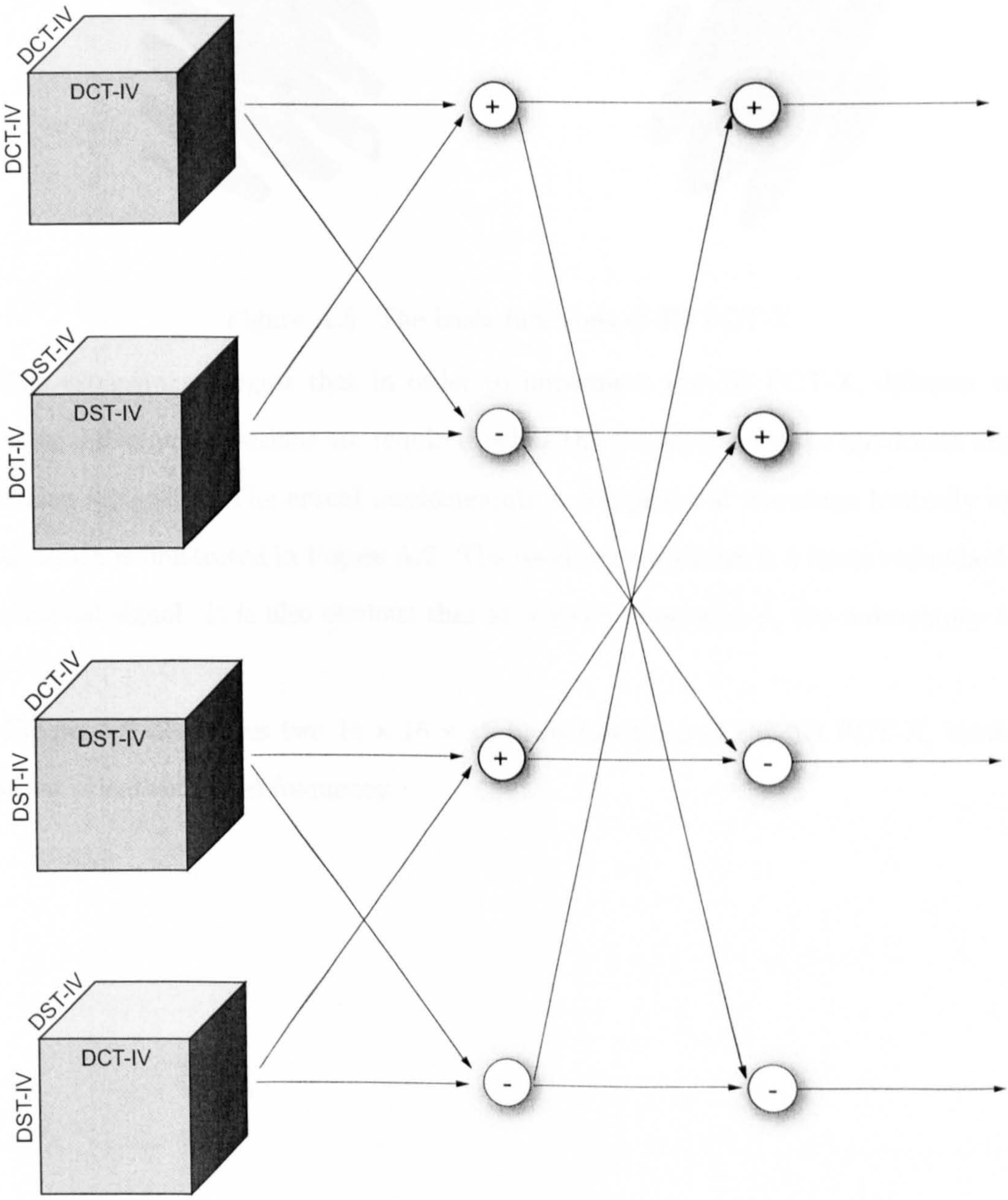


Figure A.2: The 3D PCT-X implementation

Figure A.3: The basis functions of 3D PCT-X

The expansions suggest that in order to implement the 3D PCT-X, different transforms on different dimensions are required, while the transforms for the quadrants are still Cartesian separable. The actual implementation comprises of two-stage butterfly operations, which is illustrated in Figure A.2. The resulting transform is 4 times redundant over the original signal. It is also obvious that at a given dimension n, the redundancy factor is 2^{n-1}.

Figure A.3 illustrates two $16 \times 16 \times 16$ basis functions of the 3D PCT-X, located at different orientation and frequency.

Appendix B

Publications

Relevant publications produced by the author of this thesis are included here:

1. **Planelets: A New Analytical Tool For Planar Feature Extraction**, with Nasir Rajpoot, Roland Wilson, presented at the WIAMIS 2004.

2. **Adaptive Wavelet Restoration of Noisy Video Sequences**, with Nasir Rajpoot, Roland Wilson, presented at the IEEE ICIP 2004.

3. **Radon/Ridgelet Signature for Image Authentication**, with Nasir Rajpoot, presented at the IEEE ICIP 2004.

4. **Image Denoising using Multiscale Directional Cosine Bases** with Nasir Rajpoot, presented at the IEEE ICIP 2005.

5. **Directional Wavelet Analysis with Fourier-type Bases for Image Processing**, in Wavelet Analysis and Applications, book series Applied and Numerical Harmonic Analysis, Springer-Verlag, 2007.

6. **Image Denoising with Directional Bases**, with Heechan Park and Graham Martin, presented at the IEEE ICIP 2007.

PLANELETS: A NEW ANALYSIS TOOL FOR PLANAR FEATURE EXTRACTION

Nasir Rajpoot *Roland Wilson* *Zhen Yao*

Department of Computer Science
University of Warwick
Coventry CV4 7AL, United Kingdom

e-mail: nasir,rgw,yao@dcs.warwick.ac.uk

ABSTRACT

Locally planar structures, formed by sweeping edges of objects, are commonly found in video sequences and convey most of the useful information. In this paper, the issue of efficient representation of such structures is addressed. We propose a novel representation tool which uses basis functions, termed as *planelets*, resembling planar structures and having compact support in space-time and spatiotemporal frequency. The representation is translation invariant, offers good directional selectivity, and can be computed efficiently.

1. INTRODUCTION

Wavelets have gained significant popularity as a signal analysis and processing tool over the last decade or so. This is largely due to their ability to provide a localised, sparse representation of a signal (or image) which is inherently multiresolution in nature and can deal with point singularities. It should come as no surprise that wavelet-based solutions to many problems in the analysis and processing of video sequences have been proposed. These include denoising, coding, and motion compensation [1]. However, the performance of such algorithms is severely restricted due to the following observation. While the wavelet transform in higher dimensions can be conveniently computed separably, separability also seriously limits the ability of wavelets to efficiently represent higher dimensional features (such as lines in images or planes in 3D image volumes). For 2D images, non-separable representations such as ridgelets [2] and curvelets [3] have recently been developed, motivated by the same observation. Ridgelets have also been shown to be optimal for representing functions with linear singularities. Furthermore, the lack of frequency selectivity remains an elusive problem with most techniques operating in the wavelet domain.

In this paper, we present a novel representation designed specifically for efficiently representing 3D functions with planar singularities. Locally planar structures, such as moving luminance edges, are commonly found in video sequences and often convey most of the information. The new representation, termed as the *planelet* basis, has a combination of scale, translation, and directional characteristics which are well matched to the locally planar surfaces of interest in many applications. Extraction of such planar features may be useful in various applications, such as video denoising, video coding, geometry estimation [4] and tracking of objects in video sequences. The planelet representation offers translation invariance, good directional selectivity, and yet can be computed efficiently. The computational complexity of a planelet transform is $O(n)$, where n is the number of points in analysis window. In its current form, the representation provides a non-orthogonal basis and is redundant by less than 14%.

In the next section, the new basis is briefly described. The ability of planelets to extract planar surfaces is demonstrated in Section 3. Experimental results for restoring video sequences in very noisy environments show the superiority of our representation over the state-of-the-art method of denoising. The paper concludes with some remarks about the current work and directions for future work.

2. REPRESENTATION WITII PLANELETS

A prototypical planelet function in 1D is of the following form

$$f_{\xi,\omega,a}(x) = w(\frac{x-\xi}{a}) \exp[-\jmath\frac{\omega(x-\xi)}{a}] \qquad (1)$$

where ξ, ω and a are respectively the location, frequency and scale parameters of the function. The function $w(.)$ is a window function, chosen alongwith the sampling interval to ensure invertibility of the discrete form of the transform. In 2D, the planelet basis can be regarded as a modification of the complex wavelet bases proposed in [5, 6], which show both translation invariance and directional selectivity, and may be used as an alternative to the ridgelet representation. In 3D, the basis comprises of the set of Cartesian products over ξ, ω at each scale a. That the continuous transform defined by (1) is invertible follows directly from the observation that it is simply the multiresolution Fourier transform (MFT) [5].

The discrete planelet transform (DPT), however, is significantly different from that described in [5]. It is a combination of two well known image transforms: the Laplacian pyramid [7] and the windowed Fourier transform (WFT). In some ways, it is similar to a 3D extension of the octave band Gabor representation proposed in [8], but avoids some of the more unpleasant numerical properties of the Gabor functions. The DPT of a video sequence x, in vector form, is given at scale m by

$$\hat{x}_m = \mathcal{F}_n(I - G_{m,m+1}G_{m+1,m})x_m \qquad (2)$$

where $\hat{x}_m$ denotes the DPT at scale m, $\mathcal{F}_n$ is the WFT operator with window size $n \times n \times n$, I is the identity operator, f_m is the Gaussian pyramid representation of x at level m

$$x_m = \prod_{0}^{m-1} G_{l+1,l} x \tag{3}$$

and $G_{m,m+1}, G_{m+1,m}$ are the raising and lowering operators associated with transitions between levels in the Gaussian pyramid. Invertibility follows directly from equations (2) and (3):

Theorem 1 *The representation defined by equation (2) is invertible.*

Proof

First we note that the WFT operator $\mathcal{F}_n$ has an inverse, which can be denoted by $\mathcal{F}_n^{-1}$. Secondly, we know from Burt and Adelson that the Laplacian pyramid is invertible, since, trivially,

$$x_m = x_m - G_{m,m+1}x_{m+1} + G_{m,m+1}x_{m+1} \tag{4}$$

and the proof is completed by induction on m.

Importantly, although both the pyramid and WFT operators are Cartesian separable, the closeness of the Burt and Adelson filter to a Gaussian function gives the pyramid virtually isotropic behaviour, which can be exploited well by the high frequency resolution of a Fourier basis. The planelet basis functions resemble planar structures and have compact support in both space-time and spatiotemporal frequency.

3. PLANAR FEATURE EXTRACTION

Planelets provide an ideal tool for representing local planes in a video sequence (or an image volume, in general) due to their ability to localise planar surfaces which correspond to lines in the Fourier domain. The presence of planar surface in a local analysis window can be inferred by computing the eigenvalues of the local inertia tensor in the window and analysing them. The parameters for orientation of the local planar surface and translation from centre of the window can also be estimated by analysing the most significant coefficients in the locality. Consider a video sequence synthesised by moving the centre of a circle on a sinusoidal wave in the time direction. Nonlinear approximations of this sequence using only 0.07% of the wavelet and planelet coefficients are shown in Fig. 1(*a*) and 1(*b*), respectively. It is clear from this example that the planelet approximation of a video sequence containing locally planar surfaces can result in a smaller approximation error as compared to that using wavelets. Planelets, therefore, can also be used for a piecewise planar approximation of a video sequence. Moreover, the approximation can be made to be adaptive to the local scale of the planar surfaces.

4. EXPERIMENTAL RESULTS

The ability of planelets to capture locally planar structures can be demonstrated in various applications, one of which is

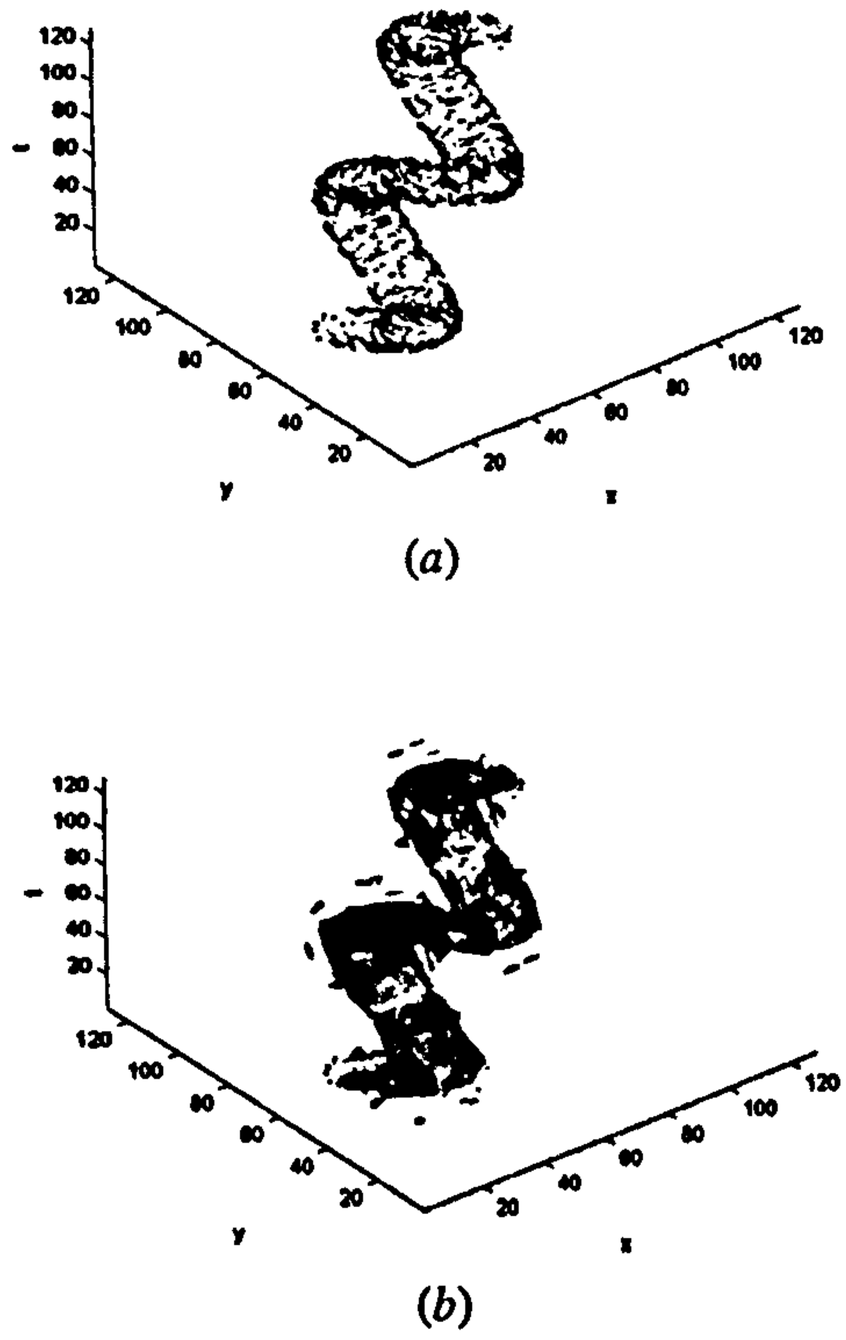


Fig. 1. Nonlinear approximation of a video sequence containing a moving circle using (*a*) wavelets (*b*) planelets

video denoising. In video sequences acquired in extremely noisy situations, it can be assumed that the coefficients which are relatively small in magnitude most likely correspond to the noise. This leads to a simple thresholding strategy in the planelet domain, which is akin to wavelet shrinkage method commonly used for denoising 2D images. Since the presence of additive Gaussian white noise means that almost all the planelet coefficients are affected by it, soft thresholding would provide an estimation of the original uncorrupted video sequence. The choice of threshold is crucial to the performance of a transform domain denoising algorithm [9]. We use either or both of the following thresholds: **(1)** a modification of the universal threshold proposed by Donoho and Johnstone [10]; $\theta_i = \mathcal{L}(\sigma)\sqrt{2\log n_i}$, where n_i denotes the number of coefficients at level i of the planelet decomposition, σ is the noise variance, and $\mathcal{L}(\sigma)$ is a suitably chosen function of σ, and **(2)** the *SUREshrink* (or simply *SURE*) threshold [1].

The algorithm described above for video denoising was tested on four standard video sequences reduced to a resolution of 128^3: *Miss America*, *Football*, *Tennis* and *Hall*. The image data was corrupted with additive Gaussian noise, and adaptive thresholding was applied to the transform coefficients of the noisy sequence represented in a 3-level planelet domain using a 16^3 window. Experimental results for the planelet denoising and translation invariant wavelet (TIW) denoising [11] of the noisy sequences for signal-to-noise-ratio (SNR) values of 0dB, 5dB, 10dB, and 15dB are presented in Table 1. The function $\mathcal{L}(\sigma)$ was chosen to be $\mathcal{L}(\sigma) = a\log_{10}\sigma + b$ where $a, b \in \Re$ and $b = 2a$. A value

of $a = 0.46$ was chosen empirically using least squares fitting. As is evident from these results, planelet denoising outperforms TIW denoising by a wide margin in terms of both visual quality and SNR gain. Denoising using planelet representation with *SURE* threshold generally outperforms the other two methods. In particular, the SNR gain of 9.5dB over TIW denoising for the noisy *Hall* sequence (containing two persons walking down a hallway) of 0dB SNR shows the promise that planelet representation offers for efficiently representing planar structures. Selected frames, for each of these sequences, restored by planelet denoising are shown in Fig. 2. Overall, these results compare favourably with those found in the literature [12], in terms of both visual quality and SNR gain. Two types of artifacts can be observed from these results: blocking artifacts due to the use of a 16^3 window, and *fake textures* which sometimes persist within these windows, due to suppression of a significant amount of high frequency energy.

The computational complexity of our algorithm is $O(n)$ as compared to $O\left(N \log_2(N)\right)$ for TIW denoising, where n and N respectively denote the size of analysis window and the size of each frame. It is to be noted, however, that the TIW denoising with a soft thresholding was applied to the individual 2D frames as opposed to the planelet denoising which benefits from representation of the noisy image volume in a 3D planelet domain, although implying higher storage requirements.

Video Sequence	Noisy (dB)	TIW (dB)	Planelets (dB)	
			θ_i	*SURE*
Miss America	0	9.5	17.1	17.3
	5	12.6	19.0	19.6
	10	15.3	20.8	21.5
	15	18.1	23.2	23.5
Football	0	5.9	11.7	12.1
	5	8.5	12.8	13.2
	10	11.0	14.4	14.7
	15	13.4	16.6	16.6
Tennis	0	6.7	14.8	14.6
	5	11.3	16.5	16.7
	10	14.0	18.2	18.4
	15	16.5	19.9	20.1
Hall	0	5.3	14.5	14.8
	5	9.4	16.7	17.2
	10	11.7	19.2	19.6
	15	14.2	21.7	21.8

Table 1. SNR (in dB) values for four standard video sequence

5. CONCLUSIONS

In this paper, planelets were proposed as an efficient representation tool for 3D functions with planar singularities. Such singularities are commonly found in video sequences in the form of moving luminance edges. It was shown that a piecewise planar approximation of a video sequence can be obtained by using a very small fraction of transform coefficients in the planelet domain. The ability of planelets to extract planar features from a video sequence makes them an attractive tool for analysis in various applications. For instance, decent restoration of video sequences captured in extreme noise is possible by thresholding of the planelet coefficients. The effects of oversampling and use of tapered windows remain to be investigated. Although the discrete planelet transform can be computed efficiently, its redundancy and high storage requirements may be of concern in some applications. Future work will address these issues and a further investigation into the usefulness of planelets in a wide range of video analysis applications.

6. REFERENCES

[1] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[2] E. Candes. *Ridgelets: Theory and Applications*. PhD thesis, Stanford University, Department of Statistics, 1998.

[3] E. Candes. Curvelets and curvilinear integrals. *Journal of Approximation Theory*, 113:59–90, 2000.

[4] A. Bhalerao and R. Wilson. A Fourier approach to 3D local feature estimation from volume data. In *Proceedings British Machine Vision Conference*, 2001.

[5] R.G. Wilson, A.D. Calway, and E.R.S. Pearson. A generalized wavelet transform for Fourier analysis: The multiresolution Fourier transform and its applications to image and audio signal analysis. *IEEE Transancation on Information Theory*, 38(2):674–690, March 1992.

[6] N.G. Kingsbury. Image processing with complex wavelets. *Phil. Trans. Royal Soc. A*, 1999.

[7] P.J. Burt and E.H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532–540, 1983.

[8] M. Porat and Y.Y. Zeevi. The generalized Gabor scheme of image representation in biological and machine vision. *IEEE Trans. PAMI*, 10:452–468, 1988.

[9] M. Jansen. *Noise Reduction by Wavelet Thresholding*. Springer-Verlag, 2001.

[10] D.L. Donoho and I.M. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 31:425–455, 1994.

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

[12] A. Kokaram. *Motion Picture Restoration*. Springer-Verlag, London, 1998.

Fig. 2. Denoising results for four standard video sequences

(*a*) Frame# 90 of *Miss America* (*b*) Noisy (SNR=0dB) (*c*) TIW Denoised (SNR=9.5dB) (*d*) Planelets Denoised (SNR=17.3dB)

(*e*) Frame# 60 of *Football* (*f*) Noisy (SNR=5dB) (*g*) TIW Denoised (SNR=8.5dB) (*h*) Planelets Denoised (SNR=13.2dB)

(*i*) Frame# 57 of *Tennis* (*j*) Noisy (SNR=10dB) (*k*) TIW Denoised (SNR=14.0dB) (*l*) Planelets Denoised (SNR=18.4dB)

(*m*) Frame# 106 of *Hall* (*n*) Noisy (SNR=15dB) (*o*) TIW Denoised (SNR=14.2dB) (*p*) Planelets Denoised (SNR=21.8dB)

ADAPTIVE WAVELET RESTORATION OF NOISY VIDEO SEQUENCES

Nasir Rajpoot *Zhen Yao* *Roland Wilson*

Department of Computer Science
University of Warwick
Coventry CV4 7AL, UK
email: nasir,yao,rgw@dcs.warwick.ac.uk

ABSTRACT

In this paper, we report work on a novel algorithm for restoration of noisy video sequences by thresholding in the transform domain. A video sequence is first transformed into an optimal 3D wavelet domain using basis functions adapted to the contents of the sequence. Assuming that all the major spatiotemporal frequency phenomena present in the sequence would produce high amplitude transform coefficients, a modified form of the BayesShrink thresholding method is used to suppress the noise. In order to reduce the effects of Gibbs phenomenon in the restored sequence, translation dependence is removed by averaging the restored instances of the shifted sequence. The algorithm yields promising results in terms of both objective and subjective quality of the restored sequence.

1. INTRODUCTION

It is often desirable to remove noise from video sequences captured in noisy environments or corrupted by noise during transmission, in broadcast and surveillance applications to name only a few. Noise removal by thresholding in the wavelet domain, a method also known as the *wavelet shrinkage* [1, 2], has become increasingly popular in recent years. The wavelet thresholding approach works in three steps: taking the discrete wavelet transform of a noisy signal, thresholding the wavelet coefficients, and taking the inverse discrete wavelet transform to estimate the original signal. Two methods are commonly used: *hard* and *soft* thresholding. In hard thresholding, all wavelet coefficients below a threshold are set to zero, assuming that the suppressed coefficients were the only ones most affected by noise. In the case of *soft* thresholding, all coefficients below a chosen threshold are set to zero, and the magnitude of the remaining coefficients is decreased by the threshold value, the assumption being that all coefficients were affected by noise. Regardless of which thresholding method is employed for denoising the signal, the algorithm is fast and offers the advantage that both compression and restoration of a signal can be achieved simultaneously. This general approach to signal restoration can also be applied to the removal of noise in a noisy video sequence by thresholding the coefficients of three-dimensional (3D) spatiotemporal wavelet transform of the sequence. The case for thresholding in spatiotemporal wavelet domain is supported by the fact that certain errors in motion estimation can be overcome by including the temporal direction in the realm of wavelet domain. Recent attempts to solve the video restoration problem have included combined spatial and temporal wavelet denoising [3], and the use of thresholding in non-separable transform domains, such as oriented 3D ridgelets [4] and 3D complex wavelets [5].

Although wavelet shrinkage performs significantly better than most other commonly used denoising methods, visual quality of the restored image (or video) can sometimes suffer from *ridge* type of artifacts, vallyes around the edges, due to the Gibbs phenomenon. The shift-variant nature of the wavelet transform worsens the effect of the Gibbs phenomenon, resulting in unpleasant artifacts. Translation invariant (TI) wavelet denoising of Coifman and Donoho [6] was developed to counter such artifacts by averaging out the translation dependence. Another feature of wavelet denoising is that it imposes a fixed dyadic wavelet basis on all types of input signals. Not only can the use of dyadic wavelets result in a blurred reconstruction, it can also limit the analysis of a locally occurring phenomenon in the frequency domain. The solution to this problem lies in the use of basis functions which are well localized in frequency as well as time (for 1D signals in time). For a video sequence, basis functions (separable or otherwise) with good localization in both space-time and spatiotemporal frequency are sought. In another relevant paper [7], we have developed a non-separable 3D representation termed as the *planelet* basis and studied its application to the video denoising problem. In this paper, we present a novel algorithm based on the 3D extension of translation invariant denoising using an adaptive wavelet packet representation for restoration of noisy video sequences. Comparative results show that our algorithm achieves significant gains over the state-of-the-art denoising techniques in terms of both SNR and visual qual-

ity of the restored video sequence for all three standard test video sequences at different levels of noise.

2. WAVELET PACKETS IN 3D

The ability of wavelet packets to capture locally occurring frequency phenomena in a signal has led to their successful application to many problems including image coding [8, 9]. The fundamental idea is to relax the restricted decomposition of only the lowpass subband and allow the exploration of all frequency bands up to the maximum depth. The discrete wavelet packet transform (DWPT) of a 1D signal x of length N can be computed as follows

$$\begin{aligned} w_{2n,d,l} &= \sum_k g_{k-2l}\, w_{n,d-1,k} && l = 0,1,\ldots,N2^{-d}-1 \\ w_{2n+1,d,l} &= \sum_k h_{k-2l}\, w_{n,d-1,k} && l = 0,1,\ldots,N2^{-d}-1 \\ w_{0,0,l} &= x_l && l = 0,1,\ldots,N-1 \end{aligned}$$

where $d = 1,2,\ldots,J-1$ is the scale index, with $J = \log_2 N$, n and l respectively denote the frequency and position indices, $\{h_n\}$ and $\{g_n\}$ correspond to the lowpass and highpass filters respectively for a two-channel filter bank and the transform is invertible if appropriate dual filters $\{\tilde{h}_n\}$, $\{\tilde{g}_n\}$ are used on the synthesis side. These equations can be used to compute full wavelet packet (FWP) tree of the signal decomposition. However, this implies that a large number of combinations of basis functions is now available to completely represent the signal. A tree-prunning approach such as [10] can be used to efficiently select the *best basis* with respect to a cost function.

The 3D DWPT can be computed by applying above equations separably in all three directions to get the FWP decomposition up to the coarsest resolution of subbands. The best basis can be selected in $O(N \log N)$ time, where N denotes the number of samples (frame resolution times the number of frames) in the video sequence. Given the goal here is to capture the significant spatiotemporal frequency phenomena in a video sequence, we used the Coifman-Wickerhauser entropy [10] as a cost function to select the best basis.

3. THE RESTORATION ALGORITHMS

The effect of the Gibbs phenomenon can be weakened by averaging the restored signal over a range of circular shifts [6]. For this reason, we apply soft threhsolding to the 3D wavelet packet coefficients of the shifted (in all three directions) noisy video sequence. A modified BayesShrink [11] method is using to compute the optimal value of threshold adaptively for each subband. Threshold θ_b for a subband of length N in an L-level WP decomposition is given by

$$\theta_b = \sqrt{\log N/L}\left(\frac{\sigma^2}{\sqrt{\max(\sigma_b^2-\sigma^2,0)}}\right)$$

where σ_b^2 is the subband variance, and σ^2 is the noise variance. If σ^2 is not known, a robust median estimate for noise standard deviation $\hat{\sigma}$ is obtained as follows

$$\hat{\sigma} = \mathcal{E}\{\hat{\Sigma}\}, \quad \hat{\sigma}_i = \frac{Median(|Y_i|)}{0.6745}$$

where $\hat{\sigma}_i \in \hat{\Sigma}$, $Y_i \in \{\mathcal{Y}\}$, set of all HHH bands in the decomposition tree, and the mean $\mathcal{E}$is taken only on the smaller half of the sorted $\hat{\Sigma}$ excluding the smallest value.

4. EXPERIMENTAL RESULTS

The above algorithm was tested against a number of other algorithms for restoration of several standard video sequences, three of which are included here: *Miss America*, *Hall*, and *Football*, all at a resolution of 128^3. The video sequences were corruped with additive white Gaussian noise, with the SNR of the noisy sequences being 0dB, 5dB, 10dB, and 15dB. Table 1 gives denoising results in terms of SNR for these noisy sequences using the following algorithms: TI soft thresholding in 2D wavelet domain (TIW2D), TI soft and hard thresholding in 3D wavelet domain (TIW3D), 3D wavelet packet (WP3D) with BayesShrink [11], both non-TI and TI 3D wavelet packet (TIWP3D) with the modified form of BayesShrink described in the previous section, and non-separable planelet [7] domain thresholding using SUREShrink [2] method. Comparative SNR curves for individual frames for two of the test sequences are provided in Figure 1. For all our experiments, the proposed algorithm produces by far the best results in terms of both overall and individual SNR. Some of the frames of the test sequences restored by our algorithm and TIW3D-Hard, a 3D realization of the algorithm in [6], are shown in Figure 2. While TIW3D restores clean and smooth version of the original frames, some of the details are restored by TIWP3D.

For comparison purposes, computational complexity for each of the algorithms considered is also provided in Table 1. It is clear from this table that the planelet algorithm of [7] is the least computationally expensive, whereas the TI implementations of 3D wavelet and 3D WP are towards the more expensive side with TIWP3D being the most expensive due to the additional one-off cost of best basis selection.

5. CONCLUSIONS

In this paper, a novel algorithm for restoration of noisy video sequences was presented. The algorithm works by finding an optimal wavelet packet representation for a sequence and averaging the results of thresholding by a modified form of BayesShrink [11] method on shifted transform coefficients. Experimental results suggest that the performance of the algorithm is by far the best as compared to other methods

Video Sequence	Noise (dB)	Denoising Algorithm (Transform+Thresholding)						
		TIW2D Soft	TIW3D Soft	TIW3D Hard	WP3D Bayes	WP3D Proposed	TIWP3D Proposed	Planelet SURE
Miss America	0	9.5	12.4	17.9	17.2	17.4	**18.9**	17.3
	5	12.6	15.6	19.5	19.0	19.3	**20.7**	19.6
	10	15.3	18.1	21.5	21.1	21.5	**23.0**	21.5
	15	18.1	20.3	23.9	23.1	23.8	**25.2**	23.5
Hall	0	5.3	11.0	14.7	14.8	15.0	**16.7**	14.8
	5	9.4	13.4	16.6	17.2	17.3	**18.9**	17.2
	10	11.7	15.5	19.0	19.5	19.8	**21.3**	19.5
	15	14.2	17.7	21.7	22.1	22.5	**24.1**	21.8
Football	0	5.9	9.4	11.9	11.9	12.0	**12.8**	12.1
	5	8.5	11.1	13.1	12.9	13.3	**14.3**	13.2
	10	11.0	12.5	15.0	13.9	15.2	**16.9**	14.7
	15	13.4	14.3	18.0	15.3	17.9	**20.0**	16.6
Computational Complexity	–	$O(N+l^2N)$	$O(N+l^3N)$	$O(N+l^3N)$	$O(N\log N)$	$O(N\log N)$	$O(N\log N+l^3N)$	$O(n)$

Table 1. SNR results for three standard video sequences and algorithms' complexity
N and n respectively denote sequece size and planelet window size, and l denotes length of the wavelet filter.

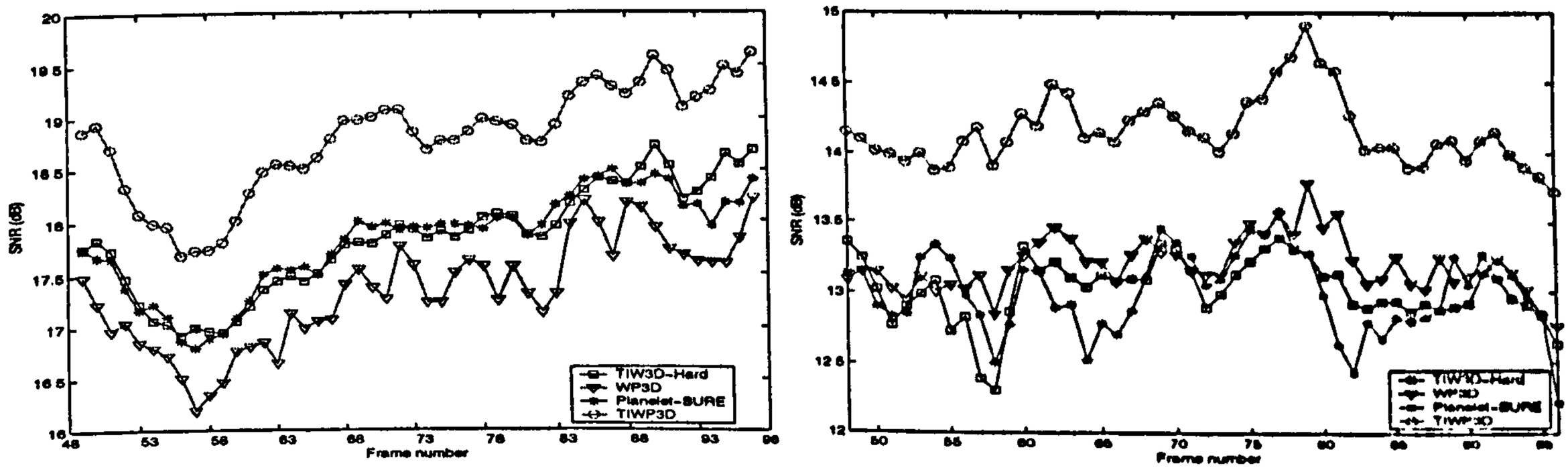


Fig. 1. Frame-by-frame comparative results (*a*) *Miss America* and (*b*) *Football*

found in the literature, to the best of our knowledge. Although its computational cost may be a limiting factor in some applications, a less expensive version of the algorithm (WP3D) with the proposed thresholding method produces results comparable to the state-of-the-art at a lower cost. It is perhaps worth noting that while being the least expensive, the planelet algorithm [7] produces SNR results which are still comparable to those of TIW3D-Hard. These results suggest that the localisation of spatiotemporal frequency is a desirable feature of the domain in which video sequences are represented.

6. REFERENCES

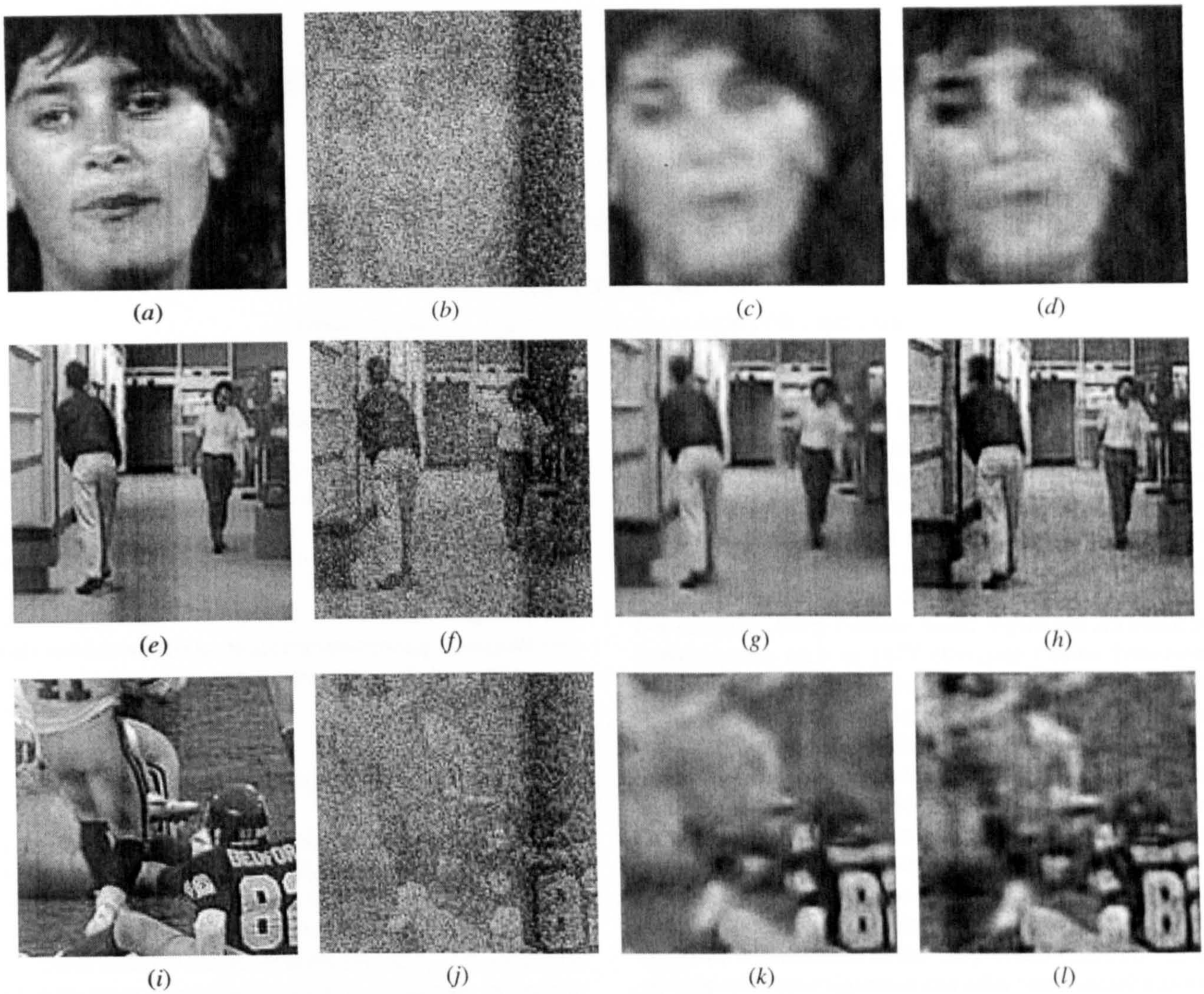
[1] D.L. Donoho and I.M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[2] M. Jansen, *Noise Reduction by Wavelet Thresholding*, Springer-Verlag, 2001.

[3] A. Pizurica, V. Zlokolika, and W. Philips, "Combined wavelet domain and temporal video denoising," in *Proc. IEEE Intl. Conf. on Advanced Video and Signal based Surveillance (AVSS)*, July 2003.

[4] P. Carre, D. Helbert, and E. Andres, "3-D fast ridgelet transform," in *Proc. IEEE Intl. Conf. on Image Processing (ICIP)*, September 2003.

[5] I.W. Selesnick and K.Y. Li, "Video denoising using 2D and 3D dual-tree complex wavelet transforms," in *Proc. SPIE Wavelets X*, August 2003.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Fig. 2. Denoising results for three standard video sequences
(*a*) Frame# 90 of *Miss America* (*b*) Noisy (SNR=0dB) (*c*) TIW3D-Hard (SNR=17.9dB) (*d*) TIWP3D (SNR=18.9dB)
(*e*) Frame# 106 of *Hall* (*f*) Noisy (SNR=10dB) (*g*) TIW3D-Hard (SNR=19.0dB) (*h*) TIWP3D (SNR=21.3dB)
(*i*) Frame# 60 of *Football* (*j*) Noisy (SNR=5dB) (*k*) TIW3D-Hard (SNR=13.1dB) (*l*) TIWP3D (SNR=14.3dB)

[6] R.R. Coifman and D.L. Donoho, "Translation-invariant denoising," in *Wavelets and Statistics. Lecture Notes in Statistics*, 1995.

[7] N.M. Rajpoot, R.G. Wilson, and Z. Yao, "Planelets: A new analysis tool for planar feature extraction," in *Proc. 5th Intl. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, April 2004.

[8] F.G. Meyer, A.Z. Averbuch, and J-O. Strömberg, "Fast adaptive wavelet packet image compression," *IEEE Trans. on Image Processing*, vol. 9, May 2000.

[9] N.M. Rajpoot, R.G. Wilson, F.G. Meyer, and R.R. Coifman, "Adaptive wavelet packet basis selection for zerotree image coding," *IEEE Trans. on Image Processing*, vol. 12, December 2003.

[10] R.R. Coifman and M.V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. on Info. Th.*, vol. 38, March 1992.

[11] G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. on Image Processing*, vol. 9, September 2000.

RADON/RIDGELET SIGNATURE FOR IMAGE AUTHENTICATION

Zhen Yao *Nasir Rajpoot*

Dept. of Computer Science, University of Warwick
Coventry, CV4 7AL
United Kingdom
email: {yao, nasir}@dcs.warwick.ac.uk

ABSTRACT

In this paper, we describe a novel content-based image signature for authentication using the ridgelet transform. The signature is extracted from the Radon domain and entropy coded after a 1D wavelet transform, which is essentially the so-called "ridgelet transform". Unlike traditional authentication signatures, it has the ability to localise tampering at a high resolution, robust to content-preserving manipulations such as compression and allows a progressive authentication.

1. INTRODUCTION

Traditional data authentication and integrity verification are done by appending a hashed signature of the file, usually encrypted with a secret key. The signature is practically "unique" and distributed with the file. However, the proliferation of multimedia data over the Internet poses some new authentication requirements, such as localising tampering and semi-fragility. Unfortunately these are the realms that traditional signature-based system fails and alternative approaches such as authentication watermarks [1], and content-based robust signatures [2, 3] have become popular in the research community.

A watermark, usually involves some form of steganography, is a code embedded into a host image. The authenticity of the image can be verified by checking the integrity of the watermark. Since attacks on the host image also destroy the watermark correspondingly on the same position, tampering localisation can be achieved. However there is a fundamental trade-off between security and localisation. One must establish a neighbourhood dependency in the watermark otherwise it is vulnerable to counterfeiting attacks [4]. Moreover, since the process of watermarking itself introduces distortion on the host data, it is sometimes not desirable in some applications such as with medical images. Although a few reversible watermarks have been proposed [5], their localisation abilities are however constrained by the data hiding capacity.

It has been pointed out by Shannon [6] that for a perfect secrecy system, the key [1] K, must be at least as long as the message M, more precisely, that $H(K) \geq H(M)$, where $H(\cdot)$ denotes the entropy function. Intuitively, this says that in order to achieve perfect security, the key has to be long enough to describe the message. Hence a perfectly secure signature is essentially a compressed form of the image. For example, many content-based signatures are extracted from domains such as DCT (Discrete Cosine Transform) and DWT (Discrete Wavelet Transform) domain which are also popular choices for compression [7]. Such ideas were exploited in the watermarking world in the form of self-embedding, where a redundant lower-resolution copy of the image is embedded in order to detect and recover tampering. Despite the extra communication cost, the advantage of signature-based authentication is apparent: First, it does not introduce distortion on the original image. Second, it can solve the dilemma of security-localisation uncertainty, if the signature is considered as the lossy-coded version of the image, since a longer signature can offer better security as well as localisation resolution. Third, a compact signature can be combined into a watermarking system just as many watermarking schemes do employ such label-embedding approach.

Recently, motivated by the neeed for finding better representations for nature images, several geometric wavelets (eg. ridgelet and curvelet) have been proposed. The underlying Radon transform plays an essential role in providing such non-separable, directional properties. Although Radon-based signatures have been previously proposed [8], which take the advantage of invariant features of the transform to provide robustness, but few address the problem of localisation, which is the key motivation of this work.

The rest of the paper is organised as follows. In the next section, a brief review of Radon and ridgelet transforms is presented. Next we discuss how authentication and localisation can be achieved by a number of Radon vectors and how to generate a compact signature using the ridgelet idea.

[1] The *key* here refers to the authentication (public) key, which defines the authenticated space, instead of the encryption key.

Some experimental results are presented and the paper concludes with comments on the proposal and future directions of research.

2. RADON AND RIDGELET TRANSFORMS

The Radon transform, which has been mainly used in tomography reconstruction, is now gaining popularity in image processing as a general tool. Mathematically the continuous Radon transform of an integrable bivariate function $f(x, y)$ is defined by

$$R_f(\theta, t) = \int_{\mathbb{R}^2} f(x, y)\delta(x\cos\theta + y\sin\theta - t)dxdy \quad (1)$$

Discretisation has been a major difficulty in applying Radon transform to general image processing. The simplest form of discrete Radon transform is to select finite number on the angular variable of projection, then take the summation on the discrete image along the projection line. Recently, some other discrete (finite) Radon transforms have been proposed such as FRAT [9] and fast slant-stack [10].

The Radon transform is a linear transform and has several useful properties.

Property 1 *If the function $f(x, y)$ is translated, its Radon transform is*

$$f(x - x_0, y - y_0) \Leftrightarrow R_f(\theta, t - x_0\cos\theta - y_0\sin\theta) \quad (2)$$

Property 2 *If the function $f(x, y)$ is rotated by ϕ, it corresponds to a shift translation in the Radon transform.*

$$f(x\cos\phi - y\sin\phi, x\sin\phi + y\cos\phi) \Leftrightarrow R_f(\theta + \phi, t) \quad (3)$$

Property 3 *If the function $f(x, y)$ is rescaled by a factor of a, its Radon transform is*

$$f(ax, ay) \Leftrightarrow \frac{1}{|a|}R_f(\theta, at) \quad (4)$$

The ridgelet transform, introduced in [11], has the continuous form of

$$CRT_f(a, b, \theta) = \int_{\mathbb{R}^2} \psi_{a,b,\theta}(x, y)f(x, y)dxdy \quad (5)$$

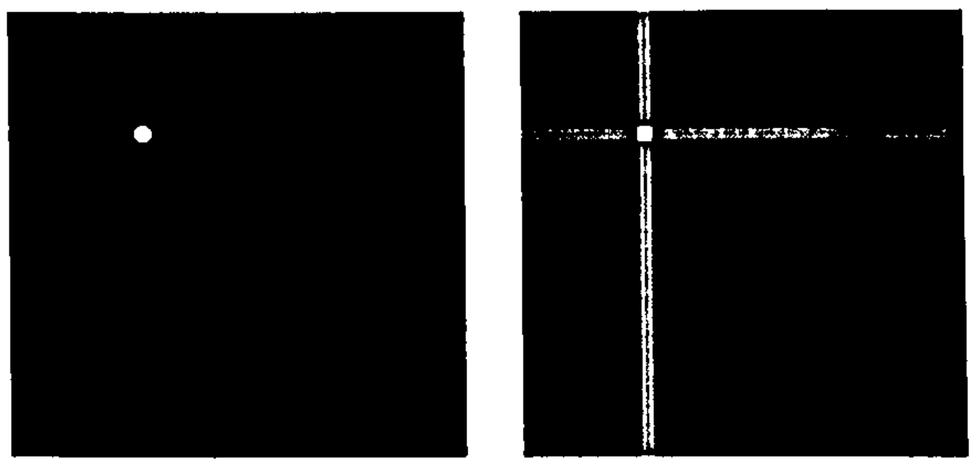
where the ridgelet $\psi_{a,b,\theta}(x, y)$ in 2-D are defined from a wavelet-type function in 1-D $\psi(x)$ as

$$\psi_{a,b,\theta}(x, y) = \frac{1}{\sqrt{a}}\psi\left(\frac{x\cos\theta + y\sin\theta - b}{a}\right) \quad (6)$$

Since Radon transform projects a linear-singularity into a point-singularity, the wavelet and ridgelet transforms are linked via the Radon transform. More precisely, the definition in equation (5) can be re-written as

$$CRT_f(a, b, \theta) = \int_{\mathbb{R}} \psi_{a,b}(t)R_f(\theta, t)dt \quad (7)$$

where $\psi_{a,b}(t) = a^{-1/2}\psi((t - b)/a)$ is a 1-D wavelet.



(a) (b)

Fig. 1. (a) A point-wise singularity and (b) its back-projected reconstruction from two Radon vectors at orthogonal angles.

3. SIGNATURE GENERATION

Although the ridgelet enjoys some elegant mathematical properties, and it does in fact exploit the fact that nature images exhibit linear regularity along contours and edges. The success of its application in image representation has been limited to geometrically regular objects [9]. However, for tampering detection, a decent representation for the image is less important than representing the tampered location and a key characteristic of malicious attacks is that they are usually locally singular. Naturally, the Radon transform is capable of capturing the location of such singularity due to its directionality in projections.

In the simplest case, consider an image with a point-wise singularity (see figure 1). Two Radon vectors (i.e. the discrete angular projection) at orthogonal angles are sufficient to determine its location. Of course, it is not sufficient when there is more than one singularity, which rises ambiguity, nor can it determine the exact geometrical shape of such singularity. However, the resolution increases as more number of Radon vectors are used in reconstruction. This motivates us to use Radon vectors as the signature of the image, but the eventual signature is ridgelet transformed, since wavelet can provide a sparse representation in a multi-resolution framework which is useful for compression.

The signature generation algorithm can be described as follows.

1. For a set of angles $\Theta = \{\theta_1, \theta_2, ..., \theta_n\}$, where $\theta_i \in [0, \pi]$, preferably evenly spaced, typically with $n > 20$, compute the normalised Radon vectors at these angles as follows.

$$\mathcal{R}_i(\theta_i) = \frac{1}{N_{\theta_i}}\sum_j f(i\cos\theta_i - j\sin\theta_i, i\sin\theta_i + j\cos\theta_i).$$

2. Apply an L-level DWT on the Radon vectors $\{\mathcal{R}_i\}$ to obtain the ridgelet representation, denoted as $\{\mathbf{w}_i\}$.
3. Entropy code the ridgelet coefficients with traditional entropy coders such as arithmetic code.

4. The signature is stored/transmitted with low-pass band first, then from smaller bands to larger bands progressively in encrypted form.

It should be noted that since Radon transform takes a summation over the image support, the Radon vectors can be considered as *global* information. There is an inherent directional *neighbourhood dependency* in the projected vectors and they are also correlated. Therefore it is not possible to perform the counterfeiting attack on the image.

The verification process is simply as follows. Once we have retrieved the image $\hat{f}$ and the signature, we apply the ridgelet transform on $\hat{f}$ as described before but without the entropy coding to obtain $\{\hat{\mathbf{w}}_i\}$, while $\{\mathbf{w}_i\}$ can be decoded from the signature. If the image is original, $\{\hat{\mathbf{w}}_i\}$ and $\{\mathbf{w}_i\}$ should be identical. If not, their reconstructions should exhibit the difference between the tampered image $\hat{f}$ and the original copy f. Since Radon and wavelet transforms are both linear, $\{\mathbf{w}_i - \hat{\mathbf{w}}_i\}$ is the (undercomplete) ridgelet representation of $f - \hat{f}$. We take the inverse ridgelet transform on $\{\mathbf{w}_i - \hat{\mathbf{w}}_i\}$ to observe the difference.

4. EXPERIMENTAL RESULTS

We have performed our experiments over a range of natural images, although the tampering detection ability of our proposed approach is image-independent. We chose the `barbara` (512×512) image to present our results in this paper. Due to the space limitation we only present results on two typical attacks: the first one is content-preserving JPEG compression, and the second one is a malicious attack with a bird-shaped stamping (see figure 2).

Figures 2(c) and 2(d) suggest that robustness against content-preserving manipulations is possible, since the difference map is a reconstruction of $\{\mathbf{w}_i - \hat{\mathbf{w}}_i\}$. If f and $\hat{f}$ are visually identical, the energy of $f - \hat{f}$ should be nearly zero. However, the contrast enhanced version of the difference, such as in figure 2(d), may be useful in determining the nature of such manipulation. Robustness against geometrical manipulations can also be addressed [8], due to the invariant properties of Radon transform described in equation (2), (3) and (4).

Figure 2(e)-(h) demonstrate the tampering localisation ability of the signature. The tampered area is accurately detected by the intersection of "ridges". Figure 2(g),(h) is further enhanced by a soft thresholding on the constructed difference map, in order to suppress some undesired linear artifact of undercomplete projection. Such artifact is completely removed in figure 2(h), where more angles of projections are used. The difference map can be used to restore the tampered area, by simply adding the difference map back into the received image.

Due to the inherent multi-resolution property of the wavelet transform, the signature also allows a progressive authentication without losing its capability of tampering localisation. Figure 3 illustrates an instance when the authentication is verified with lowpass band of the signature, which is only half of the total ridgelet coefficients. The tampering localisation is effectively identical to its counterpart in figure 2(g).

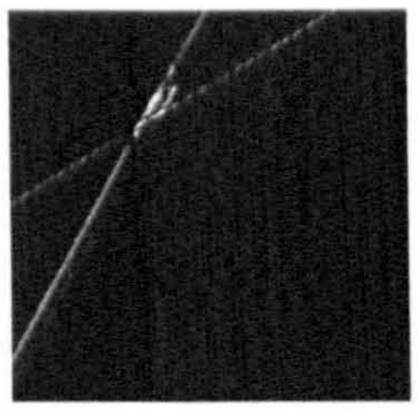


Fig. 3. Thresholded difference map constructed by the lowpass subband at resolution 256×256 (6 projections).

Image	$\lvert\Theta\rvert$=6	$\lvert\Theta\rvert$=9	$\lvert\Theta\rvert$=18
`lena`	2183	4034	7843
`barbara`	2372	4152	8012
`goldhill`	2280	4198	7911

Table 1. Signature lengths with different number of projections (in bytes).

There is certainly a degree of flexibility in controlling the signature's length by varying the number of Radon vectors. It is desirable for the signature to be compact while the trade-offs between communication cost and security / localisation always impose. However it is difficult to define exactly the length of the signature in order to be called "compact". Here we would like to propose that the signature is "compact" as long as it can be embedded obliviously into the host media with some protective redundancy. For a 512×512 image, if we use the spatial LSB bits for data embedding, the total capacity is 32768 bytes. Table 1 lists the length of signatures we obtained from various images and different number of projections. They are encoded by a zero-order arithmetic coder with simple uniform scalar quantisation, but are still within below the capacity and can be embedded as a form of self-embedding watermark.

5. CONCLUSIONS

A content-based, robust image authentication signature has been proposed. The signature, essentially a coded undercomplete ridgelet transform of the image, can localise tampering in a multi-resolution fashion. The work is also one of our first practical attempts in ridgelet encoding, although the novelty is in authentication rather than compression. Since

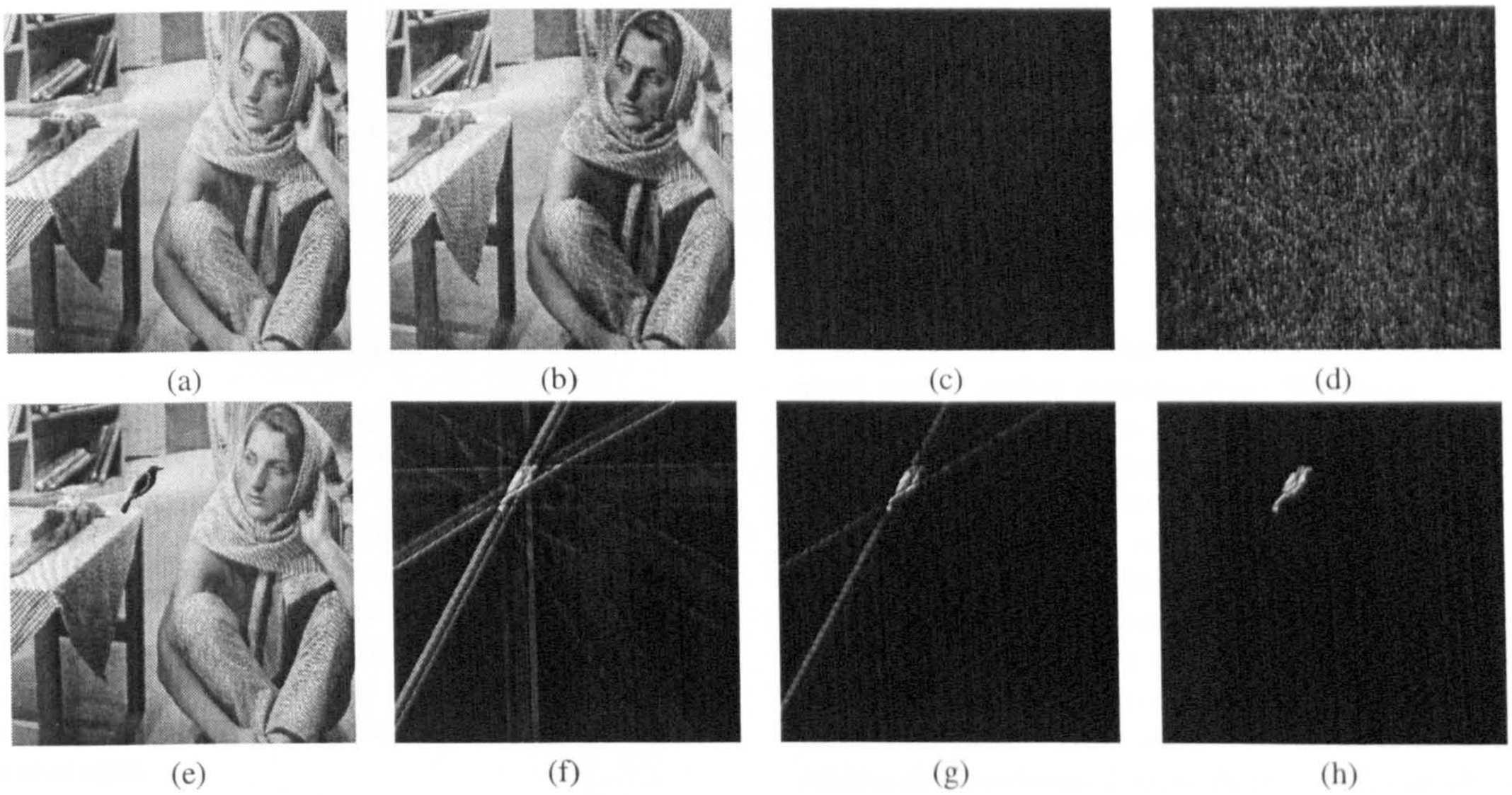


Fig. 2. Authentication results. (a) Original. **(b) JPEG** compressed. (c) The difference **map**, 6 projections (d) Contrast enhanced **(c)**. (e) The tampered image (object a**ddition**) (f) The difference map, 6 projections **(g)** The thresholded difference map with **6** projections (h) The thresholded differ**ence** map with 12 projections.

the encoder used in this work is very simple, an efficient compression algorithm for ridgelet representation can significantly improve the compactness of the signature and may benefit the image coding community on a direction beyond wavelet. It remains to be seen how interpolation algorithms can incorporated in the scheme which may help to reduce the linear artifact exhibited from the inverse Radon transform.

6. REFERENCES

[1] E.T. Lin and E.J. Delp, "A review of fragile image watermarks," in *Proc. of ACM Multimeida & Security Workshop*, Orlando, 1999, pp. 25–29.

[2] G.L. Friedman, "The trustworthy digital camera: Restoring credibility to the photographic image," *IEEE Trans. on Consumer Electronics*, vol. 39, no. 4, pp. 905–910, Nov. 1993.

[3] M. Schneider and S.F. Chang, "A robust content based digital signature for image authentication," in *Proc. of ICIP*, 1996, vol. 3, pp. 227–230.

[4] N.Holliman and N. Memon, "Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes," *IEEE Trans. on Image Processing*, vol. 9, no. 3, pp. 432–441, March 2000.

[5] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding - new paradigm in digital watermarking," *EURASIP Journ. Appl. Sig. Proc.*, vol. 2002, no. 2, pp. 185–196, Feb 2002.

[6] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, pp. 656–715, Oct. 1949.

[7] C.S. Lu and H.Y.M. Liao, "Structural digital signature for image authentication: An incidental distortion resistant scheme," *IEEE Trans. on Multimedia*, vol. 5, no. 2, pp. 161–173, 2003.

[8] F. Lefebvre, J. Czyz, and B. Macq, "A robust soft hash algorithm for digital image signature," in *Proc. of ICIP*, Barcelona, Sept. 2003, vol. II, pp. 495–498.

[9] Minh N. Do and Martin Vetterli, "The finite ridgelet transform for image representation," *IEEE Trans. Image Processing*, vol. 12, no. 1, pp. 16–28, Jan. 2003.

[10] A. Averbuch, R. Coifman, D.L. Donoho, and M. Israeli, "Fast slant stack: A notion of radon transform for data in a cartesian grid which is rapidly computible, algebraically exact, geometrically faithful and invertible," to appear in SIAM Scientific Computing.

[11] E.J. Candés, *Ridgelets: Theory and applications*, Ph.D. thesis, Dept. of Stats, Stanford Univ., Stanford, CA, 1998.

IMAGE DENOISING USING MULTISCALE DIRECTIONAL COSINE BASES

Zhen Yao *Nasir Rajpoot*

Dept. of Computer Science, University of Warwick
Coventry, CV4 7AL, United Kingdom
Email: {yao, nasir}@dcs.warwick.ac.uk

ABSTRACT

Motivated by the fact that in images, there is usually a presence of local strongly oriented harmonics, a representation which is both well-localised in frequency and orientation is desirable to efficiently describe such oriented harmonic features. Here we introduce a family of multiscale trigonometric bases for a bi-variate function called the Multiscale Directional Cosine bases for image denoising tasks. Our results show the promise of the new bases which almost consistently outperform other image representation bases on natural images.

1. INTRODUCTION

The application of transforms in image processing is often based on a separable construction. Rows and columns in an image are treated independently and the 2D basis functions are simply tensor products of the corresponding 1D functions. Such method keeps simplicity in terms of design and computation, but is not capable of capturing properly all the interesting features of an image. For example, the orthonormal separable wavelet transform [1] in higher dimensions is seriously limited in its ability to efficiently represent higher dimensional features such as lines. Furthermore, the lack of frequency selectivity remains an elusive problem with most techniques operating in the wavelet domain.

The ability to efficiently analyse and describe directional patterns is of fundamental importance for image analysis and compression. In order to represent higher dimensional image features, Candés and Donoho [2] proposed a directional multiresolution representation named *ridgelets* which deal effectively with linear singularities in 2D. The idea of *curvelets* [3] is to represent a curve as a superposition of functions of various lengths and widths obeying the scaling law. Starck et al. [4] employed the curvelet transform in the denoising task. The curvelet coefficients are more than $16J + 1$ times redundant, with J scales of decompositions. Not only does the curvelet denoising outperform wavelet-based methods in terms of PSNR, the visual appearance of the resulting images is better as well. Do and Vetterli [5] used similar ideas to construct curvelets that can be computed with a perfect reconstruction filter bank, called the *contourlet transform*, which combines the Laplacian pyramid transform with directional filter banks.

However, the assumption that natural images are characterised solely by linear edges is not true. In images, there is usually a presence of local strongly oriented harmonics (textures) separated by curvilinear edges. A sparse representation which is both well-localised in frequency and orientation is desirable to efficiently describe such oriented harmonic features. Here we describe a set of bases called *Multiscale Directional Cosine Bases* which can efficiently represent local oriented harmonics, and with a local directional cosine packet analysis, we can accommodate the "texture + edge" model by representing both directional periodic ridges and ridglets in a unified framework.

For the rest of this paper, we introduce a family of bases called the "Multiscale Directional Cosine" (MDC) Bases in section 2. Next we show some results from our denoising experiments using the MDC bases in section 3. The paper concludes with a summary in section 4.

2. MULTISCALE DIRECTIONAL COSINE BASES

First, we define the directional cosine basis in 2D as

$$C_{k,\theta} = \lambda_k \cos(\pi k(x\cos\theta + y\sin\theta)) \qquad (1)$$

where $\lambda_k = \begin{cases} 1 & \text{if } k = 0 \\ \sqrt{2} & \text{if } x \neq 0. \end{cases}$

The directional 2D continuous cosine transform is defined as

$$\begin{aligned} Cf(k,\theta) &= \langle f, C_{k,\theta} \rangle \\ &= \int_{\mathbb{R}^2} \lambda_k f(x,y) \cos(\pi k(x\cos\theta + y\sin\theta)) dxdy \end{aligned}$$

The directional cosine basis vectors are indexed by frequency k and direction θ, as can be seen in Figure 1. It is obvious that the basis vectors look similar to the Fourier basis. In fact, if we substitute the cosine basis function with $e^{ix\xi}$, we obtain the polar Fourier basis representation. However,

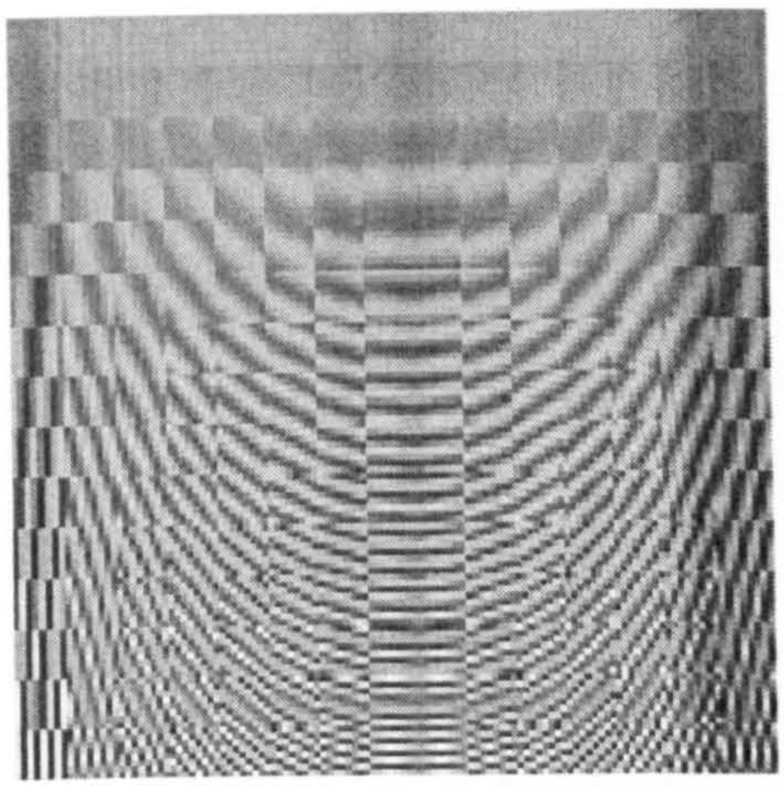


Fig. 1. The 8×8 directional cosine basis vectors

the proposed transform has several advantages when compared with traditional Fourier basis. First, the transform is real-to-real instead of real-to-complex. Second, since the cosine basis assumes a symmetric extension, its approximation error decays more rapidly than Fourier series. Lastly, unlike in the Fourier domain, the Cartesian-to-polar coordinate conversion has already been done in the Radon transform to get the sense of directionality.

For an image representation basis to be useful, the basis vectors should be localised both in space and frequency, and they should have certain orientation selectivity. More importantly, to capture patterns of interest at different scales, the basis need to be multiresolutional. A prototypical MDC function has the form

$$\psi_{k,\theta,s,\mathbf{t}}(\mathbf{x}) = b\left(\frac{\mathbf{x}-\mathbf{t}}{s}\right)\mathcal{C}_{k,\theta}\left(\frac{\mathbf{x}-\mathbf{t}}{s}\right). \qquad (2)$$

where k, θ, $\mathbf{t}$ and s denotes the frequency, orientation, location and scale parameters of the function respectively and $b(\cdot)$ is the smooth bell function chosen along with the sampling interval to ensure invertibility of the discrete form of the transform.

The discrete implementation of MDC transform is similar to the digital curvelet construction. While the discrete cosine transform and discrete Radon transform [6] are well studied in the literature, a combination of these two transforms gives us the discrete directional cosine operator. Unlike the digital curvelet implementation in [4], which is very redundant, the multiresolution property of the MDC transform is given by the well-known Laplacian pyramid [7]. Therefore, the transform is over-complete by some 33% in 2D if non-overlapping windows are used. The discrete MDC of a 2D vector $\mathbf{x}$, at scale s is given by

$$\mathbf{X}_s = \mathcal{C}_n(\mathbf{I} - \mathbf{G}_{s,s+1}\mathbf{G}_{s+1,s})\mathbf{x}_s. \qquad (3)$$

where $\mathbf{X}_s$ denotes the transform at scale s, $\mathcal{C}_n$ is the discrete directional cosine transform operator with window size $n \times n$, $\mathbf{I}$ is the identity operator, $\mathbf{x}_s$ is the Gaussian pyramid representation of $\mathbf{x}$ at scale s

$$\mathbf{x}_s = \prod_{l=0}^{s-1} \mathbf{G}_{l+1,l}\mathbf{x}. \qquad (4)$$

and $\mathbf{G}_{s,s+1}, \mathbf{G}_{s+1,s}$ are the raising and lowering operators associated with transitions between levels in the Gaussian pyramid. The closeness of the Burt and Adelson filter to a Gaussian function gives the pyramid virtually isotropic behavior, which can be well exploited by the high frequency resolution of the cosine basis. We certainly have the choice of using the directional cosine packets as the transform operator $\mathcal{C}_n$ by substituting the cosine transform by a cosine packet analysis, forming a semi-adaptive basis. In this way, the MDC packet basis is able to capture a wide range of directional features at different resolutions.

3. IMAGE DENOISING EXPERIMENTS

A good basis for representing images should be able to capture important features of interest. In this case, the MDC bases should allow us to map directional periodic pattern patches into coefficients of significant magnitude. In order to demonstrate the efficiency of the proposed representation, we conducted image denoising experiment by simply thresholding the coefficients in the transform domain with certain threshold. The denoising experiments are performed in such settings:

1. The Laplacian pyramid is decomposed at 5 levels of subbands.
2. The window size n is chosen at 16×16, modulated with a squared cosine with 50% overlapping.
3. We used both directional cosine and directional cosine packet as the operator $\mathcal{C}_n$ on the windowed block.

The thresholding we use is a form of the universal thresholding proposed in [8], multiplied by an extra constant a $\Theta = a\sqrt{2\log N}\sigma/1.23^L$, where $N = n^2 = 256$ here and L denotes the level of decomposition, while $L = 0$ corresponds to the highest frequency subband. For directional cosine denoising, the value of $a = 0.08$ was found to give satisfactory result. For the directional local cosine packets, $a = 0.062$ was used. The lowpass subband is left intact.

The results are compared with two algorithms. The first one is a wavelet-packet based wavelet shrinkage algorithm which is described in [9], called $\mathcal{S}$-Bayes, with the thresholding function being a modified version of the *BayesShrink* [10]. The best wavelet packet basis is sought by using the Shannon entropy function and cycle-spinning [11] is used to suppress the pseudo-Gibbs artifact. Essentially such treatment gives the translation invariance to the wavelet packet basis, which is known to be good in representing some periodic signals. The second algorithm is a modified version

Image	Noise (dB)	TI $\mathcal{S}$-Bayes	Curvelet	Directional Cosine	Directional Cosine Packets
barbara	0	14.75	14.59	14.89	15.19
	5	16.08	15.93	16.42	16.85
	10	18.00	17.64	18.44	18.85
	15	21.12	19.64	20.34	20.71
	20	24.92	21.41	21.88	22.10
lena	0	17.06	17.10	17.24	17.75
	5	18.96	18.82	18.99	19.63
	10	21.08	20.83	21.05	21.89
	15	23.49	23.10	23.35	24.26
	20	26.15	25.35	25.60	26.31
grain	0	12.93	13.06	13.10	13.22
	5	13.81	13.66	13.84	14.29
	10	15.87	15.01	15.51	16.18
	15	18.72	17.28	18.00	18.84
	20	22.02	20.05	20.56	21.38

Table 1. The comparative image denoising results in SNR

of curvelet. The curvelet implementation in [4] which uses a much more redundant overcomplete wavelet frame than our MDC, is a "specialised" transform to perform denoising task instead of general-purpose image processing. In order to carry out a fair comparison, here the local ridgelets are placed on the Laplacian pyramid as well in our setting, in order to carry out a fair comparison.

Here we present results from several typical natural images: `barbara` contains some directional and non-directional periodic textures; `lena`, which can be regarded as one of the "curvelet-friendly" image, since it mainly consists of linear singularities at different scales; and `grain` image is a texture image which is considered to be very difficult to compress.

Table 1 gives denoising results in SNR using four algorithms. It seems that directional local cosine transform is the overall winner, although the TI-$\mathcal{S}$-BayesShrink sometimes outperforms in low noise levels, since the BayesShrink tends to optimise the MSE output. However, the results on other three images are more visually pleasing, preserving important directional features on the image. A detailed head-to-head comparison is presented in Figure 2 on `barbara`. It is obvious that from the TI-$\mathcal{S}$-BayesShrink denoised image, the diagonal stripes are absent on the cloth in the middle, although a few of such patterns can be seen on the trousers. The curvelet is able to recover some of those directional patterns, but incomplete nonetheless. These features are restored almost completely by our proposed methods.

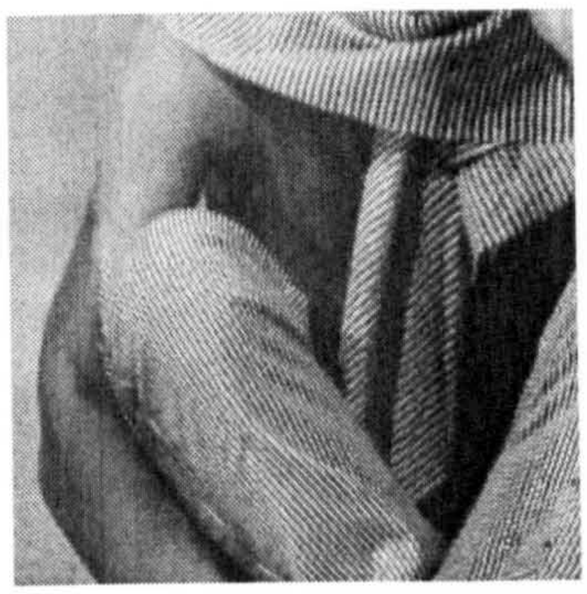
Since the directional cosine packets can be regarded as a generalisation of curvelets and directional cosine bases, it is not suprising to see it gives better results than its two counterparts. However, it involves considerable amount of extra computations, since the best basis has to be sought for each of the Radon slices. It seems that such heavy computation burden is not a fair tradeoff, at least at present, since the directional cosine transform denoised images are visually similar to directional local cosine's results.

4. CONCLUSION

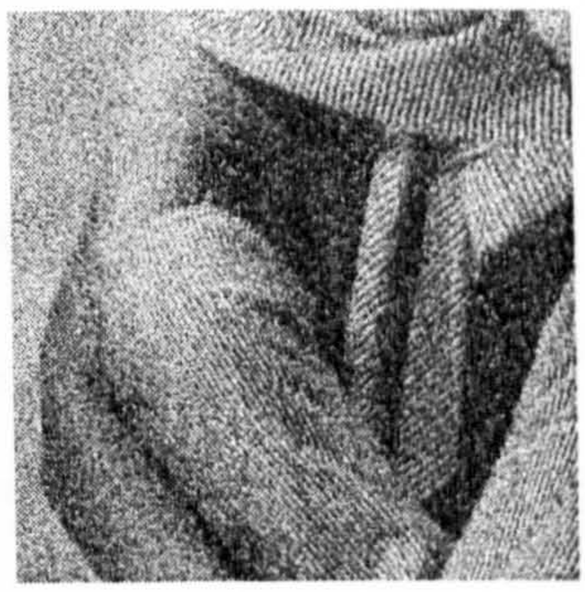
We have presented a set of new bases which have localisation in space and frequency, orientation selectivity and employ a multiresolution pyramidal framework allowing analyse of the image at different scales. In a sense the MDC bases qualify as geometric wavelets and share a lot of similarities with other directional wavelet bases proposed previously, but the construction allows us to capture local directional texture patches and linear features at ease. Unlike the traditional 2D local DCT constructed by tensor-product, the directional cosine basis provides orientation selectivity like the Fourier basis but avoids the unpleasant convergence rate of the Fourier series on intervals.

The effectiveness of the MDC bases was tested against the state-of-the-art translation-invariant wavelet packet based shrinkage method and a directional basis, the so-called *curvelet*. The new bases demonstrated a strong potential in the experiments, often outperforming the other methods. While producing much visually pleasant output than the wavelet packets with optimal threshold, the MDC seems to be able to capture a wider range of directional features than the curvelet, even without the local cosine treatment on Radon slices.

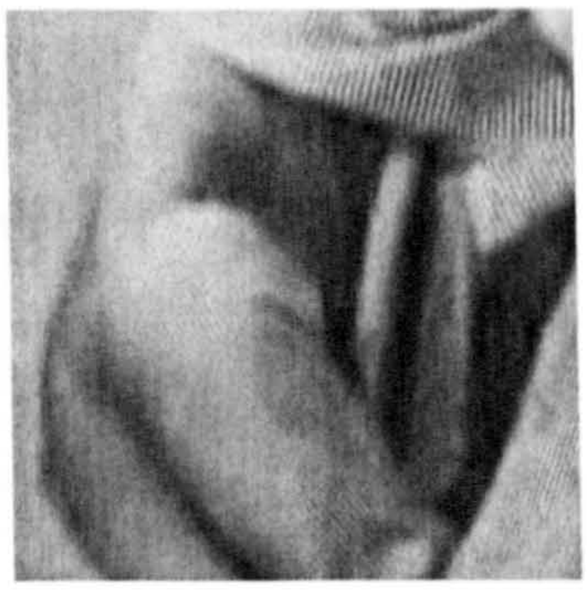
Of course the transform presented here with the Laplacian pyramid for the denoising experiments is only a taste for conducting multiresolution analysis with the directional cosine (packet) bases. A wide variation on the theme is pos-



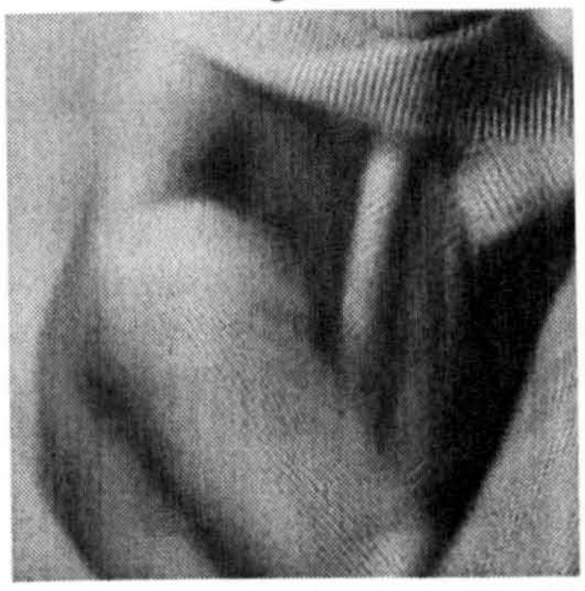
Original



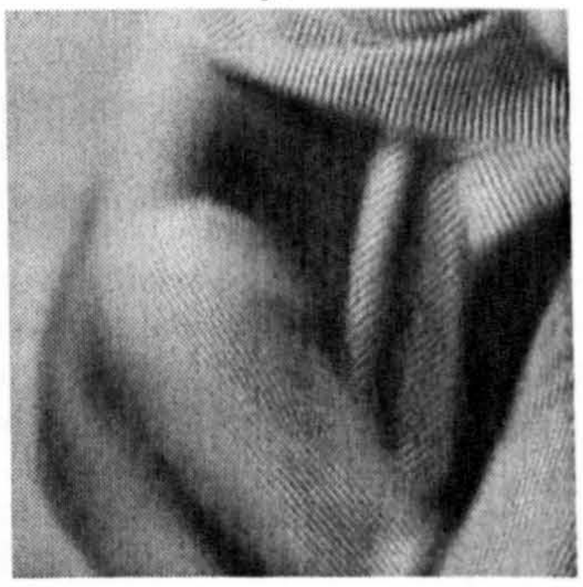
Noisy, 10dB



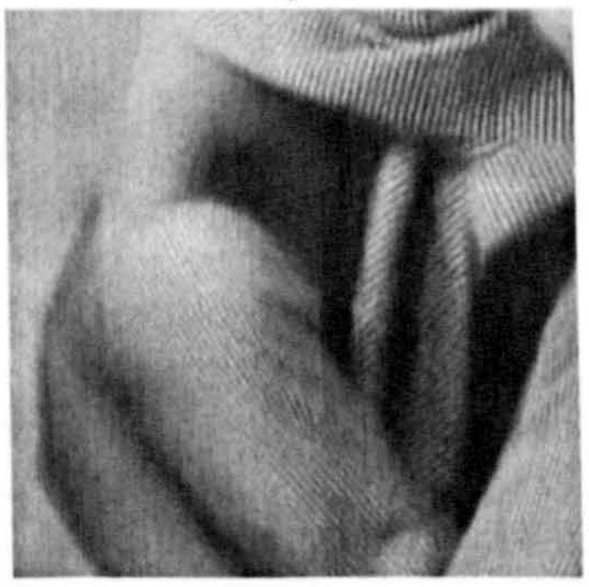
TIWP-S-Bayes, 18.00dB



Curvelet, 17.64dB



Directional cosine, 18.44dB



Directional cosine packets, 18.85dB

Fig. 2. Detailed comparative denoising results on `barbara`

sible, for example using variable sized windows on the original image might be another possibility, or to use the *algorithme á trous* subband decomposition for better denoising results. It is our intention to put forward this basis in a general way in this work to popularise its usage in various kinds of image processing tasks.

5. REFERENCES

[1] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.

[2] E.J. Candés, *Ridgelets: Theory and applications*, Ph.D. thesis, Dept. of Stats, Stanford Univ., Stanford, CA, 1998.

[3] E.J. Candés and D.L. Donoho, "Curvelets - a surprisingly effective nonadaptive representation for objects with edges," in *Curves and Surfaces*, C. Rabut, A. Cohen, and L.L. Schumaker, Eds., pp. 105–120. Vanderbilt University Press, Nashville, TN, 2000.

[4] J. Starck, E.J. Candés, and D.L. Donoho, "The curvelet transform for image denoising," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 670–684, June 2002.

[5] M.N. Do and M. Vetterli, "Contourlet," in *Beyond Wavelets*, G.V. Welland, Ed. Academic Press, 2003.

[6] A. Averbuch, R. Coifman, D.L. Donoho, and M. Israeli, "Fast slant stack: A notion of Radon transform for data in a Cartesian grid which is rapidly computible, algebraically exact, geometrically faithful and invertible," to appear in SIAM Scientific Computing.

[7] P.J. Burt and E.H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, pp. 532–540, 1983.

[8] D.L. Donoho and I.M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[9] N. Rajpoot, Z. Yao, and R. Wilson, "Adaptive wavelet restoration of noisy video sequences," in *Proceedings IEEE ICIP 2004*, Singapore, 2004.

[10] S.G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Transaction on Image Processing*, vol. 9, no. 9, pp. 1532–1546, September 2000.

[11] R.R. Coifman and D.L. Donoho, "Translation invariant denoising," in *Wavelets in Statistics*, A. Antoine and G. Oppenheim, Eds., pp. 125–150. Springer, New York, 1995.

Directional Wavelet Analysis with Fourier-type Bases for Image Processing

Zhen Yao, *Student Member, IEEE* , Nasir Rajpoot, *Member, IEEE* and Roland Wilson

Abstract—**Motivated by the fact that in natural images, there is usually a presence of local strongly oriented features such as directional textures and linear discontinuities, a representation which is both well-localised in frequency and orientation is desirable to efficiently describe those oriented features. Here we introduce a family of multiscale trigonometric bases for image processing using Fourier-type constructions, namely, the multiscale directional cosine transform and the multiscale Fourier transform. We also show that by seeking an adaptive basis locally, the proposed bases are able to capture both oriented harmonics as well as discontinuities, although the complexity of such adaptiveness varies significantly. We conducted denoising experiments with the proposed bases and the results show great promise of the proposed directional wavelet bases.**

Index Terms—**Directional wavelets, curvelets, Fourier transform, cosine transform, denoising, restoration.**

I. Introduction

THE application of transforms in image processing is often based on a separable construction. Rows and columns in an image are treated independently and the two-dimensional basis functions are simply tensor products of the corresponding one-dimensional functions. Such method keeps simplicity in terms of design and computation, but is not capable of capturing properly all the interesting features of an image. For example, the orthonormal separable wavelet transform [27] in higher dimensions is seriously limited in its ability to efficiently represent higher dimensional features such as lines. Furthermore, the lack of frequency selectivity remains an elusive problem with most techniques operating in the wavelet domain.

Edges and textures in an image can exist at all possible locations, orientations, and scales. The ability to efficiently analyse and describe directional patterns is thus of fundamental importance for image analysis and image compression. The idea that biological visual systems might analyse image along dimensions such as orientation, scale and frequency (ie. bandpass) dates back to the work by Hubel and Wiesel [25] in the 1960's. In the computational vision literature, the idea of analysing images along multiple orientations appears at the beginning of the seventies with the Binford-Horn's line-finder [2], [23] and later work by Granlund [22]. Many edge-based image representations have then been elaborated [20], [42] with different edge detection procedures and image approximations using jump models along these edges. To refine these models, multiscale edge representations using wavelet maxima [29] or an edge-adapted multiresolution representation [10] have also been studied. Edge based image representations with complete orthonormal families of foveal wavelets in [32] and footprints [19] have been introduced and studied to reconstruct the main image edge structures. To stabilize the edge detection, global optimization procedures have also been elaborated by Donoho [17], Shukla et al. [36] and Wakin et al. [40]. The optimal configuration of edges is then calculated with an image segmentation over dyadic squares using fast dynamic programming algorithms over quadtrees. Instead of describing the image geometry through edges, which are most often ill-defined, Le Pennec and Mallat later proposed a basis named *"bandelets"* [33] which characterises the image geometry with a geometric flow of vectors. Recently, Peyré and Mallat presented the *second generation bandelets* [34]. The decomposition is computed first by the standard wavelet transform, followed by adaptive geometric orthogonal filters. The compression results are significantly better than wavelet-based coders.

All the approaches previously discussed are *adaptive* representations, in the sense that the bases are adapted to the signal/image contents. Meanwhile, from a different heuristic principle, a number of researchers have been working on developing *fixed* directional representation bases for natural images. The idea of curvelets [7] is to represent a curve as a superposition of functions of various lengths and widths obeying the scaling law $width \approx length^2$. Several different methods were proposed to construct the curvelets. A digital implementation for the curvelet transform, more commonly referred as the *curvelet-99* was used in [37] for noise removal by Starck et al. The transform first decomposes the image into subbands, i.e., separating the object into a series of disjoint scales, using the *algorithme á trous* wavelet transform. Each scale is then analysed by means of a local windowed ridgelet [6] transform. The proposed transform is $16J+1$ times redundant, with J being the number of scales for decomposition. The same authors later proposed a combined approach with curvelets and wavelets in denoising [38]. Such joint sparse representation idea is related to the idea of *Matching Pursuit* (MP) and *Basis Pursuit* (BP) [28], and another application in image deconvolution was presented in [39].

While the redundancy certainly is a advantage in the area of image restoration, it is by no means an ideal transform for compression and other tasks. In order to construct a form of discrete curvelet frame with less redundancy, Do and

Z. Yao is with the Computer Science Department, University of Warwick, CV4 7AL UK. E-mail: yaozhen@ieee.org

N. Rajpoot is with the Computer Science Department, University of Warwick, CV4 7AL UK. E-mail: nasir@dcs.warwick.ac.uk

R. Wilson is with the Computer Science Department, University of Warwick, CV4 7AL UK. E-mail: rgw@dcs.warwick.ac.uk

Vetterli [15], [16] pioneered the *"contourlet"* transform by marrying the Laplacian pyramid and a directional filter bank. Such approach is called *"double filter bank"* structure. The Laplacian pyramid mainly is used for separating isotropic features into different resolutions, then the directional filter links the point discontinuities into linear structures. This allows contourlets to efficiently approximate a smooth contour at different scales. The double filter bank design certainly allows the contourlet to be flexibly constructed. In [26], Lu and Do developed a critically sampled contourlet transform called *"CRISP-contourlet* using a combined iterated non-separable filter bank for both multiscale and directional decomposition. A non-subsampled contourlet transform was recently proposed [14]. The Laplacian pyramid was substituted with a 2-channel non-subsampled 2D filter bank which is similar to the *á trous* wavelet expansion. However, with J levels of decomposition, it has $J+1$ redundancy. By contrast, the 2-D *á trous* algorithm by tensor product has $3J+1$ redundancy. The whole transform has $1+\sum_{j=1}^{J} 2^{l_j}$, where l_j denotes the number of levels in the transform at the j-th scale. Experimental results suggest that the transform compares favourably to other existing denoising and enhancement methodds reported in literature.

The curvelets can also be conveniently constructed from a frequency tiling approach. Such idea later adopted by Candés and Donoho [8] in constructing *second generation curvelets* which do not require ridgelets. Such tight frame can be computed more efficiently than the previous curvelet-99 implementation. A recent report [5] details its implementation using unequally-spaced fast Fourier transforms (USFFT) and the wrapping of specially selected Fourier samples. Both implmentations are improved in the sense that they are conceptually simpler, faster and far less redundant. The same strategy was used in constructing a 3D curvelet transform [43] whose basis functions are planar patches. These digital implementations can be found in the `CurveLab` distribution.

However, the assumption that natural images are characterised solely by linear edges is not true. Evidently we have seen attempts to separate the image into additive ingredients [31] - usually one is textural and the other is piecewisely smooth. This suggests that there is usually a presence of local strongly oriented harmonics (textures) separated by curvilinear edges. Sparse representations which are both well-localised in frequency and orientation is desirable to efficiently describe such oriented harmonic features. Also, it would be ideal to accommodate both directional linear features as well as directional periodic textures in a unified manner according to the *"image=texture+edge"* model. In this paper, we show that directional wavelet analysis can be performed with directional trigonometric transforms localised in a multiscale framework. We introduce the *Multiscale Directional Cosine Bases* in section II which can efficiently represent local oriented harmonics, and with a local directional cosine packet analysis, we can accommodate both directional periodic ridges and ridgelets which is a dual basis to the ridgelet packets. In section III, we show that directional singularities and harmonics can also be captured by the *Multiresolution Fourier transform* using a Gaussian model of its magnitude spectrum with less computational burden. Next we show some results from our denoising experiments with both transforms and compare them with other wavelet transforms in section IV. The paper concludes with a summary in section V.

II. The Multiscale Directional Cosine Transform

Like the 2D orthonormal wavelet transform, the discrete cosine transform (DCT) in 2D is also formed by tensor product, resulting in basis functions which look like "chessboard" patterns. Therefore we need to define a directional cosine operator in order to bring the orientation parameter into the transform. Also, we will need to localise the basis spatially in order to capture local features. This section describes the construction of the Multiscale Directional Cosine Transform (MDCT).

A. The Directional Cosine Basis

First we define the parametric space $\Gamma = \{\gamma = (k, \vec{\theta})\}$ where $k \in [0, 2\pi), \vec{\theta} \in \mathrm{S}^{d-1}$, $\vec{\theta}$ is on the unit sphere S^{d-1} in dimension d which indicates orientation and k indicates the frequency. Consider a family of orthonormal trigonometric basis for $L^2([0,1])$, derived from Fourier transform $\hat{f}(\xi) = \int e^{-ix\xi} f(x) dx$.

1) $\{\sqrt{2}\sin(\frac{2k+1}{2}\pi x)\}$, $k = 0, 1, 2, 3, \ldots$
2) $\{\sqrt{2}\sin(k\pi x)\}$, $k = 1, 2, 3, \ldots$
3) $\{\sqrt{2}\cos(\frac{2k+1}{2}\pi x)\}$, $k = 0, 1, 2, 3, \ldots$
4) $\{1, \sqrt{2}\cos(k\pi x)\}$, $k = 1, 2, 3, \ldots$

We denote such a trigonometric basis as $c_k(x)$, and the corresponding transform can be written as $\langle f, c_k \rangle$. Now we define the continuous directional trigonometric transform on a multi-variate function $f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d$:

$$C_\gamma(\mathbf{x}) = c_k(\vec{\theta} \cdot \mathbf{x}) \tag{1}$$

Since $\langle f, c_k \rangle$ is essentially a Fourier transform, we have the admissibility condition

$$K_C = \int \frac{|\hat{C}(\xi)|^2}{|\xi|^d} d\xi < \infty \tag{2}$$

and the reconstruction is

$$f = \int \langle f, C_\gamma \rangle C_\gamma \mu(d\gamma) \tag{3}$$

The Parseval relation holds

$$\|f\|_2^2 = \int |\langle f, C_\gamma \rangle|^2 \mu(d\gamma) \tag{4}$$

For a general image representation, we choose the $c_k = \{1, \sqrt{2}\cos(k\pi x)\}$, known as the cosine II basis, which has faster decay on interval $[0, 1]$ than the Fourier transform. We then have the directional 2D basis

$$C_{k,\theta} = \lambda_k \cos(\pi k(x\cos\theta + y\sin\theta)) \tag{5}$$

where $\lambda_k = \begin{cases} 1 & \text{if } k = 0 \\ \sqrt{2} & \text{if } x \neq 0. \end{cases}$

The directional 2D continuous cosine transform is defined as

$$\begin{aligned} Cf(k,\theta) &= \langle f, C_{k,\theta} \rangle \\ &= \int_{\mathbf{R}^2} \lambda_k f(x,y) \cos(\pi k(x\cos\theta + y\sin\theta)) dxdy \end{aligned}$$

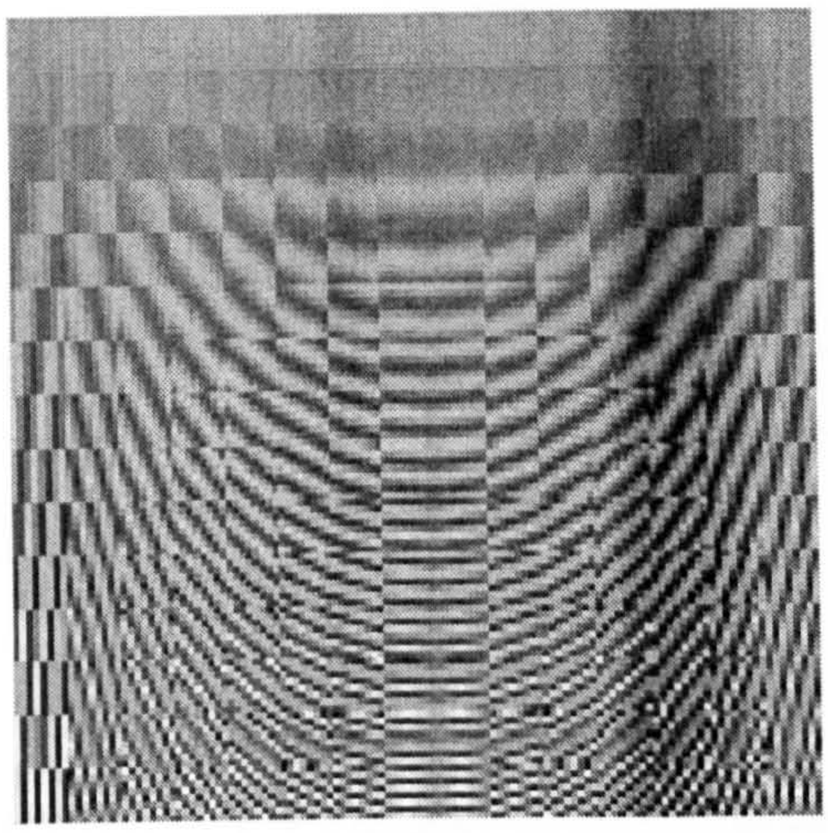


Fig. 1. The 8×8 directional cosine basis vectors

The directional cosine basis vectors are indexed by frequency k and direction θ, as can be seen in Figure 1. It is obvious that the basis vectors look similar to the Fourier basis despite the fact that the directional cosine transform is real-to-real instead of real-to-complex and its approximation error decays more rapidly than the Fourier counterpart, due to its symmetrical boundary extension.

B. Directional Cosine Packets

Smooth local trigonometric bases proposed by Coifman and Meyer [11] and by Malvar [30] use smooth window functions to split the signal and to fold overlapping parts back into the pieces so that the orthogonality is preserved. Therefore, the folded signal is suited for representation by a trigonometric basis. At its simplest, a typical local cosine basis function has the form :

$$\phi_{n,k}(x) = b_n(x)\cos(k+\frac{1}{2})\pi x. \tag{6}$$

where b_n is a smooth window or a *bell function*.

The usefulness of applying smooth local trigonometric bases to focus on local interesting properties of a signal is well studied, and their applications such as MP3 audio compression have been demonstrated to be successful. As we have discussed, image usually consists regions of homogenous textures separated by linear edges and contours. It is then natural to consider applying local cosine basis on Radon projected slices in order to represent both periodic patterns and some linear singularities. In this work, we use the Coifman-Wickerhauser's entropy-based best basis algorithm [13] to look for the best local cosine basis with dyadic interval. The resulting adaptive basis is similar to one possible "*ridgelet packets*" construction mentioned in [21]. We will thereafter refer to such dictionary of bases in the Radon domain as the "*directional cosine packets*".

C. Multiscale Digital Implementation

For an image representation basis to be useful, the basis vectors should be localised both in space and frequency, and they should have certain orientation selectivity. More importantly, to capture patterns of interest at different scales, the basis need to be multiresolution. A prototypical MDC function has the form

$$\psi_{k,\theta,s,\mathbf{t}}(\mathbf{x}) = b\left(\frac{\mathbf{x}-\mathbf{t}}{s}\right) C_{k,\theta}\left(\frac{\mathbf{x}-\mathbf{t}}{s}\right). \tag{7}$$

where k, θ, $\mathbf{t}$ and s denotes the frequency, orientation, location and scale parameters of the function respectively and $b(\cdot)$ is the smooth bell function chosen along with the sampling interval to ensure invertibility of the discrete form of the transform.

The discrete implementation of MDCT is similar to the digital curvelet-99 construction. While the discrete cosine transform and discrete Radon transform [1] are well studied in the literature, a combination of these two transforms gives us the discrete directional cosine operator. Unlike curvelet-99 which is very redundant, the multiresolution property of the MDC transform is given by the well-known decimated Laplacian pyramid [3]. The discrete MDC of a 2D vector $\mathbf{x}$, at scale s is given by

$$\mathbf{X}_s = C_n(\mathbf{I} - \mathbf{G}_{s,s+1}\mathbf{G}_{s+1,s})\mathbf{x}_s. \tag{8}$$

where $\mathbf{X}_s$ denotes the transform at scale s, C_n is the discrete directional cosine transform operator with window size $n \times n$, $\mathbf{I}$ is the identity operator, $\mathbf{x}_s$ is the Gaussian pyramid representation of $\mathbf{x}$ at scale s

$$\mathbf{x}_s = \prod_{l=0}^{s-1} \mathbf{G}_{l+1,l}\mathbf{x}. \tag{9}$$

and $\mathbf{G}_{s,s+1}, \mathbf{G}_{s+1,s}$ are the raising and lowering operators associated with transitions between levels in the Gaussian pyramid. We certainly have the choice of using the directional cosine packets as the transform operator C_n by substituting the cosine transform by a cosine packet operator ψ_n, forming a semi-adaptive basis. In this way, the MDC packet basis is able to capture a wide range of directional features at different resolutions.

III. The Multiresolution Fourier Transform

The MDCT is similar to the curvelet-99 transform, only the wavelet ridge function is replaced by the cosine basis. With the local cosine analysis on Radon slices, the MDC packet bases fits well to the " *image = edges + textures*" model.

However, the proposed bases have two limitations. The first problem is that best basis for local cosine packets has to be sought on every Radon slice, making the computation extremely expensive. Secondly, the Radon transform is a redundant transform and its inverse introduces some numerical errors. One might note that the Radon transform is directly related to the Fourier transform by the Fourier Slice Theorem, briefly stated as below:

Theorem 1: (Fourier Slice Theorem). The 1D Fourier transform with respect to t of the projection $Rf(t,\theta)$ is equal to a central slice, at angle θ, of the 2D Fourier transform of the function $f(x,y)$, that is,

$$\hat{Rf}(t,\theta) = \hat{f}(\xi\cos\theta, \xi\sin\theta). \tag{10}$$

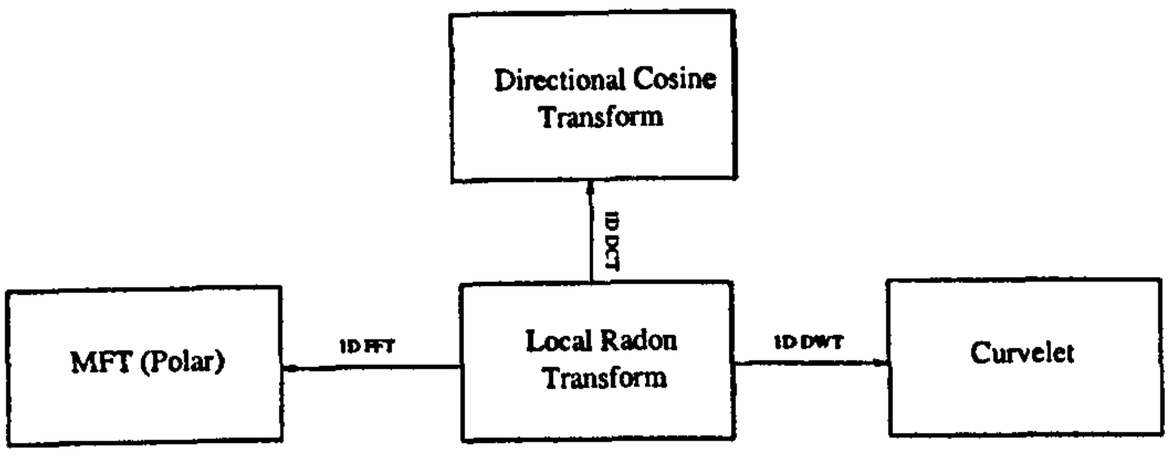


Fig. 2. The relationships between MFT, Radon, Curvelet and directional cosine basis

where

$$\hat{f}(\xi_1, \xi_2) = \int \int f(x, y) e^{-2\pi i(x\xi_1 + y\xi_2)} dx dy.$$

is the 2D Fourier transform of $f(x, y)$.

Since many discrete Radon transform are implemented via this theorem, it suggests that we may be able to perform such directional analysis using the standard Fourier transform.

A. The MFT Implementation

The *Multiresolution Fourier Transform* (MFT) [4], [41] has been proposed as a combination of STFT and wavelet methods, which inherits many of the desired features of both. With the windowing function $g(t)$, the transform of a function $f \in L^2(\mathbb{R})$ at position u frequency ξ and scale s is defined as:

$$Mf(u, \xi, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(t) g(s(t-u)) e^{-i\xi t} dt. \quad (11)$$

In effect, it is simply a stack of windowed Fourier transforms, in which the scale of the analysis window is varied systematically with the stack index. As a general image analysis tool, it has been applied in feature extraction and segmentation with music and image analysis, such as music note segmentation and extracting boundary curves in a multiresolution fashion [41]. It has also been used in texture synthesis and analysis [24] and many other areas.

The discrete implementation of MFT can take many forms. Similar to the construction of the digital MDCT described before. We build the MFT on top of the Laplacian pyramid, then on each level of the pyramid, windowed Fourier transform is performed with the same window size regardless of the scale. The discrete MFT of a 2D vector $\mathbf{x}$, at scale s is given by

$$\hat{\mathbf{x}}_s = \mathcal{F}_n(\mathbf{I} - \mathbf{G}_{s,s+1}\mathbf{G}_{s+1,s})\mathbf{x}_s. \quad (12)$$

where $\mathcal{F}_n$ is the discrete Fourier transform operator with window size $n \times n$. The closeness of the Burt and Adelson filter to a Gaussian function gives the pyramid virtually isotropic behavior, which can be well exploited by the high frequency resolution of the Fourier basis. The whole transform is some 5.33 times redundant if overlapping window is used.

We can see that the only difference between MFT and the MDCT is that the operator used here is just a Fourier transform. In fact, the polar separablity of the Fourier transform suggests that it is also a directional trigonometric transform and Radon transform was implemented via the Fourier-slice theorem by inverse Fourier transform on Fourier polar slices. The relations between MFT, MDCT, Radon and curvelet transform can be illustrated in Figure 2. It is obvious that it requires an inverse Fourier transform and a cosine transform to convert the Fourier domain into the directional cosine domain. Although it has some advantage in approximation convergence, the extra computation is 2 times more than the conventional Fourier transform.

B. Gaussian Modelling of Fourier Spectrum

The Fourier basis is a natural representation for directional periodic patterns, although it decays slower than a cosine basis in terms of approximation. In order to perform some sort of *"curvelet"* analysis, we need a model for linear features in the Fourier domain. Fortunately, such model is not difficult to derive, since a line in the spatial domain will be transformed into another line in its Fourier domain perpendicular to its direction.

The magnitude intensity of the local Fourier spectrum can be modelled as a single 2D Gaussian function with its centroid fixed at the origin, which means the Gaussian is zero-mean :

$$G(\mathbf{x}) = \frac{1}{2\pi} \exp\left(\frac{-\mathbf{x}^T C^{-1} \mathbf{x}}{2}\right). \quad (13)$$

The covariance matrix C of the Gaussian $G(\cdot)$ can be obtained from the inertia tensor of the spectrum.

$$C = \sum_{\vec{\omega}} |\hat{f}(\vec{\omega})|^2 \vec{\omega}\vec{\omega}^T. \quad (14)$$

where the $\hat{f}(\vec{\omega})$ denotes the Fourier coefficients. With the covariance matrix C, the shape and the orientation of the Gaussian is determined. The centroid of the feature $\mathbf{x}_0$ can then be estimated by taking the pairwise average correlations between neighbouring coefficients in each of the horizontal and vertical directions :

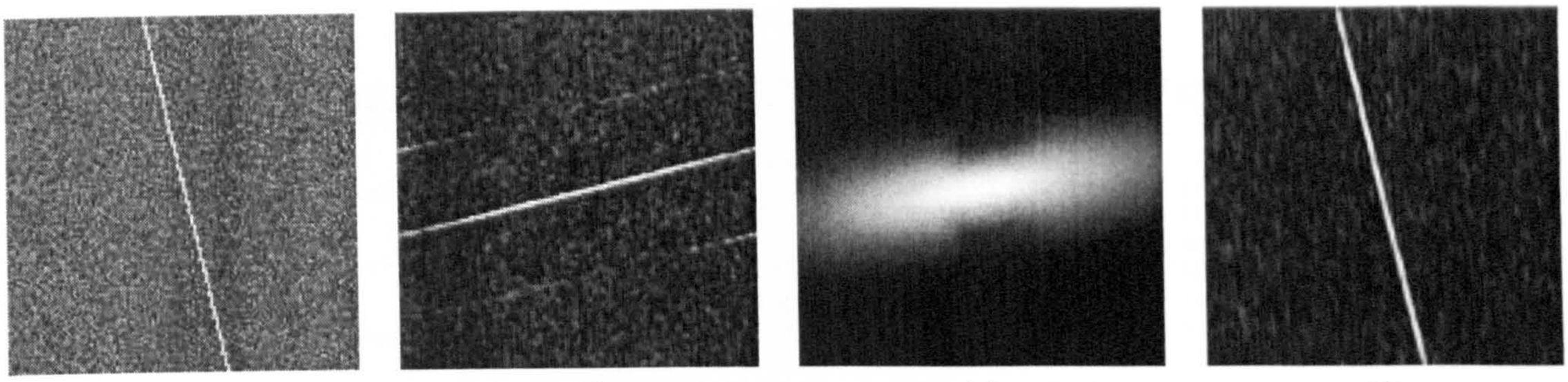
$$\mathbf{x}_0 = \frac{B}{2\pi} \arg\left(\sum_{\vec{\omega}} \hat{f}(\vec{\omega} - 1)\hat{f}(\vec{\omega})^*\right). \quad (15)$$

where B is the windowing size.

The choice of using the Gaussian function is due to several reasons. First, the uncertainty principle states that the Gaussian function can achieve optimal spread in space and frequency, and it is smooth in both domains. Secondly, the shape of the 2D Gaussian function can be both isotropic and anisotropic. When the covariance matrix gives an anisotropic Gaussian distribution, this suggests that the spatial feature is a linear shape. In other cases, the Gaussian blobs will tend to be isotropic. A simple measure of the anisotropy can be obtained by performing the Principal Components Analysis (PCA) on the covariance matrix C, yielding two eignvalues λ_1 and λ_2, where $\lambda_1 \leq \lambda_2$. The measure is simply:

$$\mathcal{A} = \left|\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right| \quad (16)$$

To test the effectiveness of our model, we use a simple linear feature with Gaussian white noise. The noisy line (see Figure



(a) (b) (c) (d)

Fig. 3. (a) a noisy line, (b) the Fourier transform of the noisy line, (c) The Gaussian filter estimated from inertia tensor, (d) the denoised line.

3.(a)) is transformed into its Fourier domain (see Figure 3.(b)). In order to suppress noise, we can use the $G(\mathbf{x})$ as a frequency filter in the Fourier domain. However, the problem is that the inertia tensor itself is easily affected by noise. Our solution is to apply thresholding on the noisy transformed data as a stage of pre-processing, in order to suppress the noise energy. The inertia tensor C then can be more reliably estimated from the thresholded data. Therefore, the resulting Gaussian frequency filter is estimated from a thresholded version of Figure 3.(b). The inversion is a clean line image without most of the noise energy. In this way, we have achieved a directional *ridgelet*-like analysis with the Fourier basis, based on the single-feature hypothesis. While the assumption is not realistic for a natural image, such library of wave packets will work well locally in a multiresolution setting. A combination of this model to the MFT allows us to analyse the signal adaptively, so that many features including contours and textures can be captured effectively.

IV. Image Denoising Experiments

Good bases for reprensenting images should be able to capture important features of interest, so that the reconstruction requires as few basis functions as possible. The bases' effectiveness can be tested by performing denoising experiments by simple thresholding in the transformed domain. For the MDCT and MDC packet transform, the denoising experiments are performed in such settings:

- The Laplacian pyramid is decomposed at 5 levels of subbands.
- The window size n is chosen at 16×16, modulated with a squared cosine.
- The windows are overlapped by 50%.

The thresholding we use is a form of the universal thresholding proposed in [18], multiplied by an extra constant a, $\Theta = a\sqrt{2 \log N}\sigma / 1.23^L$, where $N = n^2 = 256$ here and L denotes the level of decomposition, while $L = 0$ corresponds to the highest frequency subband. For directional cosine denoising, $a = 0.08$ was found to give satisfactory result. For the directional local cosine packets, $a = 0.062$ was used. The lowpass subband is left intact.

The settings for the MFT are generally the same as for the MDCT, only with a little sophistication on estimating the filter and $a = 0.8$

1) Within each Fourier transformed block $\hat{B}$, we apply the threshold $8\sqrt{2 \log 256}\sigma$ on that to obtain $\hat{B}_T$, from which the inertia tensor C will be estimated.
2) If $\mathcal{A}_C > 0.43$, which means there is a strong directional feature present, the Gaussian filter generated from C will apply on the original noisy spectrum $\hat{B}$ to obtain $\hat{B}_G$, the inverse Fourier transform is taken on $(\hat{B}_G + \hat{B}_T)/2$ as the denoised block.
3) Otherwise, we take the $\hat{B}_T$ as the denoised block.

The Fourier-type transforms are compared with two algorithms. The first is a wavelet-packet based wavelet shrinkage algorithm which is described in [35], called S-Bayes in which the thresholding function is a modified version of the *BayesShrink* [9]. The best wavelet packet basis is sought by using the Shannon entropy function and cycle-spinning [12] is used to suppress the pseudo-Gibbs artifact. Essentially such treatment gives the translation invariance to the wavelet packet basis, which is known to be good in representing periodic signals as well as discontinuities.

The second is a modified version of curvelet. The curvelet-99 implementation reported in [37], which uses a much more redundant overcomplete wavelet frame than our MDCT and MFT, is a "specialised" transform to perform denoising task instead of general-purpose image processing. Therefore, here the local ridgelets are placed on the Laplacian pyramid as in our setting, in order to carry out a fair comparison.

We have conducted experiments on a wide range of natural images. Three of them present some typical characteristics: `barbara` contains some directional and non-directional periodic textures; `lena`, which can be regarded as one of the "curvelet-friendly" image, since it mainly consists of linear discontinuities at different scales; the `grain` image is a texture image which was considered to be very difficult to compress. It contains many directional components, however very irregular.

Table I gives denoising results in SNR by those five bases, where the best numbers are stressed in bold. We see that the best results are always among the MDC packet and the MFT, while the MDC packet seems to be more effective in more noisy situations. This is due to the fact that the Gaussian filters are estimated from the noise-sensitive inertia tensor. When the spectrum are dominated by the noise energy, a simple thresholding would fail to preserve the signal information. However, the MFT can be considered as the overall winner: it compares

TABLE I

THE COMPARATIVE IMAGE DENOISING RESULTS IN SNR

Image	Noise (dB)	TIWP	Curvelet	MDCT	MDC Packet	MFT
barbara	0	14.75	14.59	14.89	**15.19**	14.80
	5	16.08	15.93	16.42	16.85	**17.10**
	10	18.00	17.64	18.44	18.85	**19.68**
	15	21.12	19.64	20.34	20.71	**22.46**
	20	24.92	21.41	21.88	22.10	**24.65**
lena	0	17.06	17.10	17.24	17.75	16.63
	5	18.96	18.82	18.99	**19.63**	19.02
	10	21.08	20.83	21.05	**21.89**	21.60
	15	23.49	23.10	23.35	24.26	**24.43**
	20	26.15	25.35	25.60	26.31	**26.94**
grain	0	12.93	13.06	13.10	13.22	12.85
	5	13.81	13.66	13.84	**14.29**	14.01
	10	15.87	15.01	15.51	16.18	**16.30**
	15	18.72	17.28	18.00	18.84	**19.66**
	20	22.02	20.05	20.56	21.38	**22.82**

well with other denoising methods and its computational cost is much lower than the rest of the methods. We also notice that TIWP-S-BayesShrink sometimes outperforms at some low noise levels, since the BayesShrink tends to optimise the MSE output. However, the visual qualities of other candidates are more pleasing, preserving important directional features on the image.

A detailed head-to-head comparison is presented in Figure 4 on `barbara`. It is obvious that from the TIWP-S-BayesShrink thresholded image, the diagonal strips are absent on the cloth in the middle, although a few of such patterns can be seen on the trousers. The curvelet is able to recover some of those directional patterns, but incomplete nonetheless. These features are restored almost completely by our proposed methods.

Since the directional cosine packets can be regarded as a generalisation of the curvelets and directional cosine bases, it is not suprising to see it gives better results than these two counterparts. However, it introduces considerable amount of extra computations, since the best basis has to be sought for each of the Radon slices. Also, the inverse Radon transform from incomplete data introduces numerical errors in the reconstruction. Although visually MFT and MDC packet both captured linear and oscillating patterns, MFT's reconstruction is much sharper and outperforms by quite a margin in SNR. On the other hand, the best-basis search ensures that the denoised image from MDC packet is smoother and almost "artifact-free". Considering the visual/statistical performance and the complexity, the MFT with Gaussian filter is the overall winner.

V. CONCLUSION

In this paper, we have reviewed a growing literature body on directional wavelets' construction, analysis and their applications. The contribution of this paper is to introduce sets of Fourier-type bases which have localisation in space and frequency, orientation selectivity, and employing a multiresolution pyramidal framework allowing analyses of images at different scales.

In a sense, the Fourier-type bases qualify as geometrical wavelets and share a lot of similarities with other directional wavelet bases proposed previously. But the semi-adaptiveness allows us to capture local directional texture patches and linear features at ease and we have shown that by a simple Gaussian frequency filter model of magnitude spectrum intensity, analysis of directional harmonics and linear features can be carried out much more efficiently than any other directional wavelet bases proposed to date. It can be considered as a parametric curvelet representation, or a *generalised directional wavelet packets* and the simple inertia tensor method has demonstrated to be a good substitute for the Cartesian-polar conversion. More importantly, its computational cost is a big advantage, since it does not involve a notion of Radon transform, nor the best-basis search as in some adaptive representations.

The effectiveness of the proposed bases was tested against the state-of-the-art translation-invariant wavelet packet based shrinkage method and the curvelets. The new bases demonstrated a strong potential in the experiments, outperforming the opponents by quite a margin. While producing much visually pleasant output than the wavlet packets with optimal threshold, the MDC bases seems to be able to capture a wider range of directional features than the curvelet, even without the local cosine treatment.

The denoising experiments show the effectiveness of conducting multiresolution analysis with these bases. A wide variation on the theme is possible, for example using variable sized windows on the original image might be another possibility, or to use the *algorithme á trous* subband decomposition for better denoising results. The usage of the Gaussian frequency filter and its parameter estimation in noisy environments are still under investigation. It is our intention to put forward these bases in a general way in this work to popularise their usage in various kinds of image processing tasks.

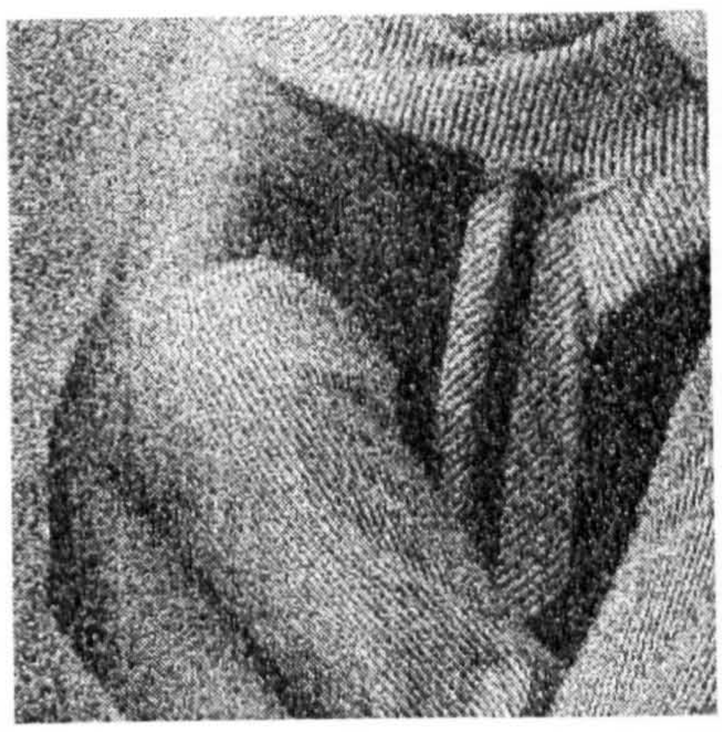
REFERENCES

[1] A. Averbuch, R. Coifman, D.L. Donoho, and M. Israeli. Fast slant stack: A notion of radon transform for data in a cartesian grid which is rapidly computible, algebraically exact, geometrically faithful and invertible. to appear in SIAM Scientifi c Computing.

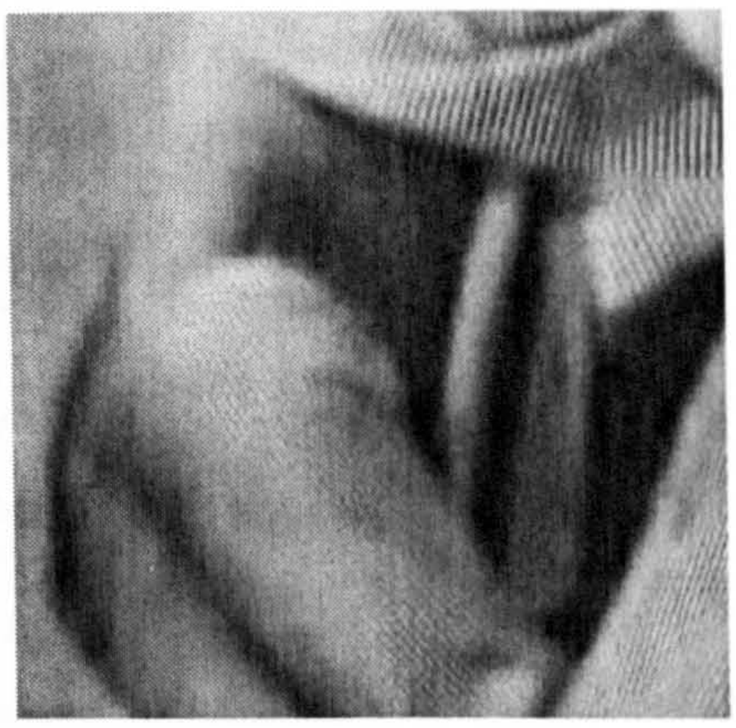
[2] T. Binford. Inferring surfaces from images. *Artifi cial Intelligence*, 17:205–244, 1981.

[3] P.J. Burt and E.H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532–540, 1983.

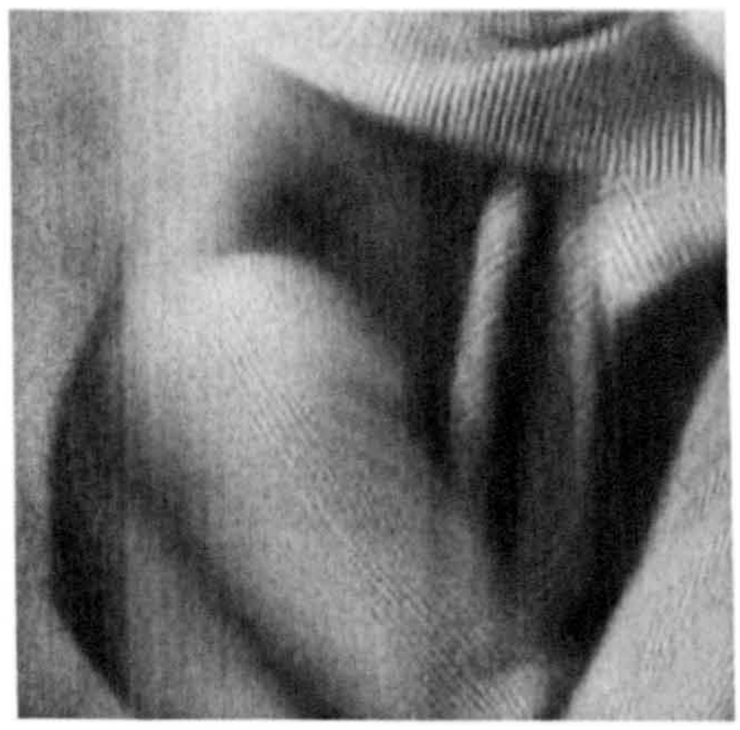
[4] A. Calway. *The Multiresolution Fourier Transform: A General Purpose Tool for Image Analysis*. PhD thesis, University of Warwick, 1989.



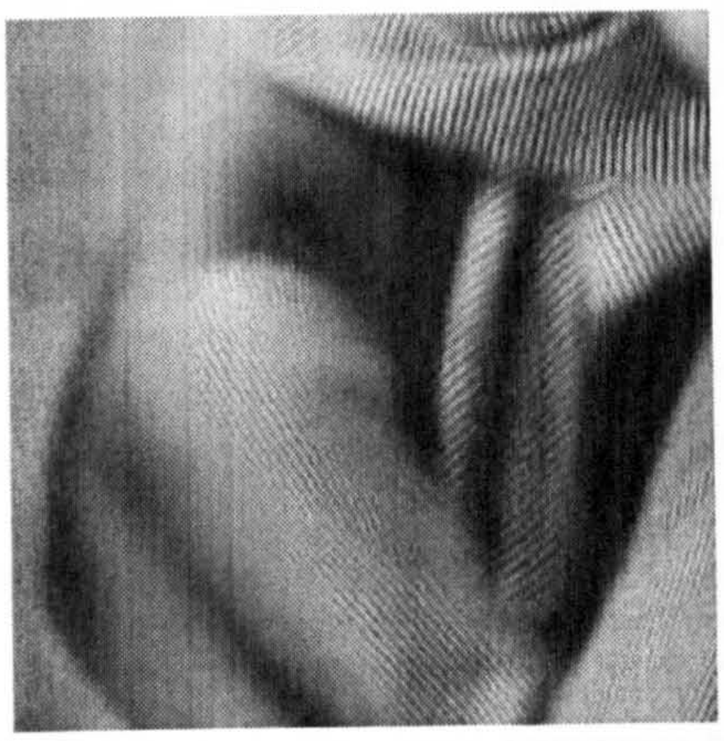
Noisy, 10dB



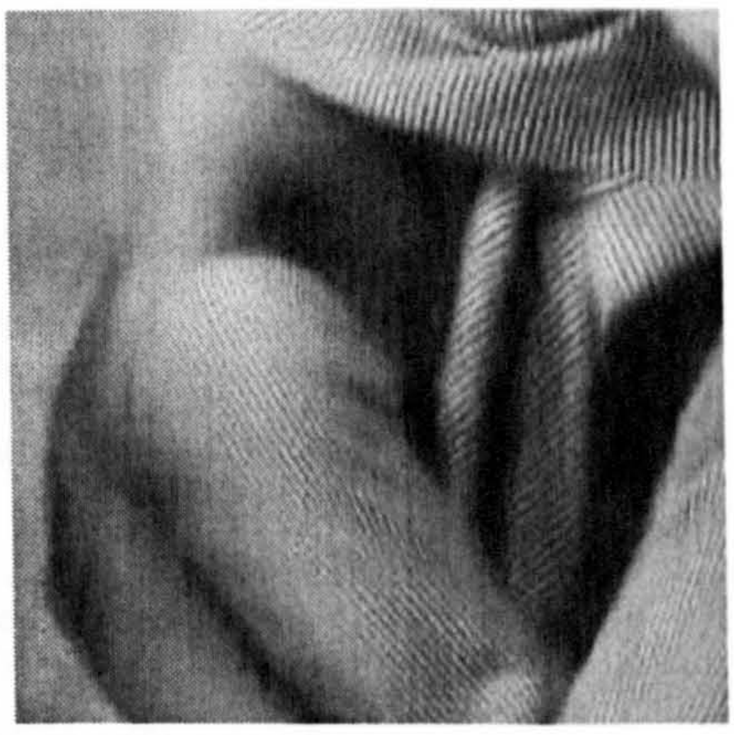
TIWP-$\mathcal{S}$-Bayes, 18.00dB



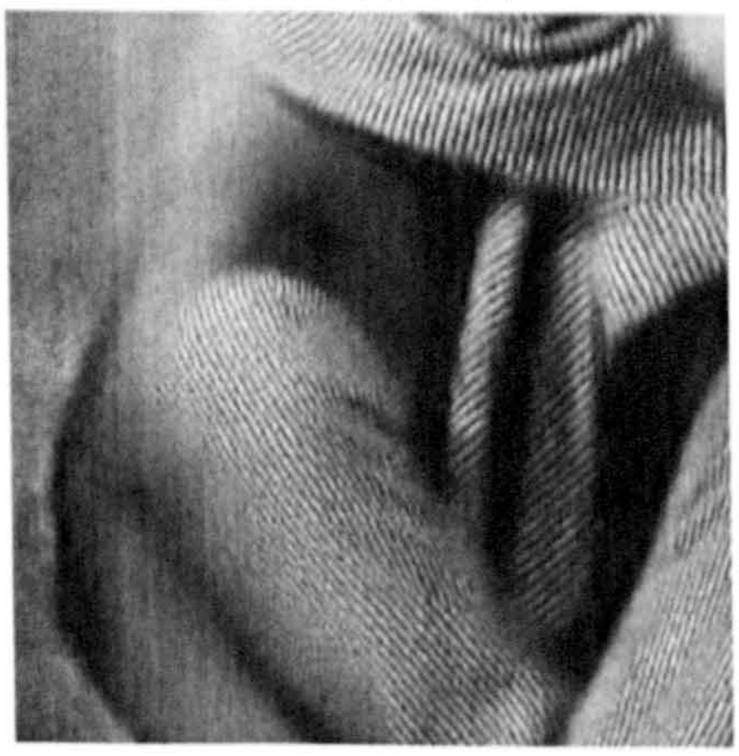
Curvelet, 17.64dB



Directional cosine, 18.44dB



Directional local cosine, 18.85dB



MFT, 19.68dB

Fig. 4. Detailed comparative denoising results on `barbara`

[5] E. Candés, L. Demanet, D. Donoho, and L. Ying. Fast discrete curvelet transforms. Technical report, California Institute of Technology, July 2005.

[6] E.J. Candés. *Ridgelets: Theory and applications*. PhD thesis, Dept. of Stats, Stanford Univ., Stanford, CA, 1998.

[7] E.J. Candés and D.L. Donoho. Curvelets - a suprisingly effective non-adaptive representation for objects with edges. In C. Rabut, A. Cohen, and L.L. Schumaker, editors, *Curves and Surfaces*, pages 105–120. Vanderbilt University Press, Nashville, TN, 2000.

[8] E.J. Candés and D.L. Donoho. New tight frames of curvelets and optimal representations of objects with C^2 smooth singularities. Technical report, Department of Statistics, Stanford University, 2002.

[9] S.G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transaction on Image Processing*, 9(9):1532–1546, September 2000.

[10] A. Cohen and B. Matei. Nonlinear subdivisions schemes: Applications to image processing. In A. Iske, E. Quack, and M. Floater, editors, *Tutorial on Multiresolution in Geometric Modelling*. Springer, New York, 2002.

[11] R. Coifman and Y. Meyer. Remarques sur l'analyse de fourier à fenêtre. *C.R. Acad. Sci. Paris Sér. I Math.*, I(312):259–261, 1991.

[12] R.R. Coifman and D.L. Donoho. Translation invariant denoising. In A. Antoine and G. Oppenheim, editors, *Wavelets in Statistics*, pages 125–150. Springer, New York, 1995.

[13] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. on Information Theory*, 38(2):713–718, 1992.

[14] A.L. Cunha, J. Zhou, and M.N. Do. The nonsubsampled contourlet transform:Theory, Design and Applications. *IEEE Transactions on Image Processing*, 2005. submitted.

[15] M.N. Do and M. Vetterli. Contourlet. In G.V. Welland, editor, *Beyond Wavelets*. Academic Press, 2003.

[16] M.N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. Submitted to IEEE Transactions Image Processing, 2003.

[17] D. Donoho. Wedgelets: Nearly-minimax estimation of edges. *Ann. Stat.*, 27:353–382, 1999.

[18] D.L. Donoho and I.M. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[19] M. Dragotti and M. Vetterli. Wavelet footprints: Theory, algorithm and applications. *IEEE Transactions on Signal Processing*, 51(5):1306–1323, May 2003.

[20] J. Elder. Are edges incomplete? *International Journal Computer Vision*, 34(2/3):97–122, 1999.

[21] A.G. Flesia, H. Hel-Or, A. Averbuch, E.J. Candés, R.R. Coifman, and D.L. Donoho. Digital implementation of ridgelet packets. In G. Welland, editor, *Beyond Wavelets*, pages 31–60. Academic Press, Sep. 2003.

[22] G.H. Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8:155–173, 1978.

[23] B. Horn. The Binford-Horn linefinder. Technical Report Memo 285, MIT AI Lab, 1971.

[24] T.I. Hsu and R.G. Wilson. A two-component model of texture for analysis and synthesis. *IEEE Transactions on Image Processing*, 7:1466–1476, 1998.

[25] D.H. Hubel and T.N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, pages 106–154, 1962.

[26] Y. Lu and M.N. Do. CRISP-contourlet: a critically sampled directional multiresolution image representation. In *Proceedings SPIE conference on Wavelet Applications in Signal and Image Processing*, San Diego, Aug. 2003.

[27] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:674–693, July 1989.

[28] S. Mallat and Z. Zhang. Atomic decomposition by basis pursuit. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[29] S. Mallat and S.S. Zhong. Wavelet transform maxima and multiscale edges. In M.B. Rskai et al., editor, *Wavelets and their applications*. Jones and Bartlett, 1992.

[30] H.S. Malvar. Lapped transforms for effi cient transform/subband coding. *IEEE Trans. Acoust. Speech Signal Processing*, 38:969–978, 1990.

[31] F. Meyer, A. Averbuch, and R. Coifman. Multilayered image representation: Application to image compr ession. *IEEE Transactions on Image Processing*, 11:1072–1080, 2002.

[32] E. Le Pennec and S. Mallat. Image compression with geometrical wavelets. In *presented at IEEE ICIP*, Vancouver, BC, Canada, Sep. 2000.

[33] E. Le Pennec and S. Mallat. Sparse geometric image representations with bandelets. *IEEE Transactions on Image Processing*, 14(4):423–438, April 2005.

[34] G. Peyré and S. Mallat. Discrete bandelets with geometric orthogonal fi lters. In *Proceedings IEEE ICIP*, 2005. submitted.

[35] N. Rajpoot, Z. Yao, and R. Wilson. Adaptive wavelet restoration of noisy video sequences. In *Proceedings IEEE ICIP 2004*, Singapore, 2004. To Appear.

[36] R. Shukla, P.L. Dragotti, M.N. Do, and M. Vetterli. Rate-distortion optimized tree structured compression algorithms. *IEEE Transactions on Image Processing*, to be published.

[37] J. Starck, E.J. Candés, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, June 2002.

[38] J. Starck, D.L. Donoho, and E.J. Very high quality image restoration by combining wavelets and curvelets. In *Proceedings of SPIE Wavelets: Applications in Signal and Image Processing IX*, volume 4478, pages 9–19, 2001.

[39] J. Starck, M.K. Nguyen, and F. Murtagh. Wavelets and curvelets for image deconvolution: a combined approach. *Signal Processing*, 83:2279–2283, 2003.

[40] M. Wakin, J. Romberg, H. Choi, and R. Baraniuk. Rate-distortion optimized image compression using wedgelets. In *Proceedings IEEE ICIP*, volume 3, pages 237–240, 2002.

[41] R. Wilson, A.D. Calway, and E.R.S. Pearson. A generalized wavelet transform for fourier analysis: the multiresolution fourier transform and its application to image and audio signal analysis. *IEEE Trans. on Information Theory*, 38:674–690, 1992.

[42] X. Xue and X. Wu. Image compression based on multi-scale edge compensation. In *Proceedings IEEE ICIP*, volume 3, pages 560–564, 1999.

[43] L. Ying, L. Demanet, and E.J. Candés. 3d discrete curvelet transform. In *Proceedings SPIE Wavelet XI*, volume 5914, 2005.

IMAGE DENOISING WITH DIRECTIONAL BASES

Heechan Park *Graham R Martin* *Zhen Yao*

Department of Computer Science
University of Warwick, Coventry United Kingdom
email: {heechan, grm, yao}@dcs.warwick.ac.uk

ABSTRACT

Directional information is an important component of both natural and synthetic images, and it is exploited in many image processing applications. Directional basis analysis is used to capture significant structural information. This paper presents an empirical study of image denoising with directional bases. We consider two distinct approaches. One involves the Multi-resolution Fourier Transform (MFT) facilitated with a multi-directional selective filter. The other is based on statistics, Independent Component Analysis (ICA) that adaptively decomposes an image into a set of directional bases. We then present a combined approach that benefits from the computational efficiency of the MFT and the data adaptiveness of ICA. Experimental results are compared with those from other recent directional transforms such as the Curvelet and Directional cosine transform.

1. INTRODUCTION

Removal of noise from noisy images to obtain the unknown original image is often referred to as *denoising*. Gaussian additive white noise has a frequency spectrum that is continuous and uniform over a specified frequency band. It is spatially uncorrelated, and the noise for each pixel is independent and identically distributed (iid).

$$\bar{f} = f + \xi \tag{1}$$

where $\bar{f}$ is the noisy image, f is the original image and ξ is i.i.d. noise. Images are assumed to be linear-shift invariant, and linear methods such as the Wiener filter and the Kalman filter are often employed for denoising. Linear denoising methods are simple and inexpensive to implement, however they tend to blur the edge structure of the image, structure that is very important to the human visual system. The Markov random field, as well as various extensions, has been utilised to model the contextual information embedded in image formation. Partial differential equation (PDE)-based techniques have also attracted much attention recently, in which image details are preserved by adding an edge detection term. The single value decomposition method decomposes the column space of the observation matrix into a dominant and a subordinate part, revealing which of its subspaces can be attributed to the noise-free signal and which can be attributed to the noise. It is often assumed that these two subspaces are orthogonal to each other, which implies that signal and noise are independent. In a similar manner, a blind source separation or the Independent Component Analysis (ICA) decomposes signals assuming the following relation between components [1].

$$P(A \wedge B) = P(A) \cdot P(B) \tag{2}$$

where A and B are the independent components of signal. Given sufficiently large number of components, a few of the components can be pure noise components.

The wavelet transform can decompose the original signal into a smooth part (lowpass) and a detailed part (highpass). For most signals, energy is mainly distributed in the smooth subband, and energy in the detail subband is clustered to a few large wavelet coefficients, corresponding to the edge structure of the original signal. Donoho and his colleagues [2] pioneered a wavelet denoising scheme by using soft thresholding and hard thresholding. This approach, with the orthonormal wavelet, thresholds the wavelet transform coefficients within the detail subband. It is well known that Donoho's method offers the advantages of smoothness and adaptation. However, as Coifman and Donoho pointed out, this algorithm exhibits visual artefacts: Gibbs phenomena in the neighbourhood of discontinuities. However, the fundamental limitation of the orthonormal wavelet transform is the limited directional subband regardless of scale. In response, Starck and colleagues [3] proposed the Curvelet transform that extracts directional features in multi-scale using the Ridgelet. Later, other similar transforms followed such as the Contourlet which also consists of directional filter banks [4]. These transforms have proven effective in denoising. Compared with these new techniques, a much older technique with inherent directional feature recognition can achieve similar performance with a directional filter, that is the Multiresolution Fourier Transform (MFT). We present the ICA-MFT combined algorithm as well as the MFT with a multi-directional filter.

The paper is organised as follows. The next section starts with an introduction to ICA in the context of the denoising task. In section 3, denoising using the MFT with a Gaussian mask is explored and a combined approach is described in section 4. Section 5 introduces the MFT facilitated with a

multi-directional filter based on Radon analysis. Section 6 reports an experimental comparison with other state-of-the-art techniques [3].

2. ICA BASED APPROACH

Independent Component Analysis (ICA) has been frequently applied to computational neuroscience and the modelling of simple and complex cells in the human primary visual cortex (V1) [1, 5] which is responsible for directional feature identification. Recently, an ICA based denoising method has been developed by Hyvarinen and his colleagues [6]. The basic motivation behind this method is that the ICA components of many signals are often very sparse so one can remove noise in the ICA domain.

$$x = As \tag{3}$$

where x, A and s are the observed data, a linear mixing matrix and the source (latent) data respectively, which are independent and nongaussian. Unlike the wavelet-based denoising methods, an ICA based method uses a representation that is estimated solely by the statistical properties of the available data. The estimation of the ICA data model can be reduced to the search for uncorrelated directions in which the components are as nongaussian as possible and as a result the independent components have a sparse (supergaussian) distribution as possible. Hyvarinen developed a sparse code based noise shrinkage method similar to the wavelet shrinkage method [6].

$$x = As + v \tag{4}$$

where v is a Gaussian noise vector and x the noisy signal. An approximate version of s, $\hat{s}$ can be obtained, applying shrinkage on $\hat{A}^{-1}x$. The components, $\hat{A}^{-1}x$ (neuron from a physiological viewpoint) with small activities are assumed as noise and shrunken, retaining only a few components with large activities [6]. This, however differs in the following aspects. The shrinkage nonlinearities are estimated separately for each component, as opposed to a single fixed model in wavelet shrinkage. Also maximum-likelihood estimation is used in the nonlinearity estimation instead of minimax estimation. This method assumes training to estimate the orthogonal basis with noise-free data that has similar statistical properties. However, we attempted denoising without prior-training with empirically optimized settings.

3. MFT BASED APPROACH

The ability to capture the directional patterns which exist at various locations, scale and orientation is a recent research trend in the image processing community. For example, the curvelet [3] represents a curve as a superposition of functions of various lengths and width controlling orientation across various scales. Various implementations have been proposed such as the curvelet, contourlet, brushlet, etc. The same ability can be found in the Multi-resolution Fourier Transform (MFT) [7]. The MFT has been proposed as a combination of STFT and the wavelet. With the windowing function $g(t)$, the transform of a function $f \in L^2(R)$ at position u, frequency ξ and scale s is defined as below.

$$Mf(u,\xi,s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(t) g(s(t-u)) e^{-i\xi t} dt \tag{5}$$

The Laplacian pyramid is used to decompose the image according to frequency which shows isotropic behavior. At each scale, the windowed Fourier transform is applied with the same window. The high frequency directional patterns can be observed in the Fourier local spectrum. This is where an elliptical shape of Gaussian filtering is suggested in [8] as follows.

$$G(x) = \frac{1}{2\pi}\left(\frac{-x^T C^{-1} x}{2}\right) \tag{6}$$

where C is a covariance matrix, which can be obtained from the inertia tensor of the spectrum. The frequency window is effectively concentrated on a narrow oriented band. Experimental denoising results are presented in section 6.

4. ICA-MFT COMBINED APPROACH

The methods in the previous two sections approach the denoising problem in a different way. The ICA-based method uses purely statistical properties of the available data, adapting to the data. As a result, it requires larger computation even without a training process as the specified number of bases increases. The second approach takes advantage of the Fourier spectrum that exhibits a directional energy pattern. The nonlinearity of the Gaussian function formed by the inertia tensor from the spectrum is used for soft-thesholding. It, however, is not sufficient to represent a multi-directional pattern with a single Gaussian model. A Gaussian mixture model could be employed [9], but estimation of the mixture model increases the computational burden and nonlinear estimation can suffer from the local minima problem. We combine the two methods, effectively achieving a semi-adaptive wave packet basis, in a way that an algorithm can be computationally efficient and also adaptive to data, in two steps.

1. Perform ICA on each subband of the MFT with a limited number of components specified, for instance a third as many as the sufficient number of bases. This ratio needs to be determined adaptively and accurately so that no component with a meaningful pattern is discarded, where the discarded components are assumed to represent pure noise. This reduces the computation significantly.

2. Apply a 2D Gaussian filter obtained from the Fourier spectrum of the bases found in step.1. As most of the components are localised in orientation as well as frequency, a much narrower and oriented Gaussian model that fits the data can be obtained from the spectrum.

Fig. 1. MFT-Slice Denoising (noisy image, a multi-directional filter on spectrum and denoised image

5. MFT-SLICE APPROACH

The last approach benefits most from the fact that the ICA decomposes a block of a specific frequency band comprising multi-directional features into a set of bases localized in orientation (similar to ridgelet basis), so that the elliptical shape of the Gaussian filter obtained from the local spectrum fits the Fourier spectrum of the basis well. In this way, the inherent denoising capability of ICA discussed earlier combines with that of the MFT. Alternatively, however, we can replace ICA by providing a multi-directional filter using Fourier slice analysis. The Fourier slice analysis involves the computation of projection $r(\theta)$ for $0 \leq \theta < \pi$.

$$r(\theta) = c_i \cdot \Sigma\Sigma |F(x,y)| \delta(xcos\theta + ysin\theta) \qquad (7)$$

where c_i indicates a normalization constant at MFT scale i, and F is the Fourier spectrum. The shape S of the multi-directional filter consists of a set of points as below.

$$S = \{[r(\theta)cos\theta, r(\theta)sin\theta]^T\} \qquad (8)$$

The resulting contour represents the energy distribution of the significant directional pattern and is illustrated in Fig.1.The shape is used instead of the elliptical shape for the Gaussian filter for hard-thresholding, i.e. coefficients outside the shape are zeroed. The novelty of the approach is that instead of performing shrinkage on the transformed coefficients, we clean up the basis functions to allow a better reconstruction.

6. EXPERIMENTAL RESULTS

To illustrate the effectiveness of the proposed algorithms, comparative results with recent directional wavelet transforms are presented at various noise levels. The MFT was implemented on the Laplacian pyramid with 3 scale levels and a 50% overlapping cos^2 window of size 16×16. The MFT-Slice approach is facilitated with a window of size 32×32 for better directional analysis. We evaluated ICA, ICA-MFT and MFT-Slice against other directional basis transforms: Curvelet [3], Directional Cosine Transform (DDCT) [8], MFT--Gaussian Filtering (MFT) [8] and the Translation Invariant Wavelet Packet (TIWP). The test images `Lena` and `Jaguar` are shown in Fig.2 and Fig.3 with comparative results on a noisy image of SNR 15dB. `Lena` has a region of fur/feathers on her hat that creates multi-directional patterns, while the rest of the image is either homogeneous or directional. Clearly most of the methods with directional bases preserve the texture while TIWP produces quite a blurred image. The MFT,

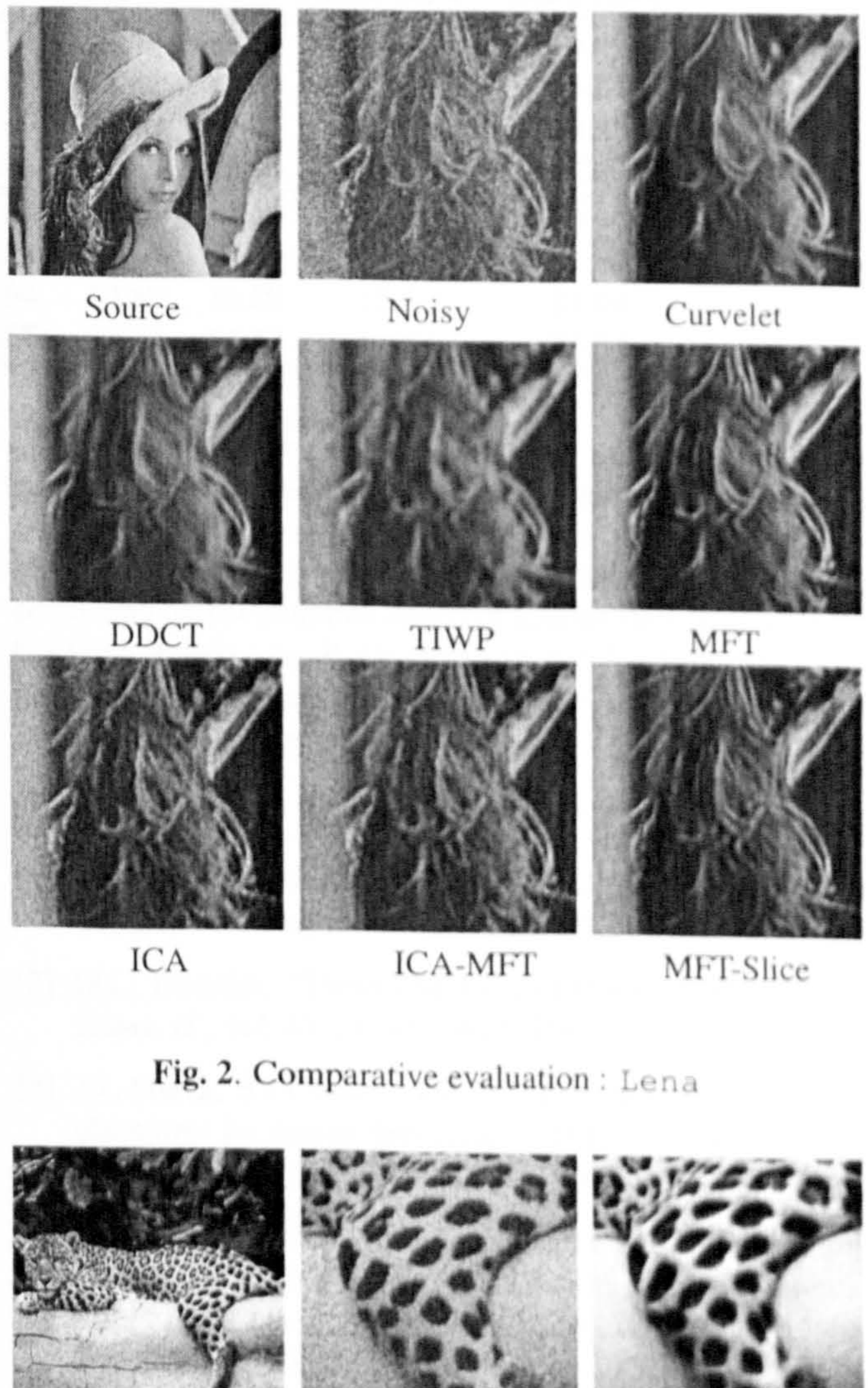


Fig. 2. Comparative evaluation : `Lena`

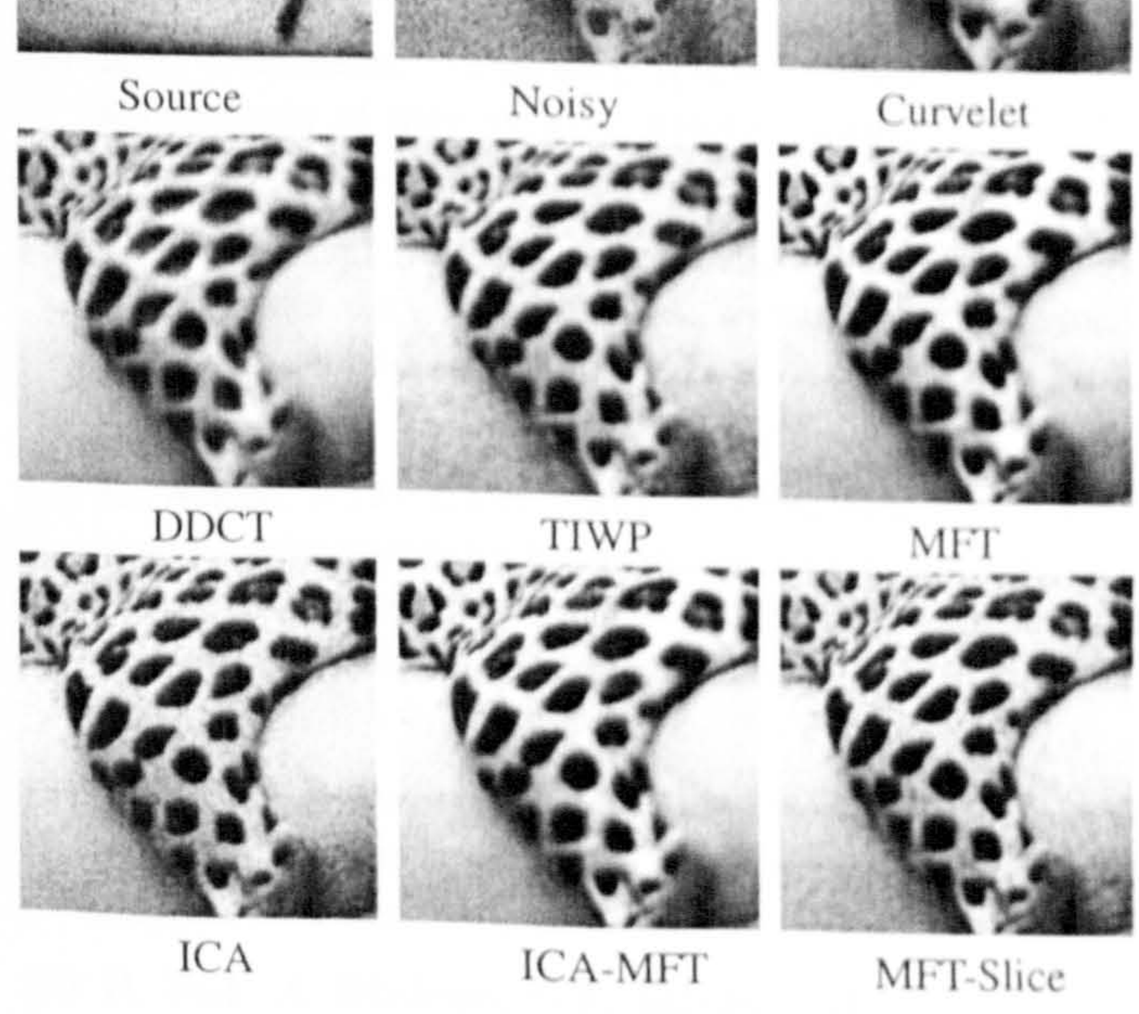


Fig. 3. Comparative evaluation : `Jaguar`

Table 1. SNR Results for different noise levels

	Noise(dB)	Curvelet	DDCT	TIWP	MFT	ICA	ICA-MFT	MFT-Slice
lena	5	17.55	17.44	18.91	18.97	18.74	**19.50**	19.43
	10	18.67	18.56	21.05	21.61	19.51	21.32	**21.82**
	15	19.40	19.37	23.49	24.42	21.48	22.65	**24.55**
	20	19.83	19.80	26.21	26.80	22.98	23.01	**27.67**
jaguar	5	12.19	12.18	15.41	**15.45**	12.24	12.66	13.36
	10	12.88	12.84	**18.11**	17.90	12.66	13.71	17.72
	15	13.24	13.26	20.94	19.89	13.28	15.23	**21.04**
	20	13.43	13.42	**23.85**	21.07	14.38	14.59	23.54

ICA-MFT and MFT-Slice show better results. Jaguar features a blob pattern on the Jaguar skin. The Curvelet and DDCT suffer from Gibb's phenomenon while both ICA-based approaches show rather blurred images. As the blob texture of the Jaguar requires various directional bases for reconstruction, most of the allocated bases of ICA are exhausted for orientation, and this leaves no basis to hold noise. The MFT and MFT-Slice generally show better results with much lower computational cost than the other methods. The MFT-Slice preserves the multi-directional pattern and removes noise better than the MFT as shown in both figures. The results of experiments at various noise levels are presented numerically in Table.1 (whole image). In Lena, MFT-Slice shows good SNR results and preserves edge structures as shown in Fig.2 and Fig.3. In Jaguar, despite TIWP providing good SNR results , the edge structures are quite blurred.

7. CONCLUSION

In this paper, we have briefly introduced two combined approaches to image denoising involving directional information. The results compare well with other proposed directional wavelet bases. It should be noted that the approach of combining different analysis methods is applicable not only to the problem of noise removal, but also it provides a new avenue to directional image analysis and numerical harmonic analysis as a whole. The MFT, originally proposed as a general image analysis tool has been around for more than a decade, with successful applications in feature extraction, motion estimation and texture analysis. By employing the idea of directional frequency filtering, the MFT finds a connection with the recent curvelet transform but with much lower complexity. The previously proposed MFT based anisotropic image denoising is limited fundamentally by its single-direction feature hypothesis, which assumes there is only one feature present in a local window. We tackled this limitation by two possible solutions. One is to use a source separation method, ICA, to decompose the signal into adaptive bases, in which the basis functions are expected to be a single directional component which can be dealt with effectively by the original MFT directional filter. The second is to introduce a multi-directional filter in the local Fourier spectrum by performing a Fourier slice integration of the magnitudes. This results in an adaptively shaped frequency mask which allows multiple components without prior knowledge of the number of components. The denoising experiments presented are intended as an example to show the power of the combined analysis although further improvement is possible by a frequency mask consisting on 1D Gaussian on every fourier slice instead of the hard-thresholding mask. On-going research is focused on employing the combined approach in other vision-related tasks such as segmentation and coding.

8. REFERENCES

[1] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley, 2001.

[2] D.L. Donoho, "Denoising by soft-thresholding," *IEEE Trans. IT.*, vol. 41, pp. 613–627, 1995.

[3] J.L Starck, E.J Candes, and D.L Donoho, "The curvelet transform for image denoising," *IEEE Trans. on Image Processing*, vol. 11, pp. 670–684, 2000.

[4] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. on Image Process.*, vol. 14, no. 12, pp. 2091–2106, 2005.

[5] B.A. Olshausen, "Learning sparse, overcomplete representations of time-varying natural images," in *IEEE ICIP.*, 2003, pp. 41–44.

[6] A. Hyvarinen, "Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation," *Neural Computation*, vol. 11, no. 7, pp. 1739–1768, 1999.

[7] R. Wilson, A.D Calway, and E.R.S Pearson, "A generalized wavelet transform for fourier analysis: themultiresolution fourier transform and its application to image and audiosignal analysis," *IEEE Trans. on Image Processing*, vol. 38, pp. 674–690, Mar. 1992.

[8] Z. Yao, N. Rajpoot, and R. Wilson, "Directional wavelet with fourier-type bases for image processing," in *Wavelet Anaysis and Applications*, 2005.

[9] H. Park, A. Bhalerao, G.R. Martin, and A.C. Yu, "An affine symmetric approach to natural image compression," in *Mobimedia*, 2006.

Bibliography

[1] T. Aach and D. Kunz. A lapped directional transform for spectral image analysis and its application to restoration and enhancement. *Signal Processing*, 80:2347–2364, 2000.

[2] G. Aharoni, R. Coifman A. Averbuch, and M. Israeli. Local cosine transform - a method for the reduction of the blocking effect in JPEG. *J. Math. Imag. Vision*, 3:7–38, 1993.

[3] N. Ahmed, T. Natarajan, and K. R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, pages 90–93, Jan. 1974.

[4] P. Auscher, G. Weiss, and M.V. Wickerhauser. Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets. In C. K. Chui, editor, *Wavelets: A Tutorial in Theory and Applications*, pages 237–256. Academic Press, San Diego, 1992.

[5] A. Averbuch, R. Coifman, D. Donoho, M. Israeli, and J. Walden. The pseudopolar FFT and its applications. Technical report, University of Yale, 1999. YaleU/DCS/RR-1178.

[6] A. Averbuch, R. Coifman, D.L. Donoho, and M. Israeli. Fast slant stack: A notion of Radon transform for data in a Cartesian grid which is rapidly computible, alge-

braically exact, geometrically faithful and invertible. to appear in SIAM Scientific Computing.

[7] D. Ballard. Generalising the Hough transform. *Pattern Recognition*, 13:111–122, 1981.

[8] D. Ballard and C. Brown. *Computer Vision*. Prentice Hall, New Jersy, 1982.

[9] A.J. Bell and T.J. Sejnowski. An information-maximization appraoch to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[10] A. Bhalerao and R. Wilson. A Fourier approach to 3D local feature estimation from volume data. In *Proceedings British Machine Vision Conference (BMVC)*, volume 2, pages 461–470, Sep. 2001.

[11] A. Bhalerao and R. Wilson. Affine invariant image segmentation. In *Proceedings British Machine Vision Conference (BMVC)*, 2004.

[12] A. Bhalerao and R. Wilson. Warplets: An image-dependent wavelet representation. In *Proceedings of IEEE ICIP*, volume 2, pages 490–493, Sept. 2005.

[13] T. Binford. Inferring surfaces from images. *Artificial Intelligence*, 17:205–244, 1981.

[14] H. Le Borgne and N. O'Connor. Ridgelet-based signatures for natural image classification. In *Proceedings 2nd Conference on Information Retrieval and Its Applications*, Grenoble, France, 2005.

[15] L. Boubchir and J.M Fadili. Multivariate statistical modeling of images with the curvelet transform. In *Proceedings of the Eighth International Symposium on Signal Processing and Its Applications*, volume 2, pages 747–750, August 2005.

[16] R. N. Bracewell. Numerical transforms. *Science*, 248:697–704, 1990.

[17] P.J. Burt and E.H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532–540, 1983.

[18] A. Calway. *The Multiresolution Fourier Transform: A General Purpose Tool for Image Analysis.* PhD thesis, University of Warwick, 1989.

[19] A.D. Calway. Image representation based on the affine symmetry group. In *Proceedings IEEE ICIP*, pages 189–192, 1996.

[20] A.D. Calway and R. Wilson. Curve extraction in images using a multiresolution framework. *CVGIP: Image Understanding*, 59(3):359–366, May 1994.

[21] P. Campisi, D. Kundur, and A. Neri. Robust digital watermarking in the ridgelet domain. *IEEE Signal Processing Letters*, 11(10):826–830, Oct. 2004.

[22] E. J. Candés, L. Demanet, D. Donoho, and L. Ying. Fast discrete curvelet transforms. Technical report, California Institute of Technology, July 2006.

[23] E. J. Candés and F. Guo. New multiscale transforms, minimum total variation synthesis: Applications to edge-preserving image reconstruction. *Signal Processing*, 82:1519–1543, 2002.

[24] E.J. Candés. *Ridgelets: Theory and applications.* PhD thesis, Dept. of Stats, Stanford Univ., Stanford, CA, 1998.

[25] E.J. Candés. Ridgelets and the representation of mutilated Sobolev functions. *SIAM J. Math. Anal.*, 33:197–218, 1999.

[26] E.J. Candés. Ridgelets: estimating with ridge function. *Ann. Statist.*, 31:1561–1599, 1999.

[27] E.J. Candés and D.L. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. Royal Soc. Lond. A.*, pages 2495–2509, 1999.

[28] E.J. Candés and D.L. Donoho. Curvelets - a suprisingly effective nonadaptive representation for objects with edges. In C. Rabut, A. Cohen, and L.L. Schumaker, editors, *Curves and Surfaces*, pages 105–120. Vanderbilt University Press, Nashville, TN, 2000.

[29] E.J. Candés and D.L. Donoho. New tight frames of curvelets and optimal representations of objects with C^2 smooth singularities. Technical report, Department of Statistics, Stanford University, 2002.

[30] J.F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, Nov. 1986.

[31] S. Carlsson. Sketch based coding of gray level images. *IEEE Transactions on Image Processing*, 15(1):57–83, Jan. 1988.

[32] S.G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transaction on Image Processing*, 9(9):1532–1546, September 2000.

[33] G.Y. Chen and B. Kégl. Image denoising with complex ridgelets. *Pattern Recognition*, 40:578–585, 2007.

[34] S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal of Scientific Computing*, 20(1):33-61, 1999.

[35] M. Choi, R.Y. Kim, and M.G. Kim. The curvelet transform for image fusion. In *Proceedings ISPRS Congress*, pages 59–64, Istanbul, July 2004.

[36] R. Claypoole and R. Baraniuk. A multiresolution wedgelet transform for image processing. In *Proceedings of SPIE: Wavelet Applications in Signal and Image Processing VIII*, volume 4119, pages 253–262, 2000.

[37] S. Clippingdale. *Multiresolution image modelling and estimation.* PhD thesis, Department of Computer Science, University of Warwick, 1988.

[38] A. Cohen, W. Dahmen, I. Daubechies, and R. DeVore. Tree approximation and optimal encoding. *Journal of Applied Computational Harmonic Analysis*, 11, 2001.

[39] R. Coifman and Y. Meyer. Remarques sur l'analyse de Fourier à fenêtre. *C.R. Acad. Sci. Paris Sér. I Math.*, I(312):259–261, 1991.

[40] R.R. Coifman and D.L. Donoho. Translation invariant denoising. In A. Antoine and G. Oppenheim, editors, *Wavelets in Statistics*, pages 125–150. Springer, New York, 1995.

[41] R.R. Coifman and Y. Meyer. Orthonormal wave packet bases. Technical report, Dept. of Mathematics, Yale University, New Haven, 1990.

[42] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. on Information Theory*, 38(2):713–718, 1992.

[43] A.L. Cunha, J. Zhou, and M.N. Do. The nonsubsampled contourlet transform:Theory, Design and Applications. *IEEE Transactions on Image Processing*, 15(10):3089–3101, Oct. 2006.

[44] I. Daubechies. *Ten lectures on wavelets.* SIAM, Philadelphia, 1992.

[45] A.R. Davies. *Image feature analysis using the Multiresolution Fourier Transform.* PhD thesis, University of Warwick, 1993.

[46] S.R. Deans. *The Radon Transform and Some of Its Applications.* Wiley, New York, 1983.

[47] S.R. Deans. *The Radon Transform and Some of Its Applications.* Krieger Publishing Company, revised edition edition, 1993.

[48] L. Demanet and L. Ying. Curvelets and wave atoms for mirror-extended images. In *Proceedings of SPIE: Conference on Wavelets XII*, San Diego, August 2007.

[49] L. Demanet and L. Ying. Wave atoms and sparsity of oscillatory patterns. *Applied Computational Harmonic Analysis*, 2007.

[50] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistic Society Ser. B*, 39, 1977.

[51] R. DeVore. Nonlinear approximation. *Acta Numer.*, 7:51–150, 1998.

[52] M. N. Do and Martin Vetterli. The finite ridgelet transform for image representation. *IEEE Trans. Image Processing*, 12(1):16–28, Jan. 2003.

[53] M.N. Do and M. Vetterli. Contourlet. In G.V. Welland, editor, *Beyond Wavelets.* Academic Press, 2003.

[54] M.N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *IEEE Transactions Image Processing*, 14(12):2091–2106, Dec. 2005.

[55] D. Donoho. Wedgelets: Nearly-minimax estimation of edges. *Ann. Stat.*, 27:353–382, 1999.

[56] D. Donoho and X. Huo. *Multiscale and Multiresolution Methods*, chapter Beamlets and multiscale image analysis. Lecture Notes in Computational Science and Engineering. Springer, 2001.

[57] D.L. Donoho. Orthonormal ridgelets and linear singularities. *SIAM Journal on Mathematical Analysis*, 31(5):1062–1099, 2000.

[58] D.L. Donoho. Ridge functions and orthonormal ridgelets. *Journal of Approximation Theory*, 111(2):143–179, August 2001.

[59] D.L. Donoho and I.M. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[60] D.L. Donoho and I.M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, December 1995.

[61] M. Dragotti and M. Vetterli. Wavelet footprints: Theory, algorithm and applications. *IEEE Transactions on Signal Processing*, 51(5):1306–1323, May 2003.

[62] R.O. Duda and P.E. Hart. Use of the Hough transform to detect lines and curves in pictures. *Communication ACM*, 15:11–15, Jan 1972.

[63] G. Easley, W. Lim, and D. Labate. Sparse directional image representations using the discrete shearlet transform. Submitted to Elsevier, 2006.

[64] M. Elad. Why simple shrinkage is still relevant for redundant representations? *IEEE Transactions on Information Theory*, 52(12):5559–5569, Dec. 2006.

[65] J. Elder. Are edges incomplete? *International Journal Computer Vision*, 34(2/3):97–122, 1999.

[66] R. Eslami and H. Radha. The contourlet transform for image de-noising using cycle spinning. In *Proceedings of the 37th Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1982– 1986, Nov. 2003.

[67] R. Eslami and H. Radha. Wavelet-based contourlet transform and its application to image coding. In *Proceedings IEEE ICIP*, Singapore, Oct. 2004.

[68] R. Eslami and H. Radha. Wavelet-based contourlet packet image coding. In *Proceedings Conference on Information Science and Systems*, The Johns Hopkins University, March 2005.

[69] A. Federico and G. Kaufmann. Denoising in digital speckle pattern interferometry using wave atoms. *Optics Letters*, 32(10):1232–1234, 2007.

[70] A.G. Flesia, H. Hel-Or, A. Averbuch, E.J. Candés, R.R. Coifman, and D.L. Donoho. Digital implementation of ridgelet packets. In G. Welland, editor, *Beyond Wavelets*, pages 31–60. Academic Press, Sep. 2003.

[71] K. Fourmont. Non-equispaced fast Fourier transforms with applications to tomography. *Journal of Fourier Analysis and Applications*, 9(5):431–450, 2003.

[72] H. Freeman. On the encoding of arbitrary geometric configuration. *IRE Transactions on Electronic Computing*, 10:260–268, 1961.

[73] D. Gabor. Theory of communication. *Journal of IEE*, 93:429–457, 1946.

[74] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[75] G.H. Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8:155–173, 1978.

[76] K. Guo, G. Kutyniok, and D. Labate. Sparse multidimensional representations using anisotropic dilation and shear operators. In *Wavelets and Splines*, pages 189–201, Athens, 2005.

[77] B. Horn. The Binford-Horn linefinder. Technical Report Memo 285, MIT AI Lab, 1971.

[78] T.I. Hsu. *Texture Analysis and Synthesis using the Multiresolution Fourier Transform.* PhD thesis, University of Warwick, Sept. 1994.

[79] T.I. Hsu and R.G. Wilson. A two-component model of texture for analysis and synthesis. *IEEE Transactions on Image Processing*, 7:1466–1476, 1998.

[80] D.H. Hubel and T.N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, pages 106–154, 1962.

[81] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Wiley, New York, 2001.

[82] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.

[83] A. E. Jacquin. Fractal image coding: A review. *Proceedings of the IEEE*, 81(10):1451–1465, Oct. 1993.

[84] G. Jiang, M. Yu, W. Yi, F. Li, and Y. Kim. A new denoising method with contourlet transform. *Lecture notes in control and information sciences*, 345:626-630, 2006.

[85] B. Julesz. Textons, the elements of texture perception and their interactions. *Naturr*, 290:91–97, 1981.

[86] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35–45, 1960.

[87] N.G. Kingsbury. Image processing with complex wavelets. *Phil. Trans. Royal Society London*, A(357):2543–2560, Sept. 1999.

[88] H. E. Knutsson, R. Wilson, and G. H. Granlund. Anisotropic nonstationary image estimation and its applications: Part I-restoration of noisy images. *IEEE Transactions on Communications*, 31(3):388–397, 1983.

[89] H. Krim, D. Tucker, S. Mallat, and D. Donoho. On denoising and best signal representation. *IEEE Transactions on Information Theory*, 45(7):2225–2238, Nov. 1999.

[90] D. Labate, W-Q. Lim, G. Kutyniok, and G. Weiss. Sparse multidimensional representation using shearlets. In *Proceedings of SPIE: Wavelets XI*, pages 254–262, Bellingham, 2005.

[91] J.S. Lim and A.V. Oppenheim. Enhancement and bandwidth compression of noisy speech. *Proceedings of IEEE*, 67(12):1586–1604, 1979.

[92] Y. Lu and M.N. Do. CRISP-contourlet: a critically sampled directional multiresolution image representation. In *Proceedings SPIE conference on Wavelet Applications in Signal and Image Processing*, San Diego, Aug. 2003.

[93] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:674–693, July 1989.

[94] S. Mallat. Geometrical grouplets. *Journal of Applied and Computational Harmonic Analysis*, 2006. submitted.

[95] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[96] S. Mallat and S.S. Zhong. Wavelet transform maxima and multiscale edges. In M.B. Rskai et al., editor, *Wavelets and their applications*. Jones and Bartlett, 1992.

[97] H.S. Malvar. Lapped transforms for efficient transform/subband coding. *IEEE Trans. Acoust. Speech Signal Processing*, 38:969–978, 1990.

[98] D. Marr. *Vision.* W.H. Freeman and Company, 1982.

[99] B. Matalon, M. Elad, and M. Zibulevsky. Improved denoising of images using modelling of a redundant contourlet transform. In *Proceedings of the SPIE*, volume 5914, pages 587–598, Jan. 2005.

[100] F. Matúš and J. Flusser. Image representation via a finite Radon transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):996–1006, Oct. 1993.

[101] P.R. Meulemans. *Hierarchical Image Sequence Analysis and Segmentation.* PhD thesis, University of Warwick, United Kingdom, May 2001.

[102] F. Meyer, A. Averbuch, and R. Coifman. Multilayered image representation: Application to image compression. *IEEE Transactions on Image Processing*, 11:1072–1080, 2002.

[103] F.G. Meyer. Image compression with adaptive local cosines: A comparative study. *IEEE Trans. Image Processing*, 11(6):616–629, June 2002.

[104] F.G. Meyer, A.Z. Averbuch, and J-O. Strömberg. Fast adaptive wavelet packet image compression. *IEEE Trans. on Image Processing*, 9(5):792–800, 2000.

[105] F.G. Meyer and R.R. Coifman. Brushlets: a tool for directional image analysis and image compression. *Applied and Computational Harmonic Analysis*, pages 147–187, 1997.

[106] F.G. Meyer and R.R. Coifman. Brushlets: Steerable Wavelet Packets. In J. Stoeckler and G.V. Welland, editors, *Beyond Wavelets*, pages 1–25. Academic Press Inc., 2001.

[107] Y. Meyer. *Oscillating patterns in image processing and non linear evolution equations.* University Lecture Series. AMS, 2002.

[108] E. Montiel, A. S. Aguado, and M. S. Nixon. Improving the Hough transform gathering process fo affine transformations. *Pattern Recognition Letters*, 22:959-969, 2001.

[109] M.S. Nixon. Improving an extended Hough transform. *Signal Processing*, 19:321-335, 1990.

[110] B.A. Olshausen and D.J. Field. Emergence of simple-cell receptive field properities by learning a sparse code for natural images. *Nature*, 381:607-609, 1996.

[111] B.A. Olshausen and D.J. Field. Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vision Research*, 37:3311-3325, 1997.

[112] A. Papoulis. *Signal Analysis.* McGraw-Hill, 1984.

[113] H. Park, G. Martin, and Z. Yao. Image denoising with directional bases. In *Proceedings IEEE ICIP*, volume 1, pages 301-304, San Antonio, Sept 2007.

[114] E. Le Pennec and S. Mallat. Image compression with geometrical wavelets. In *presented at IEEE ICIP*, Vancouver, BC, Canada, Sep. 2000.

[115] E. Le Pennec and S. Mallat. Bandelet image approximation and compression. *SIAM Journal of Multiscale Modeling and Simulation*, 4(3):992-1039, 2005.

[116] E. Le Pennec and S. Mallat. Sparse geometric image representations with bandelets. *IEEE Transactions on Image Processing*, 14(4):423-438, April 2005.

[117] D.B. Percival and A.T. Walden. *Wavelet Methods for Time Series Analysis.* Cambridge University Press, Cambridge, 2000.

[118] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629-639, July 1990.

[119] G. Peyré and S. Mallat. Discrete bandelets with geometric orthogonal filters. In *Proceedings IEEE ICIP*, Genova, Italy, 2005. submitted.

[120] G. Peyré and S. Mallat. Orthogonal bandlet bases for geometric images approximation. Preprint Ceremade, Jan. 2007.

[121] G. Peyré and S. Mallat. A review of bandlet methods for geometrical image representation. *Numerical Algorithms*, 44(3):205–234, Mar. 2007.

[122] G. Peyré, E. Le Pennec, C. Dossal, and S. Mallat. Geometrical image estimation with orthogonal bandlet bases. In *Proceedings of SPIE Wavelet XII*, Aug. 2007.

[123] A. Pinkus. Approximating by ridge functions. *Surface Fitting and Multiresolution Methods*, pages 279–292, 1997.

[124] D.D. Po and M.N. Do. Directional multiscale modeling of images using the contourlet transform. *IEEE Transactions on Image Processing*, 15(6):1610–1620, June 2003.

[125] P. Pongpiyapaiboon. Development of efficient algorithms for geometrical representation based on arclet decomposition. Master's thesis, Technische Universität München, 2005.

[126] I. Popovici and W.D. Withers. Curve parametrization by moments. Private Communication.

[127] I. Popovici and W.D. Withers. Custom-built moments for edge location. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):637-642, April 2006.

[128] I. Popovici and W.D. Withers. Locating edges and removing ringing artifacts in JPEG images by frequency-domain analysis. *IEEE Transactions on Image Processing*, 16(5):1470–1474, May 2007.

[129] J. Portilla, V. Strela, J. Wainwright, and E.P. Simoncelli. Image denoising using scale mixture of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, Nov. 2003.

[130] J. Princen, J. Illingworth, and J. Kittler. A hierarchical approach to line extraction based on the Hough transform. *Computer Vision, Graphics, and Image Processing*, 52(1):57–77, Oct. 1990.

[131] L. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings IEEE*, 77:257–285, Feb. 1989.

[132] N. Rajpoot, R. Wilson, and Z Yao. Planelets: A new analysis tool for planar feature extraction. In *Proceedings 5th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'2004)*, Lisbon, Portugal, April 2004.

[133] N. Rajpoot, Z. Yao, and R. Wilson. Adaptive wavelet restoration of noisy video sequences. In *Proceedings IEEE ICIP 2004*, Singapore, 2004.

[134] K. Ramchandran and M. Vetterli. Best wavelet packet bases in a rate distortion sense. *IEEE Trans. on Image Processing*, 2(2):160–175, April 1993.

[135] J. Romberg, M. Wakin, H. Choi, and R. Baraniuk. A geometric hidden Markov tree wavelet model. In *Proceedings of SPIE: Wavelets: Applications in Signal and Image Processing X*, pages 80–86, Nov. 2003.

[136] A. Said and W. Pearlman. A new, fast, and efficient image codec based on set partitioing in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–250, June 1996.

[137] J. Shapiro. Embedded image coding using zerotree of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, Dec 1993.

[138] B. Shen and I.K. Sethi. Direct feature extraction from compressed images. In *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Database IV*, pages 404–414, 1996.

[139] Y. Shkolnisky and A. Averbuch. 3D Fourier based discrete Radon transform. *Applied and Computational Harmonic Analysis*, 15:33–69, 2003.

[140] R. Shukla, P.L. Dragotti, M.N. Do, and M. Vetterli. Rate-distortion optimized tree structured compression algorithms. *IEEE Transactions on Image Processing*, to be published.

[141] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D.J.Heeger. Shiftable multiscale transforms. *IEEE Transactions on Information Theory*, 38(2):587–607, 1992.

[142] B.C. Song and J.B. Ra. Fast edge map extraction from MPEG compressed video data for video sharing. In *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Database VII*, pages 710–721, Jan. 1999.

[143] J. Starck, E.J. Candés, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, June 2002.

[144] J. Starck, D.L. Donoho, and E.J. Candés. Very high quality image restoration by combining wavelets and curvelets. In *Proceedings of SPIE Wavelets: Applications in Signal and Image Processing IX*, volume 4478, pages 9–19, 2001.

[145] J. Starck, M. Elad, and D.L. Donoho. Image decomposition via the combination of sparse representations and a variational approach. *IEEE Transactions on Image Processing*, 14(10):1570–1582, Oct. 2005.

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

[147] J.L. Starck, F. Murtagh, E.J. Candés, and D.L. Donoho. Gray and color image contrast enhancement by the curvelet transform. *IEEE Transactions on Image Processing*, 12(6):706–717, June 2003.

[148] C.M. Stein. Estimation of the mean of a multivariate normal distribution. *Ann. Statist.*, 9(6):1135–1151, 1981.

[149] S. Tan and L. Jiao. Ridgelet bi-frame. *Applied Computational Harmonic Analysis*, 20:391–402, 2006.

[150] P. Vary. Noise suppression by spectral magnitude estimation. *Signal Processing*, 8(4):387–400, 1986.

[151] L. Vese and S. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. *Journal of Scientific Computing*, 19:553–577, 2003.

[152] L. Villemoes. Wavelet packets with uniform time-frequency localization. *Comptes-Rendus Mathematique*, 335(10):793–796, 2002.

[153] M. Wakin, J. Romberg, H. Choi, and R. Baraniuk. Rate-distortion optimized image compression using wedgelets. In *Proceedings IEEE ICIP*, volume 3, pages 237–240, 2002.

[154] G.K. Wallace. The JPEG still picture compression standard. *Commun. Assoc. Comput. Mach.*, 34:30–44, April 1991.

[155] L. Wang, A. Bhalerao, and R. Wilson. Robust modelling of local image structures and its application to medical imagery. In *Proceedings of ICPR*, volume 3, pages 534–537, 2004.

[156] L. Wang, A. Bhalerao, and R. Wilson. Analysis of retinal vasculature using a multiresolution hermite model. *IEEE Transactions on Medical Imaging*, 26(2):137–152, Feb. 2007.

[157] N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series.* Wiley, New York, 1949.

[158] R. Wilson, A.D. Calway, and E.R.S. Pearson. A generalized wavelet transform for Fourier analysis: the multiresolution Fourier transform and its application to image and audio signal analysis. *IEEE Trans. on Information Theory*, 38:674–690, 1992.

[159] R. Wilson and G. H. Granlund. The uncertainty principle in image processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(11):758–767, 1984.

[160] R. Wilson and C.-T. Li. A class of discrete multiresolution random fields and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):42–56, January 2003.

[161] R. Wilson and M. Spann. *Image Segmentation and Uncertainty.* Wiley, New York, 1996.

[162] X. Xue and X. Wu. Image compression based on multi-scale edge compensation. In *Proceedings IEEE ICIP*, volume 3, pages 560–564, 1999.

[163] Z. Yao and N. Rajpoot. Radon/ridgelet signature for image authentication. In *Proceedings IEEE ICIP 2004*, Singapore, 2004.

[164] Z. Yao and N. Rajpoot. Image denoising using multiscale directional cosine bases. In *Proceedings IEEE ICIP 2005*, Genova, Italy, Sep. 2005.

[165] Z. Yao, N. Rajpoot, and R. Wilson. Directional wavelet analysis with Fourier-type bases for image processing. In T. Qian, M. I. Vai, and Y.S. Xu, editors, *Wavelet Analysis and Applications*, Applied and Numerical Harmonic Analysis, pages 123–142. Springer, 2006.

[166] L. Ying, L. Demanet, and E.J. Candés. 3D discrete curvelet transform. In *Proceedings SPIE Wavelet XI*, volume 5914, 2005.