Image Feature Analysis using the Multiresolution Fourier Transform

Andrew R. Davies B.Eng.

Department of Computer Science

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

August 1993

Image Feature Analysis using the Multiresolution Fourier Transform

Andrew R. Davies B.Eng.

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

August 1993

Summary

The problem of identifying boundary contours or line structures is widely recognised as an important component in many applications of image analysis and computer vision. Typical solutions to the problem employ some form of edge detection followed by line following or, more commonly in recent years, Hough transforms. Because of the processing requirements of such methods and to try to improve the robustness of the algorithms, a number of authors have explored the use of multiresolution approaches to the problem. Non-parametric, iterative approaches such as relaxation labelling and "Snakes" have also been used.

This thesis presents a boundary detection algorithm based on a multiresolution image representation, the *Multiresolution Fourier Transform* (MFT), which represents an image over a range of spatial/spatial-frequency resolutions. A quadtree based image model is described in which each leaf is a region which can be modelled using one of a set of feature classes. Consideration is given to using linear and circular arc features for this modelling, and frequency domain models are developed for them.

A general model based decision process is presented and shown to be applicable to detecting local image features, selecting the most appropriate scale for modelling each region of the image and linking the local features into the region boundary structures of the image. The use of a consistent inference process for all of the sub-tasks used in the boundary detection represents a significant improvement over the ad-hoc assemblies of estimation and detection that have been common in previous work. Although the process is applied using a restricted set of local features, the framework presented allows for expansion of the number of boundary feature models and the possible inclusion of models of region properties. Results are presented demonstrating the effective application of these procedures to a number of synthetic and natural images.

Key Words:
Image Analysis, Multiresolution, Feature Modelling, Fourier Methods

Contents

1 Introduction **1**
- 1.1 Image Analysis . . . 1
- 1.2 Uncertainty . . . 3
- 1.3 Detecting Image Boundaries . . . 5
- 1.4 Thesis Outline . . . 12

2 Towards the MFT **15**
- 2.1 Scale in Image Analysis . . . 15
- 2.2 Phase-Space Representations . . . 17
  - 2.2.1 The Windowed Fourier Transform . . . 18
  - 2.2.2 The Wavelet Transform . . . 22
- 2.3 The Multiresolution Fourier Transform . . . 26
  - 2.3.1 What is the MFT? . . . 26
  - 2.3.2 Choice of Window Function for the MFT . . . 31
  - 2.3.3 Multiresolution Image Modelling using the MFT . . . 48
- 2.4 Examples . . . 53

3 Local Image Features **59**
- 3.1 Frequency Domain Feature Modelling . . . 59
  - 3.1.1 Linear Features . . . 60
  - 3.1.2 Arc Features . . . 64
- 3.2 Model Based Decision Making . . . 67
- 3.3 Feature Detection . . . 70
  - 3.3.1 Linear Features . . . 74
  - 3.3.2 Arc Features . . . 88
- 3.4 Summary . . . 92

4 Feature Detection: Implementation **93**
- 4.1 Implementation . . . 93
- 4.2 Estimating Linear Feature Parameters . . . 95
  - 4.2.1 Parameter estimation . . . 95
  - 4.2.2 Multiple Features . . . 99
  - 4.2.3 Feature Extent . . . 100

4.3 Estimating Arc Feature Parameters 104
4.3.1 Parameter Estimation 104
4.3.2 Feature Extent 108
4.4 Computational Requirements 111
4.5 Experimental Results 115

5 Segmentation and Linking **122**
5.1 Introduction 122
5.2 Segmentation Algorithm 123
5.3 Scale Consistency 128
5.4 Linking Algorithm 132
5.5 "Missed" Features 136
5.6 Discrete Spectrum Transformation 140
5.7 Spectrum Correlation 145
5.8 Computational Requirements 148
5.9 Experimental Results 150

6 Conclusions **157**
6.1 Thesis Summary 157
6.2 Overview of Boundary Detection Algorithm 168
6.3 Limitations and Further Work 170
6.4 Concluding Remarks 173

Appendix: Conference Paper **175**

Bibliography **180**

List of Figures

1.1 A simple $\sigma = 2$ pyramid 4
1.2 Modelling with single linear feature vs. modelling with multiple features or circular arcs 11

2.1 Local spectrum interpretation of WFT 19
2.2 Filter bank interpretation of WFT 20
2.3 WFT tessellation of time/frequency plane 21
2.4 WT tessellation of time/frequency plane 24
2.5 MFT tessellation of time/frequency plane 28
2.6 Spatial and Frequency sampling points 29
2.7 Non-relaxed Window: Frequency Domain 36
2.8 Non-relaxed Window: Spatial Domain 37
2.9 Non-relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB. 38
2.10 Non-relaxed Window: Space/Frequency Plot 39
2.11 Frequency Relaxed Window: Frequency Domain 40
2.12 Frequency Relaxed Window: Spatial Domain 41
2.13 Frequency Relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB. 42
2.14 Frequency Relaxed Window:Space/Frequency Plot 43
2.15 Spatial/Frequency Relaxed Window: Frequency Domain 44
2.16 Spatial/Frequency Relaxed Window: Spatial Domain 45
2.17 Spatial/Frequency Relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB. 46
2.18 Spatial/Frequency Relaxed Window:Space/Frequency Plot 47
2.19 Parent - Child Relationship for Regions 48
2.20 MFT structure - 2-dimensional 49
2.21 Image Model: Grey areas represent regions that cannot be modelled at a given level 51
2.22 Shapes: Original 256×256 image 56
2.23 Shapes: MFT Level 2 56
2.24 Shapes: MFT Level 3 56
2.25 Shapes: MFT Level 4 56

2.26 Lena: Original 256 × 256 image 57
2.27 Lena: MFT Level 3 57
2.28 Lena: MFT Level 4 57
2.29 Vineleaf: Original 256 × 256 image 58
2.30 Vineleaf: MFT Level 3 58
2.31 Vineleaf: MFT Level 4 58

3.1 Linear feature centroid 63
3.2 Arc 65
3.3 Arc projection onto t-axis 66
3.4 Estimation, Synthesis and Decision 70
3.5 Feature Detection/Estimation Process 71
3.6 Feature Detection/Estimation Process 73
3.7 Adjacent frequency displacement vectors 75
3.8 Shaded area indicates Θ_{θ_0} 81

4.1 Adjacent Regions 94
4.2 Calculation of end points 102
4.3 Two possible arcs with the same energy spectrum 106
4.4 Calculation of arc end points 108
4.5 Shapes: Level 2 116
4.6 Shapes: Level 2 116
4.7 Shapes: Feature detection - Level 2 119
4.8 Shapes: Feature detection - Level 3 119
4.9 Shapes: Feature detection - Level 4 119
4.10 Lena: Feature detection - Level 3 120
4.11 Lena: Feature detection - Level 4 120
4.12 Lena: Feature detection - Level 5 120
4.13 Vine leaf: Feature detection - Level 3 121
4.14 Vine leaf: Feature detection - Level 4 121
4.15 Vine leaf: Feature detection - Level 5 121

5.1 Segmentation Algorithm 125
5.2 Feature missed at large scale 126
5.3 Feature in parent and child 129
5.4 Arc feature in parent and child 131
5.5 Linking Primitive Feature Segments 133
5.6 Detecting "Missed" Features 137
5.7 Bilinear Interpolation 142
5.8 Sample points at levels n and $n+1$ 144
5.9 Spectrum Spreading 146
5.10 Coefficient Distance 147
5.11 Shapes: Feature detection - over levels 2 - 4 154

5.12 Shapes: Feature detection - with scale consistency 154
5.13 Shapes: Feature linking 154
5.14 Vine leaf : Feature detection - over levels 3 - 5 155
5.15 Vine leaf: Feature linking 155
5.16 Lena: Feature detection - over levels 3 - 5 156
5.17 Lena: Feature linking 156
5.18 Lena: Feature linking - with backtracking for missed features. 156

6.1 Segmentation and Linking Algorithm 169

Acknowledgements

This work was funded by UK SERC, and conducted within the Image and Signal Processing Research Group in the Department of Computer Science at the University of Warwick.

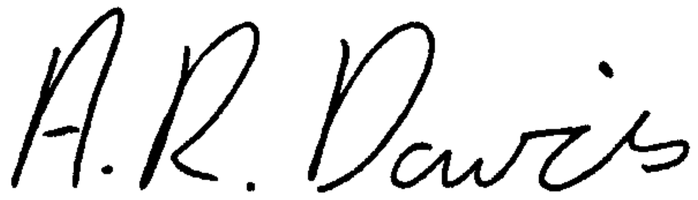
I would like to thank all the staff of the Computer Science Department, in particular my friends and colleagues, past and present, of the Image and Signal Processing Group at Warwick: Abhir Bhalerao, Andrew Calway, Simon Clippingdale, Tao-I. Hsu, Andy King, Wooi Boon Goh, Edward Pearson, Hugh Scott, Tim Shuttleworth, Martin Todd, June Wong and Horn-Zhang Yang. Their ideas and encouragement provided a stimulating environment in which to work, and I am indebted to them. I would also like to thank Jeff Smith for invaluable software support.

In particular my thanks go to my supervisor, Dr. Roland Wilson. This work would not have been possible without his profound expertise in the subject and the constant support which he has provided.

I would like to thank my parents and brother for their much valued encouragement throughout my education.

Declaration

I declare that, except where specifically 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.



A. R. Davies

This work is dedicated to my wife, Sarah.

Chapter 1

Introduction

1.1 Image Analysis

As noted by Marr [64], many writers, including Gibson [39], have been "misled by the apparent simplicity of the act of seeing". Our daily experience of using the human visual system (HVS) suggests that vision is a simple, almost trivial process. Because learning to see is a largely unconscious process, seeing appears to be easy. In fact, millions of year of evolution have produced a system that is so well developed and suited to its tasks that we are lead to underestimate its complexity.

Input to the HVS is provided by light from the environment being focused onto the retina at the back of eye by the lens, forming an image of the external world[46]. This continuous retinal image is then sampled by a dense array of photo-receptive cells, producing an array of 'intensities' that is passed onto the higher level processing of the HVS[81]. Vision may be considered as the transformation of this raw, iconic image representation into a description of the scene conveyed by the image. Such a description may act as an aid to navigation, defence or any number of human activities that require information about the immediate environment [49, 89].

Computer vision may be seen as the artificial equivalent of the HVS. The eye

is replaced by a camera and some sampling process to produce a discrete $2-d$ representation of the $3-d$ scene in front of the camera. The image is represented by an array of picture elements (*pixels*) that represent the intensity of the image at a lattice of regularly spaced sample points. The more densely the image is sampled, the more closely it represents the continuous image projected onto the focal plane of the camera. Increasing the resolution of the image, however, increases the amount of data that the system must deal with. A 512×512 grey scale image contains approximately $\frac{1}{4}$ million pixels. When multiple images are required, for example when examining pairs of stereo images [5] or image sequences, the data explosion becomes even more dramatic. The first stage of any computer vision process is, therefore, often to transform the image into a more useful, and less data intensive, representation. Such a representation will be in terms of some class of objects that are applicable to the task in hand, and represent the important features of the image while eliminating any noise or irrelevant detail.

The translation of an image to a higher level representation is a complex process and as such may not be possible in a single processing step. In the words of Marr [64] it is necessary to proceed "to the desirable via the possible". Starting with a representation that can be directly computed from the image, a second representation can be derived from the first that is closer to the description sought. A further processing step can then be applied to this representation to give yet another, and so on. For Marr this process transforms the image first to a primal sketch, then to a $2\frac{1}{2}$-d sketch and finally a 3-d model based representation. Each representation makes explicit certain information, while hiding or transforming other information in such a way that it cannot, if at all, be easily recovered. The sequence of transformations

must be carefully chosen, therefore, to ensure they are suited to the desired result.

1.2 Uncertainty

As noted by various authors [95, 97, 29, 93, 84] such a transformation gives a description in terms of "what" is "where" in an image, and therefore immediately runs up against the uncertainty principle: for a given image position $\vec{\xi}_i$ and its neighbourhood $\mathcal{N}_i$ is there an object of class C at $\vec{\xi}_i$? Intuitively the dilemma is clear. The larger the neighbourhood the more evidence is available on which to base the decision. If, for example, a noisy image is being segmented by grey level value, the grey level at a particular point can be estimated by local averaging. The best estimate can be made by averaging the grey levels over the largest possible area. Thus increasing the size of the neighbourhood increases the accuracy of the mean estimate. This brings with it problems, however, since using larger neighbourhoods leads to reduced spatial resolution. There is therefore a trade off between determining "what" and "where". The achievement of localisation in the class space is at the expense of localisation in the position space.

In order to try to 'cheat' the uncertainty principle, a number of authors have proposed analysis methods based upon multiple resolution representations. The image is represented over a range of scales, varying from the initial image, which has low feature space resolution, to a low spatial resolution representation that has a high resolution in the feature space. Information can then be combined over a range of scales to infer high resolution in both domains. The key to such an approach is the model that relates the different scales. In cases where such a model and a method of multiscale combination based upon this model can be applied, it may appear that

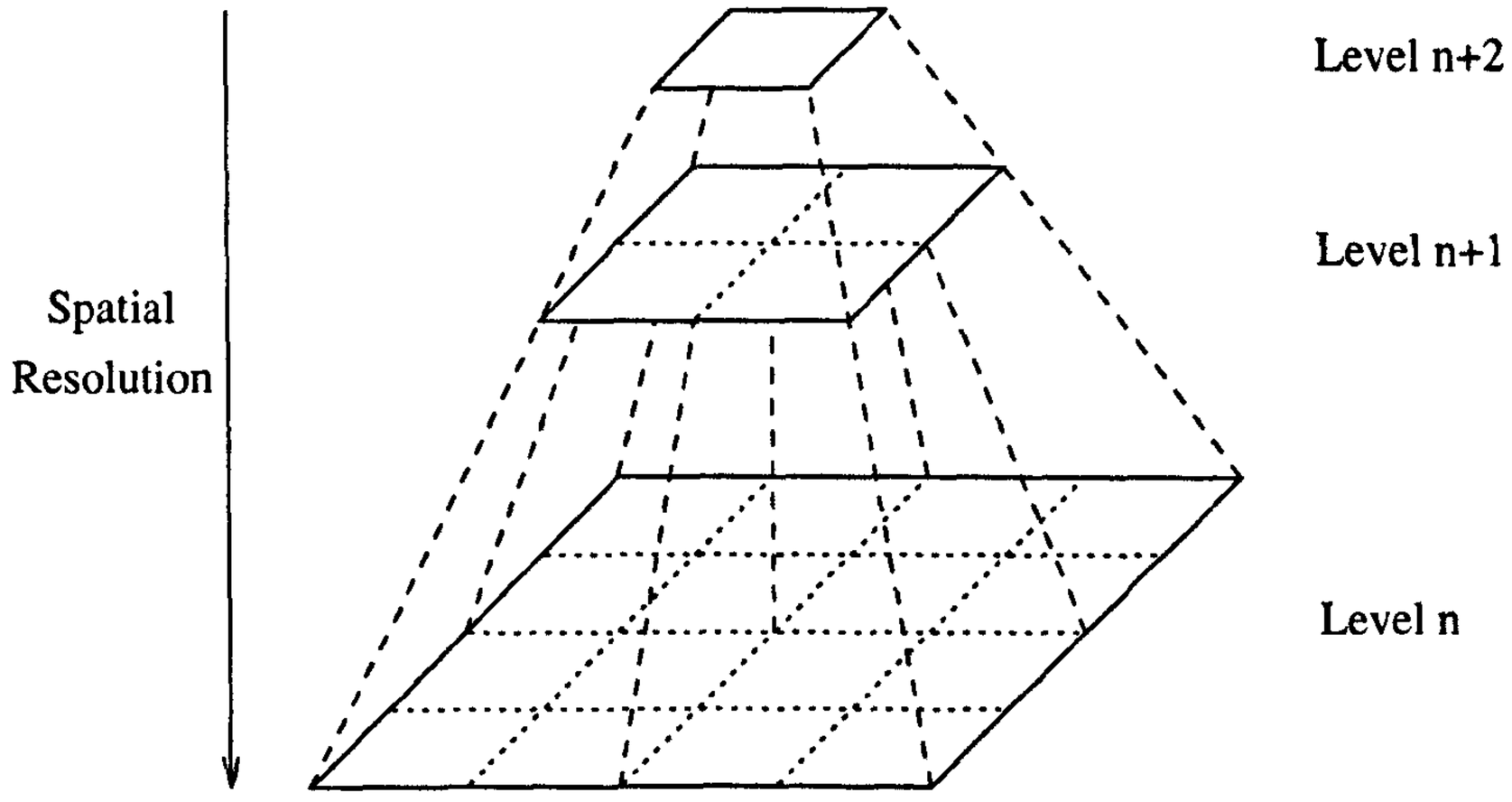


Figure 1.1: A simple $\sigma = 2$ pyramid

the uncertainty principle is "beaten". If the model does not fit the image, however, the combination over scale will fail to integrate the different image representations coherently. The successful adoption of such an approach therefore requires knowledge of where the model fits in order to be successful.

The simplest such representation is the pyramid [14, 15], where the resolution is reduced by a constant factor, σ, between one level and the next. To calculate a level from the one below it, each block of $2^k\sigma \times 2^k\sigma$ samples is replaced by its weighted average. Choosing a value of $k > 0$ allows these blocks to overlap. At each scale a region of the image is represented by its weighted average. A simple ($\sigma = 2, k = 0$) pyramid is illustrated by figure 1.1, where each sample at level $n + 1$ is simply the arithmetic mean of its four children at level n.

A related multiresolution representation is the grey level quadtree [86]. This is constructed by calculating the mean and variance of the entire image. If the variance is below some threshold value then the image can be approximated by the mean, if

not the image is split into four quadrants and the test reapplied to each in turn. This continues until the image has been segmented into a set of regions that model the image by their mean grey levels, within a margin of error defined by the variance threshold.

More complex multiresolution representations such as Wavelets[27, 28, 61] and the multiresolution Fourier transform[16, 90], provide more information for each region. The MFT represents the image as a stack of local spectrum estimates over a range of spatial resolutions. Each level is a complete and invertible representation of the image. Indeed each local spectrum provides a complete frequency domain representation of the corresponding image region - unlike the local average provided by the simple pyramid described above.

1.3 Detecting Image Boundaries

The aim of the present work is boundary detection. It is clear that boundaries in images give important information to the HVS, as we can often recognise an object from a line drawing of it. As noted by Marr and Hildreth [65], these boundaries may correspond to the edges of objects in an image or may be the result of varying illumination in the image, such as shadows, and their detection amounts to computing the "raw primal sketch" of the image [65]. The work of Campbell and Robson [19] suggests that there are cells in the early stages of the HVS that respond to the areas of the image plane containing high spatial frequencies, ie. the rapid changes in intensity that correspond to boundaries in the image. Additionally, the majority of cells that respond to line and edge stimulus are sensitive to orientation[81, 47, 11].

Traditional boundary detection is usually a two stage process. First, some form

of edge detection operator is applied to the image to detect points in the image of rapid intensity change. A large number of techniques for local edge detection have been developed during the past thirty years. The earliest, and most common, is the gradient operator as described by Roberts [80], which approximates the first partial derivatives in orthogonal directions. This is just one of a class of difference operators that have been developed (eg. Prewitt, Sobel and Canny[20]). An overview of these methods can be found in [40] or [51].Related to the edge gradient are edge templates such as those used by Kirsch[53]. These methods provide an indication at each point of whether the point is on an edge, as well as indication of the edge orientation at that point. Another closely related method is based upon the Laplacian[40, 4]. As noted in [4], however, this has two major disadvantages in that it does not give orientation information and, because it approximates a second derivative, it enhances any noise present in the image. Since edges correspond to high spatial frequencies it is clear that edge detection operations can also be performed in the frequency domain[54, 16, 61].

A general problem facing all methods of edge detection is that of noise, which is rapidly changing and therefore produces relatively large responses from differential operators. In an attempt to counter this, it is desirable to perform some smoothing on the image, but the smoothing should be such that, while it reduces the noise component of the image, it does not blur the edges significantly. To overcome the limitations of fixed scale smoothing, ie. uncertainty, a number of authors have proposed using smoothing filters at various scales (eg. Marr and Hildreth[65], Rosenfeld and Thurston[82] and Witkin[98]). A robust method of combining edge and orientation information across scale, developed by Wilson, Clippingdale and Bhalerao is described in [8, 9, 92]

A further consideration is that of regions with multiple orientations. Many of the edge detection processes described give an indication of the dominant orientation for each edge point. Such information is clearly useful for linking or other higher level processes based upon the edge detection, such as direction-adaptive filtering (eg. [56]). However at corners, or other junction points, edges in a number of orientations meet and so it is desirable to produce a set of the orientations associated with the point. A number of authors (including Freeman[34, 35], Andersson[1] and Perona[74]) have considered the use of *steerable* (or *controllable*) filters in order to test for edges over a set of different orientations, as well as proposing methods for detecting junction points. At points where there are two dominant orientations, Freeman considers the energy profiles along, and perpendicular to, these orientations to decide what sort of junction is present, eg. 'L','T' or 'X' junctions. Andersson has developed filters for detecting line endpoints, based upon dual quadrature filters[54].

The application of some edge detection operator(s) provides a representation of the image in terms of local edges and orientations. The edge elements, or *edgels*, that result from the local edge detection are simply a set of disjoint points that lie on the image boundaries. These points form the input to the next stage of processing in order to segment the edgels into "coherent one dimensional (*edge*) features"[4].

A number of methods have been applied to this task. These approaches include using similarity in the magnitude, gradient and position information provided by the edge detector to link edgels, such as graph theoretic approaches [4, 68], pyramid based grouping [26] and relaxation methods[58, 48]. The latter approach is exemplified by the work of Parent and Zucker [72], who argue that orientation and curvature constraints can be used to refine an initial coarse set of trace point estimates, us-

ing relaxation labelling [48], before synthesising the curve from the trace [101]. An approach similar to this, based upon salient contour analysis, is also considered by Freeman[34].

A more global approach is the Hough transform (HT) [33, 50]. Applying this to small image regions the edge data is transformed into a parameter array based on some parametric feature representation. Although originally used for detecting straight segments, these methods have been generalised to consider other parametric [4] or non-analytic [3] curves. A hierarchical version [77] has been developed which combines local and more global information, by combining line segments detected in small spatial regions into larger segments where possible. A problem encountered with the Hough transform is that of connectivity. The transform uses accumulator arrays to measure the colinearity of edge points. A number of short line segments, or texture elements with the same orientation may therefore be detected as a straight line, even though they are disconnected. A method of building a connectivity check into the HT has been considered in [100].

A similar hierarchical approach is used by Shneier [83], which is based upon using two hierarchical representations of edge information. The first is the edge pyramid, which is constructed by grouping together edgels into edge segments in small, overlapping 4×4 regions. Higher levels are calculated by combining these segments to form edge segments that represent larger regions of the image, along with an error term indicating how well they do so. This representation is similar to that of the Laplacian pyramid [14], but represents a region by using a linear feature, rather than a weighted average of the image intensity. Shneier's second representation is the edge quadtree, which is constructed in a similar manner to the grey level quadtree [86].

First, an attempt is made to model the image using a single edge. If this is unsuccessful the image is split into four quadrants and the modelling process is reapplied. This continues until each region has been successfully modelled.

Another approach to boundary detection is based upon active contour models or "Snakes"[52]. Snakes are energy minimising splines that 'lock onto' local features in an image. An energy functional is defined as the sum of three components : internal energy to ensure continuity and smoothness of the spline, image energy defining the image features onto which the snake should lock, and an external constraint term provide by an operator or higher level process. The image energy term may be based upon image intensity (line detection) or gradient (edge detection), and may include a scale space element allowing "minimisation by scale continuation" [98, 99, 52]. The problem with this approach is its reliance on some external process to guide it. Some outside agent, either a human operator or some other "intelligent" overseer must decide on the snake's starting point. Zucker's linking algorithm [101] attempts to overcome these problems by using local dynamic splines that initially minimise a local energy functional, but may also join together if they begin to overlap.

All of the above methods follow the processing framework suggested by Marr in that the image is transformed from its initial representation (a pixel array) to an array of edge points, which is then transformed, or segmented, into a set of line segments. The work of Calway [16, 18] combined both of these processing steps - edge detection and feature detection - into a single step. By first transforming the image into a set of local spectrum estimates over a range of scales, the MFT, a linear feature detector can be directly applied to a complete and invertible representation of any part of the image. The resulting feature estimate can then be used to produce a measure

of how well this feature models the image region concerned, and hence whether the feature fits the model. Since an image region may fit a model at some scales, but not at others, such a decision allows the selection of an appropriate scale at which to represent it.

The above methods are based on edges rather than regions. It should be noted, however, that there is an implicit link between boundaries of image regions and the regions themselves. Much work in the area of image region segmentation has included some form of boundary information, such as the Markov random field (MRF) approaches described in [38, 42]. A multiresolution approach to combining edge and region information has also been explored by Bhalerao and Wilson [9, 7]. It is therefore desirable that, although segmentation based upon regional properties is outside the scope of the current work, the framework adopted for the boundary detection should be of a form that does not exclude the possibility of the incorporation of region information.

The object of the current work is to extract from a grey level image a representation of the line and boundary structures within it. The approach used follows that of both Calway[16] and Shneier[83], in that the initial problem may be regarded as that of constructing some form of quadtree, where each leaf represents an area of the image in terms of some feature model. Shneier notes that quadtrees constructed for a closed boundary and for the region enclosed by that boundary have closely related shapes. Furthermore, it has been shown that a multiple scale representation related to that used for boundary detection in the current work can be successfully applied to the segmentation and analysis of feature regions based upon their textural properties[85]. This indicates that the processing framework used should be readily extendible to

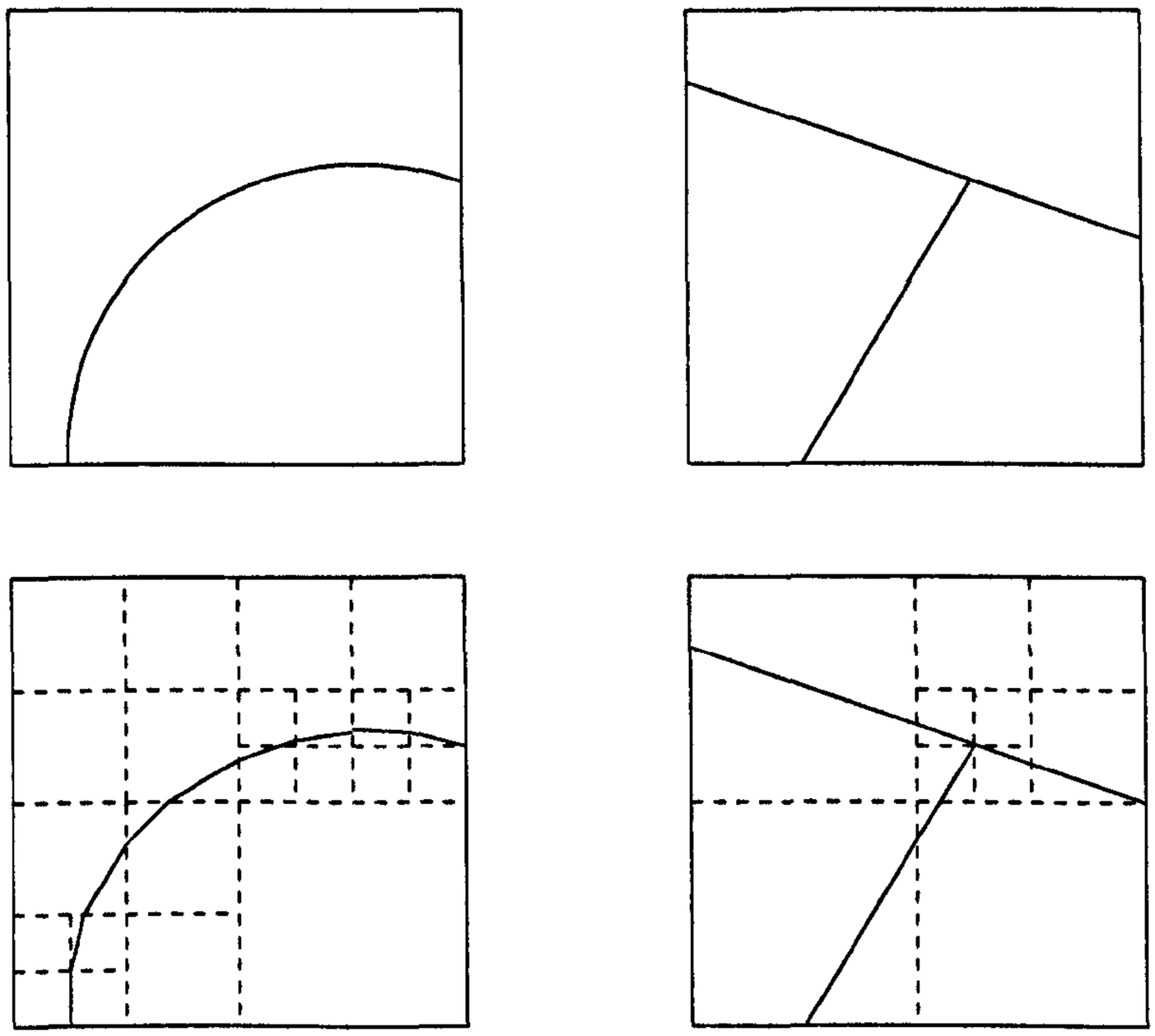


Figure 1.2: Modelling with single linear feature vs. modelling with multiple features or circular arcs

allow detection of both local (edge) features and regional (textural) features.

A limitation of Calway's work is the simplicity of the local feature model used - only regions containing a single linear feature are considered. This implies that regions with high curvature boundaries must be split into smaller regions, in which the segments can be approximated using linear features. As previously mentioned, it is desirable to perform the feature detection at a high feature space resolution in order to increase the accuracy of the parameter estimates. One way in which to increase the scale at which the region can be modelled is to use a feature model that fits curved boundaries. In the current work, this is done by including a model for segments of a circular arc. Figure 1.2 illustrates the difference in scale of feature detection that may result from including such a feature model.

A similar problem occurs for regions containing corners or junction points. Restricting the model to regions containing only a single feature forces corner regions to be modelled at a high spatial resolution, as shown in figure 1.2. To overcome this, the modelling can be extended to allow regions containing a number of linear features in different orientations. This extension parallels that used in edge detection (eg. Freeman [34]) to detect multiple orientations at a given image point.

One motivation for attempting to model an image region directly is that it is then possible to build consistency constraints into the local feature model, rather than apply an additional relaxation stage to the output of an edge detection process (cf. Parent and Zucker [72]). It is clear that if a region can be modelled by a linear feature the edge points within that region must have a constant orientation. Similarly if a circular arc can be used to model the edge in the region, the orientations at the edge points must be consistent, and the curvature constant, over the region.

By extending the range of models within the same general framework as Calway, it is intended to increase the power and flexibility of the methods, without sacrificing the advantages of speed and representational completeness of his work.

1.4 Thesis Outline

Chapter 2 considers the importance of representation in image analysis. Specifically it deals with the importance of representing an image at a scale, or range of scales, most suited to the required analysis. The trade-off between spatial resolution and some "property space" (or *phase-space*) is considered. Two representative phase-space representations are considered and the multiresolution Fourier transform (MFT) is presented. The MFT may be considered as a form of generalised pyramid,

each level representing the image at a different spatial resolution. However instead of representing each region by a single mean grey level[14], it is represented by an estimate of its local Fourier spectrum. Additionally each level completely represents the image and could therefore be used to reconstruct it. A multiresolution image model, representing an image as a set of disjoint regions over a range of scales, is presented.

The use of local spectrum estimates as a basis for region modelling is considered in chapter 3. A framework is described for detecting local image features and estimating their parameters. Models are derived for the spectra of regions containing one or more linear features (such as straight lines or edges) or a single arc feature. Parameter estimators are derived for these regions and a method of determining how well the models fit the actual image data, using correlations, is presented. Chapter 4 presents the implementation of these estimation and detection procedures using single levels of the MFT to provide a set of local spectrum estimates. Results are presented for several synthetic and natural test images to demonstrate the effectiveness of the estimators.

In chapter 5, the combination of features estimated over a range of scales is examined. An algorithm, based upon the image model described in chapter 2, is developed, for segmenting an image into regions that fit the local feature models described in chapters 3 and 4. The implementation of the algorithm using the MFT is considered and results of applying the segmentation algorithm to some test images are presented. This segmentation may be compared to the construction of an edge quadtree [83]. Each region is represented by a set of coefficients from the MFT, and as such the segmentation provides an invertible representation of the image[16].

In most real images, local features will be parts of some larger scale boundary structure. An algorithm is presented for linking the local features into larger, compound features by comparison of the local spectra representing them. A method for "back-tracking" to search for features that may have been missed by the feature detection process is also presented. This uses features that have been detected in neighbouring regions to suggest the existence and approximate position of the missing features.

The results presented at the end of chapter 5 demonstrate that the approach developed in this work has the potential to provide an effective method of detecting object boundaries within a general image analysis framework. However it is clear that before the potential of this approach can be fully realised a number of issues need further consideration. Some of these are mentioned in chapter 6, along with a synopsis of the work and discussion of future areas of development.

Chapter 2

Towards the MFT

2.1 Scale in Image Analysis

Much has been made in recent years of the usefulness of local processing in image analysis. The use of global image properties for analysis assumes that these give a good indication of what is going on in all parts of the image, whereas in many cases different image localities may have very different properties. By analysing regions in an image, the properties local to those regions can be determined or utilised in a large number of image processing tasks. Each locality can be treated in the way most appropriate to it - as opposed to using some global, compromise method which may not be right for any given region of the image.

In image restoration [51, 6], for example, knowledge of the signal structure can aid in the choice of filter that will reduce the noise content, while, as much as possible, leaving the signal intact - such as the derivation of Wiener filters [71, 70, 60] based on statistical signal/noise models. However, considering the image as a set of local regions [25], or sub-images, the restoration process can vary over the image. In flat, or slowly varying regions, some form of averaging would be sufficient to remove the noise, but in a region of a more rapid change, such as near an edge, such averaging

may cause blurring of perceptually important image features and so an alternative must be chosen.

The present work is concerned with the detection of local image features, from a number of different feature classes. To this end, it is desirable to represent these feature classes in a similar way, allowing their detection to take place within a coherent framework. Although the present work is concerned with edge and boundary detection it would be desirable to have a representation that is sufficiently expressive to include additional feature classes such as texture elements, allowing the detection process to be generalised whilst keeping the same basic framework.

Since the aim is to detect local features, it is necessary to consider the scale at which the analysis should take place. The image can be viewed through a set of analysis windows, each of which provides a view of just a small portion of the image. What size should these windows be? For any given analysis window there may be multiple features present and each of these could be from any of the feature classes. Choosing a fixed window size, or analysis scale, therefore, requires that the feature models and detection processes be sufficiently complex to deal with regions containing these multiple features.

Another way to deal with this is to allow the analysis window size to vary independently of the feature model. The scale and model can then be selected so that they give the best fit to the image data. Since any feature can in general exist at any scale, it is necessary that the representation provides for any compact region of the image to be represented without interference from any features outside of the region. This allows the image to be segmented into differently sized regions, each of which can be modelled using the available feature classes [16].

The rest of this chapter is concerned with explaining why, among possible representations, the multiresolution Fourier transform (MFT) provides an effective framework for tackling this problem.

2.2 Phase-Space Representations

Local analysis is conveniently performed using a *phase-space* image representation [27]. Such a representation conjoins some 'property coordinate' to the spatial coordinate giving an indication of both the local properties of each region of the image and of each region's location. Although there are many such representations, there are two main classes.

The first may be considered as the class of *space-frequency* representations. Both linear representations, which are closely related to the windowed Fourier transform (WFT) [76] and Gabor [37] transforms, and bilinear (quadratic) representations, eg. Wigner distributions [22, 23, 24] and Ambiguity Functions[70], have been examined by many authors and applied to numerous applications. A review of this class is given in [44]. For these representations, the conjoined co-ordinate is frequency - for each of a set of image neighbourhoods the representation provides an indication of the local frequency content. Although these representations provide local image information, the size of the neighbourhood over which this information is calculated is fixed and built into the representation. Thus while the representation is general, it does not allow the desired variation of analysis scale.

The second class are *scale-space* representations - with scale being the conjoined image co-ordinate. This includes the Wavelet Transform (WT) [27, 41, 61], and pyramid based representations, such as the Laplacian pyramid [14]. In both cases,

the image is represented as a coarse (low pass) image and a set of detail images that must be added to reconstruct the original image. These detail levels represent the 'features' in the image at increasingly fine scales. The features detected at a given scale are determined by the decomposition used. For the Laplacian pyramid [14] the features are spatial points of rapid change, such as edges or lines, that do not occur at coarser levels. In Mallat's wavelet decomposition, this is expanded to take account of the orientation of these features [61]. Although the image is represented at a range of scales, the feature class is built into the representation through the choice of wavelet, thus reducing the generality of any processing framework. Such a representation is tied to a specific class of local image features, consideration of regions containing features of different classes would therefore require having multiple representations of the image.

The next sections describe the most common representatives of these classes, the WFT and WT, in more detail.

2.2.1 The Windowed Fourier Transform

The windowed Fourier transform (WFT) (often the short time Fourier transform (STFT) in 1-d) [76, 44] is a phase-space signal representation, where a 1-d signal (in space) is transformed into a 2-d space/frequency representation. For a continuous 1-d signal $x(\xi)$ the forward transform $\hat{x}_s(t,\omega) \in L^2(R)$ is given by the equation

$$\hat{x}_s(\xi,\omega) = \int_{-\infty}^{\infty} w^*(\chi - \xi)x(\chi)\exp[-\jmath\omega\chi]d\chi \tag{2.1}$$

where $w(\xi)$ is the forward transform window, or *analysis window.*

This transform may be considered in two ways. First assume that, for a given position ξ the signal is premultiplied by the window function shifted so that it is

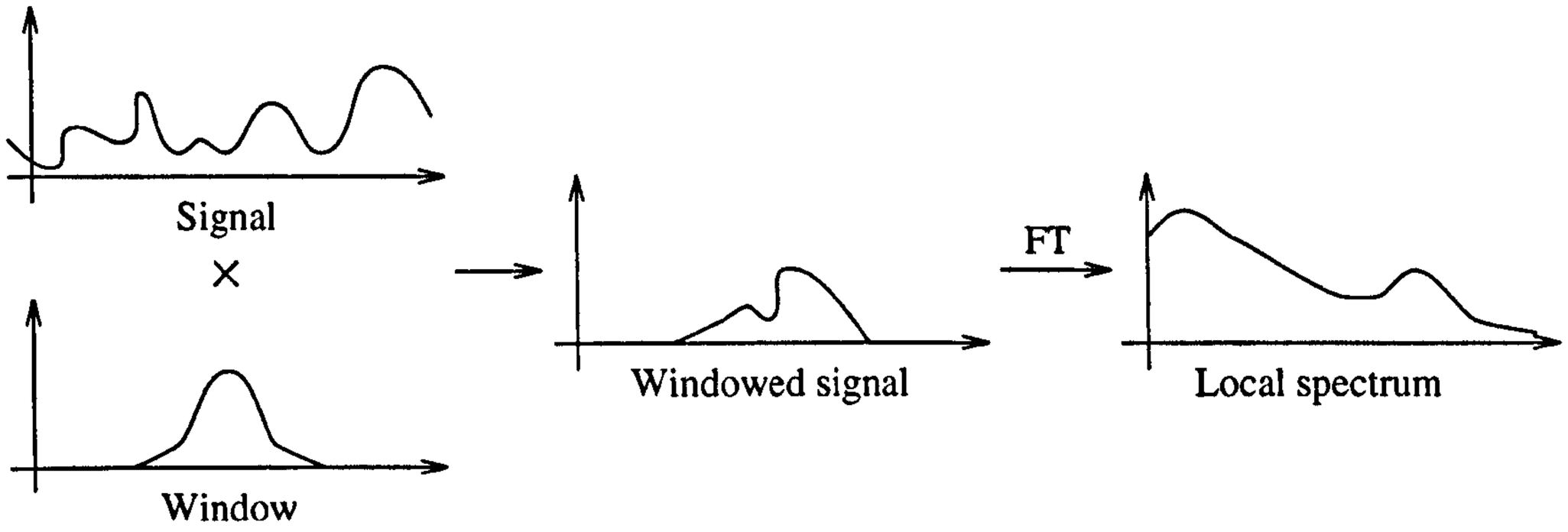


Figure 2.1: Local spectrum interpretation of WFT

centred at $\chi = \xi$, as shown in figure 2.1. Since the window decays away from the origin, the signal is windowed so that only the signal in the neighbourhood of ξ has any significant energy. The size of this neighbourhood can be defined in terms of the concentration of the analysis window energy, and a measure of this, such as the variance, can be used to indicate the scale at which the signal is being analysed [70]. Denoting this windowed signal by $x_w(\xi, \chi)$ the WFT is then simply its Fourier transform, ie.

$$\hat{x}_s(\xi, \omega) = \int_{-\infty}^{\infty} x_w(\xi, \chi) \exp[-\jmath\omega\chi] d\chi \tag{2.2}$$

Thus for a given value of ξ, the WFT may be regarded as a 'local spectrum', representing the spectral properties of the signal in the windowed region.

A second way of viewing the WFT is to regard it as the output of a bank of bandpass filters, as shown in figure 2.2. Equation (2.1) can be rewritten in terms of the Fourier transforms of both the signal and the window, $\hat{x}(\omega)$ and $\hat{w}(\omega)$ respectively, as

$$\hat{x}_s(\xi, \omega) = \exp[-\jmath\omega\xi] \int_{-\infty}^{\infty} \hat{x}(\omega') \hat{w}(\omega' - \omega) \exp[\jmath\omega'\xi] d\omega' \tag{2.3}$$

For a given ω, the signal spectrum is windowed by a spectral window centred upon

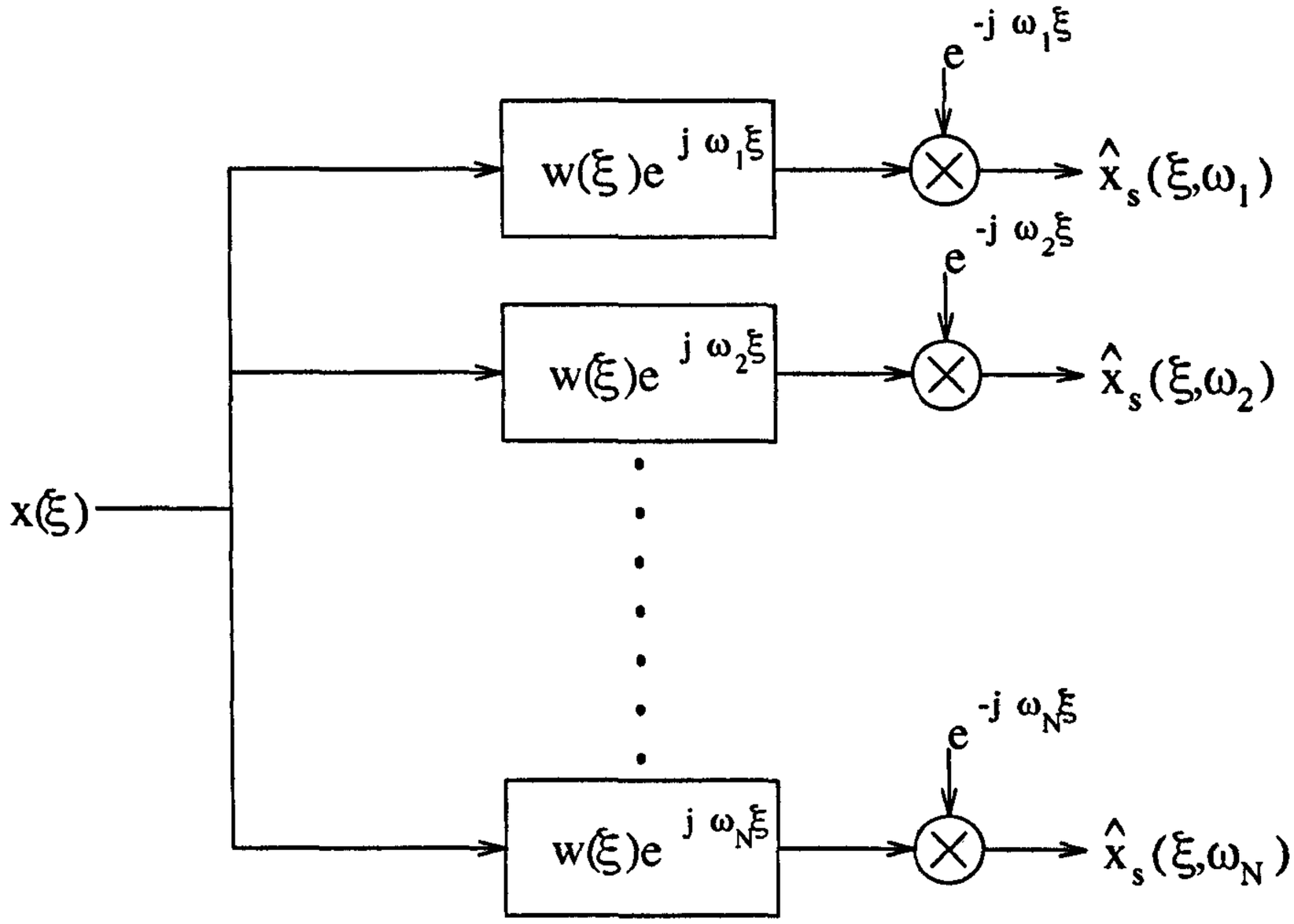


Figure 2.2: Filter bank interpretation of WFT

ω, the passband of which is dependent upon the concentration of the window in the frequency domain.

The WFT can be easily extended to two (or more) dimensions. For an image $x(\vec{\xi})$, where $\vec{\xi}$ is a 2-d position vector, and $\vec{\omega}$ is a 2-d frequency vector, the WFT is given by

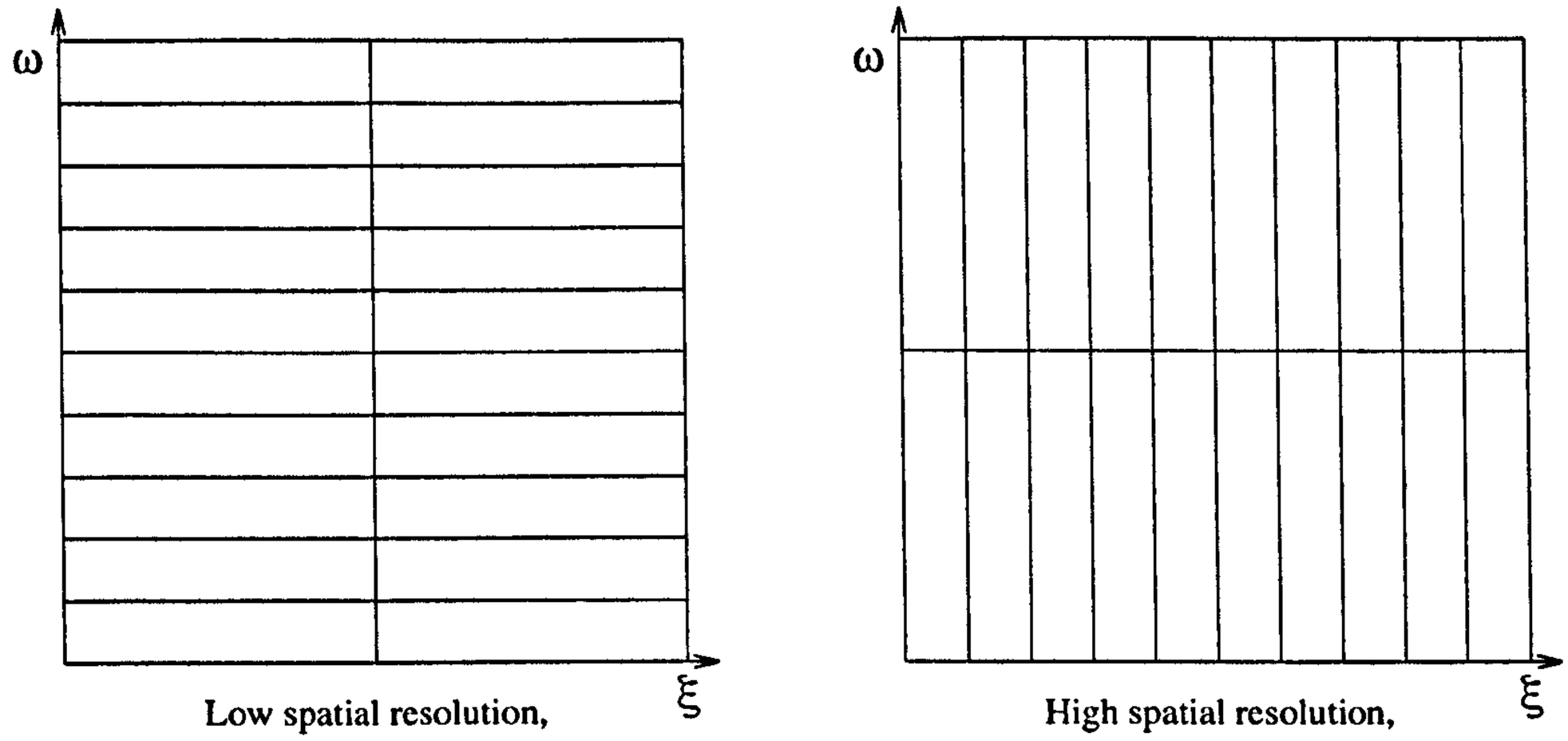
$$\hat{x}_s(\vec{\xi},\vec{\omega}) = \int_{-\infty}^{\infty} w^*(\vec{\chi}-\vec{\xi})x(\vec{\chi})\exp[-\jmath\vec{\omega}\cdot\vec{\chi}]d\vec{\chi} \tag{2.4}$$

where '$\cdot$' denotes scalar product.

For any type of window, the energy concentrations of the windows in the two domains are inversely proportional, since from the similarity theorem [70]

$$x(at) \leftrightarrow \frac{1}{a}\hat{x}\left(\frac{\omega}{a}\right) \tag{2.5}$$

A decision therefore has to be made about a suitable analysis scale. In order to



Low spatial resolution, high frequency resolution.

High spatial resolution, low frequency resolution.

Figure 2.3: WFT tessellation of time/frequency plane

model an image region as a single feature, an analysis window must be used that is sufficiently concentrated that the given region contains no other features. The similarity theorem, however, indicates that in the frequency domain the window will be spread out, leading to low frequency resolution. The scale of the analysis window must therefore be chosen to give an acceptable trade-off between resolution in the two domains. Using a measure such as window standard deviation in the spatial and frequency domains, σ_ξ and σ_ω respectively, a *logon* can be defined, following Gabor [37], of size $\sigma_\xi \times \sigma_\omega$ which represents the area of the phase-space covered by each analysis window. Choosing a set of sample points on the space/frequency plane about which these cells are centred, such that they provide a complete tessellation of the plane, a description of the signal at a given scale is provided. Figure 2.3 shows two tessellations of the space/frequency plane at different scales. The tessellation on the left has $\sigma_\xi = 5\sigma_\omega$, and therefore gives high frequency resolution and low spatial resolution, whereas the other has $\sigma_\xi = \frac{1}{5}\sigma_\omega$ and therefore gives high spatial resolution

and low frequency resolution.

Other problems arise from the need to choose the analysis scale before the analysis of the image starts, so that it cannot be adapted to the local structure of a particular image. A choice can only be made using heuristic knowledge of the class of images under consideration. Since the scale is constant over the entire image plane, the WFT is a single resolution representation, and as such the scale must be chosen to give a good compromise between the different scales that could best be used for different image regions.

2.2.2 The Wavelet Transform

One attempt to tackle the problems inherent in a single resolution approach such as the WFT is to use a multiresolution representation, in which different features of the signal are represented at different scales. In the Wavelet Transform (WT) [27, 41, 62], the analysis scale varies with frequency. The WT has been used for a number of applications such as image coding [2], active vision [21] and multi-orientation edge detection [75].

The WT can be regarded as a decomposition onto a set of basis functions called a family of wavelets. From Rioul [79], the Continuous Wavelet Transform (CWT), $\hat{x}_{cw}(\xi, a)$, of a 1-d signal $x(\xi) \in L^2(R)$ is given by

$$\hat{x}_{cw}(\xi, a) = \int_{-\infty}^{\infty} x(\xi) h^*_{a,\xi}(\chi) d\chi \tag{2.6}$$

where $h_{a,\chi}(\xi)$ is the analysing wavelet. Each of the analysing wavelets is a scaled, shifted version of a single prototype function or *mother wavelet*, $h(\xi)$, ie. each basis

function is given by

$$h_{a,\chi}(\xi) = \frac{1}{\sqrt{|a|}} h\left(\frac{\chi - \xi}{a}\right) \tag{2.7}$$

where χ represents the position about which the wavelet is centered and a the dilation of the window. Since a and χ are independent, this gives a free choice of both scale (dilation of wavelet) and position (the point about which the signal is analysed). In this case, the WT can be regarded as a scale-space representation.

Although the wavelet can be any of a large class of bandpass functions, choosing a modulated window function as the mother wavelet allows the WT to be regarded as a space-frequency representation and shows the similarities between the WT and the WFT. Choosing

$$h(\xi) = w(\xi) \exp\left[j\omega_0 \xi\right] \tag{2.8}$$

where $w(\xi)$ is a real valued window function, as used for the WFT, and ω_0 is the central frequency for the mother wavelet in the frequency domain. Given that the scaling factor is related to the central frequency by

$$a = \frac{\omega_0}{\omega} \tag{2.9}$$

the Wavelet transform can then be rewritten in the form

$$\hat{x}_{cw}(\xi, \omega) = \int_{-\infty}^{\infty} x(\chi) h^*_{\omega,\xi}(\chi) d\chi \tag{2.10}$$

where

$$h_{\omega,\xi}(\chi) = \sqrt{\left|\frac{\omega}{\omega_0}\right|} h\left(\frac{\omega}{\omega_0}(\chi - \xi)\right) \tag{2.11}$$

The advantage of the wavelet transform over the WFT is that the scale parameter a determines the resolution in the spatial and frequency domains. The wavelets have the property that if we define the standard deviation of energy (about 0) for $h(\xi)$

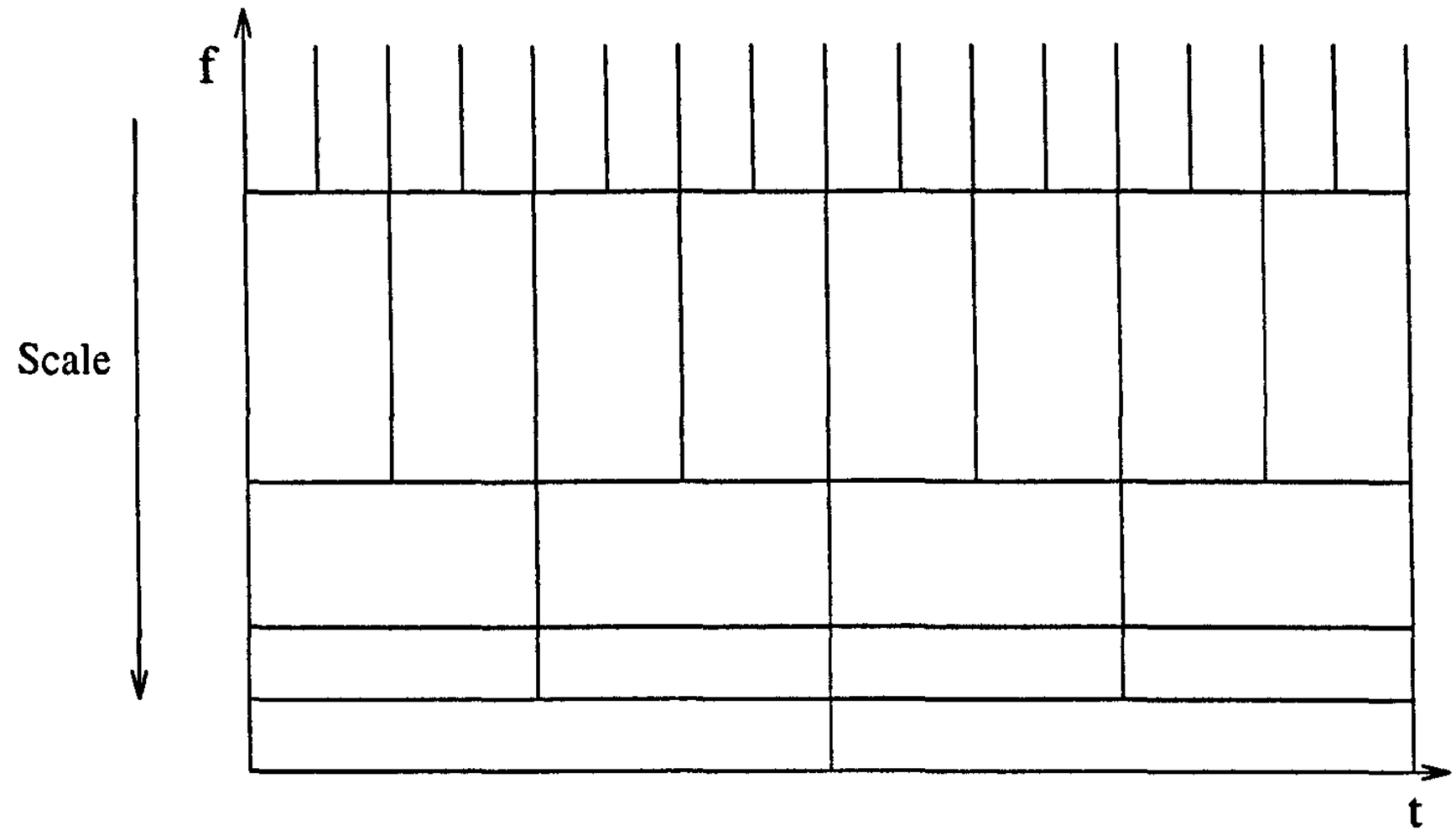


Figure 2.4: WT tessellation of time/frequency plane

in the spatial domain as σ_ξ, then the energy of $h_{\omega,\xi}(\xi)$ is concentrated about ξ with standard deviation σ_ξ/a. Similarly if we define $\hat{h}(\omega)$ as the Fourier transform of $h(\xi)$, then the energy of $\hat{h}_{\omega,\xi}(\omega)$ is concentrated about $a\omega$ with standard deviation $a\sigma_\omega$. Thus, as shown in figure 2.4, it can be seen that increasing a increases the concentration in the spatial domain, while spreading out the wavelet in the frequency domain. Conversely in order to increase concentration in the frequency domain, a must be reduced. The wavelet therefore has a "resolution cell" [62] in the phase-space equal to

$$[\xi - \frac{\sigma_\xi}{a}, \xi + \frac{\sigma_\xi}{a}] \times [a\omega - a\sigma_\omega, a\omega + a\sigma_\omega]. \tag{2.12}$$

Like the WFT, this may be considered as filtering the signal with a set of bandpass filters. Unlike the WFT, however, the filters do not have constant bandwidth but a constant ratio of centre frequency to bandwidth. The wavelet transform can therefore be regarded as a form of "constant-Q" analysis [79].

A variation of wavelets, called *chirplets* have been proposed by several authors [67, 63]. Wavelets are generated from a mother wavelet by shifting and scaling in the time-frequency plane. The generation of a chirplet additionally allows a mother chirplet to be sheared in time and frequency.

Extension of the WT to two dimensions is simple, and may be defined for signal $x(\vec{\xi}) \in L^2(R^2)$, as

$$\hat{x}_{cw}(\vec{\xi}, \vec{\omega}) = \int_{-\infty}^{\infty} x(\vec{\chi}) h^*_{\vec{\xi},\vec{\omega}}(\vec{\chi}) d\vec{\chi} \tag{2.13}$$

where $h_{\vec{\xi},\vec{\omega}}(\vec{\chi})$ is a 2-dimensional window function, centred on $\vec{\xi}$ in the spatial domain and $\vec{\omega}$ in the frequency domain. In practice, however, most approaches have used a set of 'mother wavelets', often based on cartesian separable functions. This is done for computational and representational reasons.

Mallat's work with the wavelet transform [61, 62] has used wavelets as an orthonormal basis for computing the level of detail present at each level of a multiresolution representation of an image. Following Witkin [98], Mallat defines the detail at a level as that information which is present at that level, but not at any level of coarser resolution. In doing this he provides a description of the original, sampled image, which consists of a coarse representation of the image, at some low level of resolution, and an orthogonal set of levels of detail that must be added back to the image, as its resolution is increased, in order to recover the original image. Unlike the Laplacian pyramid [14], however, there are multiple detail images at each scale - each representing a different range of feature orientations.

Since the detail levels are orthogonal, the representation is non-redundant (cf. Laplacian pyramid [14]): only N^2 samples are needed to represent an $N \times N$ image.

The restrictions of the Wavelet representation become apparent when considering

the resolution cell of the 1-dimensional case. Cells at high frequencies have low frequency resolution. For example, for frequencies centered on ω_0 the resolution cell is

$$[\xi_0 - \sigma_\xi, \xi_0 + \sigma_\xi] \times [\omega_0 - \sigma_\omega, \omega_0 + \sigma_\omega]. \tag{2.14}$$

In order to then consider frequencies around $2\omega_0$, a, the scale parameter, must be made equal to 2. This gives the following resolution cell

$$[\xi_0 - \frac{\sigma_\xi}{2}, \xi_0 + \frac{\sigma_\xi}{2}] \times [2\omega_0 - 2\sigma_\omega, 2\omega_0 + 2\sigma_\omega]. \tag{2.15}$$

It can be seen that although the frequency information is now centered on $2\omega_0$, the frequency resolution has been halved and the spatial resolution doubled.

In two dimensions, if the wavelets are considered in terms of angle and radial frequency, the angular resolution is fixed in the frequency domain. A long line (or edge) segment in the spatial domain has a narrow, oriented spectrum, whereas a shorter segment has a spectrum that is less concentrated. It is desirable to be able to alter angular resolution to take account of this.

2.3 The Multiresolution Fourier Transform

2.3.1 What is the MFT?

When considered as a space-frequency representation, the wavelet transform ties together scale and frequency. In order to analyse high frequencies a low frequency resolution is used. The multiresolution Fourier transform (MFT) is a representation that gives space/frequency information over a range of scales, but the scale and frequency variables are independent. A 1-d signal is transformed into a 3-d representation in space, frequency and scale. In general, an n-d signal has a $(2n + 1)$-d

representation. The MFT has been successfully used as the basis for image feature analysis [16], estimation of stereo disparity [17] and polyphonic audio analysis [73].

The continuous MFT of a continuous 1-d signal $x(\xi)$ is defined by the equation

$$\hat{x}(\xi,\omega,\sigma) = \sigma^{1/2}\int_{-\infty}^{\infty} x(\chi)w(\sigma(\chi-\xi))\exp\left[-\jmath\chi\omega\right]d\chi \tag{2.16}$$

where $w(\chi)$ is the window function, ω is the Fourier coordinate and σ is the scale variable. For a given σ, it can be be seen that the MFT equation reduces to that of the windowed Fourier transform, and that varying the value of σ increases or decreases the size of the analysis window used. If a finite set of scale parameters is used, the MFT may be considered as a stack of WFTs, over a range of scales. As such, the generality of the WFT representation is combined with the multiple scales of the WT. As shown in [90], the MFT allows any region of the image to represented in an interference free manner: any image feature at any scale and position can be represented independently of any surrounding features.

Figure 2.5 shows the different tilings of the time/frequency for a number of different scales. Since the representation provides a range of scales for analysis of the image, there is no need to make an *a priori* decision about the scale at which to represent the signal. During the analysis of the signal the "best" scale can be chosen for each region. It is also clear that for a given scale, the resolution cell is a constant size and shape in the two domains.

The continuous MFT of a continuous 2-d signal $x(\vec{\xi})$, where $\vec{\xi}$ is a position vector, is defined by the equation

$$\hat{x}(\vec{\xi},\vec{\omega},\sigma) = \sigma^{1/2}\int_{-\infty}^{\infty} d\vec{\chi}x(\vec{\chi})w(\sigma(\vec{\chi}-\vec{\xi}))\exp[-\jmath\vec{\chi}\cdot\vec{\omega}] \tag{2.17}$$

where $w(\vec{\chi})$ is a 2-dimensional window function and $\vec{\omega}$ is the Fourier coordinate.

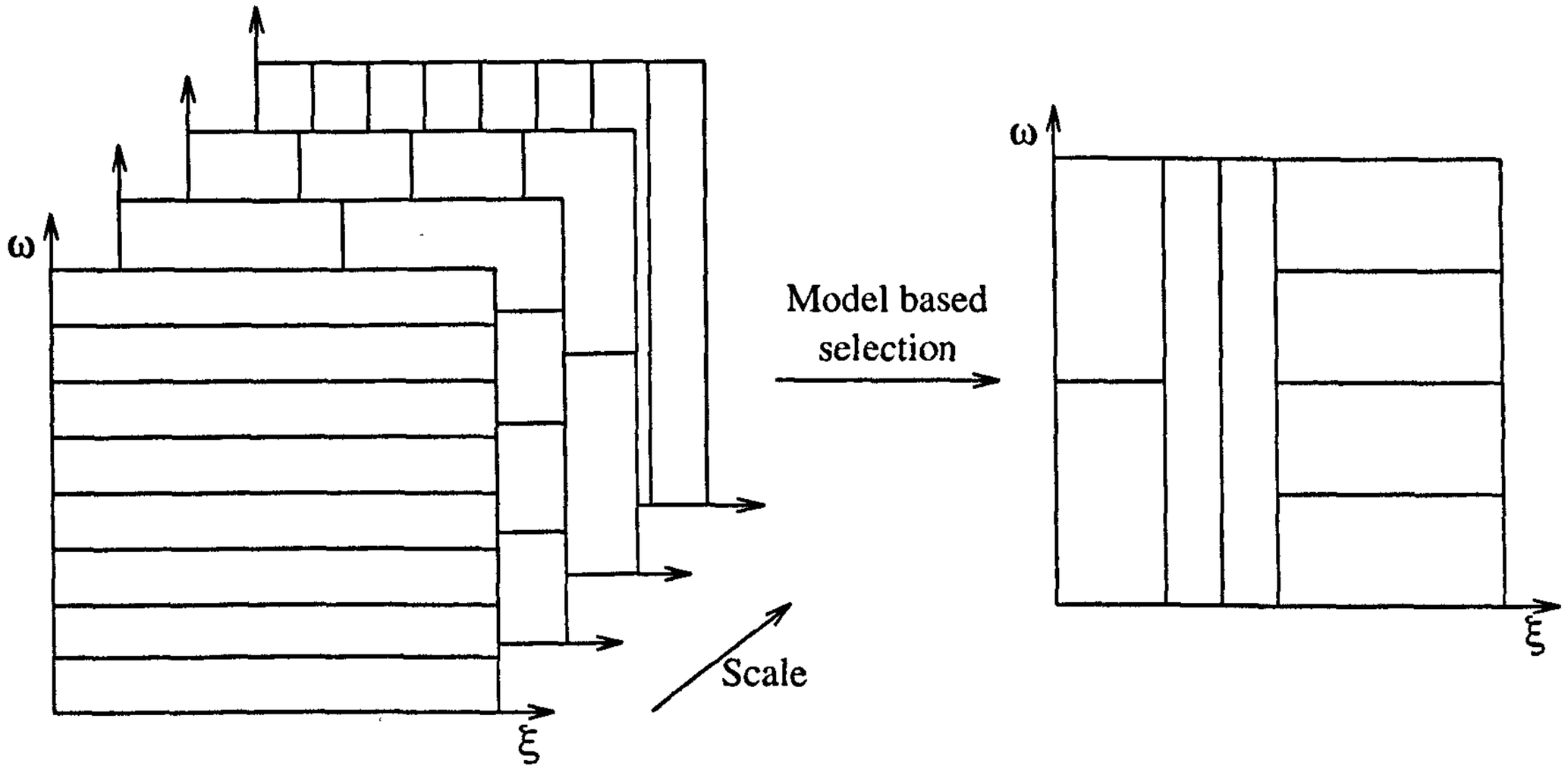


Figure 2.5: MFT tessellation of time/frequency plane

The discrete two dimensional MFT may be regarded as a sampling of the continuous case, and is given by

$$\hat{x}(\vec{\xi}_i(n), \vec{\omega}_j(n), \sigma(n)) = \sum_k x(\vec{\xi}_k) w_n(\vec{\xi}_k - \vec{\xi}_i(n)) \exp[-\jmath \vec{\xi}_k \cdot \vec{\omega}_j(n)] \tag{2.18}$$

where n is the discrete scale parameter, and $\vec{\xi}_i(n)$ and $\vec{\omega}_j(n)$, the spatial and frequency co-ordinates respectively, are sampled from the continuous domain.

In order that any given level of the MFT provides a complete representation of the signal, the number of sampling points along each spatial axis, $N_\xi(n)$, and the number along each frequency axis, $N_\omega(n)$, must obey the inequality

$$N_\xi(n) N_\omega(n) \geq N \tag{2.19}$$

assuming that the original signal is m-dimensional, and of size N^m. Given that $N = 2^M$, and that the sampling is regular in both spatial and frequency domains, the spatial and frequency sampling intervals for scale n, $\Xi_k(n)$ and $\Omega(n)$ respectively,

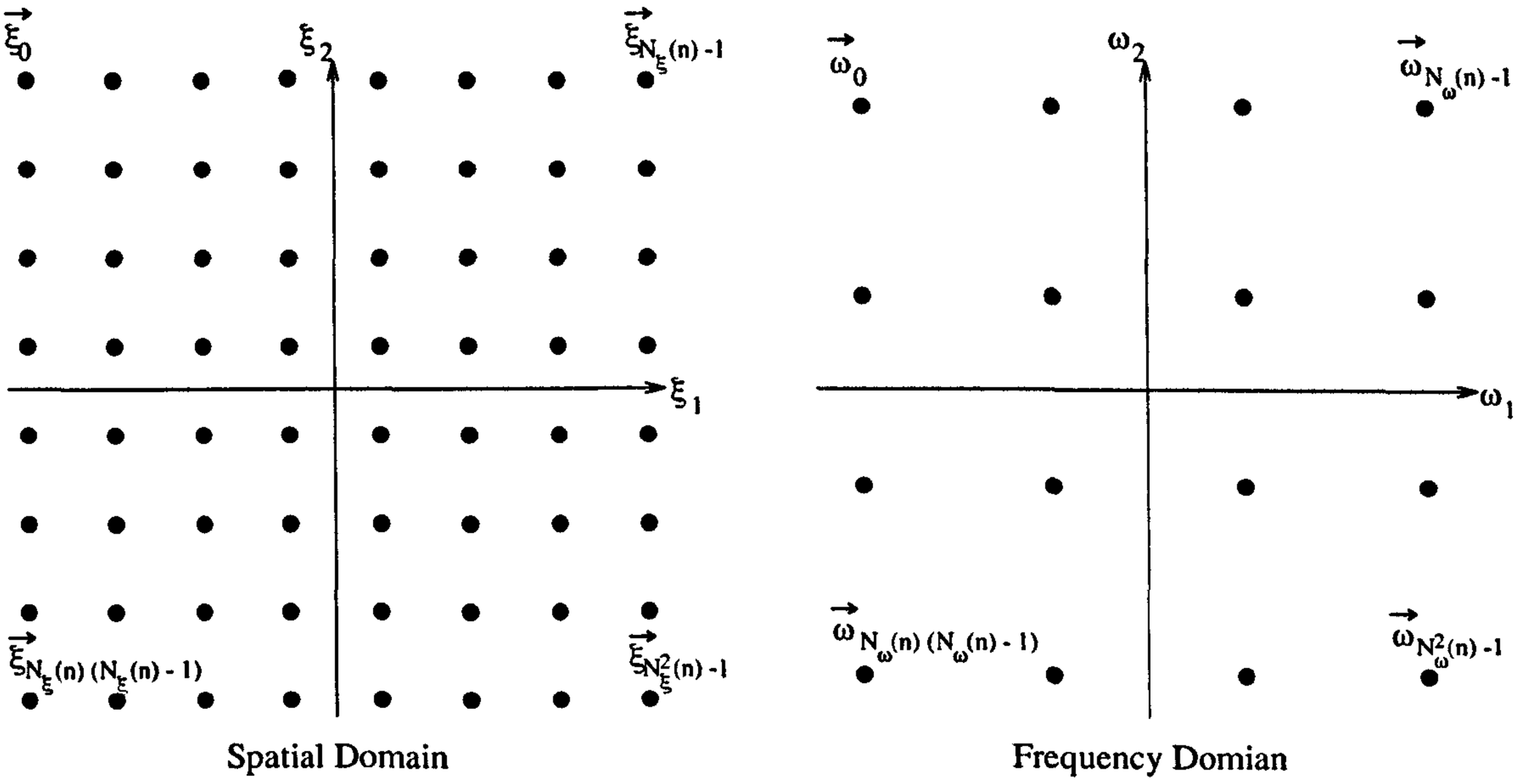


Figure 2.6: Spatial and Frequency sampling points

can be chosen as

$$\Xi_k(n) = 2^{M-k-n} \tag{2.20}$$

and

$$\Omega(n) = 2^{n+1-M}\pi \tag{2.21}$$

where $k \geq 0$ represents the degree of oversampling.

Regarding the origin in both spatial and frequency domains to be in the centre of the image/spectrum, between the four central coefficients, the sampling is symmetrical about the origin, as shown in figure 2.6. Since, as described in section 2.3.2, a relaxed ($k = 1$) version of the MFT is being used, the spatial sampling interval is $\Xi_1(n)$ in each spatial dimension and so for an $N \times N$ image, the number of samples along each spatial axis is

$$N_\xi(n) = N/\Xi_1(n) \tag{2.22}$$

For simplicity it is useful to label the sample points starting at the top left and

scanning a row at a time from left to right with the index i, in the range 0 to $N_\xi^2(n)-1$, as shown in figure 2.6. The spatial sample vectors, relative to the image origin, can then be given by

$$\vec{\xi}_i(n) = \Xi_1(n) \begin{bmatrix} i \bmod N_\xi(n) - (N_\xi(n)-1)/2 \\ i \div N_\xi(n) - (N_\xi(n)-1)/2 \end{bmatrix} \tag{2.23}$$

where '$\div$' denotes integer division.

In the frequency domain, the sampling interval is $\Omega(n)$ as given by equation (2.21). The number of samples along each spectral axis is therefore

$$N_\omega(n) = 2\pi/\Omega(n) \tag{2.24}$$

or alternatively

$$N_\omega(n) = 2^k N/N_\xi(n) \tag{2.25}$$

$$= 2^k \Xi_1(n) \tag{2.26}$$

The frequency samples can be labelled in a similar manner to that for the spatial samples, and the sample frequencies are given by

$$\vec{\omega}_j(n) = \Omega(n) \begin{bmatrix} j \bmod N_\omega(n) - (N_\omega(n)-1)/2 \\ j \div N_\omega(n) - (N_\omega(n)-1)/2 \end{bmatrix} \tag{2.27}$$

for $0 \le j < N_\omega^2(n)$.

When dealing with the discrete MFT, it makes sense to consider the local spectrum of a region as a two dimensional array. In this case the notation is defined by

$$\hat{x}_{i,n}(u,v) \triangleq \hat{x}(\vec{\xi}_i(n), \vec{\omega}_j(n), \sigma(n)), \quad 0 \le j < N_\omega^2(n) \quad 0 \le u,v < N_\omega(n) \tag{2.28}$$

where u,v and j are related by the expressions

$$u = j \bmod N_\omega(n) \tag{2.29}$$

and

$$v = j \div N_\omega(n) \tag{2.30}$$

Consequently the frequency vector represented by the indices (u, v) is, from equation (2.27),

$$\vec{\omega}_{u,v}(n) = \Omega(n) \begin{bmatrix} u - (N_\omega(n) - 1)/2 \\ v - (N_\omega(n) - 1)/2 \end{bmatrix} \tag{2.31}$$

2.3.2 Choice of Window Function for the MFT

For the present work a bandlimited window function is chosen. Since the feature models of interest are based in the frequency domain, it is desirable to have each coefficient corresponding to a precisely defined range of frequencies. This choice means that due to the uncertainty principle the window will be unconfined in the spatial domain [97, 95]. However, since the feature modelling is concerned with frequency spectra corresponding to spatial localities within the image, it is desirable to have a window function that is also highly concentrated in the spatial domain.

The analysis window functions used for the 1-d MFT are 'relaxed' forms of the Finite Prolate Spheroidal Sequence's (FPSS's) [94, 96], which can be shown to have the maximum energy concentration in the spatial interval $\Xi(n)$ for a bandlimited function with bandwidth $\Omega(n)$. From [90, 16] the window sequences are defined as solutions to the eigenvalue problem

$$\mathbf{B}(2^k\Omega(n))\mathbf{I}(\alpha\Xi_0(n))\mathbf{B}(\Omega(n))\mathbf{w_n} = \lambda\mathbf{w_n} \tag{2.32}$$

where $\mathbf{I}(\Xi)$ is the index limiting operator

$$I_{ij}(\Xi) = \begin{cases} \delta_{ij} & \text{if } |\xi_i| < \Xi/2 \\ 0 & \text{else} \end{cases} \tag{2.33}$$

and $\mathbf{B}(\Omega)$ is a bandlimiting operator given by

$$\mathbf{B}(\Omega) = \mathbf{F}^{-1}\mathbf{I}(\Omega)\mathbf{F} \tag{2.34}$$

Three window sequences generated as solutions to these equations are shown in figures 2.7 - 2.18. All were generated at scale $n = 3$, with $\Xi_0(n) = 32$ and $\Omega(n) = \pi/16$ assuming an image size of 256×256.

If $\alpha = 1$, then the window functions, for $k = 0, 1$ are the FPSS and frequency relaxed FPSS, described in [90]. Figures 2.7 - 2.10 show the non-relaxed version of the window, ie. the solution to equation (2.32) for $\alpha = 1$ and $k = 0$. Although this is strictly bandlimited (fig. 2.7), the spatial response (fig. 2.8 and 2.9) has large sidelobes right out to the edge. Following Wilson [90] a measure of how well concentrated the window is in the spatial domain is γ, defined as the ratio of the peak value to the value of the largest peak outside of the spatial concentration region, $\Xi_0(n)$. For this window $\gamma = -11.78$dB. Figure 2.10 shows the window plotted in the time-frequency plane, ie. the outer product of the 1-d space and frequency responses.

The choice of $k = 1$ improves the spatial energy concentration of the window by relaxing the bandwidth constraint. This frequency relaxed window is shown in figures 2.11 - 2.14. As can be seen in figures 2.11 and 2.12 although the bandwidth relaxation reduces the size of the sidelobes, the frequency domain function still has a discontinuity at the edges. Once again figure 2.13 show the 2-d spatial response of the window on a logarithmic scale, down to -50 dB. For this window function $\gamma = -25.42$dB, a large improvement over the non-relaxed window.

As the windows overlap, each region of the signal is included within more than one windowed version of signal. To ensure that any region of the image plane is well

represented, it is therefore necessary to ensure that any region is represented in at least one of the windowed signals, with minimum interference from other regions. Denoting by $x_w(\xi_i, \xi)$ the original function windowed about the position ξ_i, given by

$$x_w(\xi_i, \xi) = w_n(\xi_i - \xi)x(\xi) \tag{2.35}$$

the local spectrum corresponding to this windowed sequence is represented by the set of MFT coefficients $\hat{x}(\xi_i(n), \omega_j(n), \sigma(n))$, $0 \leq j < \Xi_0(n)$. Since the window is not spatially limited, and the local spectrum is subsampled, the spectrum actually corresponds to a sequence, $f(\xi_i, \xi)$, given by

$$f(\xi_i, \xi) = \sum_i x_w(\xi_i, \xi + k\Xi_0(n)), \quad \text{for } k = 0, \pm 1, \pm 2, ... \tag{2.36}$$

Assuming that $k = 1$, the spatial sampling interval is, from equation (2.20),

$$\Xi_1(n) = 2^{M-n-1} \tag{2.37}$$

for a sequence of length 2^M. The spatial concentration interval, however, is

$$\Xi_0(n) = 2^{M-n} \tag{2.38}$$

$$= 2\Xi_1(n) \tag{2.39}$$

and so the two adjacent spatial window functions $f(\xi_j, \xi)$ and $f(\xi_{j-1}, \xi)$ overlap by 50%. Each local spectrum may therefore be considered as corresponding primarily to the central region of the windowed signal, ie. the interval $[\xi_j - \Xi_0(n)/4, \xi_j + \Xi_0(n)/4]$. Anything outside of this will be in the central region of one of the neighbouring windows. Any sidelobes that can affect this central region must be outside of the central region of the neighbouring windows, ie. they must be in of the intervals $[0, \xi_j - 3\Xi_0(n)/4]$ or $[\xi_j + 3\Xi_0(n)/4, N]$. Defining ρ to be the ratio of the maximum

n	ρ (dB.)			
	$k=0$	$k=1$		
	$\alpha=1$	$\alpha=1$	$\alpha=1.125$	$\alpha=1.25$
2	-16.06	-27.31	-28.63	-27.60
3	-13.79	-26.67	-27.88	-27.52
4	-15.41	-26.68	-27.73	-27.49
5	-22.51	-26.86	-	-27.80

Table 2.1: Values of ρ for various window parameters

sidelobe magnitude in these intervals to the maximum window magnitude, a measure can therefore be given to the largest sidelobe that affects the central portion of the windowed signal.

If $\alpha = 1$, the values of ρ for $k = 0, 1$ are -13.79dB and -26.67dB respectively. Using this measure of window concentration, it is possible to improve the window sequence. By selecting an appropriate value of $\alpha > 1$ and allowing some relaxation of the windows spatial energy concentration, the window in the frequency domain can be seen to approach zero more smoothly. The window function shown in figures 2.15 - 2.18 were generated using $\alpha = 1.125$, allowing a relaxation of the spatial energy concentration factor of 1/8. For this window function $\gamma = -21.57$dB, and as such the maximum sidelobe outside of the concentration interval is larger than for the frequency relaxed FPSS. However $\rho = -27.88$dB, and this indicates that less interference will be caused by the window sidelobes to the central portion of the windowed sequence.

Table 2.1 shows the values of ρ for scales $2 - 5$ and for various values of the parameters k and α. For scale $n = 5$, ie. where $\Xi(n) = 8$, it is not sensible to expand by 1/8 since it desirable to use a window with an even number of samples. For the other scales choosing $\alpha = 1.125$ gives an improvement in the value of ρ over windows

with $\alpha = 1$. Increasing the relaxation, eg. to $\alpha = 1.25$ reduces the value of ρ and in all the tabulated cases leads to an undesirable zero crossing in the frequency response.

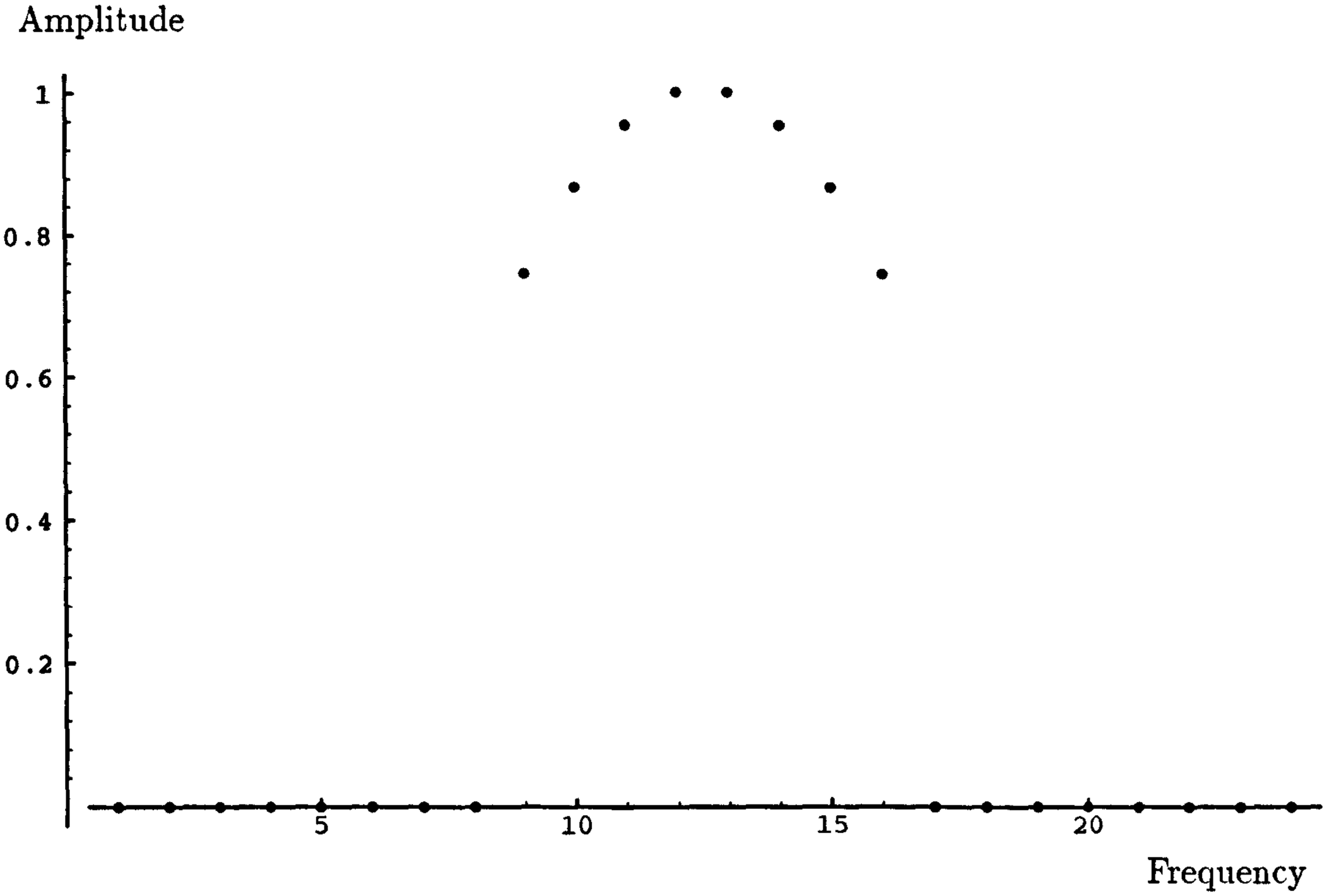


Figure 2.7: Non-relaxed Window: Frequency Domain - origin between samples 12 and 13

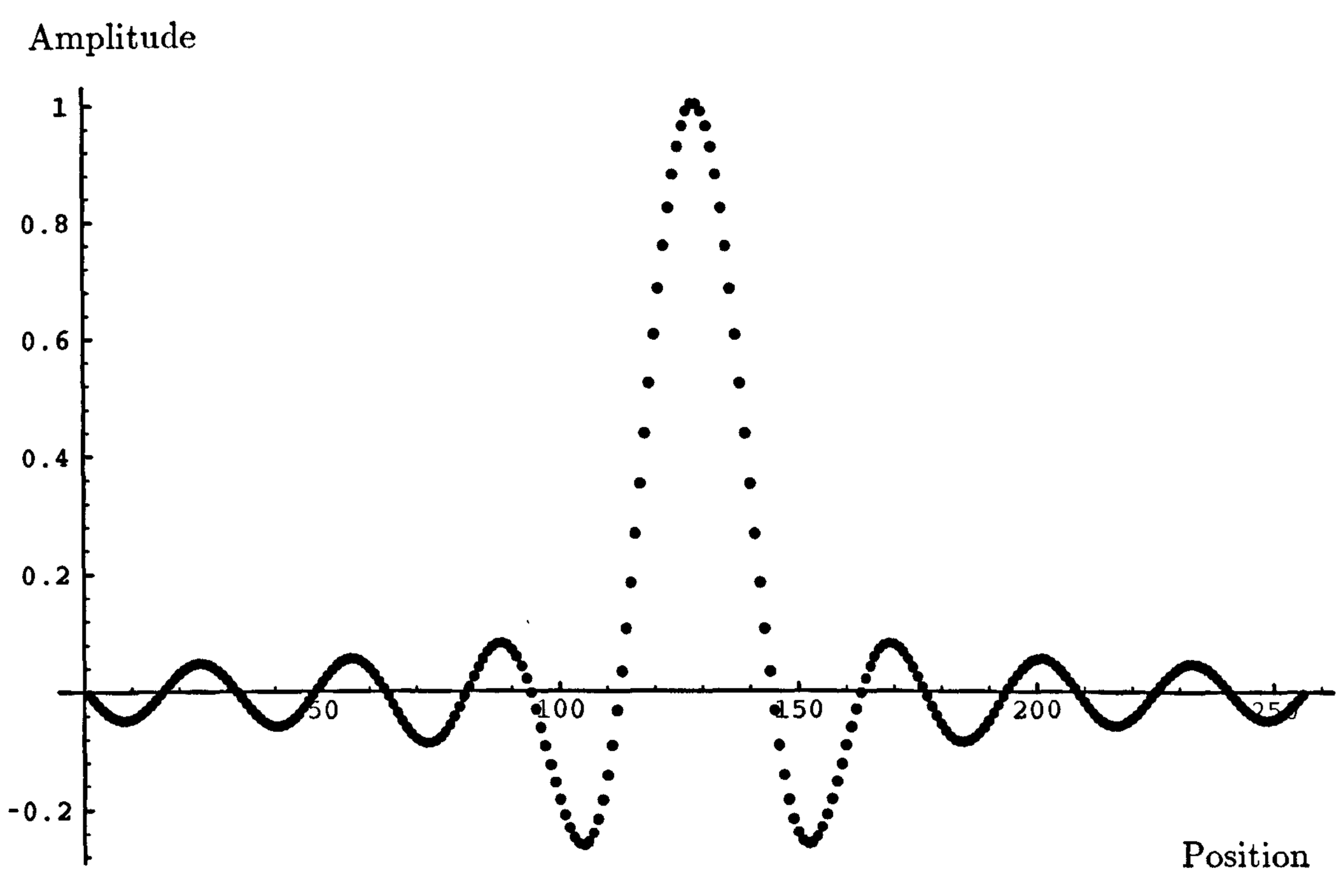


Figure 2.8: Non-relaxed Window: Spatial Domain - origin at sample 128

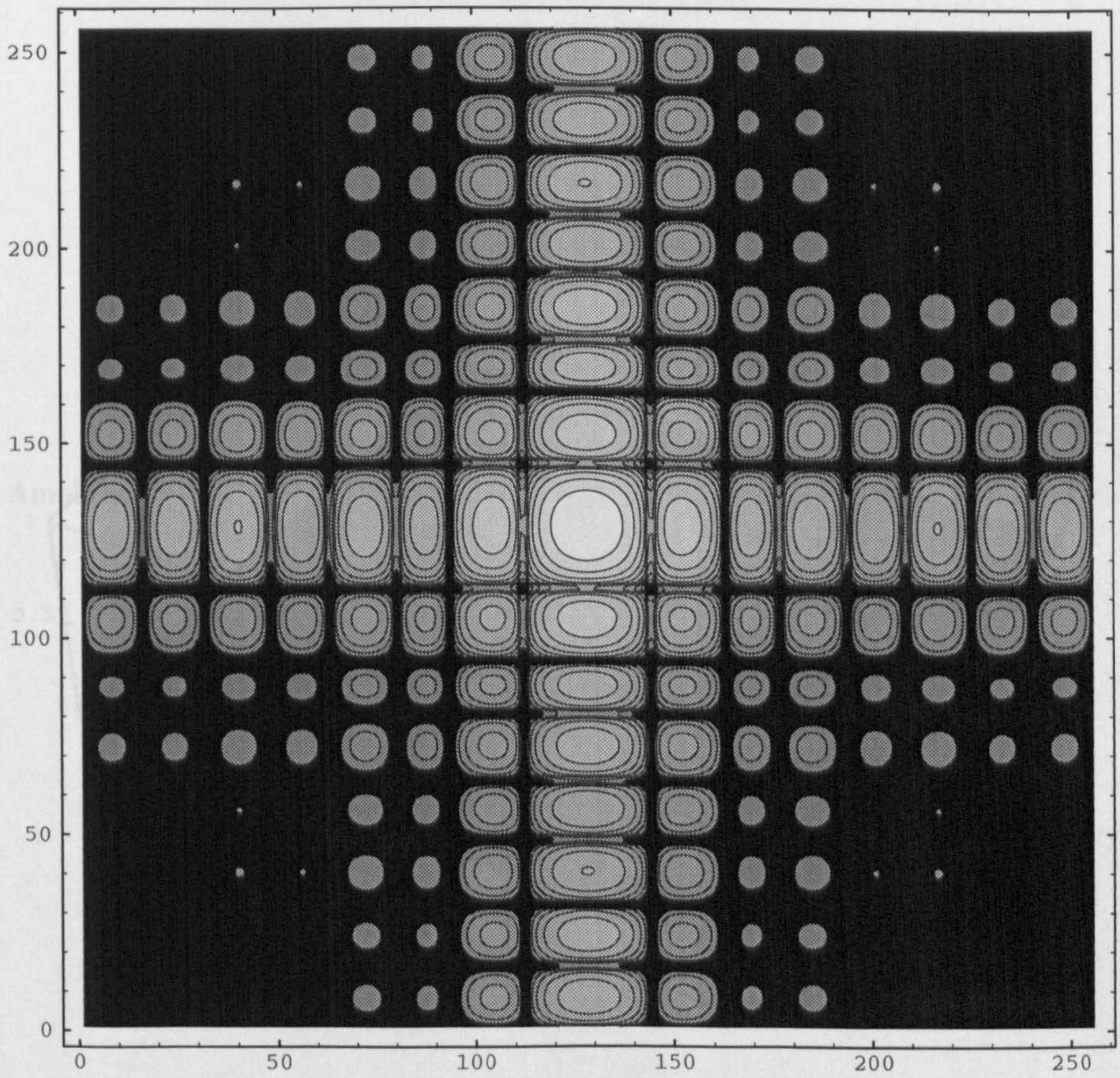


Figure 2.9: Non-relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB., origin at (128, 128)

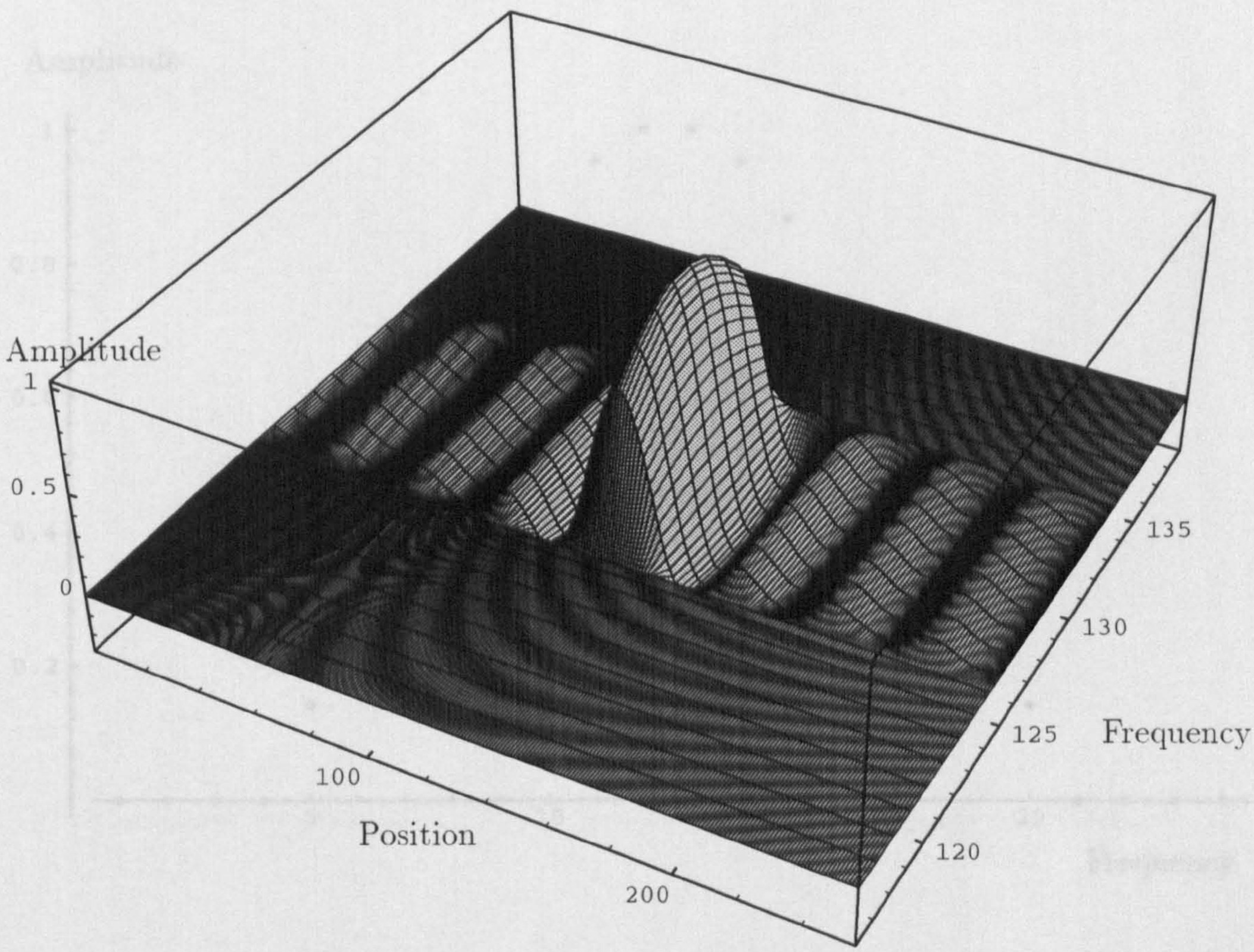


Figure 2.10: Non-relaxed Window: Space/Frequency Plot - origin in space-frequency plane at $(128, 128)$

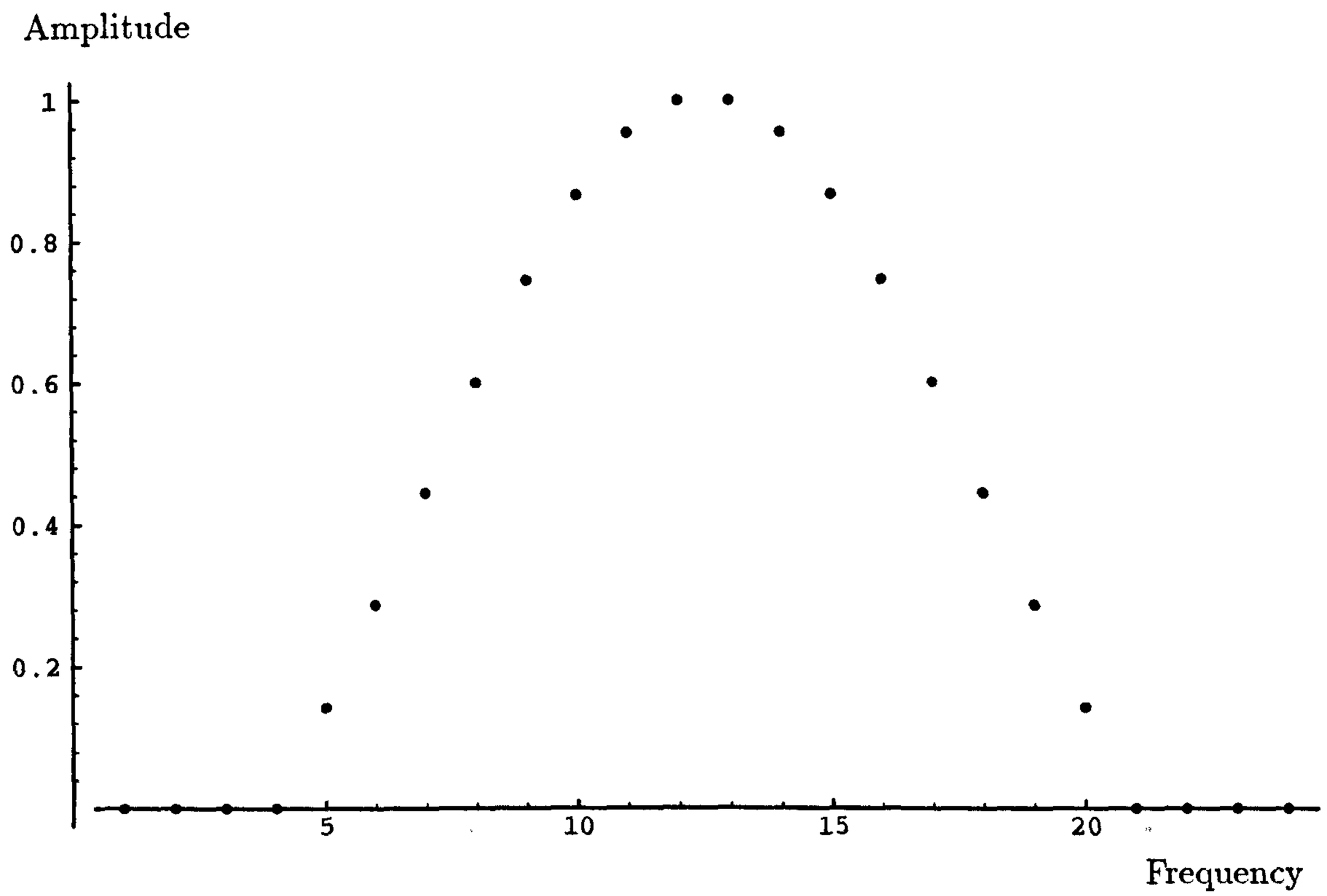


Figure 2.11: Frequency Relaxed Window: Frequency Domain - origin between samples 12 and 13

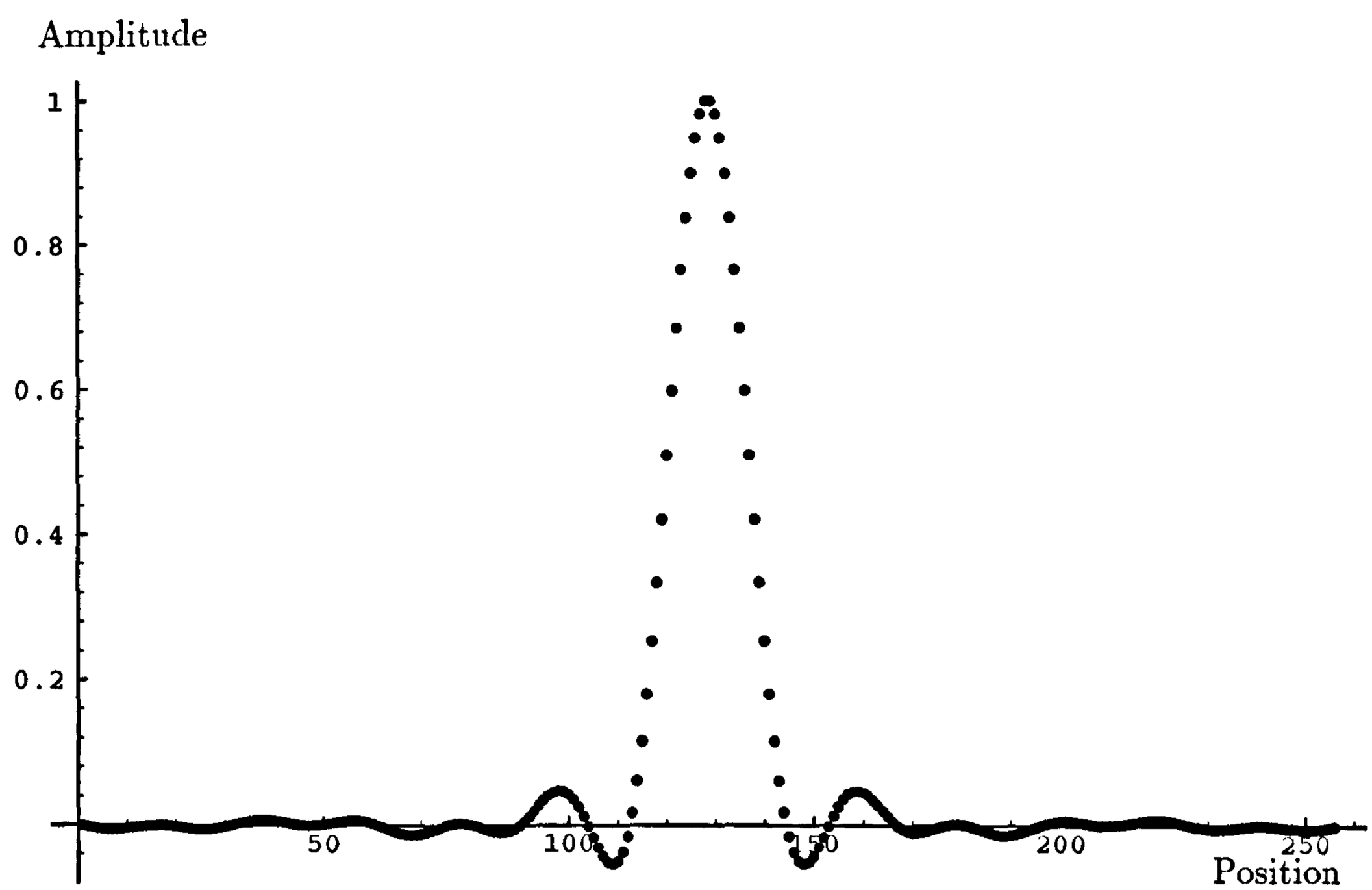


Figure 2.12: Frequency Relaxed Window: Spatial Domain - origin at sample 128

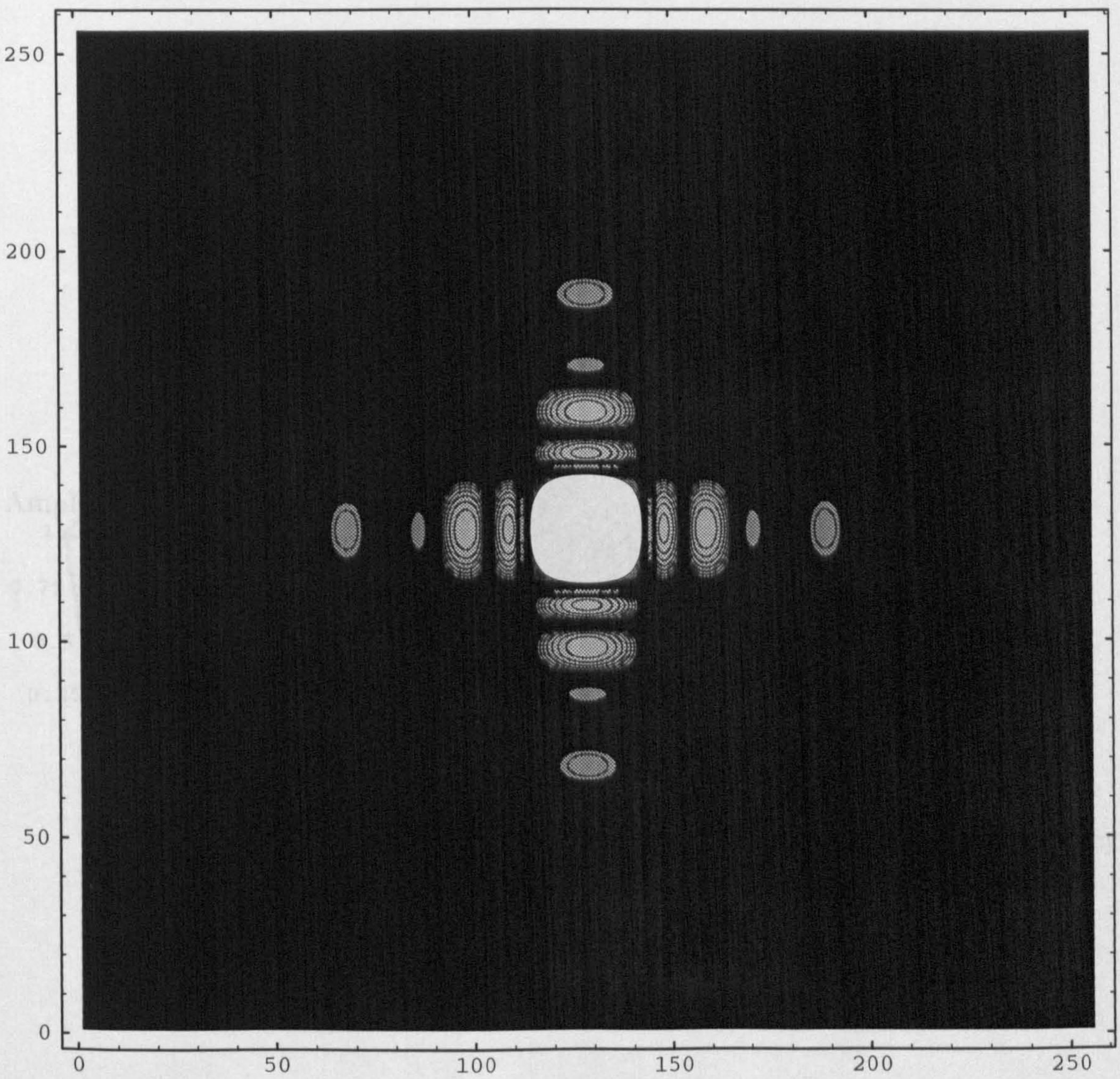


Figure 2.13: Frequency Relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB., origin at (128, 128)

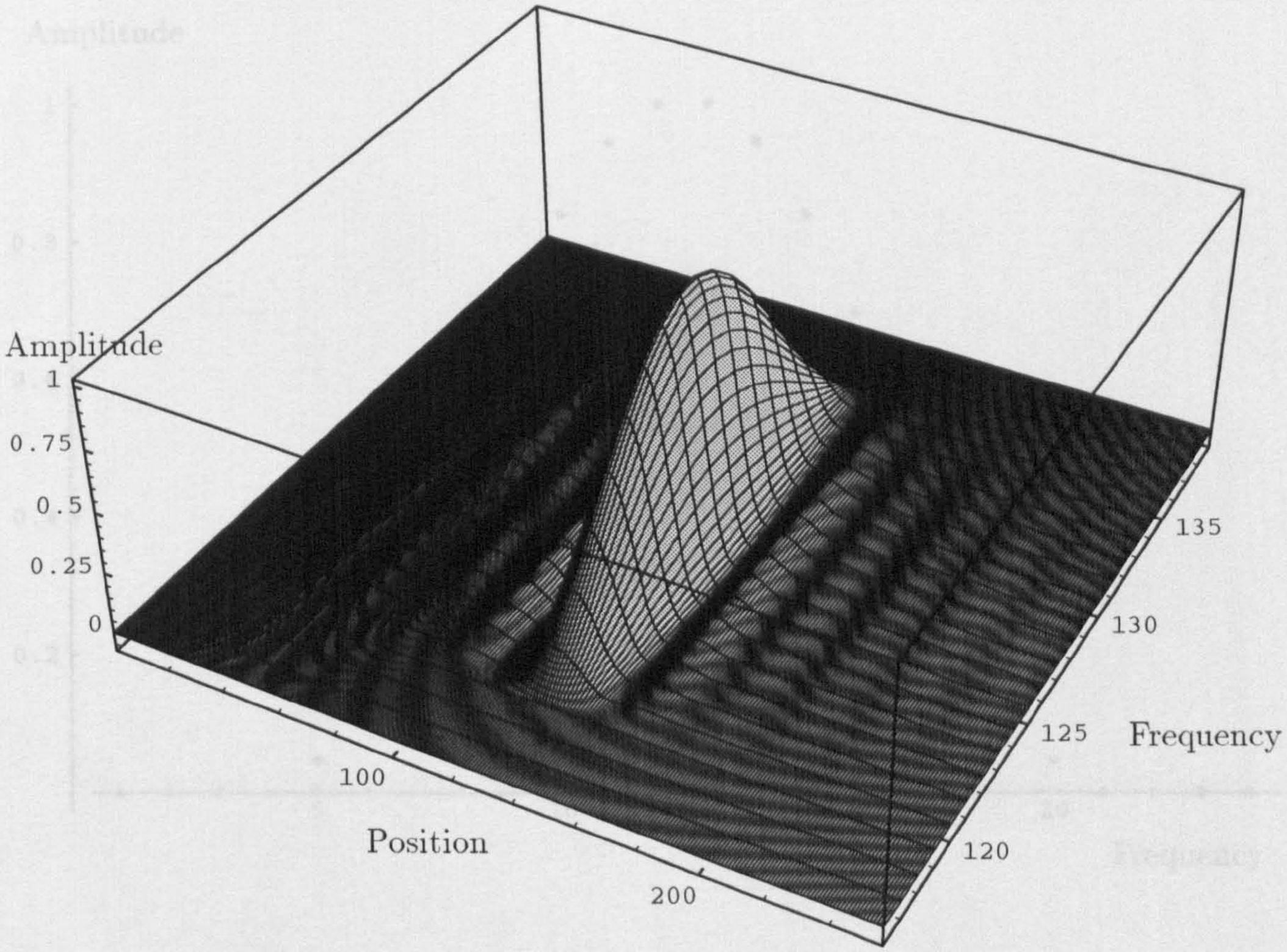


Figure 2.14: Frequency Relaxed Window:Space/Frequency Plot - origin in space-frequency plane at $(128, 128)$

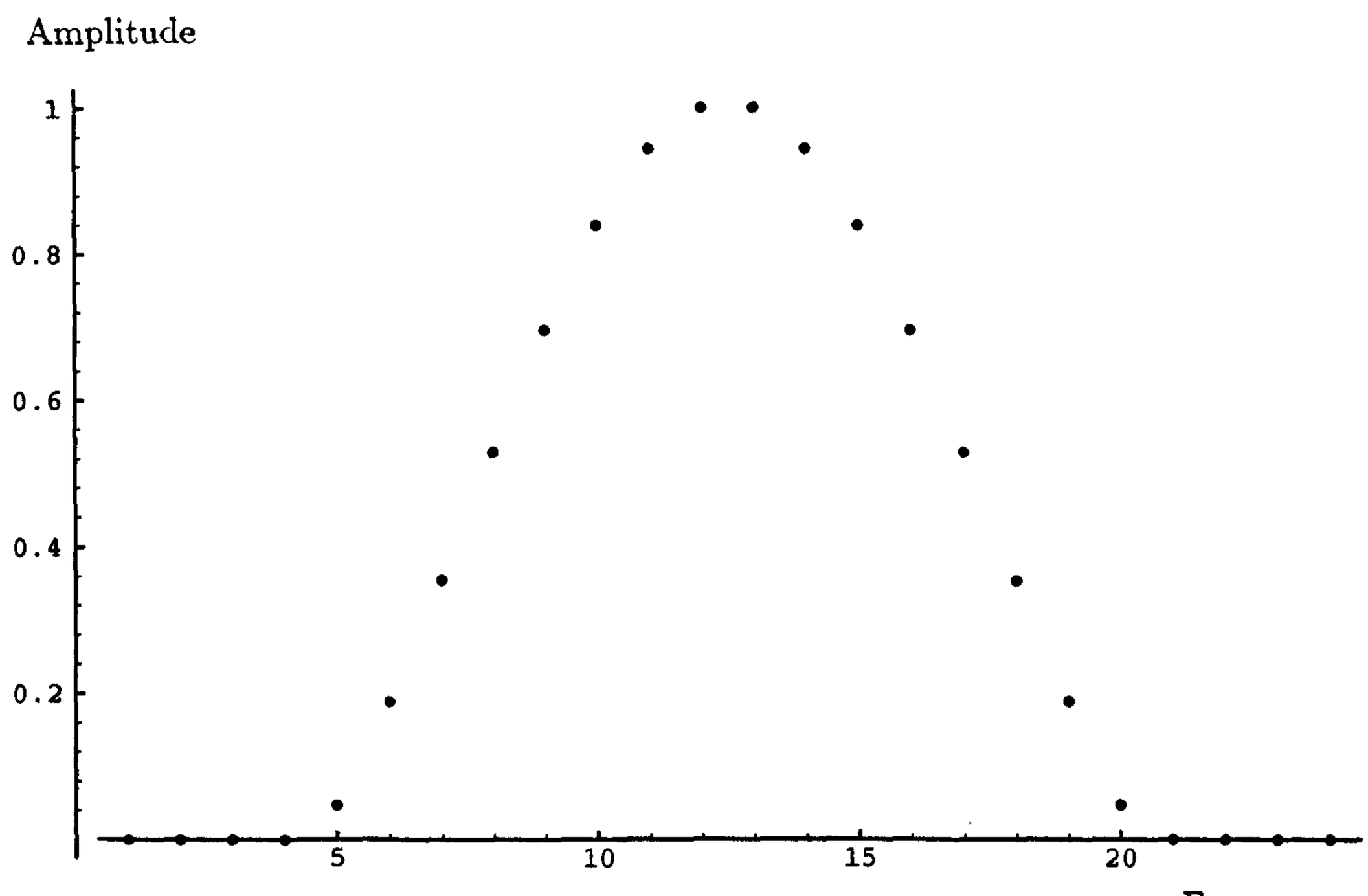


Figure 2.15: Spatial/Frequency Relaxed Window: Frequency Domain - origin between samples 12 and 13

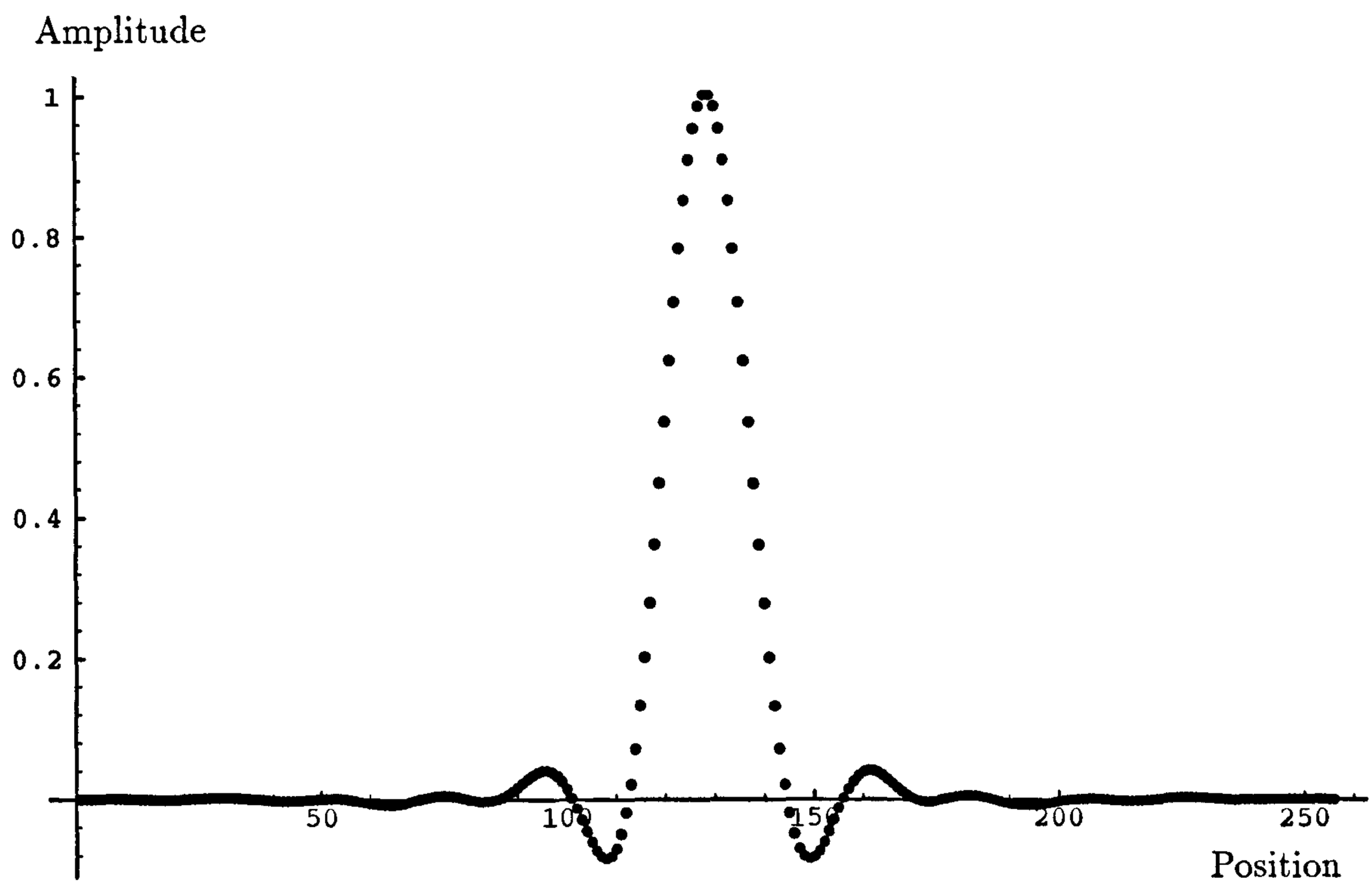


Figure 2.16: Spatial/Frequency Relaxed Window: Spatial Domain - origin at sample 128

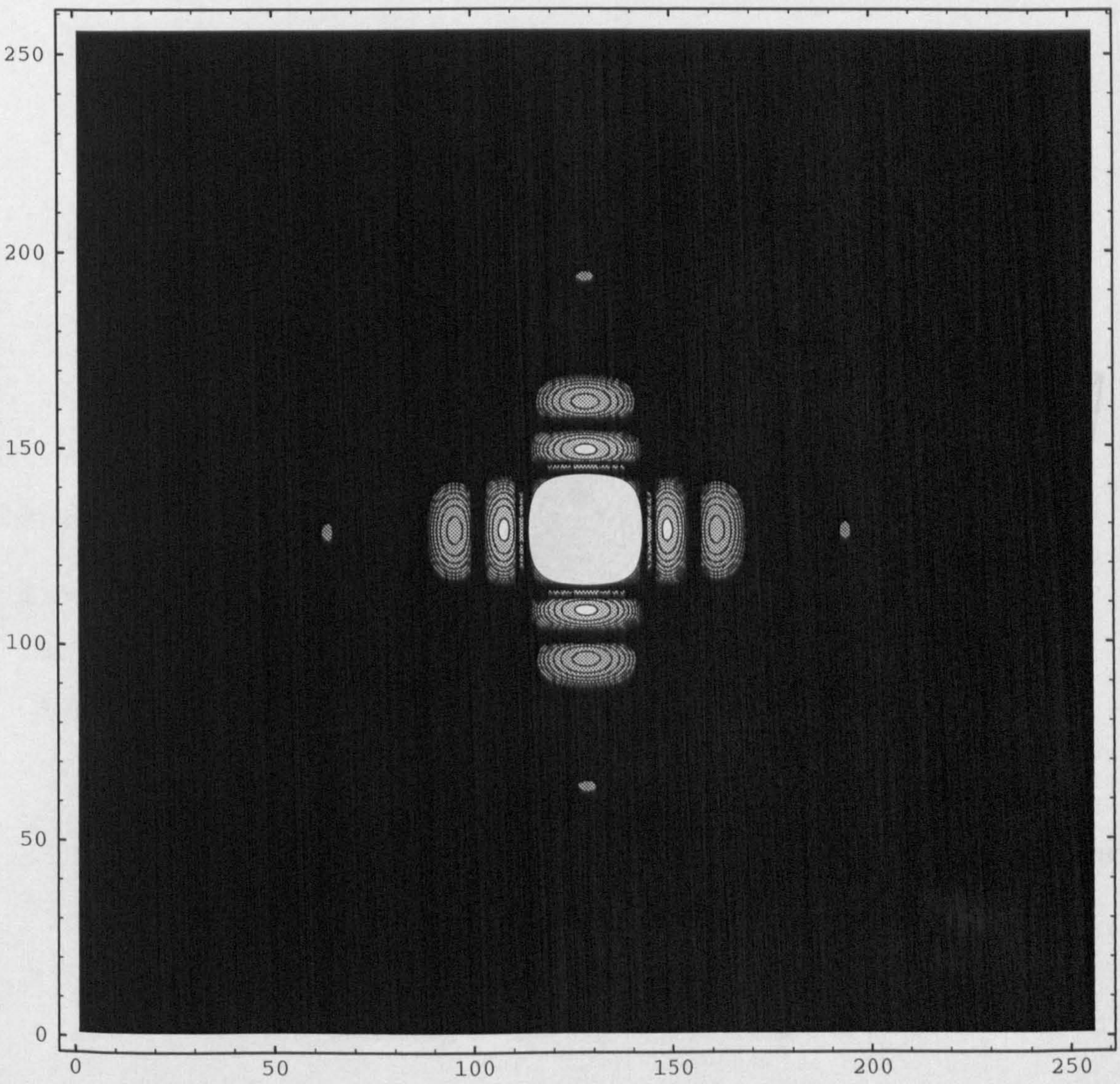


Figure 2.17: Spatial/Frequency Relaxed 2-d Window in Spatial Domain: Contour Plot - Contours at 5 dB. intervals, down to -50 dB., origin at $(128, 128)$

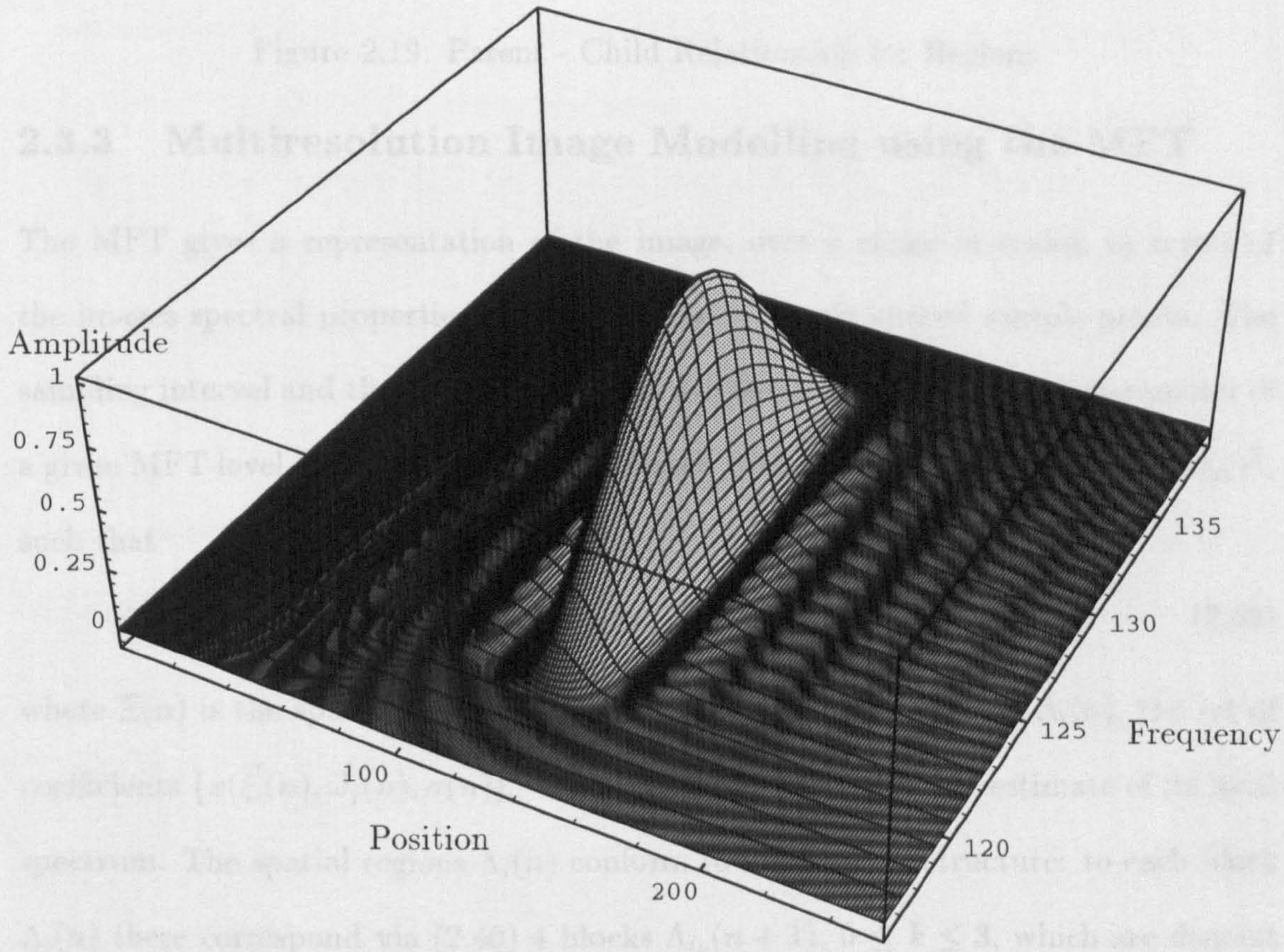


Figure 2.18: Spatial/Frequency Relaxed Window:Space/Frequency Plot - origin in space-frequency plane at $(128, 128)$

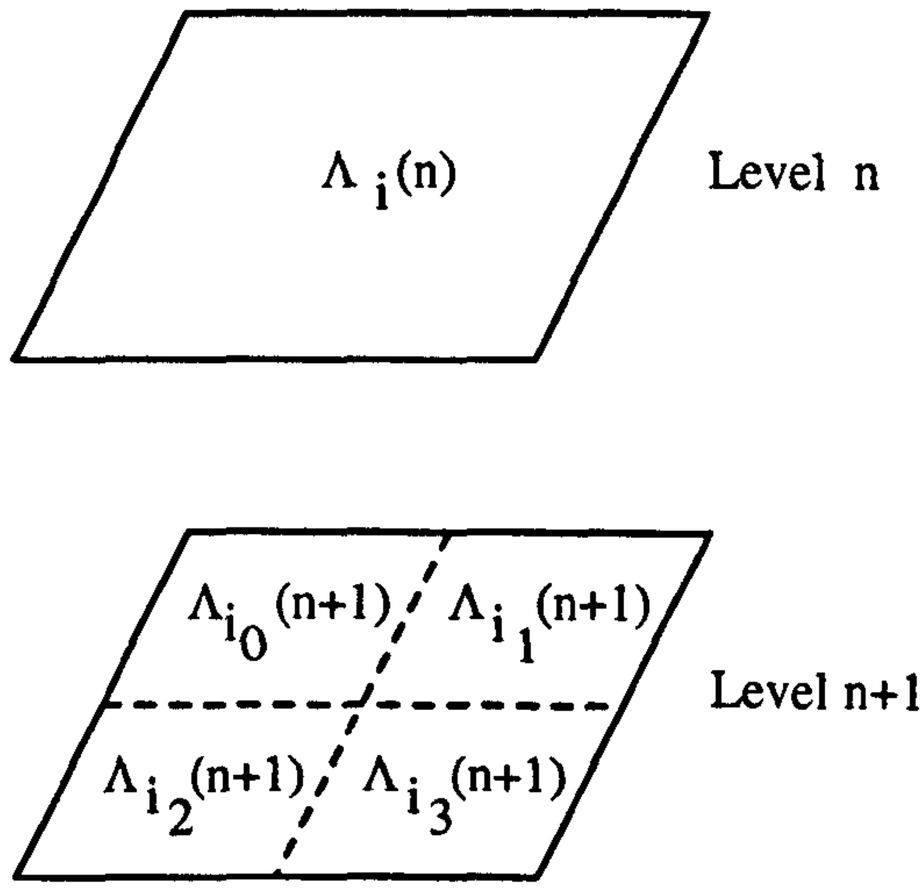


Figure 2.19: Parent - Child Relationship for Regions

2.3.3 Multiresolution Image Modelling using the MFT

The MFT gives a representation of the image, over a range of scales, in terms of the images spectral properties in the region of regularly spaced sample points. The sampling interval and the size of the regions are determined by the scale parameter of a given MFT level. A region $\Lambda_i(n)$ is associated with each sample point $\vec{\xi_i} = (\xi_{i1}\, \xi_{i2})^T$, such that

$$\vec{\xi} \in \Lambda_i(n) \iff |\xi_l - \xi_{il}| < \Xi(n)/2, l = 1, 2 \tag{2.40}$$

where $\Xi(n)$ is the spatial sampling interval at level n. For a region $\Lambda_i(n)$, the set of coefficients $\{\hat{x}(\vec{\xi_i}(n), \vec{\omega}_j(n), \sigma(n)),\ \ 0 \leq j < 4\Xi^2(n)\}$, is a discrete estimate of its local spectrum. The spatial regions $\Lambda_i(n)$ conform to a quad-tree structure: to each block $\Lambda_i(n)$ there correspond via (2.40) 4 blocks $\Lambda_{i_k}(n+1)$, $0 \leq k \leq 3$, which are disjoint and whose union is equal to $\Lambda_i(n)$,(see Fig. 2.19),

$$\Lambda_i(n) = \bigcup_{k=0}^{3} \Lambda_{i_k}(n+1) \tag{2.41}$$

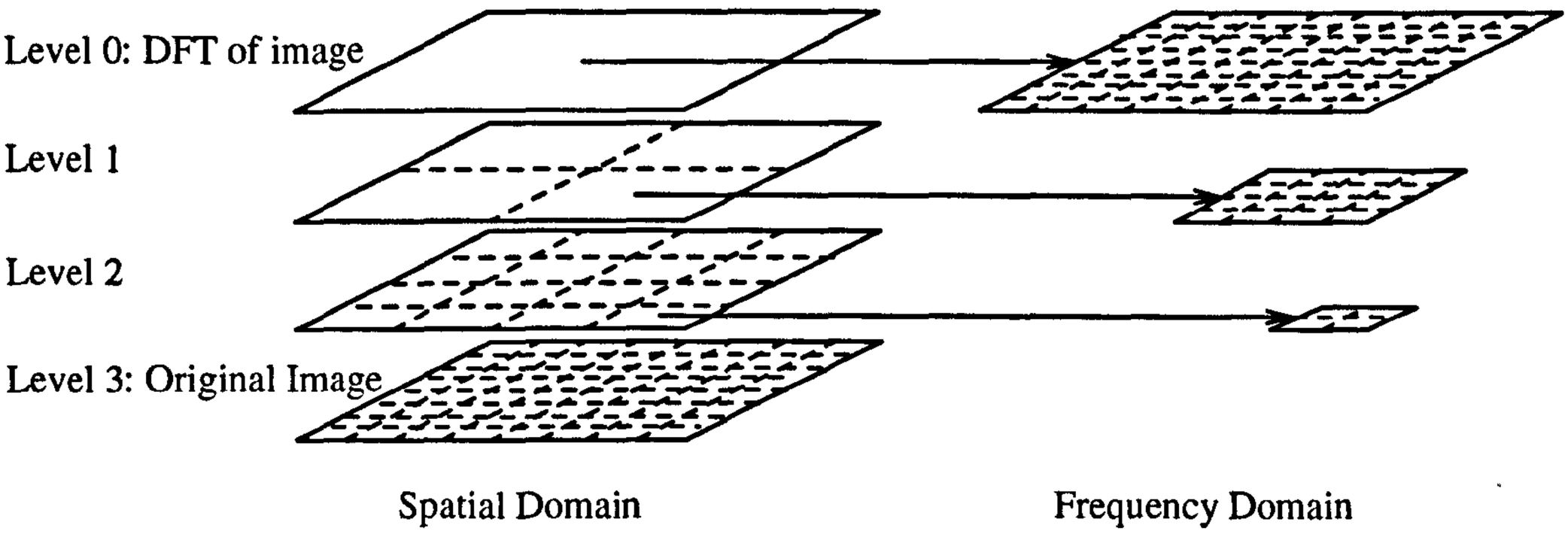


Figure 2.20: MFT structure - 2-dimensional

Ancestors of a region $\Lambda_i(n)$ can be denoted by $\Lambda_{i^l}(n-l)$, $1 \leq l \leq n$, where $\Lambda_{i^1}(n-1)$ is the parent and $\Lambda_{i^2}(n-2)$ the grandparent etc.

Figure 2.20 shows the MFT for a small image. In the spatial domain, each region at level n is split into 4 at level $n+1$ (its children); thus spatial resolution increases with n. From equation (2.20)

$$\Xi(n+m) = 2^{-m}\Xi(n) \tag{2.42}$$

and therefore the size of the spatial region is halved by each increase in level. The set of local Fourier coefficients representing the spectrum of each of the children, however, is a quarter of the size of that for the parent, i.e. the frequency resolution decreases with n. It is clear from equation (2.21) that

$$\Omega(n+m) = 2^{m}\Omega(n) \tag{2.43}$$

and therefore that the frequency sampling interval, in each dimension, doubles with each increase in level.

The image model in this work follows that described in [88, 9, 16], in that it is a variety of a class of linear multiresolution image models first described by Clippingdale

[25]. An image can be considered as consisting of a tessellation of differently sized square regions, each of which contains features from one of a set of classes. The image at a given level, or resolution, will consist of a tessellation of regions that are the same size. A linear, recursive version of this model [91], which may be considered as a Markov model, is given by

$$x(\vec{\xi}, n) = A(\vec{\xi}, n)x(\vec{\xi}, n-1) + B(\vec{\xi}, n)w(\vec{\xi}, n), \;\; 0 \geq n \geq N \tag{2.44}$$

where $x(\vec{\xi}, n)$ is the image at level n, $w(\vec{\xi}, n)$ represent the innovations at level n and the functions $A(\vec{\xi}, n)$ and $B(\vec{\xi}, n)$ 'build' the image at level n by selecting features from the innovations at the current level, and the features at previous levels. The function $B(\vec{\xi}, n)$ may be considered as a "detector" function, indicating regions that can be modelled at level n using instances of the feature classes. The other function, $A(\vec{\xi}, n)$, may be considered a scale selection function, choosing which regions should be modelled at the current level n and which at lower levels $n' < n$. An example of this model is given in figure 2.21. Given

$$x(\vec{\xi}, 0) = B(\vec{\xi}, 0)w(\vec{\xi}, 0) \tag{2.45}$$

the complete image containing all features at all scales is given by

$$x(\vec{\xi}) = x(\vec{\xi}, N) \tag{2.46}$$

In terms of the local (regional) representation given by the MFT, the image at a given level can be considered in terms of the regions at that level, and lower levels, which can be modelled by features from a set of feature classes. Associated with each region, $\Lambda_i(n)$, is a set of features $\Phi_i(n)$, that describe, to some satisfactory level of accuracy, the features present in the region. Each member of $\Phi_i(n)$, $\phi_{i_k}(n)$, is a vector

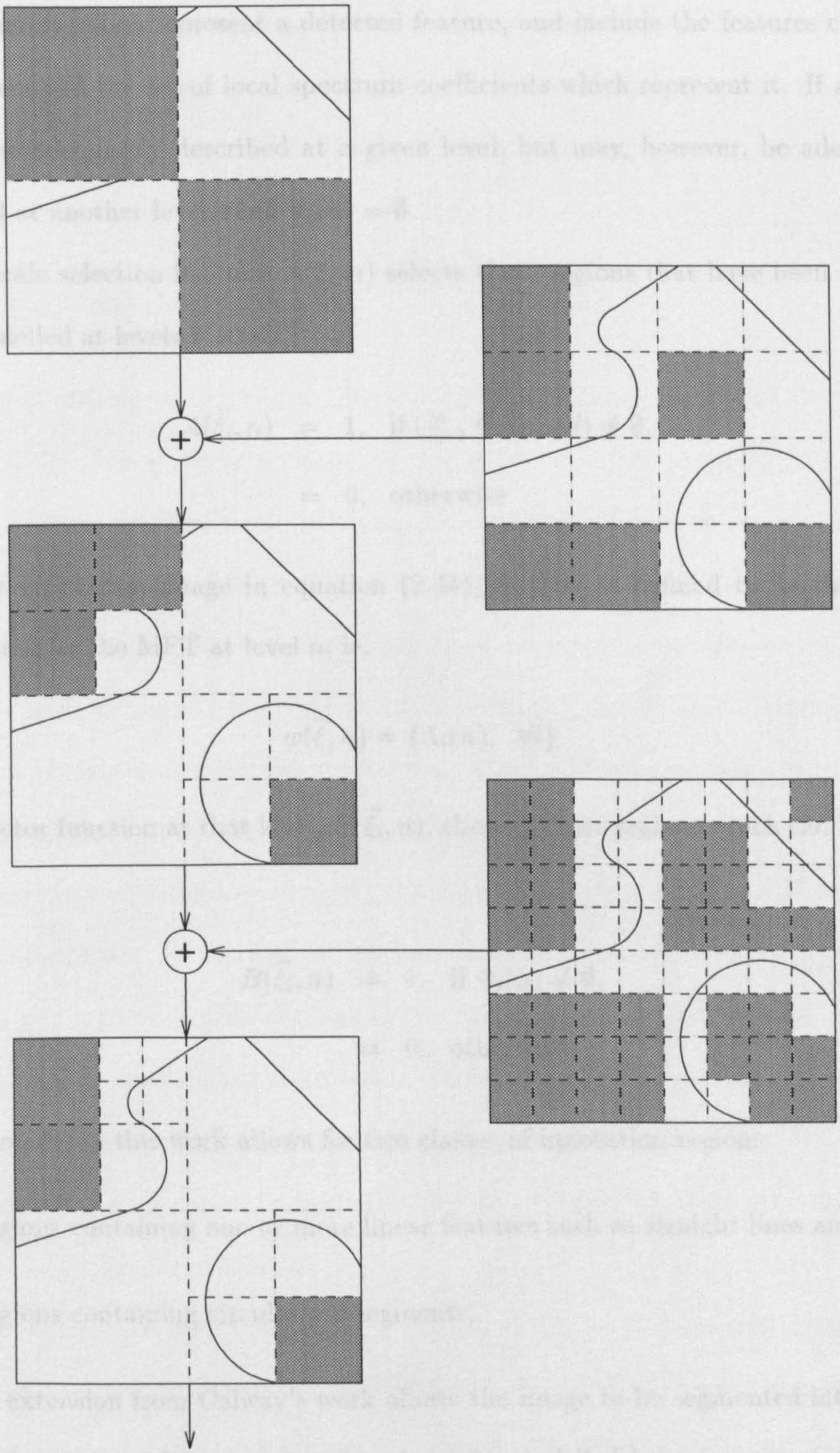


Figure 2.21: Image Model: Grey areas represent regions that cannot be modelled at a given level

of parameters which represent a detected feature, and include the features class, its parameters and the set of local spectrum coefficients which represent it. If a region cannot be adequately described at a given level, but may, however, be adequately modelled at another level, then $\Phi_i(n) = \emptyset$.

The scale selection function $A(\vec{\xi_i}, n)$ selects those regions that have been successfully modelled at levels $n' < n$.

$$\begin{aligned} A(\vec{\xi_i}, n) &= 1, \quad \text{if } \bigcup_{l=1}^{n} \Phi_{il}(n-l) \neq \emptyset, \\ &= 0, \quad \text{otherwise} \end{aligned} \tag{2.47}$$

If the innovation image in equation (2.44), $w(\vec{\xi}, n)$, is defined to be the image represented by the MFT at level n, ie.

$$w(\vec{\xi}, n) = \{\Lambda_i(n), \ \forall i\} \tag{2.48}$$

the detector function at that level, $B(\vec{\xi_i}, n)$, chooses those regions which can be modelled, ie.

$$\begin{aligned} B(\vec{\xi_i}, n) &= 1, \quad \text{if } \Phi_i(n) \neq \emptyset, \\ &= 0, \quad \text{otherwise} \end{aligned} \tag{2.49}$$

The model in this work allows for two classes of innovation region:

1. Regions containing one or more linear features such as straight lines and edges.

2. Regions containing circular arc segments.

This extension from Calway's work allows the image to be segmented into higher level regions, since the curves can be piecewise modelled by arc segments that are

larger than straight line segments performing the same task. The use of regions containing multiple linear features also allows for the modelling of corners within regions. As in Calway's work the MFT is the vehicle used in the estimation of model parameters and hypothesis testing[16].

2.4 Examples

This section contains a number of different MFT levels for three images. Figure 2.22 shows a simple 256×256 test image containing five geometric figures. The original figure was binary but has been slightly smoothed, using a low pass Gaussian filter, to reduce aliasing. Figures 2.23 to 2.25 show three levels of the MFT for the "Shapes" image, at levels 2 to 4 respectively. Each level corresponds to a windowed Fourier transform of the image with a window size related to the scale by equation (2.20).

The images show the MFT levels as sets of local spectra regularly spaced on the image plain. Each spectrum represents a region of the image about one of a grid of regularly spaced sample points. As the local spectra are complex, colour is used to represent phase. It can be clearly seen, especially at the smaller scales, that the energy is highest in those regions of the image which contain a large, relatively localised change in intensity - an edge. In these regions, the energy of the local spectra is concentrated about an orientation perpendicular to the edge. At the larger scales the spectra are generally more complex and spread out, since they correspond to larger, more complex regions.

The next example is a natural image. Figure 2.26 shows the original 256×256 grey scale "Lena" image. Figures 2.27 and 2.28 show the MFT of "Lena" at levels 3 and 4 respectively. Again it is clear, particularly at level 4, that the energy is greatest in

those regions in the neighbourhood of an edge, and that in these neighbourhoods the spectrum energy is concentrated in an orientation perpendicular to the edge orientation.

It is also clear that there is a lot of energy at the edge of the image, implying that there is a discontinuity at the image boundary. This boundary discontinuity arises from the periodic nature of the discrete Fourier transform (DFT) used in the generation of the MFT (see [70] or [13]). When generating the DFT the image is regarded as a torus with the top edge joined to the bottom, and the left edge joined to the right. Therefore the difference in grey level between the top and bottom, or left and right, image boundaries will effectively be an edge. This edge effect is further enhanced by centering the local spectra between the four lowpass coefficients when generating the MFT. In order to perform this centering, the image is pre-multiplied by a complex exponential in order to shift the spectrum by half a sample (shifting theorem [13]), ie.

$$x'(i,j) = x(i,j)\exp\left[-\jmath\frac{\pi(i+j)}{N}\right], \quad i,j = 0,1,...,N-1 \tag{2.50}$$

The effect of this multiplication may be seen by considering for a given j that the left edge ($i = 0$) is given by

$$x'(0,j) = x(0,j)\exp\left[-\jmath\frac{j\pi}{N}\right] \tag{2.51}$$

while the right edge is given by

$$\begin{aligned} x'(N-1,j) &= x(N-1,j)\exp\left[-\jmath\frac{\pi(N-1+j)}{N}\right] && (2.52)\\ &= x(N-1,j)\exp\left[-\jmath\frac{(j-1)\pi}{N}\right]\exp\left[-\jmath\pi\right] && (2.53)\\ &= -x(N-1,j)\exp\left[-\jmath\frac{(j-1)\pi}{N}\right] && (2.54) \end{aligned}$$

Although there is a difference in phase (possibly quite small) due to the exponential term in equation (2.54), there is an additional π phase shift indicated by the negative. If both edges in the original image are the same grey level, after this pre-multiplication, one edge will be approximately the negative of the other, causing a large discontinuity at the edge of the image. The size of this discontinuity obviously depends upon the grey level at the image edges and as such the effect was not noticeable for the shapes image, which was on a black (zero intensity) background. In order to reduce this effect the mean grey level of the image was removed from the "Lena" image before the MFT was computed. This serves to lessen the edge effect, although it does not entirely remove it.

Figure 2.29 shows a third test image - a picture of a vine leaf on a slowly varying grey background. Figures 2.30 and 2.31 show levels 3 and 4 of its MFT respectively. In order to reduce the size of the edge discontinuities, the mean grey level of the image was removed before the MFT was computed, but the effects of this on the local spectra near the edge of the image can still be seen.

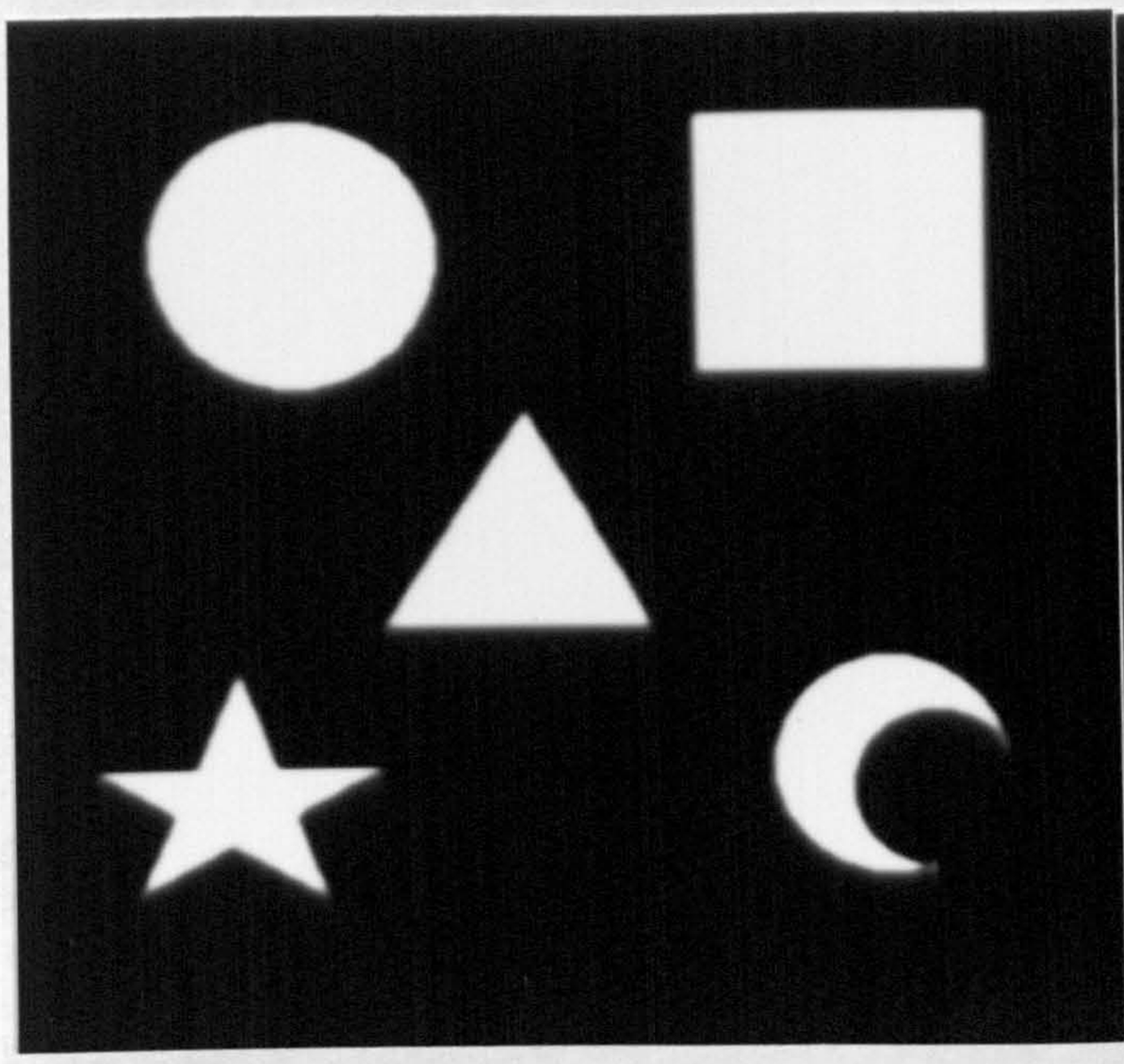


Figure 2.22: Shapes: Original 256×256 image

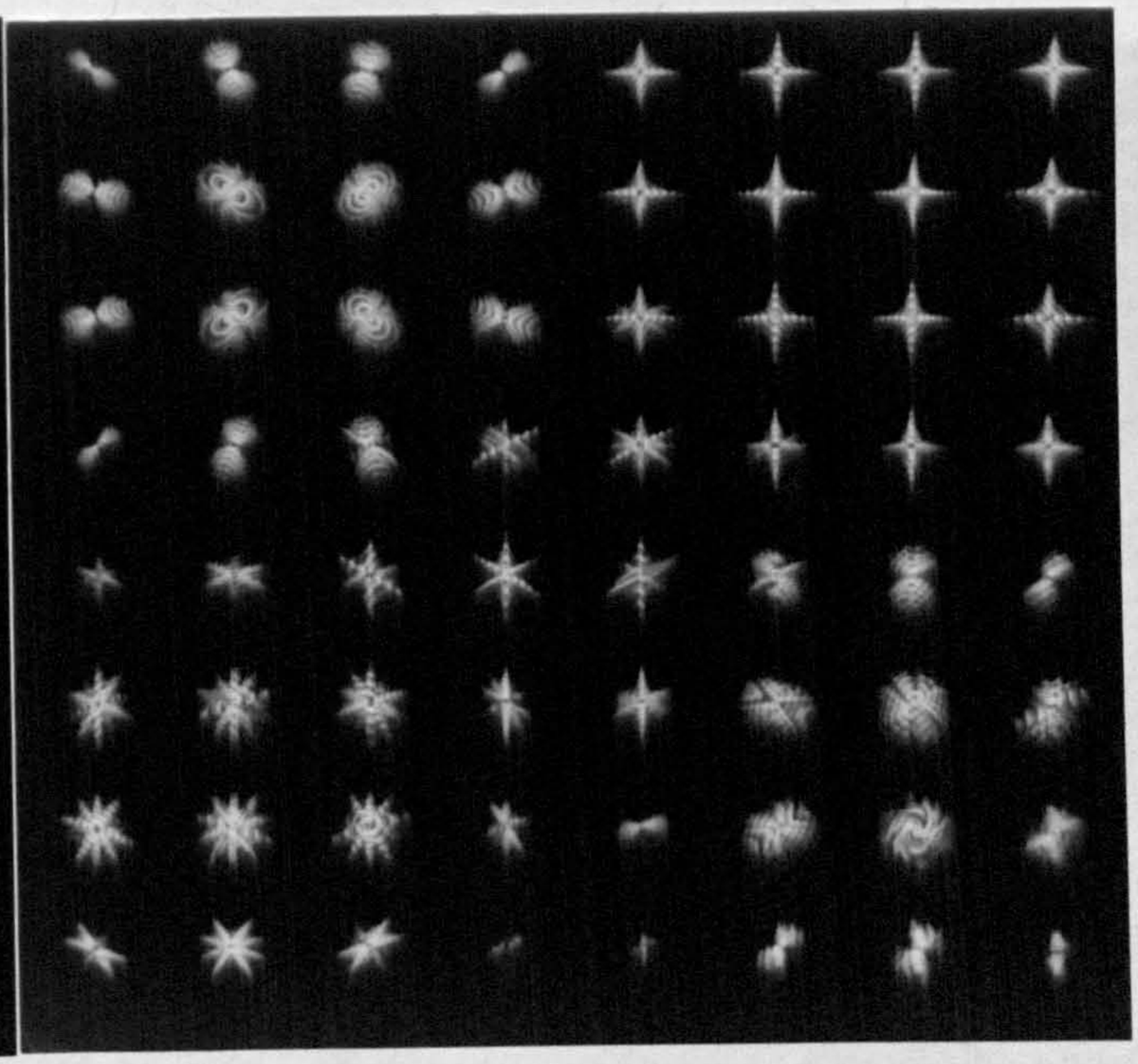


Figure 2.23: Shapes: MFT Level 2

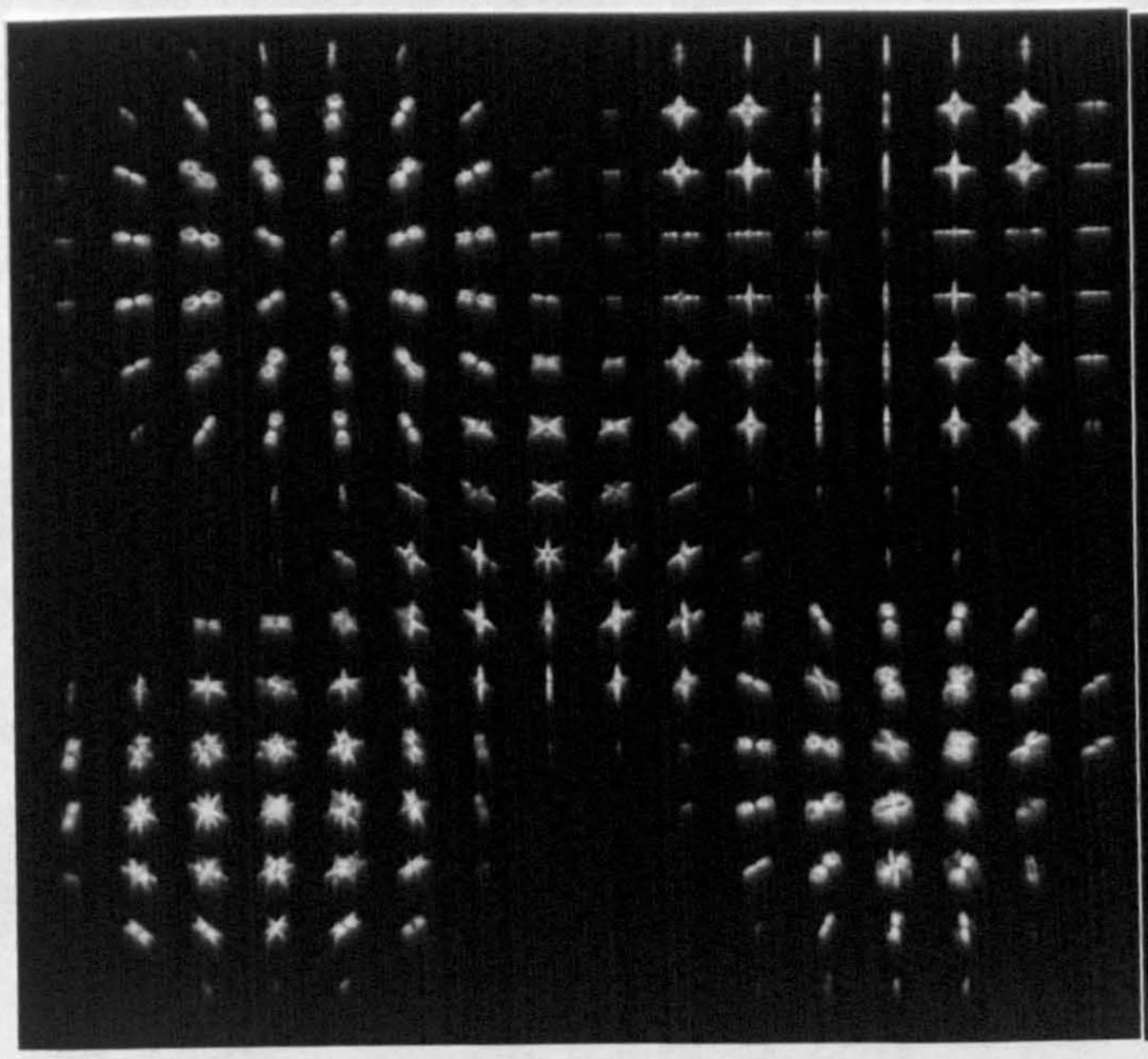


Figure 2.24: Shapes: MFT Level 3

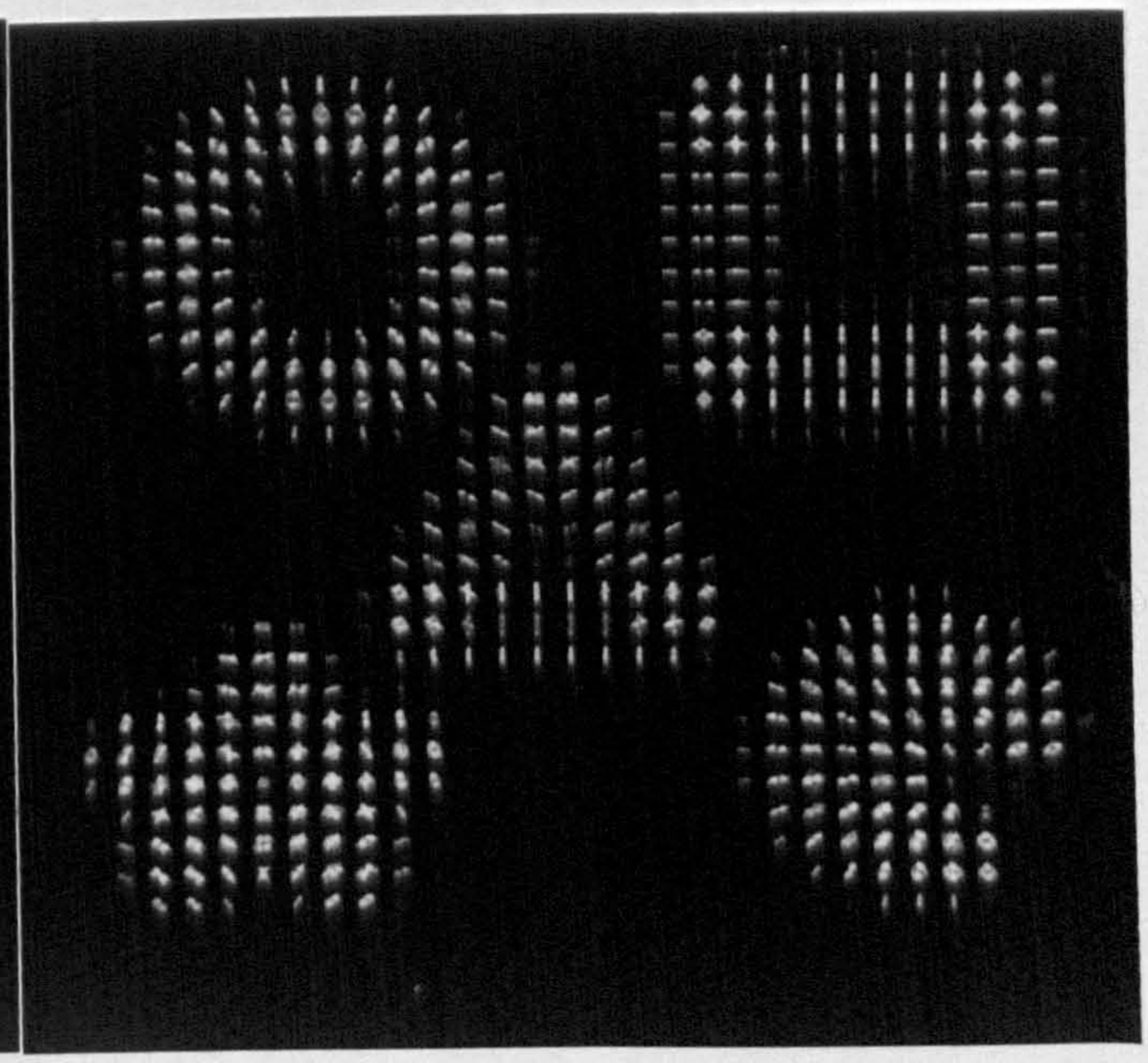


Figure 2.25: Shapes: MFT Level 4

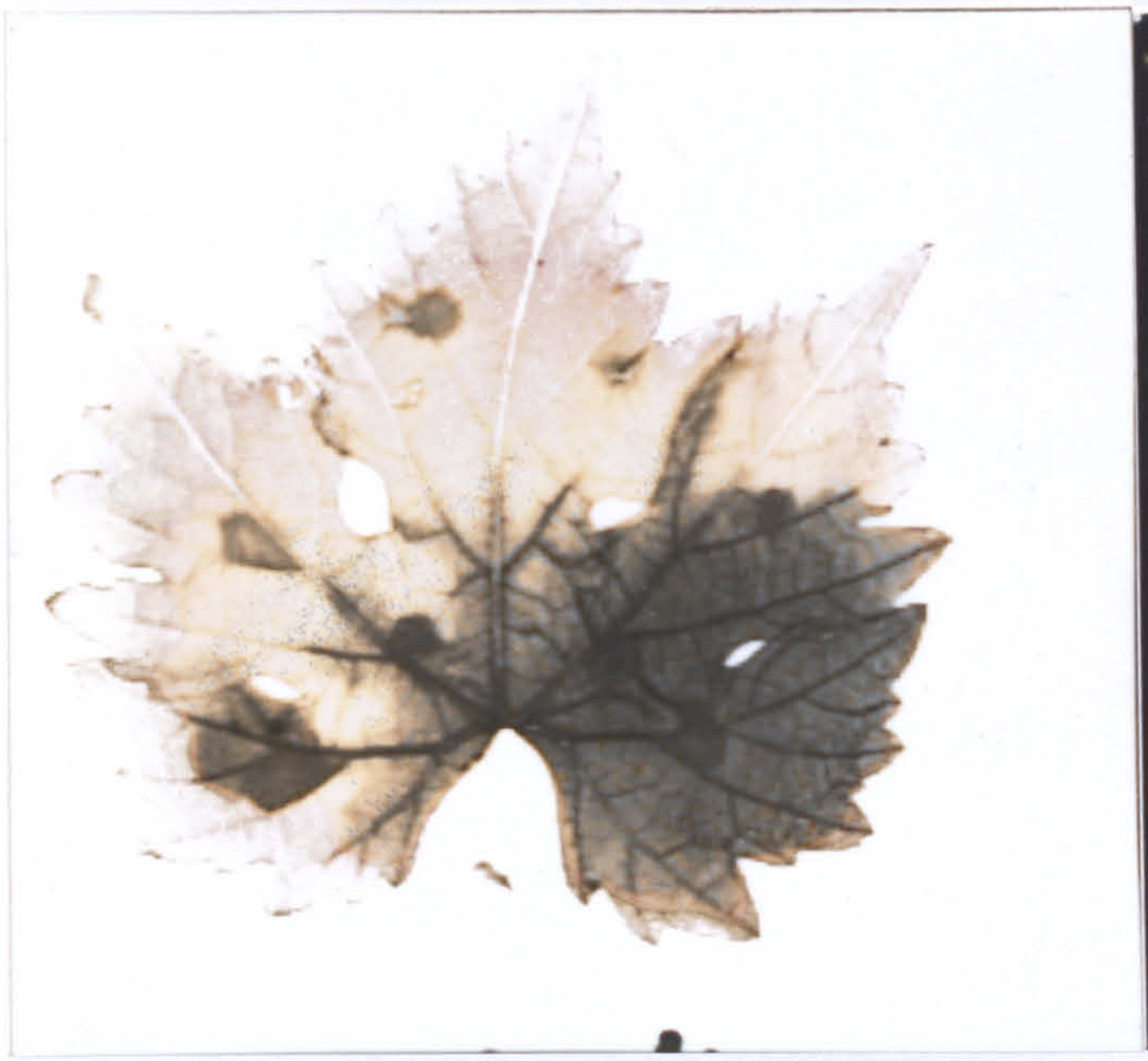


Figure 2.29: Vineleaf: Original 256 × 256 image

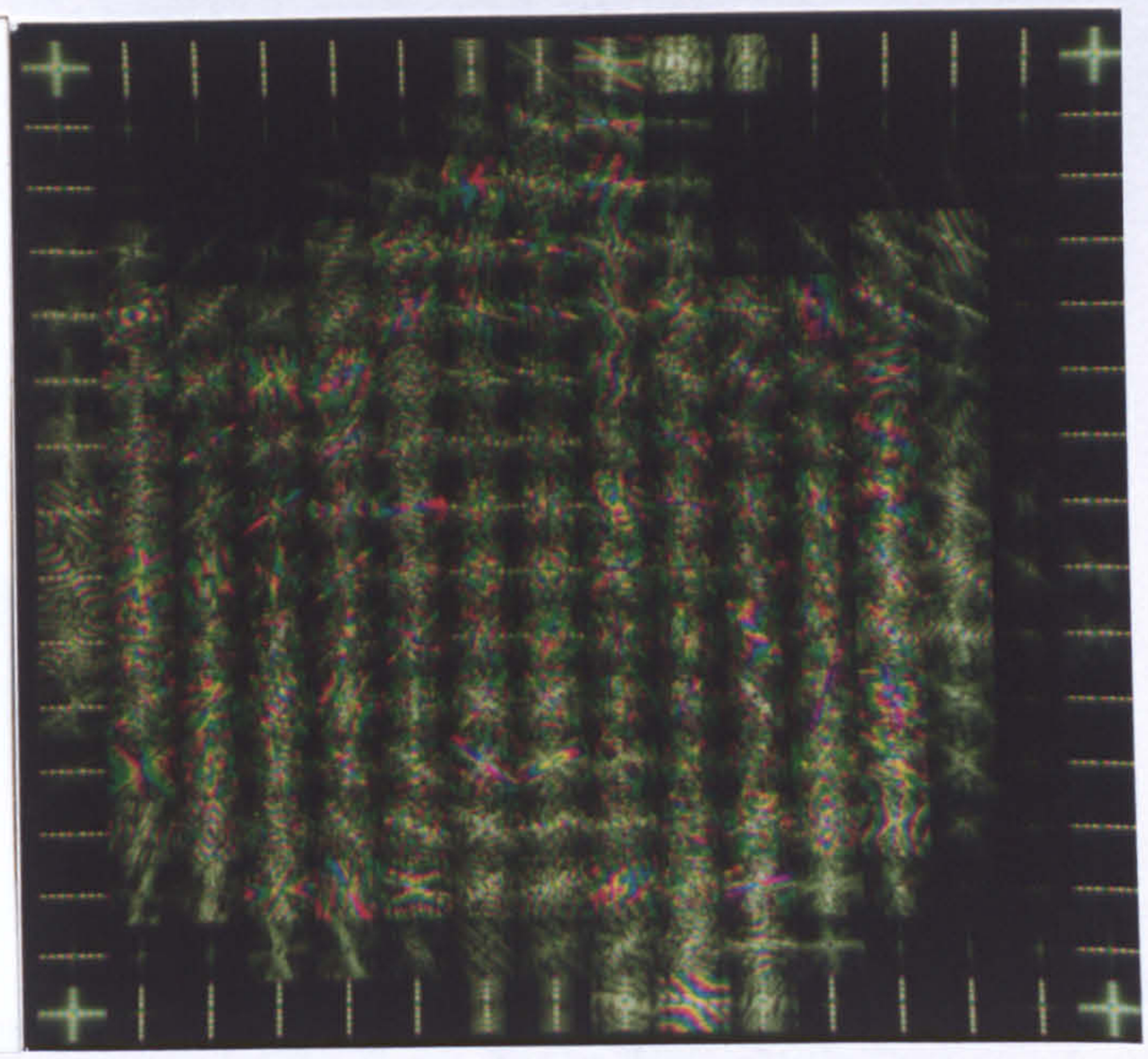


Figure 2.30: Vineleaf: MFT Level 3

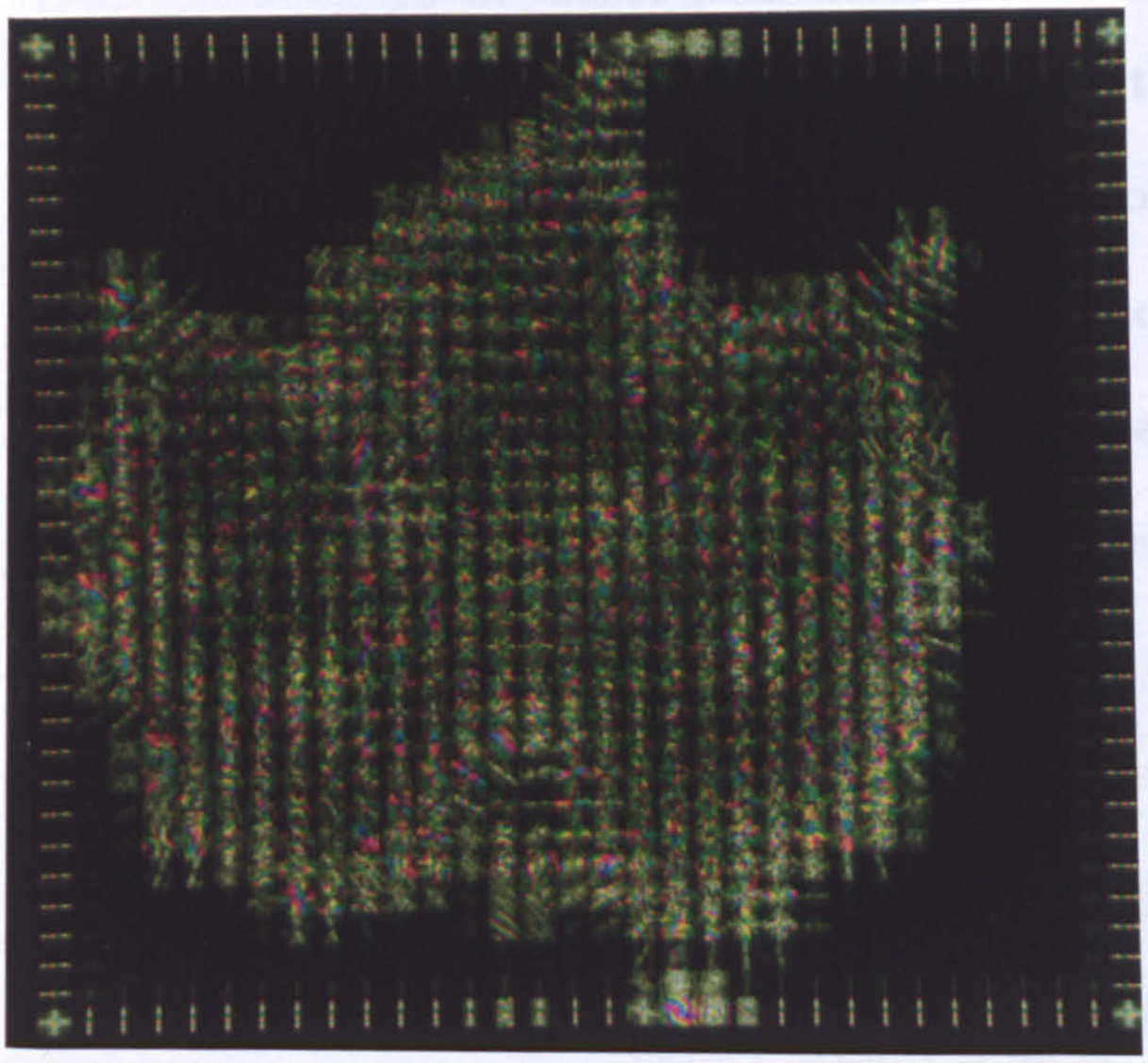


Figure 2.31: Vineleaf: MFT Level 4

Figure 2.26: Lena: Original 256 × 256 image

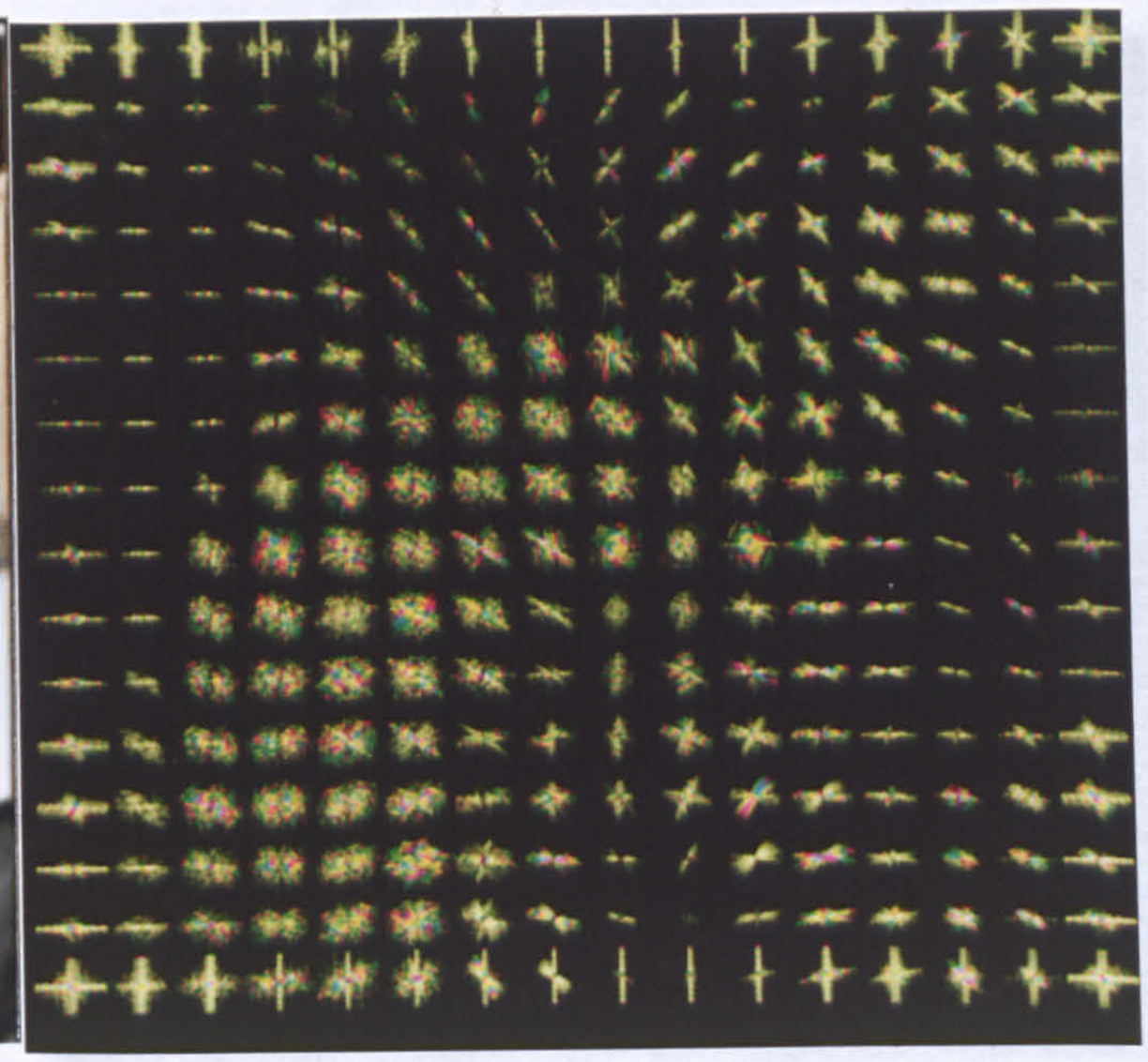


Figure 2.27: Lena: MFT Level 3

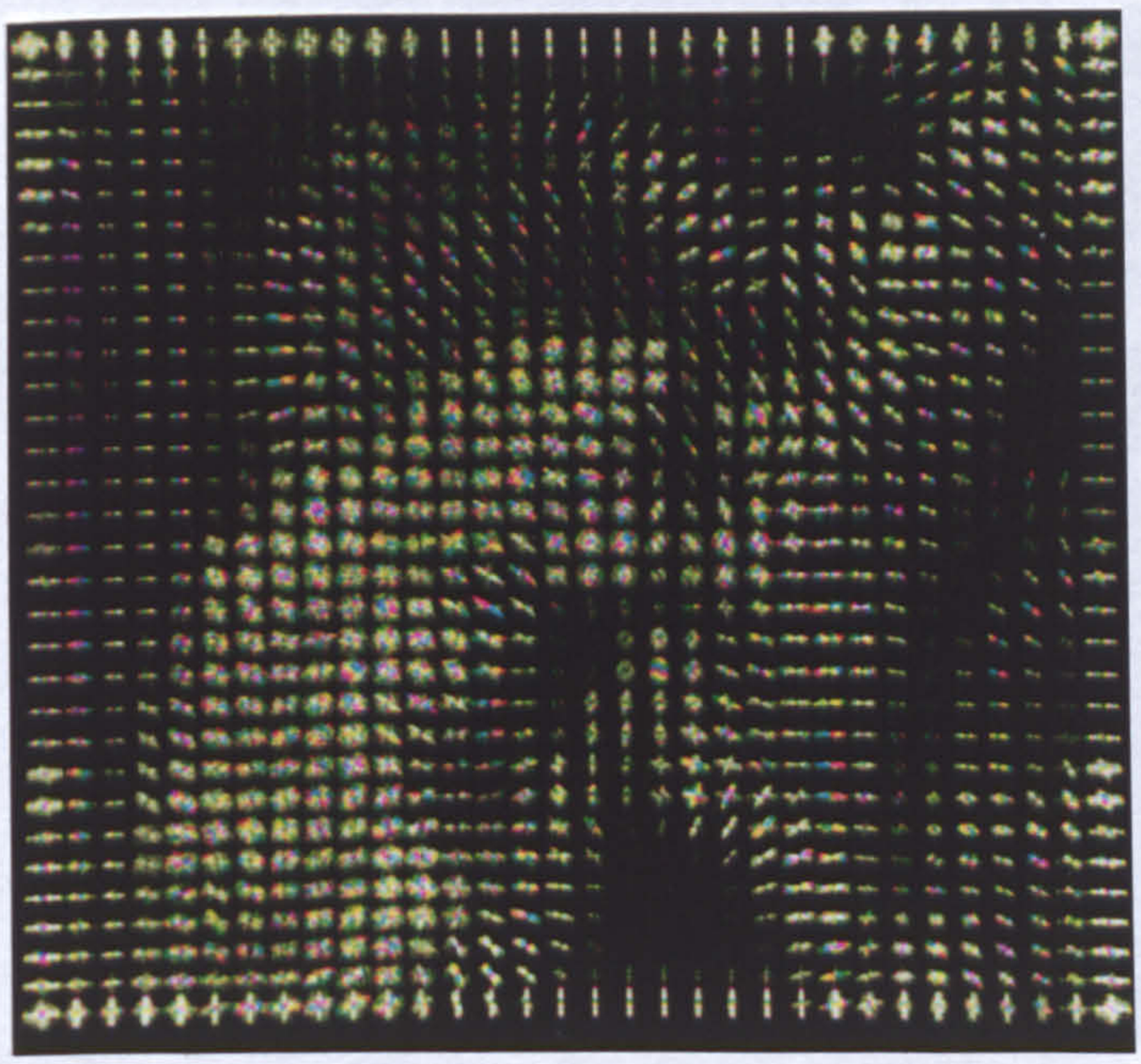


Figure 2.28: Lena: MFT Level 4

Chapter 3

Local Image Features

3.1 Frequency Domain Feature Modelling

The local feature models derived in this section are based upon the characteristics of local spectra. Such an approach models an image region directly using a representation of that image region - its local spectrum - and therefore allows for the feature to be detected, and its parameters estimated, from that representation. Thus the feature detection is a single stage process, unlike the two step approach of an edge detection [82, 20] followed by some form of line following [4] or Hough transform [33, 50, 77].

The feature models are primarily based upon the phase behaviour of local spectrum, which has been shown to be important in representing the spatial organisation of an image. Indeed Oppenheim and Lim [69] have shown that under certain constraints it is possible to reconstruct an image from its Fourier phase.

The original work of Calway has been extended in two main ways. First, multiple linear features are modelled in a similar way to the simple linear features of Calway's work. This requires a more complex hypothesis test, but has the considerable advantage that it can deal with the 'junction points' caused by occlusions in images of 3-d scenes, which are an important cue to 3-d structure [64].

Secondly, the inclusion of a circular arc model allows boundaries of high curvature to be modelled at a larger scale than that which can be achieved with piecewise linear approximations. Since most natural objects have curved boundaries, this has an obvious attraction in terms of the expressiveness of the model, without a large corresponding increase in computation.

As in Calway's work, the fundamental principle of identifying the largest scale at which one of the models fits is retained. A method of combining estimates from a range of scales in the framework of a multiresolution image model will be presented in chapter 5. In this chapter, the emphasis is on the modelling and associated estimation at a given scale.

3.1.1 Linear Features

An ideal linear feature of infinite length can be regarded as a 1-dimensional feature, since there is only variation orthogonal to the feature orientation. After Calway [16], such a feature in the continuous spatial domain can be represented by

$$x(\vec{\xi}) = x(\xi_1 \cos\theta + \xi_2 \sin\theta) \tag{3.1}$$

or, by defining an orientation vector

$$\vec{v}_\theta = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \tag{3.2}$$

it can be written in vector form, and shown to be dependent upon only the inner product between the position vector and the orientation vector

$$x(\vec{\xi}) = x(\vec{\xi} \cdot \vec{v}_\theta) \tag{3.3}$$

The one dimensional function $x(\cdot)$ represents the profile of the feature.

The Fourier spectrum of such a region is concentrated in an orientation perpendicular to feature orientation

$$x(\vec{\xi}) \leftrightarrow \hat{x}(\vec{\omega}) = \hat{x}(\vec{\omega} \cdot \vec{v}_\theta)\delta(\vec{\omega} \cdot \vec{v}_{\theta_\perp}) \tag{3.4}$$

where '$\leftrightarrow$' denotes 'transforms to', and $\vec{v}_{\theta_\perp}$ is a unit vector perpendicular to $\vec{v}_\theta$. The position of this feature with respect to the origin of the image plane is related linearly to the phase of the spectrum[70]. Denoting the phase by $\phi(\vec{\omega})$, the spectrum can be written in the form

$$\hat{x}(\vec{\omega}) = |\hat{x}(\vec{\omega})| \exp[-\jmath\phi(\vec{\omega})] \tag{3.5}$$

For a spectrum corresponding to a single linear feature

$$\phi(\vec{\omega}) = \phi(\vec{\omega} \cdot \vec{v}_\theta) = \eta\vec{\omega} \cdot \vec{v}_\theta + \varepsilon(\vec{\omega} \cdot \vec{v}_\theta) \tag{3.6}$$

where η is the centroid of the 1-d function $x(\xi)$ and $\varepsilon(\omega)$ is a function dependent upon the profile of the edge in $x(\xi)$. In the cases of interest for the current work, assuming that there will be a discontinuity of the feature at $\xi = \xi_0$, $x(\xi)$ will be of one of two forms:

1. If $x(\xi)$ corresponds to a line feature it is real and even in the spatial domain, ie.

$$x(\xi - \xi_0) = x(-(\xi - \xi_0)) \tag{3.7}$$

then (see Bracewell[13])

$$\varepsilon(\omega) = 0 \tag{3.8}$$

2. If $x(\xi)$ corresponds to an edge feature it is real and odd, ie.

$$x(\xi - \xi_0) - \overline{x(\xi)} = -(x(-(\xi - \xi_0)) - \overline{x(\xi)}) \tag{3.9}$$

where $\overline{x(\xi)}$ is the mean value of the function, then

$$\varepsilon(\omega) = \pi/2, \text{if } \omega < 0 \tag{3.10}$$

$$= -\pi/2, \text{if } \omega > 0 \tag{3.11}$$

In practice, such features will be local and confined to regions of the image plane and hence not be of infinite length, as assumed in the above case. Again following Calway, a local feature of finite extent can be considered as a windowed infinite length feature. Such a local feature may be expressed by

$$x(\vec{\xi}) = x(\vec{\xi} \cdot \vec{v}_\theta) w(\vec{\xi} \cdot \vec{v}_{\theta_\perp}) \tag{3.12}$$

where $w(\xi)$ is a window function. This gives in the frequency domain[70]

$$\hat{x}(\vec{\omega}) = \hat{x}(\vec{\omega} \cdot \vec{v}_\theta) \hat{w}(\vec{\omega} \cdot \vec{v}_{\theta_\perp}) \tag{3.13}$$

where $\hat{w}(\omega)$ represents the spreading of the signal in the frequency domain caused by the windowing in the spatial domain. The phase of this function is related to the centroid of $w(\xi)$ by

$$\arg(\hat{w}(\omega)) = \phi'(\vec{\omega} \cdot \vec{v}_{\theta_\perp}) \tag{3.14}$$

$$= \eta' \vec{\omega} \cdot \vec{v}_{\theta_\perp} + \varepsilon'(\vec{\omega} \cdot \vec{v}_{\theta_\perp}) \tag{3.15}$$

If it is assumed that the window function is real and even, then $\varepsilon'(\omega) = 0$.

The phase for the spectrum of the windowed feature can therefore be written in the form

$$\phi(\vec{\omega}) = \eta \vec{\omega} \cdot \vec{v}_\theta + \eta' \vec{\omega} \cdot \vec{v}_{\theta_\perp} + \varepsilon(\vec{\omega} \cdot \vec{v}_\theta) \tag{3.16}$$

$$= \vec{\omega} \cdot (\eta \vec{v}_\theta + \eta' \vec{v}_{\theta_\perp}) + \varepsilon(\vec{\omega} \cdot \vec{v}_\theta) \tag{3.17}$$

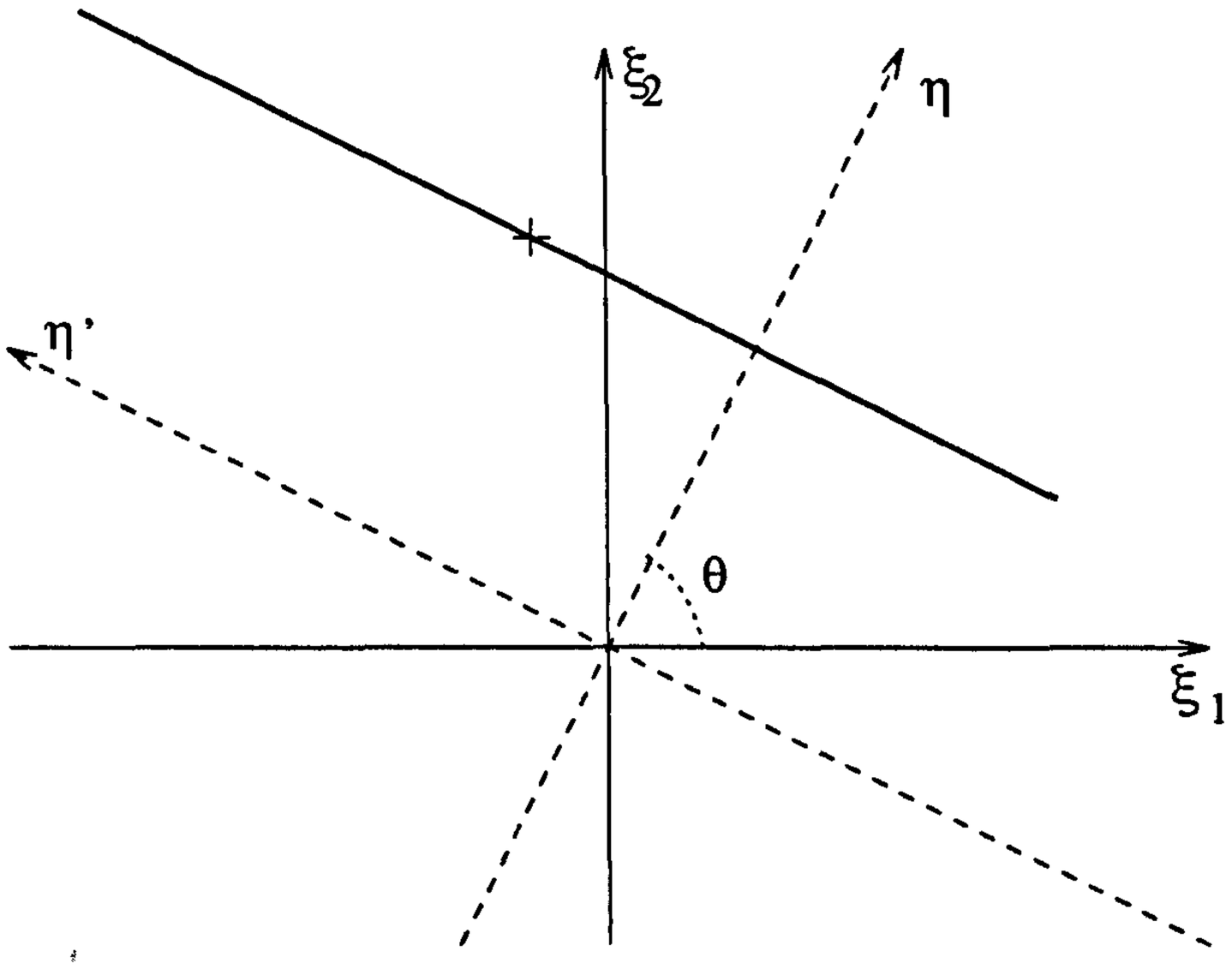


Figure 3.1: Linear feature centroid

Defining a centroid vector $\vec{\eta}$ by

$$\vec{\eta} = \begin{bmatrix} \eta \\ \eta' \end{bmatrix} \tag{3.18}$$

this gives the centroid relative to a pair of axes parallel and perpendicular to the feature orientation, as shown in figure 3.1. Transforming this to coordinates relative to the $\vec{\xi}$-axes, the phase is given by

$$\phi(\vec{\omega}) = \vec{\xi_0} \cdot \vec{\omega} + \varepsilon(\vec{\omega} \cdot \vec{v}_\theta) \tag{3.19}$$

where $\vec{\xi_0}$ is the centroid vector for the feature, given by

$$\vec{\xi_0} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \vec{\eta} \tag{3.20}$$

$$= \mathbf{R}_{(-\theta)}\vec{\eta} \tag{3.21}$$

If the image region contains more than one feature, the image spectrum can be modelled as the sum of a set of component spectra - one for each component feature.

Assuming that there are K features, and that the component spectrum due to the k-th feature is $\hat{x}_k(\vec{\omega})$, the local spectrum is given by

$$\hat{x}(\vec{\omega}) = \sum_{k=0}^{K-1} \hat{x}_k(\vec{\omega}) \tag{3.22}$$

where each component spectrum can be modelled by the equations

$$\hat{x}_k(\vec{\omega}) = |\hat{x}_k(\vec{\omega})| \exp\left[-\jmath\phi_k(\vec{\omega})\right] \tag{3.23}$$

and

$$\phi_k(\vec{\omega}) = \vec{\xi}_{k0} \cdot \vec{\omega} + \varepsilon_k(\vec{\omega} \cdot \vec{v}_\theta) \tag{3.24}$$

3.1.2 Arc Features

Regions containing arc features can be modelled in a similar, phase based way to that used for linear features. For an ideal linear feature, the phase of its spectrum, along an axis perpendicular to the feature orientation, is determined by the centroid of the feature, as given in equation 3.19. Similarly, it can be shown that the phase of the spectrum corresponding to a circular arc feature, at a given frequency, $\vec{\omega}$, is also determined by position of the section of arc in the orientation $\angle\vec{\omega}$.

An arc centered on $\vec{\xi}_0$ with radius R, can be represented as the locus of the point $\vec{\xi}_\theta$, given by the equation

$$\vec{\xi}_\theta = R\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} + \vec{\xi}_0 \tag{3.25}$$

where θ is the orientation of the radial to the arc at $\vec{\xi}_\theta$, as illustrated by figure 3.2.

Defining a pair of rotated coordinates (t, u), given by

$$\begin{bmatrix} t \\ u \end{bmatrix} = \mathbf{R}_\theta\vec{\xi}, \tag{3.26}$$

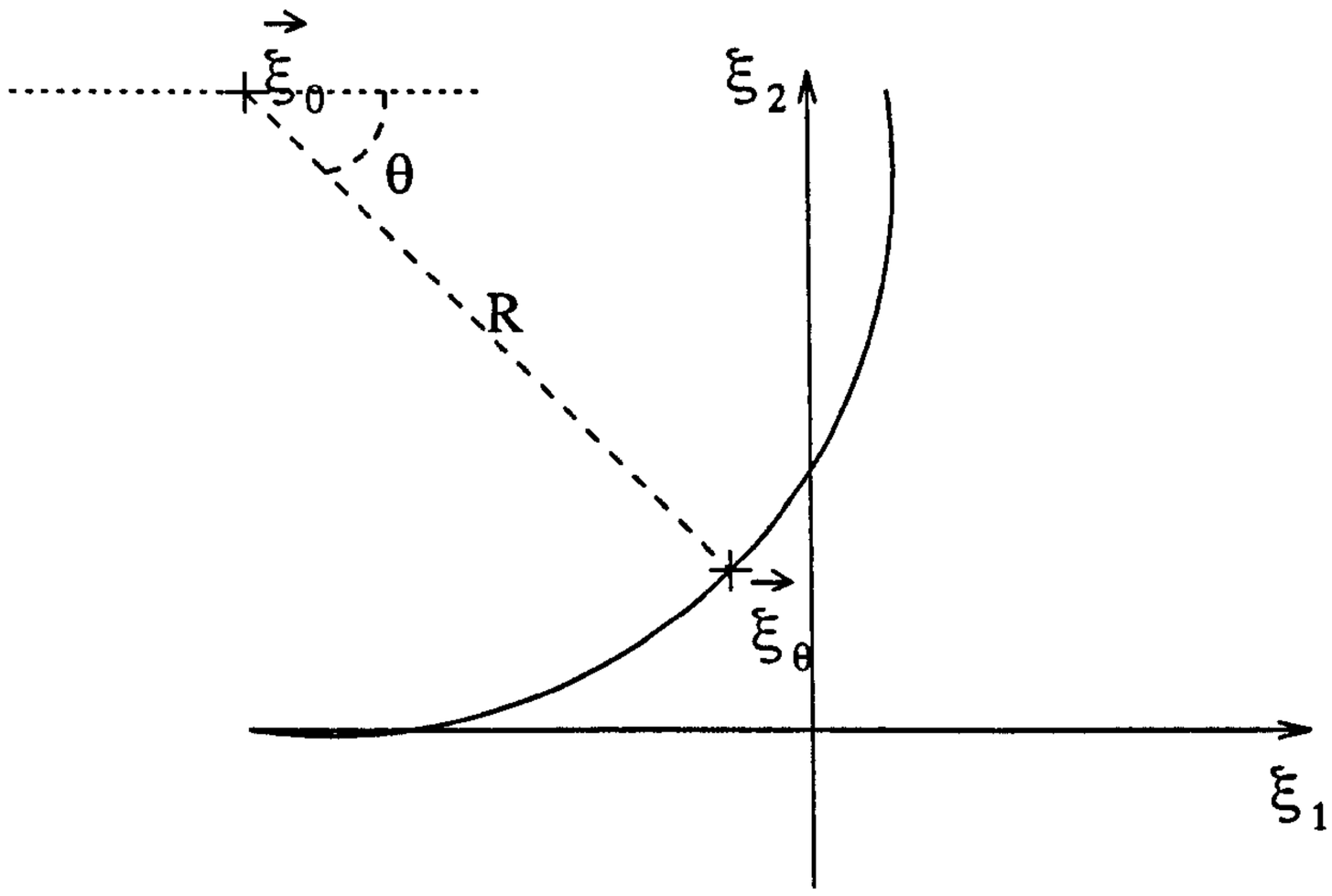


Figure 3.2: Arc

figure 3.3 shows the projection of an arc feature onto the t-axis, where the projection is given by the expression

$$x_\theta(t) = \int_{u=-\infty}^{+\infty} x(t,u)du \tag{3.27}$$

If it is assumed that the only significant feature is the arc feature, and that there is only one point on the arc in any given orientation, then the projection may be approximated by a step edge at $t = t_0$, where t_0 is the distance from the u-axis of the point on the arc with tangent orientation θ, $\vec{\xi_\theta}$. From equation (3.26), t_0 is given by

$$t_0 = \xi_{\theta 1}\cos\theta + \xi_{\theta 2}\sin\theta \tag{3.28}$$

Substituting for $\vec{\xi_\theta}$ from equation (3.25), gives

$$t_0 = R(\cos\theta\cos\theta + \sin\theta\sin\theta) + \xi_{01}\cos\theta + \xi_{02}\sin\theta \tag{3.29}$$

$$= R + \vec{\xi_0}\cdot\vec{v_\theta} \tag{3.30}$$

The Fourier transform of $x_\theta(t)$, $\hat{x}_\theta(\omega)$, is given by

$$\hat{x}_\theta(\omega) = |\hat{x}_\theta(\omega)|\exp[-\jmath t_0\omega] \tag{3.31}$$

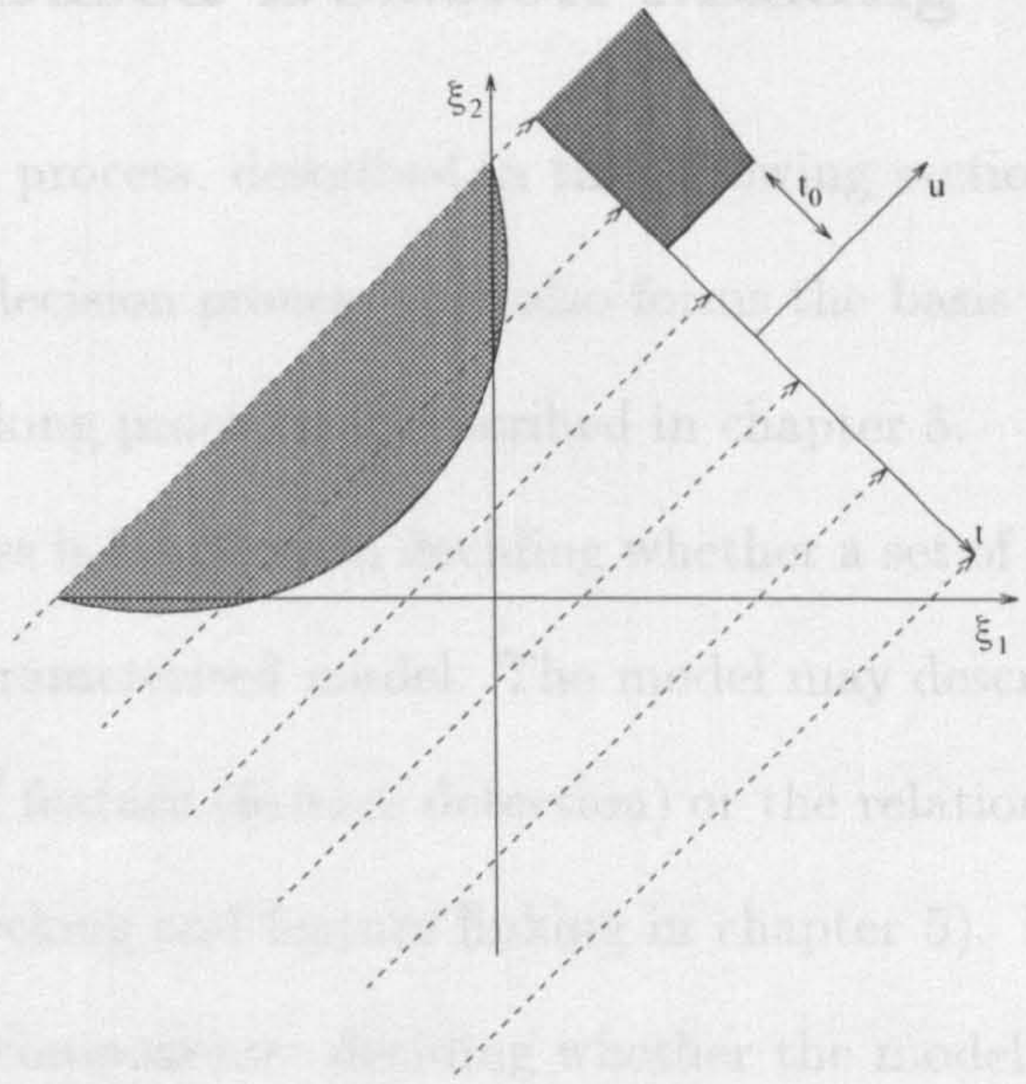


Figure 3.3: Arc projection onto t-axis

$$= |\hat{x}_\theta(\omega)| \exp\left[-\jmath(R + \vec{\xi}_0 \cdot \vec{v}_\theta)\omega\right] \quad (3.32)$$

From the projection slice theorem [60], the 2-d Fourier transform of the arc may be related to the transform of the projection by

$$\hat{x}(\vec{\omega}) = \hat{x}(\|\vec{\omega}\| \cos(\angle\vec{\omega}), \|\vec{\omega}\| \sin(\angle\vec{\omega})) \quad (3.33)$$

$$= \hat{x}_{\angle\vec{\omega}}(\|\vec{\omega}\|) \quad (3.34)$$

Substituting $\theta = \angle\vec{\omega}$ and $\omega = \|\vec{\omega}\|$ into equation (3.31), gives

$$\hat{x}(\vec{\omega}) = |\hat{x}_{\angle\vec{\omega}}(\|\vec{\omega}\|)| \exp\left[-\jmath(R + \vec{\xi}_0 \cdot \vec{v}_{\angle\vec{\omega}})\|\vec{\omega}\|\right] \quad (3.35)$$

$$= |\hat{x}_{\angle\vec{\omega}}(\|\vec{\omega}\|)| \exp\left[-\jmath(R\|\vec{\omega}\| + \vec{\xi}_0 \cdot \vec{\omega})\right] \quad (3.36)$$

The phase behaviour of the local spectrum for a region containing a circular arc contour has thus been shown to be dependent upon the radius and centre of the arc, as in (3.36).

3.2 Model Based Decision Making

The feature detection process, described in the following section, is an instance of a general model-based decision process that also forms the basis of the image segmentation and feature linking procedures described in chapter 5.

The decision process is based upon deciding whether a set of spatial regions can be described by some parameterised model. The model may describe a single region in terms of some class of feature (feature detection) or the relationship between regions (scale consistency checking and feature linking in chapter 5). In each case the decision process has two components: deciding whether the model can fit the data, and estimating the model's parameters. The process can be regarded as a form of composite hypothesis testing [30, 66]. A choice must be made between the hypothesis H_1 that the data can be modelled and the null hypothesis H_0 that it cannot. Hypothesis H_1 is composite, because it represents a set of possibilities - that the data can be modelled with a vector of parameters from the space of all possible parameters for the particular model.

An approach to deciding between two complex hypotheses is given by the generalised maximum-likelihood test (eg. Davenport and Root [30]). The process is as follows: Assuming hypothesis H_1 is true, a maximum-likelihood estimate is made for the model parameters. These estimates are used to determine the probability density under hypothesis H_1. A similar procedure is applied to hypothesis H_0. The two densities can be used to determine a likelihood ratio and decide between the hypotheses. If the density under hypothesis H_i (for $i = 0, 1$) is a function of an observed data vector $\mathbf{x}$ and parameter vector $\boldsymbol{\phi}_i$, and denoted $p_i(\mathbf{x}, \boldsymbol{\phi}_i)$, the generalised likelihood

ratio is given by

$$l(\mathbf{x}) = \frac{\max_{\phi_0} p_0(\mathbf{x}, \phi_0)}{\max_{\phi_1} p_1(\mathbf{x}, \phi_1)} \tag{3.37}$$

The generalised likelihood ratio test is therefore

$$\text{Accept } H_0 \text{ if } l(\mathbf{x}) > \eta$$

$$\text{Accept } H_1 \text{ if } l(\mathbf{x}) \leq \eta$$

It is clear that the estimation of model parameters is integral to the decision process, since the estimated parameters are required to make the decision.

In terms of modelling an image region, the choice between the two hypotheses, H_0 and H_1, may be regarded as deciding whether a region can be modelled using a given feature class or not. The null hypothesis H_0 is that there is no signal present, and that the observed signal is simply noise, ie.

$$x(\vec{\xi}) = w(\vec{\xi}) \tag{3.38}$$

and H_1 may be regarded as the hypothesis that

$$x(\vec{\xi}) = \beta s_1(\vec{\xi}, \phi_1) + w(\vec{\xi}) \tag{3.39}$$

where $x(\vec{\xi})$ is the two dimensional function that represents the region, ie. is the observed signal, and $s_1(\vec{\xi}, \phi_1)$ is a signal that fits the model with a set of parameters ϕ_1 and an amplitude $\beta > 0$.

It can be shown[30] that the likelihood ratio can be interpreted in terms of correlation. If it is assumed that ϕ_1 is known, the generalised likelihood ratio, based upon an ML estimate of the amplitude β, is given by

$$\log l(x) = c_1 \left[\int_{\vec{\xi}} g_1(\vec{\xi}, \phi_1) x(\vec{\xi}) d\vec{\xi} \right]^2 \tag{3.40}$$

where c_1 is a constant dependent upon the magnitude of $s_1(\vec{\xi}, \phi_1)$ and $g_1(\vec{\xi}, \phi_1)$ is the impulse response of a filter matched to the modelled signal $s_1(\vec{\xi}, \phi_1)$. Since the actual model parameters are unknown, an estimated parameter set, $\tilde{\phi}$, is used to calculate the ratio. Although this method relies on ML estimation to maximise the densities, it should be noted that not all of the estimation processes in the current work meet this criterion. If the noise is assumed to be white, with energy E_w, and $s_1(\vec{\xi}, \tilde{\phi})$ is normalised so that $c_1 = 1$, equation (3.40) reduces to

$$\log l(x) = \left[\frac{1}{E_w} \int_{\vec{\xi}} x(\vec{\xi}) s_1^*(\vec{\xi}, \tilde{\phi}) d\vec{\xi}\right]^2 \tag{3.41}$$

The decision rule may therefore be expressed as

$$\begin{aligned} &\text{Accept } H_1 \quad \text{if} \quad \left| \int_{\vec{\xi}} x(\vec{\xi}) s_1^*(\vec{\xi}, \tilde{\phi}) dx \right| > \eta_1(E_w) \\ &\text{Accept } H_0 \qquad \text{otherwise} \end{aligned}$$

Effectively the test is based on measuring how closely the modelled signal, generated using estimated parameters, fits the observed data. The threshold, $\eta_1(E_w)$, which is dependent upon the noise energy, determines how close this fit must be for acceptance of the model.

This suggests a general three step algorithm for performing the decision, which is shown in figure (3.4).

1. **Estimation** For a given model, assume that it may be used to represent the data set. Estimate the model parameters for this case.

2. **Synthesis** Use the set of estimated parameters in the model to synthesise a data set with those parameters.

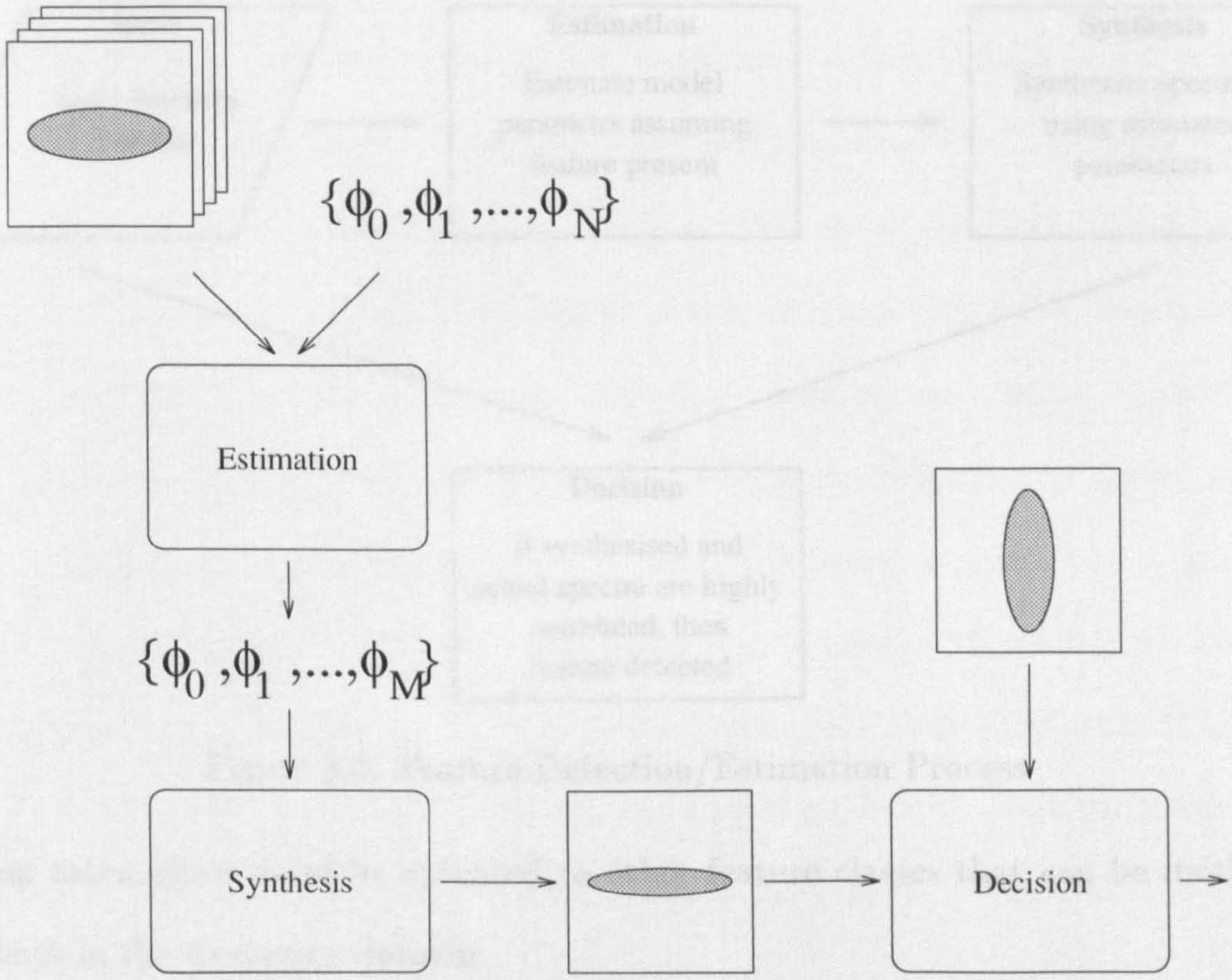


Figure 3.4: Estimation, Synthesis and Decision

3. **Decision** Use correlation to measure the similarity of the actual data to the synthesised data set. A decision can then be made as to whether the model fits the data.

The following section discusses the detection of the arc and linear edge features using this form of detection/estimation procedure.

3.3 Feature Detection

The local feature detection process is based on the feature models previously described. Although the present work is concerned with two classes of local image features - circular arcs and linear features - the general framework in which this de-

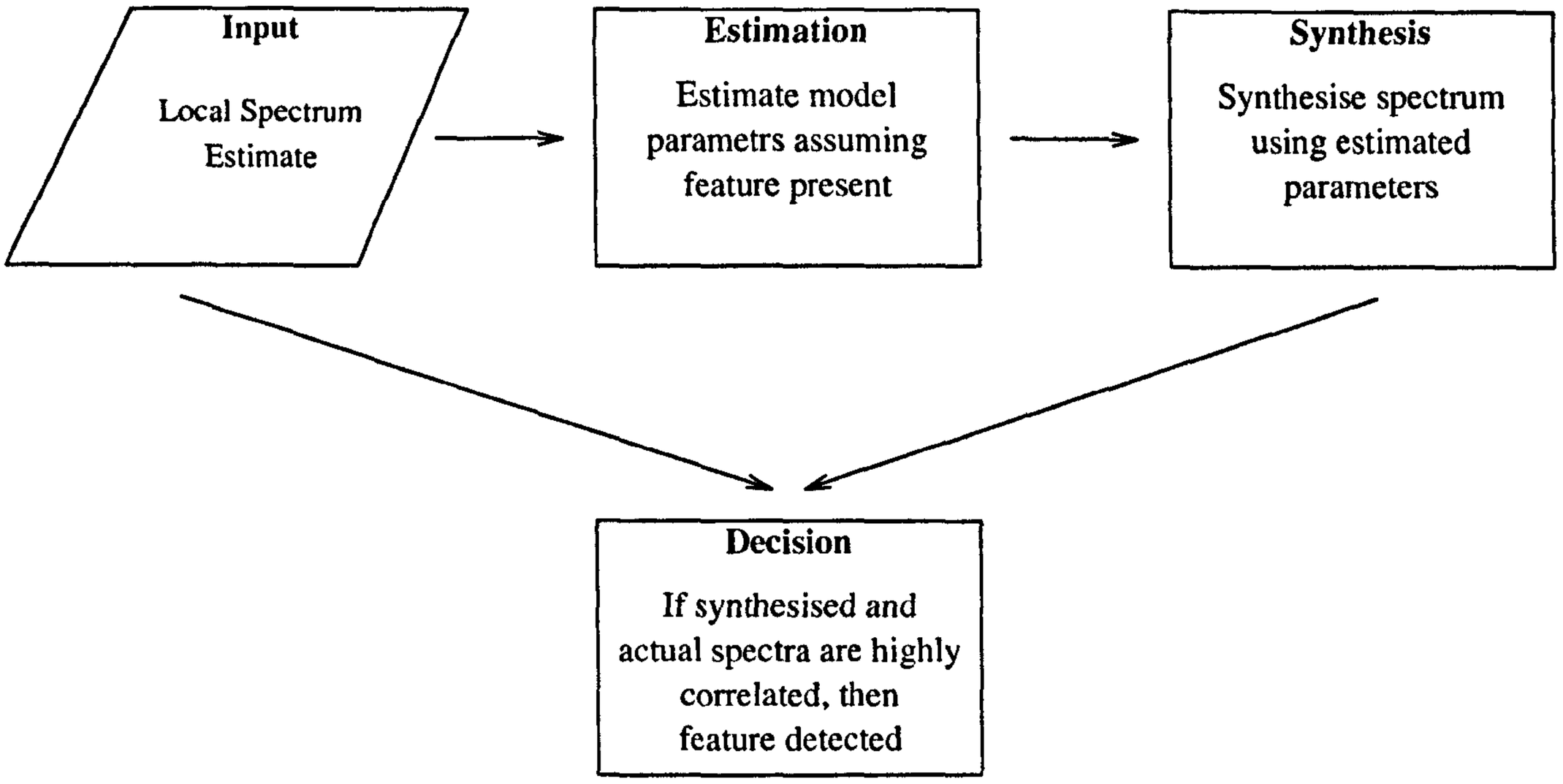


Figure 3.5: Feature Detection/Estimation Process

tection takes place could be extented to other feature classes that can be modelled similarly in the frequency domain.

The feature detection problem has two components:

1. Deciding whether a given class of feature can be used to model a spatial region to some predetermined degree of accuracy.

2. Estimating the set of parameters needed to describe the particular instance of the feature.

Clearly this problem fits into the model based decision procedure described in section 3.2. The process can be applied for each of the feature models and a decision made as to which can best model the region.

The final decision procedure can be complicated in a number of ways. First, there must be a similarity threshold for each feature class which the correlation must exceed,

before it can be considered as successfully modelling the region. If this threshold is not reached for any of the feature classes, then none of them can be used to model the region. This corresponds to accepting the 'null hypothesis' in hypothesis testing. Secondly, other factors may determine the "best" feature to use, rather than just the correlation measure. An obvious example is that of simplicity - if a simple feature can successfully model a region, then there is no need to use a more complex one. This suggests adapting the algorithm so that a local spectrum is first tested against the simplest of the feature models. If this fails, it is tested against the progressively more complex models until either one passes the similarity threshold, and is thus detected, or else it is decided that none of the feature models fits the data. If a region cannot be modelled using any of the classes of local model the detection process can be reapplied at a smaller scale, ie. by splitting the region into four quadrants and applying the detection process to each.

The order of testing used in this work is shown in figure (3.6). It is obvious that the simplest model for a region is a single linear feature and hence this is tested for first. As it is obviously desirable to use one feature to model the region, rather than several, the spectrum is tested against the arc model. Finally the spectrum is tested for the presence of multiple (two or more) linear features.

It is clear that the likelihood of being able to model a given region in terms of a set of feature models is dependent on the expressiveness of that set. In order to increase its expressiveness, the set of models can be extended to include a large number of complex models. This leads to a corresponding increase in the computational complexity of the detection process, however.

An alternative solution is to perform the detection process within a multiresolution

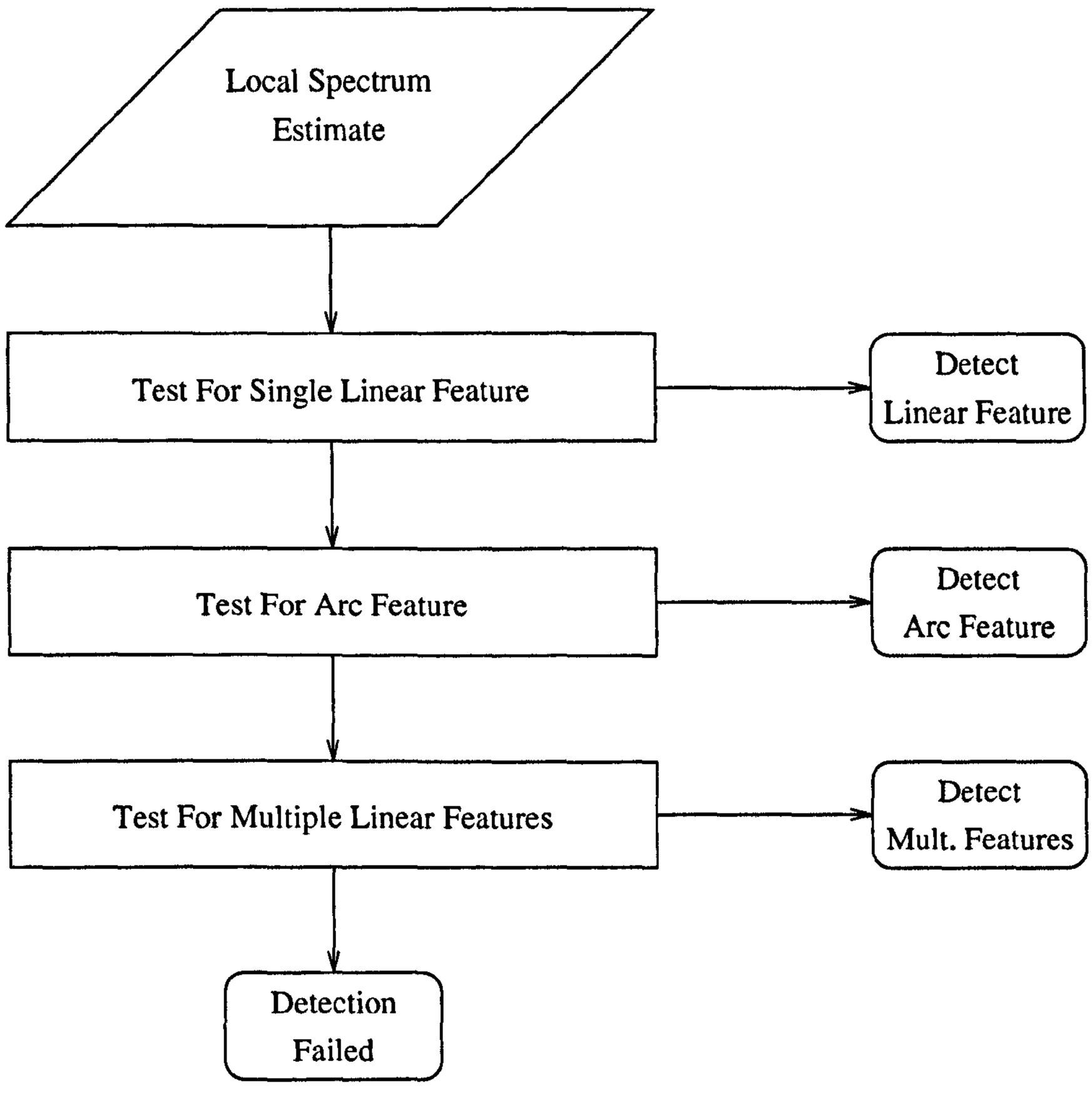


Figure 3.6: Feature Detection/Estimation Process

framework and over a range of scales. By choosing a limited set of feature models and systematically varying the analysis scale, it is possible to both guarantee that at some scale there will be an acceptable model fit, and that the largest scale at which a given region can be modelled can be found [90]. This process is discussed in chapter 5.

3.3.1 Linear Features

As described in section 3.1.1 the discrete, sampled spectrum corresponding to a region containing a single linear feature is of the form

$$\hat{x}(\vec{\omega}_i) = \begin{cases} |\hat{x}(\vec{\omega}_i)| \exp -\jmath[\vec{\xi}_0 \cdot \vec{\omega}_i + \varepsilon] & \text{if } \vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0 \\ |\hat{x}(\vec{\omega}_i)| \exp -\jmath[\vec{\xi}_0 \cdot \vec{\omega}_i - \varepsilon] & \text{otherwise} \end{cases} \quad 0 \leq i < N^2 \tag{3.42}$$

where θ_0 is the feature orientation. To simplify notation when dealing with relative frequency samples, it is convenient to define two vectors to denote adjacent samples in the ω_1 and ω_2 directions respectively, as shown in figure 3.7, where

$$\vec{\Omega}_1 = \begin{bmatrix} \Omega \\ 0 \end{bmatrix} \tag{3.43}$$

$$\vec{\Omega}_2 = \begin{bmatrix} 0 \\ \Omega \end{bmatrix} \tag{3.44}$$

and Ω is the frequency sampling interval.

For the continuous spectrum, the feature centroid is proportional to the derivative of the spectrum phase. The equivalent for a sampled spectrum is to consider the difference in phase between adjacent samples. Defining the forward and backward conjugate products between neighbouring samples as

$$d_{m1}(\vec{\omega}_i) = \hat{x}(\vec{\omega}_i)\hat{x}^*(\vec{\omega}_i - \vec{\Omega}_m) \tag{3.45}$$

$$d_{m2}(\vec{\omega}_i) = \hat{x}(\vec{\omega}_i + \vec{\Omega}_m)\hat{x}^*(\vec{\omega}_i) \tag{3.46}$$

for $m = 1, 2$, it can be clearly seen from equation (3.42) that if the region contains a single linear feature, both $d_{m1}(\vec{\omega}_i)$ and $d_{m2}(\vec{\omega}_i)$ will have a phase proportional to the feature centroid. From equation (3.42) it is clear that

$$\arg d_{m1}(\vec{\omega}_i) = \arg d_{m2}(\vec{\omega}_i) = \xi_{0m}\Omega, \ \forall i \tag{3.47}$$

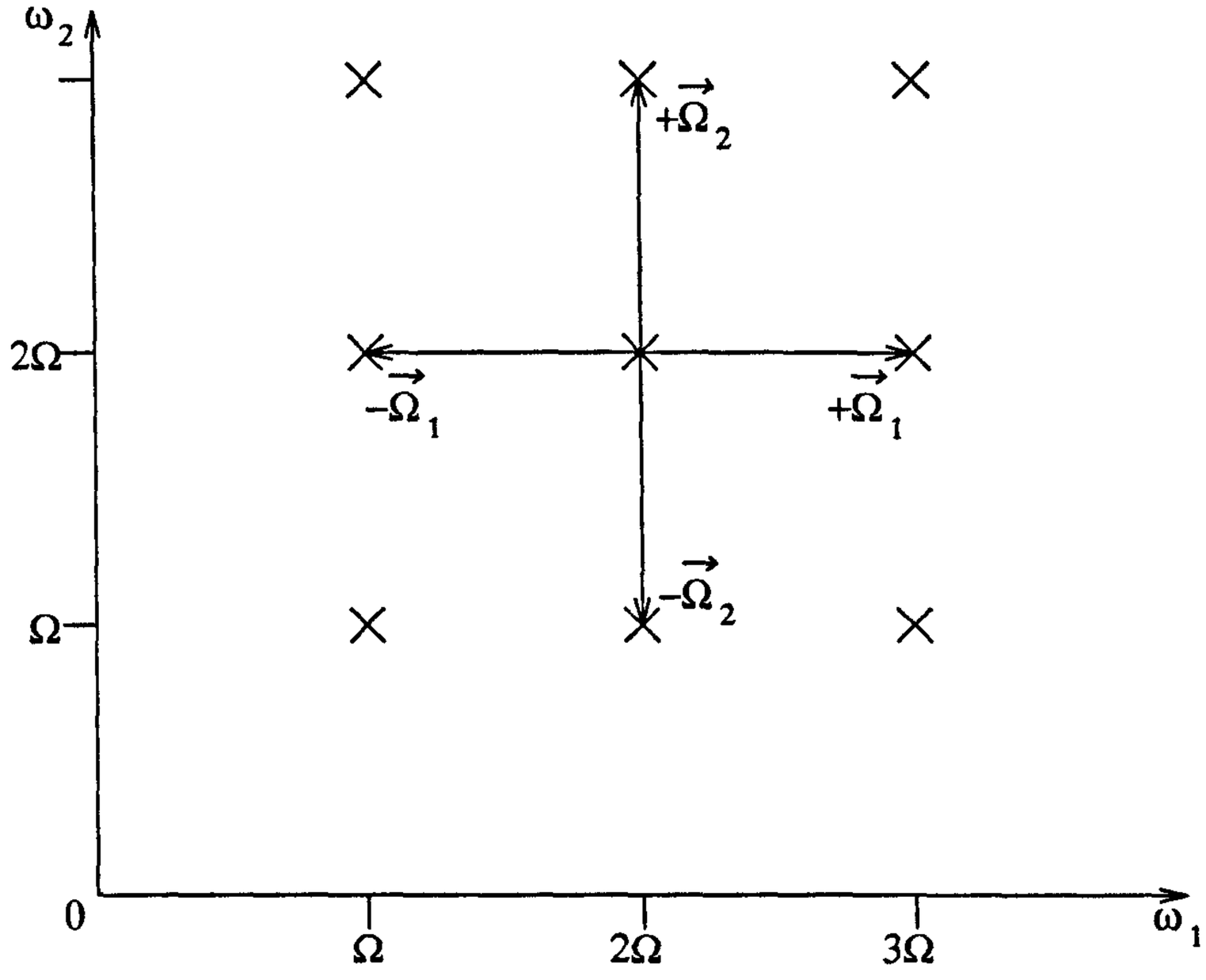


Figure 3.7: Adjacent frequency displacement vectors

In order to provide a phase estimate that is centred on a frequency coefficient, these forward and backward differences can be summed, ie.

$$d_m(\vec{\omega}_i) = d_{m1}(\vec{\omega}_i) + d_{m2}(\vec{\omega}_i) \tag{3.48}$$

A correlation statistic R_m can be defined as the sum of $d_m(\vec{\omega}_i)$ over the sampled spectrum, ie.

$$R_m = 0.5 \sum_{i=0}^{N^2-1} d_{m1}(\vec{\omega}_i) + d_{m2}(\vec{\omega}_i) \tag{3.49}$$

Assuming that the spectrum has a linear phase characteristic, and that the expected values of $d_{m1}(\vec{\omega}_i)$ and $d_{m2}(\vec{\omega}_i)$ are the same

$$E\{R_m\} = E\{0.5 \sum_{i=0}^{N^2-1} (d_{m1}(\vec{\omega}_i) + d_{m2}(\vec{\omega}_i))\} \tag{3.50}$$

$$= \exp[j\xi_{0m}\Omega] E\{\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega})||\hat{x}(\vec{\omega} - \vec{\Omega}_m)|\} \tag{3.51}$$

$$= \exp[\jmath\xi_{0m}\Omega]N^2E\{|\hat{x}(\vec{\omega})||\hat{x}(\vec{\omega}-\vec{\Omega}_m)|\} \tag{3.52}$$

where the expectation is over the magnitude distribution, it can therefore be seen that if the region contains a single linear feature, the phase constant will be proportional to the feature centroid. The phase of R_m gives an unbiased estimate of the linear phase constant, and hence a ML estimate of the feature centroid, as in [16], can be given by

$$\vec{\eta}_0 = \frac{1}{\Omega}\begin{bmatrix} \arg R_1 \\ \arg R_2 \end{bmatrix} \tag{3.53}$$

where

$$E\{\vec{\eta}_0\} = \vec{\xi}_0 \tag{3.54}$$

If the region contains multiple linear features, the corresponding spectrum can be modelled as the sum of the spectra of the individual features. If there are K features present, then

$$\hat{x}(\vec{\omega}_i) = \sum_{k=1}^{K}\hat{x}_k(\vec{\omega}_i) \tag{3.55}$$

where $\hat{x}_k(\vec{\omega})$ are the component spectra which can be modelled individually by

$$\hat{x}_k(\vec{\omega}_i) = \begin{cases} |\hat{x}_k(\vec{\omega}_i)|\exp -\jmath[\vec{\xi}_{k0}\cdot\vec{\omega}_i + \varepsilon_k] & \text{if } \vec{\omega}_i\cdot\vec{v}_{\theta_{k0}} > 0 \\ |\hat{x}_k(\vec{\omega}_i)|\exp -\jmath[\vec{\xi}_{k0}\cdot\vec{\omega}_i - \varepsilon_k] & \text{otherwise} \end{cases} \quad 0 \le i < N^2 \tag{3.56}$$

Since any coefficient will be, in general, the sum of the K component coefficients, the conjugate products $d_{m1}(\vec{\omega})$ and $d_{m2}(\vec{\omega})$ will be dependent upon all the feature spectra, and be given by

$$d_{m1}(\vec{\omega}_i) = (\sum_{j=1}^{K}\hat{x}_j(\vec{\omega}_i))(\sum_{k=1}^{K}\hat{x}_k^*(\vec{\omega}_i-\vec{\Omega}_m)) \tag{3.57}$$

$$d_{m2}(\vec{\omega}_i) = (\sum_{j=1}^{K}\hat{x}_j(\vec{\omega}_i+\vec{\Omega}_m))(\sum_{k=1}^{K}\hat{x}_k^*(\vec{\omega}_i)) \tag{3.58}$$

The correlation statistic R_m is therefore given by

$$R_m = \sum_{i=0}^{N^2-1} \left(\sum_{j=1}^{K} \hat{x}_j(\vec{\omega}_i) \sum_{k=1}^{K} \hat{x}_k^*(\vec{\omega}_i - \vec{\Omega}_m) + \sum_{j=1}^{K} \hat{x}_j(\vec{\omega}_i + \vec{\Omega}_m) \sum_{k=1}^{K} \hat{x}_k^*(\vec{\omega}_i) \right) \quad (3.59)$$

and is dependent upon the pairwise interactions among the component spectra. In order to see how this affects the total statistic, a pairwise cross correlation statistic, $R_{m(j,k)}$ can be defined to show the interference between two components

$$R_{m(j,k)} = \sum_{i=0}^{N^2-1} \hat{x}_j(\vec{\omega}_i)\hat{x}_k^*(\vec{\omega}_i - \vec{\Omega}_m) + \hat{x}_j(\vec{\omega}_i + \vec{\Omega}_m)\hat{x}_k^*(\vec{\omega}_i) \quad (3.60)$$

Substituting this into equation (3.59) gives

$$R_m = \sum_{j=1}^{K} \sum_{k=1}^{K} R_{m(j,k)} \quad (3.61)$$

The correlation statistic is therefore the sum of all the pairwise cross correlation statistics. If a region only contains linear features that differ in their position and orientation, since the features concerned are highly concentrated along their orientations, it can be assumed that to a first approximation their spectra are pairwise orthogonal and hence

$$R_{m(j,k)} = \begin{cases} \sum_{i=0}^{N^2-1} \hat{x}_j(\vec{\omega}_i)\hat{x}_j^*(\vec{\omega}_i - \vec{\Omega}_m) + \hat{x}_j(\vec{\omega}_i + \vec{\Omega}_m)\hat{x}_j^*(\vec{\omega}_i) & \text{if } j = k \\ 0 & \text{otherwise} \end{cases} \quad (3.62)$$

This can be substituted into equation (3.61), which reduces to give

$$R_m = \sum_{k=1}^{K} R_{m(k,k)} \quad (3.63)$$

The expected value of R_m is therefore given by

$$E\{R_m\} = \sum_{k=1}^{K} E\{R_{m(k,k)}\} \quad (3.64)$$

The phase of the expected value of the correlation can be seen to be the weighted mean of the phases of the expected values of the components $R_{m(k,k)}$. Thus the estimated value $\vec{\eta}_0$ will be the weighted mean of the individual feature centroids.

From the above, it can be seen that if a region has a linear phase characteristic then $|R_m|$ will be given by

$$|R_m| = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega})||\hat{x}(\vec{\omega} - \vec{\Omega}_m)| \tag{3.65}$$

In the case of multiple features, from equation (3.63) it is clear that $|R_m|$ will be smaller due to the varying phases of the component $R_{m(k,k)}$ terms. This suggests that in these cases the value of the $|R_m|$ can be used to determine whether the spectrum has a linear phase characteristic. This is not sufficient, however, to determine whether a region contains a single linear feature. Consider a region containing two parallel and similar features, modelled by the equations

$$\hat{x}_1(\vec{\omega}_i) = |\hat{x}_1(\vec{\omega}_i)| \exp[-\jmath\vec{\xi}_{01} \cdot \vec{\omega}_i] \tag{3.66}$$

$$\hat{x}_2(\vec{\omega}_i) = |\hat{x}_2(\vec{\omega}_i)| \exp[-\jmath\vec{\xi}_{02} \cdot \vec{\omega}_i] \tag{3.67}$$

where

$$|\hat{x}_1(\vec{\omega}_i)| = |\hat{x}_2(\vec{\omega}_i)| \quad 0 \leq i < N^2 \tag{3.68}$$

The local spectrum is given by the sum of these two spectra,

$$\hat{x}(\vec{\omega}_i) = \hat{x}_1(\vec{\omega}_i) + \hat{x}_2(\vec{\omega}_i) \tag{3.69}$$

$$= |\hat{x}_1(\vec{\omega}_i)|(\exp[-\jmath\vec{\xi}_{01} \cdot \vec{\omega}_i] + \exp[-\jmath\vec{\xi}_{02} \cdot \vec{\omega}_i]) \tag{3.70}$$

This can then be simplified to give

$$\hat{x}(\vec{\omega}_i) = 2|\hat{x}_1(\vec{\omega}_i)| \cos(\vec{d} \cdot \vec{\omega}_i) \exp[-\jmath\vec{\xi}_m \cdot \vec{\omega}_i] \tag{3.71}$$

where $\vec{\xi}_m$ is the mean centroid position given by

$$\vec{\xi}_m = \arg\left(\exp\left[\jmath\vec{\xi}_{01}\right] + \exp\left[\jmath\vec{\xi}_{02}\right]\right) \tag{3.72}$$

and

$$\vec{d} = 0.5\arg\left(\exp\left[\jmath\vec{\xi}_{01}\right] - \exp\left[\jmath\vec{\xi}_{02}\right]\right) \tag{3.73}$$

It can be seen from equation (3.71) that the spectrum has a linear phase characteristic with discontinuities at the points where the cosine term has a zero-crossing. As these correspond to parts of the spectrum with low energy, the phase discontinuities at these points will have only a minimal effect upon R_m. This suggests that the magnitude of R_m will not allow discrimination between single linear features and multiple parallel ones.

If the energy spectrum corresponding to the feature can be modelled as an ellipse, then the orientation about which the energy is concentrated, which is orthogonal to the feature orientation in the spatial domain, corresponds to the major axis of the ellipse. From Borisenko and Tarapov [12], a moment of inertia tensor I can be calculated using the spectrum energy in place of mass, ie.

$$I = \begin{bmatrix} I_{11} & I_{12} \\ I_{21} & I_{22} \end{bmatrix} \tag{3.74}$$

where

$$I_{11} = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega})|^2 \omega_{i1}^2 \tag{3.75}$$

$$I_{22} = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega})|^2 \omega_{i2}^2 \tag{3.76}$$

$$I_{12} = I_{21} = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega})|^2 \omega_{i1}\omega_{i2} \tag{3.77}$$

The major and minor axes of the ellipse correspond to the principal axes of the tensor, and are in the orientations given by its eigenvectors. The orientation of the feature is thus given by the eigenvector corresponding to the largest eigenvalue [57] and may be calculated by

$$\theta_0 = \arctan \frac{\lambda - I_{11}}{I_{12}} \tag{3.78}$$

where

$$\lambda = \frac{I_{11} + I_{22}}{2} + \sqrt{\left(\frac{I_{11} - I_{22}}{2}\right)^2 + I_{12}^2} \tag{3.79}$$

This application of tensor analysis to orientation estimation is similar to approaches used by both Knutsson[55] and Calway[16]. The elements of the inertia tensor can be related to the second order partial derivatives of the image (see Bigün [10]). Thus the tensor, and hence the orientation, could be calculated in the spatial domain using the image gradient at each point.

The last parameter of the model is the phase constant ε. The value of ε is constant over either half of the Fourier plane defined by the feature orientation, ie. for the set of frequency indices Θ_{θ_0}, such that

$$i \in \Theta_{\theta_0} \iff \vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0 \tag{3.80}$$

This is illustrated in figure 3.8. Once the orientation and centroid have been estimated, the constant, ε, can be estimated from

$$\tilde{\varepsilon} = \arg \sum_{i \in \Theta_{\theta_0}} \hat{x}(\vec{\omega}_i) \exp\left[j\vec{\eta_0} \cdot \vec{\omega}_i\right] \tag{3.81}$$

Since from equation (3.54), assuming there is a single linear feature in the region, the expected value of $\vec{\eta_0}$ is $\vec{\xi_0}$ this gives

$$E\{\tilde{\varepsilon}\} = E\{\arg \sum_{i \in \Theta_{\theta_0}} |\hat{x}(\vec{\omega}_i)| \exp\left[j\varepsilon\right]\}$$

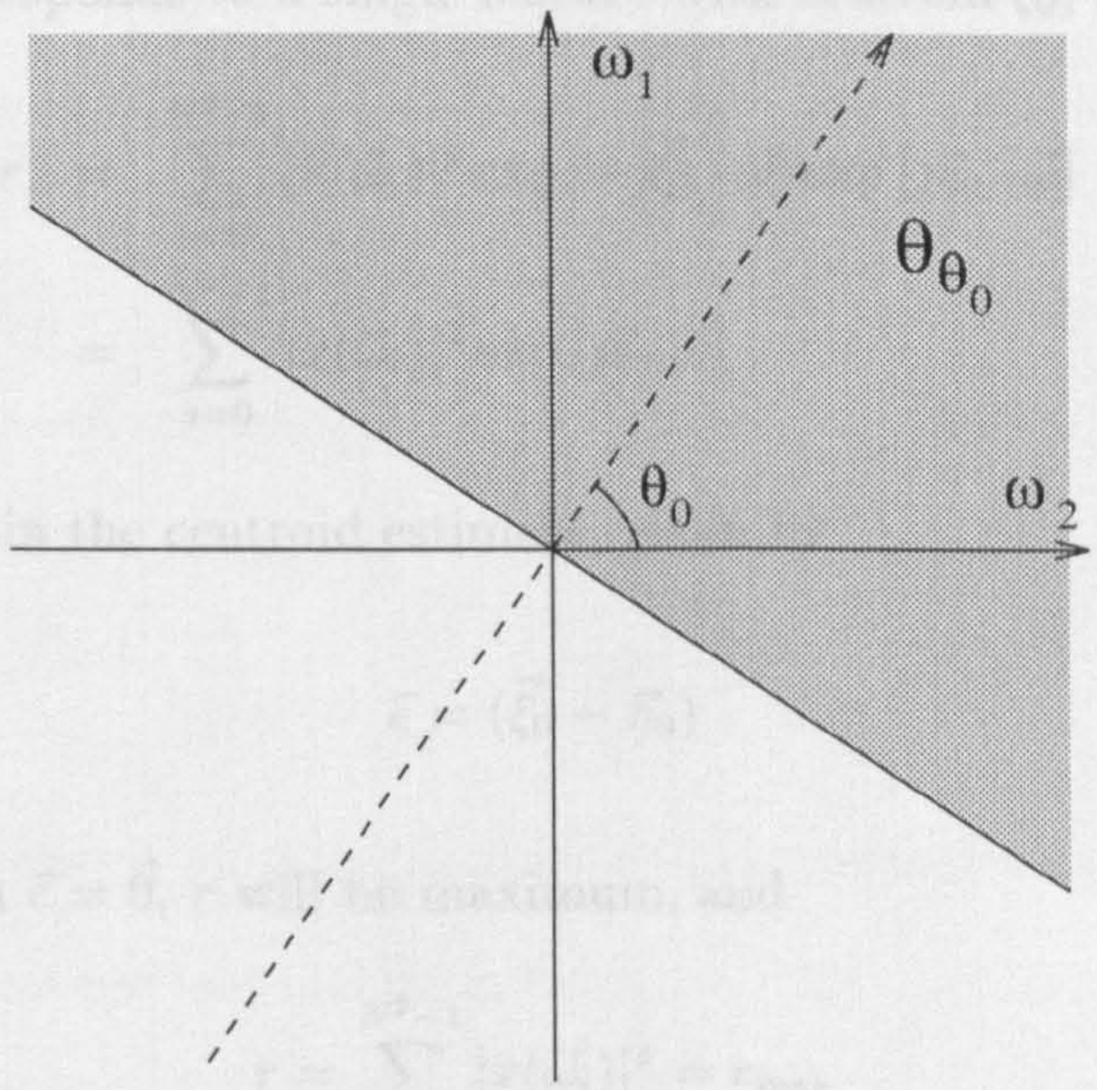


Figure 3.8: Shaded area indicates Θ_{θ_0}

$$= \varepsilon \tag{3.82}$$

Once the parameters have been estimated, it is necessary to determine whether the corresponding region model is accurate. This check is performed by synthesising the local spectrum using the estimated parameters, and then comparing this with the actual spectrum. The synthesised spectrum, denoted $\tilde{x}(\vec{\omega})$, is generated using the magnitude of the real spectrum and phase estimated using the model parameters, ie.

$$\tilde{x}(\vec{\omega}_i) = |\hat{x}(\vec{\omega}_i)| \exp \jmath[\vec{\eta}_0 \cdot \vec{\omega}_i + \hat{\varepsilon}] \tag{3.83}$$

The comparison is performed by calculating the inner product of the synthesised and actual coefficients, given by

$$r = \sum_{i=0}^{N^2-1} \tilde{x}(\vec{\omega}_i)\hat{x}^*(\vec{\omega}_i) \tag{3.84}$$

If the spectrum corresponds to a single feature with centroid $\vec{\xi}_0$, this reduces to

$$r = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \exp[-j\vec{\xi}_0 \cdot \vec{\omega}] \exp[j\vec{\eta}_0 \cdot \vec{\omega}] \quad (3.85)$$

$$= \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \exp[j\vec{e} \cdot \vec{\omega}] \quad (3.86)$$

where $\vec{e}$ is the error in the centroid estimate, given by

$$\vec{e} = (\vec{\xi}_0 - \vec{\eta}_0) \quad (3.87)$$

It is clear that when $\vec{e} = \vec{0}$, r will be maximum, and

$$r = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 = r_{\max} \quad (3.88)$$

ie. the total spectrum energy. Indeed it can be clearly seen that if a single linear feature is present in the region then since

$$E\{\vec{\eta}_0\} = \vec{\xi}_0 \quad (3.89)$$

from equation (3.54),

$$E\{\exp[j\vec{\eta} \cdot \vec{\omega}]\} = \exp[j\vec{\xi}_0 \cdot \vec{\omega}] \quad (3.90)$$

Substituting this into equation (3.85) and assuming that the magnitude and phase are uncorrelated, gives

$$E\{r\} = E\{\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2\}$$

$$= r_{\max} \quad (3.91)$$

If there is any error in the centroid estimate then the inner product will effectively be between the spectrum of the feature and that of a shifted version of the feature.

Provided these features are sufficiently concentrated and separated so that they do not overlap in the spatial domain then, using Parseval's Theorem [70], it follows that

$$r = 0 \tag{3.92}$$

If the region contains a number of linear features, K, then the inner product from equation (3.84) may be rewritten as

$$r = \sum_{i=0}^{N^2-1} \tilde{x}(\vec{\omega}_i) \sum_{k=1}^{K} \hat{x}_k^*(\vec{\omega}_i) \tag{3.93}$$

From this it is clear that if an inner product term is defined between the synthesised spectrum and each component spectrum by

$$r_k = \sum_{i=0}^{N^2-1} \tilde{x}(\vec{\omega}_i) \hat{x}_k^*(\vec{\omega}_i) \tag{3.94}$$

equation (3.84) may be simplified to give

$$r = \sum_{k=1}^{K} r_k \tag{3.95}$$

In this case it is useful to consider the expected value of r. From equation (3.95)

$$E\{r|K \text{ features present}\} = KE\{r_k\} \tag{3.96}$$

Assuming that for any given feature, the errors e_{k1} and e_{k2} are equally likely to take any value in the interval $[-N/2, N/2]$ then

$$E\{r_k\} = \sum_{i=0}^{N^2-1} |\hat{x}_k(\vec{\omega}_i)|^2 \frac{1}{N^2} \int_{-N/2}^{N/2} \int_{-N/2}^{N/2} \exp[j\vec{e} \cdot \vec{\omega}] de_1 de_2 \tag{3.97}$$

$$= \sum_{i=0}^{N^2-1} \frac{|\hat{x}_k(\vec{\omega}_i)|^2}{N^2 \omega_{i1} \omega_{i2}} \sin(N\omega_{i1}/2) \sin(N\omega_{i2}/2) \tag{3.98}$$

Substituting in the frequency values from equation (2.27) and noting that the frequency sampling interval is given by

$$\Omega = \frac{2\pi}{N} \tag{3.99}$$

then equation (3.98) reduces to

$$E\{r_k\} = \sum_{i=0}^{N^2-1} \frac{|\hat{x}_k(\vec{\omega}_i)|^2}{N^2\omega_{i1}\omega_{i2}} \tag{3.100}$$

By assuming that $|\hat{x}_k(\vec{\omega}_i)|^2$ decreases monotonically from the origin and noting that

$$N\omega_{im} > \pi \quad \forall i, m = 1, 2 \tag{3.101}$$

then

$$E\{r_k\} \leq \frac{1}{\pi^2} \sum_{i=0}^{N^2-1} |\hat{x}_k(\vec{\omega}_i)|^2 \tag{3.102}$$

and therefore, assuming that all features have the same energy and

$$\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 = \sum_{k=1}^{K} \sum_{i=0}^{N^2-1} |\hat{x}_k(\vec{\omega}_i)|^2 \tag{3.103}$$

the expected value of r becomes

$$E\{r|K \text{ features present}\} \leq \frac{1}{\pi^2} \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \tag{3.104}$$

Therefore substituting from equation (3.91)

$$E\{r|K \text{ features present}\} \leq \frac{1}{\pi^2} E\{r|1 \text{ feature present}\} \tag{3.105}$$

The statistic r can therefore be used to indicate whether the region contains a single linear feature, which can be modelled with the estimated parameters.

If a region contains more than one linear feature then, under certain conditions, it may be possible to estimate their parameters from the local spectrum. Equation (3.64) shows that for a region containing multiple orthogonal features, the value of the correlation statistic R_m for the complete spectrum is the sum of the statistics $R_{m(k,k)}$ for each of the component spectra. Therefore if it is possible to estimate these

component statistics individually, they can be used to provide centroid estimates for each of the linear features in the region. The calculation of these statistics requires knowledge of the component spectra and therefore in order to perform the estimation the spectrum must be segmented into its components.

Since the feature model is based upon the spectrum phase, this is used to segment the spectra. If a spectrum coefficient is dependent upon only one component feature, then its phase will be determined by the feature centroid of only that feature. Assuming that neighbouring coefficients are due to the same feature, the phase of the conjugate products can be used to give an estimate of the centroid of this feature. A clustering approach, similar to K-means [87] or ISODATA [36] and based upon this centroid estimate, can be used to segment the spectrum.

If each coefficient is uniquely dependent upon one feature, a partition of the set of coefficient indices may be generated with each subset corresponding to one feature. If there are K features and I_k is the set of indices corresponding to the k^{th} feature then

$$i \in I_k \iff \hat{x}_k(\vec{\omega}_i) \neq 0 \tag{3.106}$$

and

$$I_T = I_B \cup \bigcup_k I_k \tag{3.107}$$

where I_T is the set of all the indices and I_B represents coefficients that belong to no features - such as those with zero energy and those that correspond to low energy background noise.

Define the centroid estimate from a single coefficient with index i as

$$\vec{\eta}(\vec{\omega}_i) = \frac{1}{\Omega} \begin{bmatrix} \arg\left(d_{11}(\vec{\omega}_i) + d_{12}(\vec{\omega}_i)\right) \\ \arg\left(d_{21}(\vec{\omega}_i) + d_{22}(\vec{\omega}_i)\right) \end{bmatrix} \tag{3.108}$$

For any given partition of the coefficients into subsets, an error measure e can be defined as

$$e = \sum_k \sum_{i \in I_k} \|\vec{\eta}(\vec{\omega}_i) - \vec{\eta}_{k0}\|^2 \tag{3.109}$$

The best partition can therefore be defined as that for which e is minimised. Assuming that there is a known number of clusters K, the K-means algorithm can be used to generate a series of partitions that converge to one where e is a minimum. The algorithm is given by [87]:

1. Choose K arbitrary feature centroids.

2. Assign each coefficient to the cluster k for which $\|\vec{\eta}(\vec{\omega}_i) - \vec{\eta}_{k0}\|^2$ is smallest.

3. Recalculate the feature centroids for each cluster by calculating the mean value of $\vec{\eta}(\vec{\omega}_i)$.

4. If any coefficients have been reclassified since the last time this step was reached return to step 2. If not terminate.

It can be easily seen that step 2, for a fixed set of feature centroid estimates, minimises e by minimising $\|\vec{\eta}(\vec{\omega}_i) - \vec{\eta}_{k0}\|^2$ for each i. Additionally for a fixed partition, step 3 minimises e by replacing each cluster centroid estimate with the mean of the estimates of the appropriate coefficients. If the error at the end of the n^{th} iteration is denoted by e^n, then

$$e^{n+1} \leq e^n \tag{3.110}$$

This suggests that the algorithm will converge to a segmentation for which e is a minimum. The iterations will stop when no coefficients are reclassified, which is

equivalent to the condition

$$e^{n+1} = e^n \tag{3.111}$$

Since the number of features is unknown, a version of the ISODATA algorithm can be used to segment the spectrum, using the K-means algorithm, with differing values of K, until one is found that models the data sufficiently well. The measures $R_{1(k,k)}$ and $R_{2(k,k)}$, given by equation (3.62) can be used to estimate the cluster centroids and indicate whether the clusters have a linear phase property:

1. Set $K = 2$

2. Use the K-means algorithm to segment the image

3. Throw away "small" clusters by assigning their coefficients to I_B

4. If any given cluster does not have a linear phase characteristic, replace with two clusters with estimated centroids displaced from the original and then return to step 2. If all clusters pass the test terminate.

Having segmented the spectrum, each cluster is modelled as in the 'single linear feature' case and its parameters are estimated. The set of features is then used to synthesise a set of coefficients with the magnitudes from the original spectrum and phases generated from the estimated model parameters,

$$\tilde{x}(\vec{\omega}_i) = \sum_k s(i,k)|\hat{x}(\vec{\omega}_i)| \exp[-j\vec{\eta}_{0k} \cdot \vec{\omega} + \varepsilon_k] \tag{3.112}$$

where $s(i,k)$ is a function that indicates whether the i^{th} coefficient is due to the k^{th}, and is given by

$$s(i,k) = 1 \text{ if } i \in I_k \tag{3.113}$$

$$= 0 \text{ otherwise} \tag{3.114}$$

The success of the clustering can then be determined by correlating the synthesised and actual spectra - as for the single linear feature.

3.3.2 Arc Features

In section 3.1.2 the phase model of the spectrum, corresponding to a region containing a circular arc was given as

$$\hat{x}(\vec{\omega}_i) = \begin{cases} |\hat{x}(\vec{\omega}_i)| \exp -\jmath[R\|\vec{\omega}_i\| + \vec{\omega}_i \cdot \vec{\xi}_0 + \varepsilon] & \text{if } \vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0 \\ |\hat{x}(\vec{\omega}_i)| \exp -\jmath[-R\|\vec{\omega}_i\| + \vec{\omega}_i \cdot \vec{\xi}_0 - \varepsilon] & \text{otherwise} \end{cases} \quad 0 < i < N^2 - 1 \tag{3.115}$$

In order to estimate the arc parameters, it is necessary to consider the change in phase between adjacent coefficients. It suffices to use just the backward differences in this case, however. Substituting the expression for the backward difference, given by equation (3.115), into equations (3.45), assuming that $\vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0$, gives

$$d_{m1}(\vec{\omega}_i) = |\hat{x}(\vec{\omega}_i)\hat{x}(\vec{\omega}_i - \vec{\Omega}_m)| \exp \jmath[R(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_m\|) + \xi_{0m}\Omega] \tag{3.116}$$

If a vector of the phase-differenced coefficients along the two axes, $\vec{\varphi}(\vec{\omega})$, is defined as

$$\vec{\varphi}(\vec{\omega}) = \begin{bmatrix} \arg(d_{11}(\vec{\omega})) \\ \arg(d_{21}(\vec{\omega})) \end{bmatrix} \tag{3.117}$$

and $\vec{g}(\vec{\omega})$ is given by

$$\vec{g}(\vec{\omega}_i) = \begin{bmatrix} \|\vec{\omega}_i - \vec{\Omega}_2\| - \|\vec{\omega}_i\| \\ \|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\| \end{bmatrix} \tag{3.118}$$

It follows from equation (3.118) that

$$(\vec{\varphi}(\vec{\omega}_i) - \Omega\vec{\xi}_0) \cdot \vec{g}(\vec{\omega}_i) = 0 \tag{3.119}$$

Although it has been assumed that $\vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0$, it can be seen that this is equally valid for $\vec{\omega}_i \cdot \vec{v}_{\theta_0} \leq 0$.

In order to derive an estimator for the circle centre $\vec{\xi}_0$, an error measure, $\mathcal{E}(\vec{\omega}_i)$, is defined for each frequency coefficient

$$\mathcal{E}(\vec{\omega}_i) = ((\vec{\varphi}(\vec{\omega}_i) - \Omega\vec{\eta}_0) \cdot \vec{g}(\vec{\omega}_i))^2 \tag{3.120}$$

where $\vec{\eta}_0$ is an estimate of $\vec{\xi}_0$, and a total error is defined as a weighted sum over all of the coefficients, that is

$$\mathcal{E} = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|\mathcal{E}(\vec{\omega}_i) \tag{3.121}$$

$$= \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|((\vec{\varphi}(\vec{\omega}_i) - \Omega\vec{\eta}_0) \cdot \vec{g}(\vec{\omega}_i))^2 \tag{3.122}$$

The best estimate is defined as that for which the total error $\mathcal{E}$ is minimum with respect to η_1 and η_2, which may be expressed in the form

$$\frac{\partial\mathcal{E}}{\partial\eta_1} = \frac{\partial\mathcal{E}}{\partial\eta_2} = 0 \tag{3.123}$$

Substituting into the partial derivatives from equation (3.122) and taking the derivative inside the summation gives

$$\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|((\vec{\varphi}(\vec{\omega}_i) - \Omega\vec{\eta}_0) \cdot \vec{g}(\vec{\omega}_i))\frac{\partial}{\partial\eta_{01}}(\vec{\eta}_0 \cdot \vec{g}(\vec{\omega}_i)) = 0 \tag{3.124}$$

$$\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|((\vec{\varphi}(\vec{\omega}_i) - \Omega\vec{\eta}_0) \cdot \vec{g}(\vec{\omega}_i))\frac{\partial}{\partial\eta_{02}}(\vec{\eta}_0 \cdot \vec{g}(\vec{\omega}_i)) = 0 \tag{3.125}$$

which can be rearranged to give

$$\Omega\vec{\eta}_0 \cdot \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|g_1(\vec{\omega}_i)\vec{g}(\vec{\omega}_i) = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|g_1(\vec{\omega}_i)\vec{\varphi}(\vec{\omega}_i) \cdot \vec{g}(\vec{\omega}_i) \tag{3.126}$$

$$\Omega\vec{\eta}_0 \cdot \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|g_2(\vec{\omega}_i)\vec{g}(\vec{\omega}_i) = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|g_2(\vec{\omega}_i)\vec{\varphi}(\vec{\omega}_i) \cdot \vec{g}(\vec{\omega}_i) \tag{3.127}$$

or in matrix form

$$\Omega\vec{\eta}_0 \sum_{i=0}^{N^2-1} T(\vec{\omega}_i) = \sum_{i=0}^{N^2-1} T(\vec{\omega}_i)\vec{\varphi}(\vec{\omega}_i) \tag{3.128}$$

where $T(\vec{\omega}_i)$ is a tensor defined by

$$T(\vec{\omega}_i) = |\hat{x}(\vec{\omega}_i)|\vec{g}(\vec{\omega}_i)\vec{g}^T(\vec{\omega}_i) \tag{3.129}$$

Defining

$$T = \sum_{i=0}^{N^2-1} T(\vec{\omega}_i) \tag{3.130}$$

equation (3.128) simplifies to

$$\vec{\eta}_0 = \frac{1}{\Omega} T^{-1} \sum_{i=0}^{N^2-1} T(\vec{\omega}_i)\vec{\varphi}(\vec{\omega}_i) \tag{3.131}$$

A similar approach can be adopted to give an estimate $\tilde{R}$ of the radius of the arc R. Defining an error for the estimate at each coefficient as

$$\mathcal{E}_R(\vec{\omega}_i) = \varphi_1(\vec{\omega}_i) - \Omega\xi_{01} - \tilde{R}(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|)^2 \tag{3.132}$$

where $\varphi_1(\vec{\omega}_i) = \arg(d_{11}(\vec{\omega}_i))$, the best estimate can be defined as that for which the total error $\mathcal{E}_R$ is minimum, where $\mathcal{E}_R$ is given by

$$\mathcal{E}_R = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|\mathcal{E}_R(\vec{\omega}_i) \tag{3.133}$$

$$= \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|(\varphi_1(\vec{\omega}_i) - \Omega\xi_{01} - \tilde{R}(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|))^2 \tag{3.134}$$

For the total error to be minimum,

$$\frac{\partial \mathcal{E}_R}{\partial \tilde{R}} = 0 \tag{3.135}$$

and therefore

$$\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|)(\varphi_1(\vec{\omega}_i) - \Omega\xi_{01} - \tilde{R}(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|)) = 0 \tag{3.136}$$

Rearranging gives the optimum estimate of R as

$$\tilde{R} = \frac{\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|)(\varphi_1(\vec{\omega}_i) - \Omega\xi_{01})}{\sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|(\|\vec{\omega}_i\| - \|\vec{\omega}_i - \vec{\Omega}_1\|)^2} \tag{3.137}$$

Having estimated the centre and radius of the arc, the phase constant ε can be estimated in a similar manner to the linear feature case by using the expression

$$\tilde{\varepsilon} = \arg \sum_{i \in \Theta_{\theta_0}} |\hat{x}(\vec{\omega}_i)| \exp\left[\jmath \tilde{R} \|\vec{\omega}_i\| - \vec{\eta}_0 \cdot \vec{\omega}_i\right] \tag{3.138}$$

Having estimated the arc parameters the next stage is to decide whether they adequately model the region contents. As with the linear features this detection procedure is based upon taking the inner product between the actual feature coefficients and a set $\tilde{x}(\vec{\omega}_i)$ synthesised using the estimated parameters, given by

$$\tilde{x}(\vec{\omega}_i) = \begin{cases} |\hat{x}(\vec{\omega}_i)| \exp -\jmath[\tilde{R}\|\vec{\omega}_i\| + \vec{\omega}_i \cdot \vec{\eta}_0 + \tilde{\varepsilon}] & \text{if } \vec{\omega}_i \cdot \vec{v}_{\theta_0} > 0 \\ |\hat{x}(\vec{\omega}_i)| \exp -\jmath[-\tilde{R}\|\vec{\omega}_i\| + \vec{\omega}_i \cdot \vec{\eta}_0 - \tilde{\varepsilon}] & \text{otherwise} \end{cases} \quad 0 < i < N^2 - 1 \tag{3.139}$$

as with the linear features, the synthesised spectrum uses the actual spectrum magnitudes. To test the similarity between the actual and synthesised coefficients, an inner product is defined between them as

$$r = \sum_{i=0}^{N^2-1} \tilde{x}(\vec{\omega}_i)\hat{x}^*(\vec{\omega}_i) \tag{3.140}$$

If there is an arc feature present in the region with parameters R and $\vec{\xi}_0$, and assuming the phase constant $\varepsilon = 0$, this reduces to

$$r = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \exp\left[-\jmath((R - \tilde{R})\|\vec{\omega}_i\| + (\vec{\xi}_0 - \vec{\eta}_i) \cdot \vec{\omega}_i)\right] \tag{3.141}$$

$$= \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \exp -\jmath[e_R\|\vec{\omega}_i\| + \vec{e}_0 \cdot \vec{\omega}_i] \tag{3.142}$$

where e_R and $\vec{e}_0$ are the errors in the radius and arc centre estimates respectively. The inner product will be maximised when the errors are zero and will be equal to the total spectrum energy, ie.

$$r_{\max} = \sum_{i=0}^{N^2-1} |\hat{x}(\vec{\omega}_i)|^2 \tag{3.143}$$

By comparing r with this maximum value, it is possible to decide whether the estimated parameters successfully model the region.

3.4 Summary

The use of local spectral models to detect features in image regions has been discussed in this chapter. The approach taken has been to expand the "single linear feature" region model of Calway[16]. It is worth emphasising two points from this discussion.

First, the choice of feature models requires a trade-off between generality and computational demands. The local feature models that have been considered were chosen to increase the expressiveness of the image model. Arc features allow the modelling of regions containing boundaries of high, but constant, curvature - reducing the need to represent such a boundary segment as number of straight line segments. The use of the multiple linear feature model allows regions containing junction points or corners. It was noted that by using this set of feature models and systematically varying the analysis scale it is possible to guarantee that any image region can be modelled.

Secondly, a process of detecting features that can be modelled using the above feature classes was discussed. A process of estimation, synthesis and hypothesis testing, that may be compared to the statistical composite decision procedure [30, 66], was presented. Such an approach allows a decision to be made about which, if any, of the feature classes can be used to model a region, and an estimate of the feature parameters to be obtained.

Chapter 4

Feature Detection: Implementation

4.1 Implementation

In the previous chapter, feature models were defined for two classes of local image feature - linear features and arcs, and methods of detecting these and estimating their parameters were derived. The models and estimators are based on the properties of local spectrum estimates. For the purposes of this work, the estimates are provided by the Multiresolution Fourier Transform, as described in chapter 2. A given level of the MFT may be regarded as a set of estimates of local spectra corresponding to square regions of the image, the size of the regions being determined by the scale at that level. Sections 4.2 and 4.3 describe the implementation of these estimators using the discrete MFT.

The implementation of the feature detection algorithm uses the relaxed version of the MFT. Adjacent spatial regions therefore overlap by 50%, as shown in figure 4.1. The solid lines delineate the inner quarter of each region, while the dotted/dashed lines delineate the full region. Due to relaxation of the MFT window, the spatial sampling interval is N, while the size of the discrete local spectrum, and the spatial

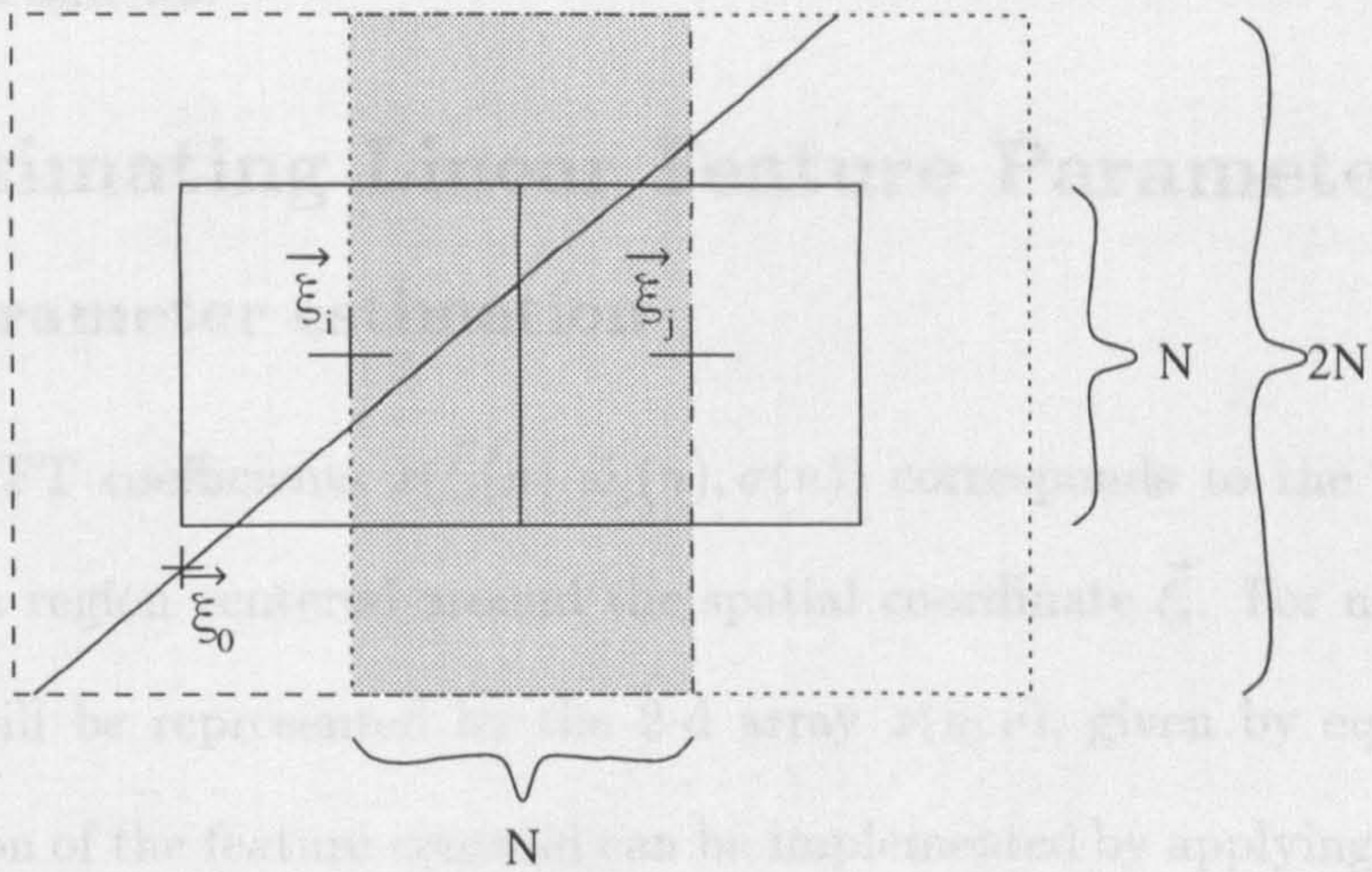


Figure 4.1: Adjacent Regions

region to which it corresponds, is $2N \times 2N$. The shaded area represents the portion of the image plane within both regions. Any feature in this region could be detected in both of the regions and so to make sure that any feature is only detected once, a feature is only regarded as detected in a region if it is at least partially in the inner quarter of that region. If it is outside this area, then it should be detected in one of its neighbours. The feature detection algorithm calculates, for a linear feature, its orientation and centroid relative to the region centre. However, this gives no indication of the length of the feature, or, if its centroid is in the outer region, whether it extends into the inner region.

Consider the feature shown in figure 4.1. Given only its centroid, $\vec{\xi}_0$, and orientation θ_0, no decision can be made as to whether it extends into the inner region and should be detected, or should be ignored by this region as it will be detected in a neighbour. In order to make this decision, estimates of the feature end points should be calculated, and methods for estimating these for the line and arc features is given

in sections 4.2 and 4.3.

4.2 Estimating Linear Feature Parameters

4.2.1 Parameter estimation

The set of MFT coefficients $\hat{x}(\vec{\xi}_i(n), \vec{\omega}_j(n), \sigma(n))$ corresponds to the sampled local spectrum of a region centered around the spatial coordinate $\vec{\xi}_i$. For notational simplicity this will be represented by the 2-d array $x(u,v)$, given by equation (2.28). The estimation of the feature centroid can be implemented by applying the equations for the correlation statistics, R_1 and R_2, given by equation (3.49), over the discrete frequency domain array, ie.

$$\begin{aligned} R_1 &= 0.5 \sum_{(u,v)} (d_{11}(u,v) + d_{12}(u,v)) && (4.1) \\ &= 0.5 \sum_{(u,v)} (\hat{x}(u,v)\hat{x}^*(u-1,v) + \hat{x}(u+1,v)\hat{x}^*(u,v)) && (4.2) \end{aligned}$$

and

$$\begin{aligned} R_2 &= 0.5 \sum_{(u,v)} (d_{21}(u,v) + d_{22}(u,v)) && (4.3) \\ &= 0.5 \sum_{(u,v)} (\hat{x}(u,v)\hat{x}^*(u,v-1) + \hat{x}(u,v+1)\hat{x}^*(u,v)) && (4.4) \end{aligned}$$

The centroid estimate is then given by equation (3.53),

$$\vec{\eta}_0 = \frac{1}{\Omega} \begin{bmatrix} \arg R_1 \\ \arg R_2 \end{bmatrix} \qquad (4.5)$$

This phase differencing approach is based upon a spectrum model that assumes linearly varying phase, given by equation (3.42). However the phase constant ε has a different sign in the two half planes defined by the feature orientation (section 3.1.1). Only when all the spectral coefficients are in the same halfplane, ie. the phase

constant for each coefficient has the same sign, does the linear model fit. Consider a vertical feature. Because of its orientation, the sign of the phase constant is different for coefficients on opposite sides of the v-axis. One of the phase difference terms, in the u-direction, is given by $\arg\left(d_{11}(N,v) + d_{12}(N,v)\right)$

$$d_{11}(N,v) = \hat{x}(N,v)\hat{x}^*(N-1,v) \tag{4.6}$$

and

$$d_{12}(N,v) = \hat{x}(N+1,v)\hat{x}^*(N,v) \tag{4.7}$$

If ψ_1 and ψ_2 represent the phase gradients, the phase of the first term is given by

$$\begin{aligned} \arg d_{11}(N,v) &= ((u-N+0.5)\psi_1 + (v-N+0.5)\psi_2 + \varepsilon) && (4.8)\\ &\quad - ((u-N+1.5)\psi_1 + (v-N+0.5)\psi_2 - \varepsilon) && (4.9)\\ &= \psi_1 + 2\varepsilon && (4.10) \end{aligned}$$

Since the sign of the phase constant is different for the two coefficients, the constants are summed. The other phase difference term, $d_{12}(N,v)$, behaves as expected since both coefficients are on the same side of the v-axis and the phase constants cancel out, ie.

$$\begin{aligned} \arg d_{12}(N,v) &= ((u-N+1.5)\psi_1 + (v-N+0.5)\psi_2 + \varepsilon) && (4.11)\\ &\quad - ((u-N+0.5)\psi_1 + (v-N+0.5)\psi_2 + \varepsilon) && (4.12)\\ &= \psi_1 && (4.13) \end{aligned}$$

As the phase of the summation of these two terms is a weighted mean of the two individual phases, the phase will be corrupted by the 2ε term in equation (4.8). Assuming that the phase difference terms have the same magnitude, that is

$$|d_{11}(N,v)| = |d_{12}(N,v)| \tag{4.14}$$

then the phase is

$$\arg\left(d_{11}(N,v) + d_{12}(N,v)\right) = \psi_1 + \varepsilon \tag{4.15}$$

If the feature concerned is an ideal line then

$$\varepsilon = 0 \tag{4.16}$$

and this term will have no effect on the centroid estimation process. For an ideal step edge however

$$\varepsilon = \pi/2 \tag{4.17}$$

and the error in the centroid estimated by this term alone would be $N/2$. The actual effect on the centroid estimate depends on the relative magnitudes of the phase difference terms. In order to avoid this error, the magnitudes of the four lowpass coefficients, ie. those given by

$$(u,v) \in (N-1,N-1) \vee (N,N-1) \vee (N-1,N) \vee (N,N) \tag{4.18}$$

are set to zero during the calculation of the feature centroid.

The orientation estimation follows directly from calculation of the moment of inertia tensor defined by equation (3.74). Since only the relative frequencies of the coefficient energies are important, the frequencies in equations (3.75) - (3.77) can be replaced by the array indices u and v shifted so as to be relative to dc. This gives the equations for the tensor components as

$$I_{11} = \sum_{u,v}(|\hat{x}(u,v)|(u-N+0.5))^2 \tag{4.19}$$

$$I_{12} = I_{21} = \sum_{u,v}|\hat{x}(u,v)|^2(u-N+0.5)(v-N+0.5) \tag{4.20}$$

$$I_{22} = \sum_{u,v}(|\hat{x}(u,v)|(v-N+0.5))^2 \tag{4.21}$$

Once the tensor has been constructed its principal axis can be calculated from equation (3.78) and the feature orientation determined.

The generation of the synthesised spectrum is performed by calculating the phase, using the feature model and estimated parameters, and taking the magnitudes from the actual coefficients. The estimation of the phase constant (equation 3.81) and the calculation of the correlation (equation 3.84) between the real and synthesised coefficients can be combined into one operation. The expression for the correlation (equation (3.84)) can be rewritten as

$$\gamma = \sum_{(u,v)\in\Theta_{\theta_0}} \tilde{x}(u,v)\hat{x}^*(u,v) \tag{4.22}$$

which differs from equation (3.84) in two ways. First, the summation is only over half the frequency plane, but since the spectrum is hermitian symmetric there is no loss of information. Secondly, the synthesised coefficients $\tilde{x}(u,v)$ are generated without the phase constant, ie

$$\tilde{x}(u,v) = |\hat{x}(u,v)| \exp[\jmath\vec{\eta}_0 \cdot \vec{\omega}_{u,v}] \tag{4.23}$$

(cf. equation (3.85)). Substituting for the real coefficients as given by equation (3.42), and rearranging the exponential terms gives

$$\gamma = \exp[\jmath\varepsilon] \sum_{(u,v)\in\Theta_{\theta_0}} |\hat{x}(u,v)|^2 \exp[\jmath\vec{\xi}_0 \cdot \vec{\omega}_{u,v}] \exp[-\jmath\vec{\eta}_0 \cdot \vec{\omega}_{u,v}] \tag{4.24}$$

Ignoring the constant exponential term, this has the same form as that in equation (3.86), and should have half the magnitude. The correlation coefficient given by equation (3.86) can therefore be calculated from γ by the expression

$$|r| = 2|\gamma| \tag{4.25}$$

Expanding the expression for the estimation of the phase constant, equation (3.81), in terms of the 2-d array and rearranging gives

$$\tilde{\varepsilon} = \arg \exp[\jmath\varepsilon] \sum_{(u,v)\in\Theta_{\theta_0}} |\hat{x}(u,v)|^2 \exp[\jmath\vec{\xi}_0 \cdot \vec{\omega}_{u,v}] \exp[-\jmath\vec{\eta}_0 \cdot \vec{\omega}_{u,v}] \quad (4.26)$$

$$= \arg \gamma \quad (4.27)$$

The phase of γ can thus be used to give the phase constant of the feature spectrum.

4.2.2 Multiple Features

The segmentation process, for detecting multiple linear features, may be implemented almost directly as described in chapter 3. For each coefficient a centroid estimate is given by equation (3.108) which can be rewritten in terms of the discrete MFT spectrum as

$$\vec{\eta}(u,v) = \frac{1}{\Omega} \begin{bmatrix} \arg(d_{11}(u,v) + d_{11}(u,v)) \\ \arg(d_{21}(u,v) + d_{22}(u,v)) \end{bmatrix} \quad (4.28)$$

Since $\vec{d}_1(u,v)$ and $\vec{d}_2(u,v)$ have been previously calculated when attempting to fit the single feature model, they need not be recalculated. Two initial feature centroid estimates are produced by adding small offset vectors to the centroid estimated while trying to detect a single linear feature. These two centroids are denoted $\vec{\eta}_1$ and $\vec{\eta}_2$. Each coefficient is initially assigned to one of two sets, I_1 or I_2, each associated with one of the initial centroids, according to which centroid it is closest to, ie. for which the error term

$$e_k(u,v) = (\eta_{k1} - \eta_1(u,v))^2 + (\eta_{k2} - \eta_2(u,v))^2, \quad \text{for } k = 1,2 \quad (4.29)$$

is minimum.

The centroids can then be recalculated by evaluating the energy weighted mean

given by

$$\vec{\eta}_k = \frac{\sum_{(u,v)\in I_k} |\hat{x}(u,v)|^2 \vec{\eta}(u,v)}{\sum_{(u,v)\in I_k} |\hat{x}(u,v)|^2} \tag{4.30}$$

over each set of coefficients. The process is repeated until only some small proportion of coefficients are moved from one set to the other. Once the segmentation has converged, each set is tested for a linear phase property using the magnitudes of R_{k1} and R_{k2}, given by equations (4.1) and (4.3) for the two coefficient sets. A given set I_k will be considered to have linear phase if, from equation (3.65),

$$R_{k1} > \rho \sum_{(u,v)\in I_k} |d_{11}(u,v)| + |d_{12}(u,v)| \tag{4.31}$$

and

$$R_{k2} > \rho \sum_{(u,v)\in I_k} |d_{21}(u,v)| + |d_{22}(u,v)| \tag{4.32}$$

where ρ is a threshold in the interval $[0,1]$, which determines the strictness of the linear phase test. If any cluster does not pass the test then its centroid is replaced with two centroids displaced from the original. The coefficient assignment process is then restarted until it reconverges. This process continues until either all the clusters pass the linear feature test, or else some maximum number of clusters is reached.

Each set of coefficients is subjected to the orientation and phase constant estimation procedures. The spectrum can then be synthesised using, for each coefficient, the set of parameters of the cluster to which it belongs.

4.2.3 Feature Extent

Having estimated the centroid and orientation of the feature, $\vec{\xi}_0$ and θ_0, the next step is to estimate which portion of it lies within the region. In many cases the feature will be part of a larger scale boundary feature and as such will pass straight through

the region, ie. its endpoints will be at the edges of the region. In other cases the feature may form part of a corner in a region, and therefore one of its endpoints will be within the region.

To avoid the complexities of building the spatial window into the feature model, it is assumed that the window may be regarded as flat, and have a size between that of the inner and outer regions. The size of this intermediate region is denoted $N' \times N'$ where

$$N \leq N' \leq 2N \tag{4.33}$$

A further simplifying assumption is that the feature occupies more than one region. Thus the length and end-points of the feature are determined by its orientation and the position of the centroid within the $N' \times N'$ region. No attempt is made to calculate these parameters directly from the local spectrum.

The estimation of the endpoints can therefore be divided into three stages, illustrated by figure (4.2):

1 Assuming that the feature is of infinite length calculate its endpoints when clipped to the spatial region, $(\vec{\eta_1'}, \vec{\eta_2'})$, by applying a version of the Liang-Barsky line clipping algorithm [59].

2 The feature centroid is midway between the endpoints of the feature. The feature should, therefore, be truncated so that the endpoints are equidistant from the centroid - $(\vec{\eta_1}, \vec{\eta_2})$.

3 Since the spatial region of interest is the $N \times N$ inner region, the feature resulting from step 2 is clipped at the boundary of this region, by a further application of the clipping algorithm, to give the line $(\vec{\xi_1}, \vec{\xi_2})$.

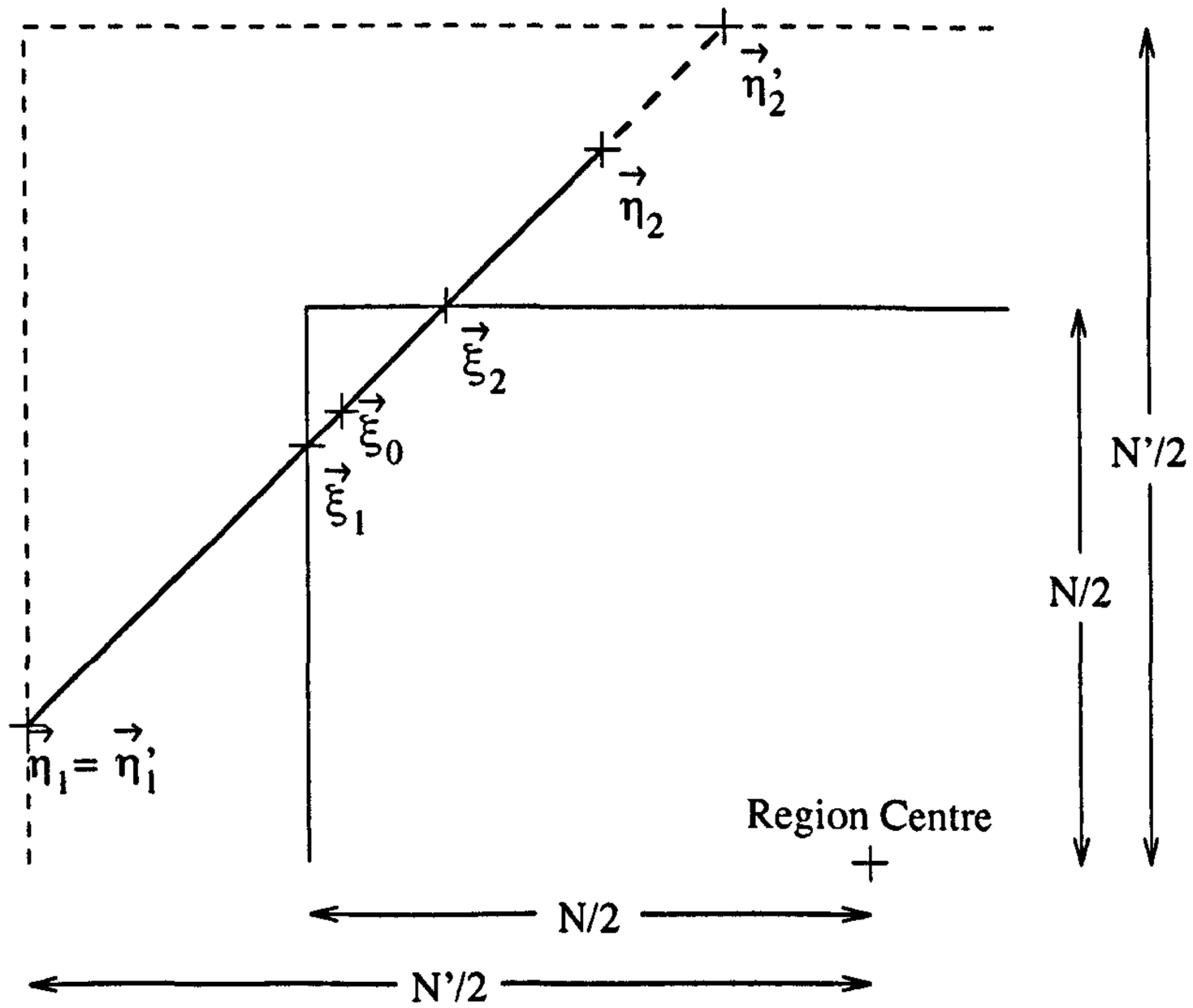


Figure 4.2: Calculation of end points

The Liang-Barsky line clipping algorithm [59, 43] is based upon a parametric expression of a straight line. Since the orientation and position of the feature have been estimated, it is possible to express the equation of the line in parametric form, ie.

$$\vec{\xi} = \vec{\xi}_0 + l\vec{v}_{\theta_0} \tag{4.34}$$

where, if L is the feature length, $-L/2 \leq l \leq L/2$. For any point on the line and in the region the following conditions must hold

$$-\frac{N'}{2} \leq \xi_{01} + l\cos\theta_0 \leq \frac{N'}{2} \tag{4.35}$$

$$-\frac{N'}{2} \leq \xi_{02} + l\sin\theta_0 \leq \frac{N'}{2} \tag{4.36}$$

These inequalities can be rewritten (cf. [43]) as

$$p_k l \leq q_k, \quad k = 1, 2, 3, 4 \tag{4.37}$$

where

$$p_1 = -\cos\theta_0, \quad q_1 = \xi_{01} + \frac{N'}{2} \tag{4.38}$$

$$p_2 = \cos\theta_0, \quad q_2 = \frac{N'}{2} - \xi_{01} \tag{4.39}$$

$$p_3 = -\sin\theta_0, \quad q_3 = \xi_{02} + \frac{N'}{2} \tag{4.40}$$

$$p_4 = \sin\theta_0, \quad q_4 = \frac{N'}{2} - \xi_{02} \tag{4.41}$$

As noted by Hearn and Baker, each value of k corresponds to one edge of the region (1 the left, 2 the right, 3 the bottom and 4 the top). If $p_k = 0$ then the feature is parallel to the corresponding edge. If additionally $q_k < 0$ then the feature is outside of the region and can be ignored. If $p_k < 0$ the line crosses the infinitely extended boundary edge from the outside to inside of the region as l increases. If $p_k > 0$ then the feature crosses from inside to outside. The parameter for the point of intersection with the k boundary is given by

$$l = \frac{q_k}{p_k} \tag{4.42}$$

Assuming that the feature passes through the region, the end points of the feature, when it is clipped to the region, $\vec{\eta}_1'$ and $\vec{\eta}_2'$, are the points where the feature enters and leaves the region respectively, and correspond to the parameters l_1' and l_2'. Therefore l_1' is the maximum value of the set of l values for which $p_k < 0$. Similarly, l_2' is the minimum of the set of l values for which $p_k > 0$. If $l_1' > l_2'$ then the line is outside of the region and can be ignored - it should be detected in a neighbouring region. The full details of implementation of this algorithm are given by Hearn and Baker [43], where it is shown to be an efficient method of line clipping.

Having estimated the points where the line, if infinitely extended, crosses the region boundary, the line should be truncated so that both endpoints are equidistant from

the centroid. Since the values of l correspond to the displacement of the point from the feature centroid the parameters l_1'' and l_2'', corresponding to the feature endpoints, $\vec{\eta}_1$ and $\vec{\eta}_2$ in figure 4.2, should have the same magnitude, and are therefore given by

$$l_1'' = \mathrm{sgn}(l_1') \min(|l_1'|, |l_2'|) \tag{4.43}$$

and

$$l_2'' = \mathrm{sgn}(l_2') \min(|l_1'|, |l_2'|) \tag{4.44}$$

These parameters correspond to the feature as detected in the larger region and so this feature must be clipped to the inner $N \times N$ region, to give the endpoint parameters l_1 and l_2. This is achieved by a second application of the clipping algorithm, this time using the smaller region. Once the endpoint parameters, l_1 and l_2, have been calculated the endpoints are given, from equation 4.34, by

$$\vec{\xi}_1 = \vec{\xi}_0 + l_1 \vec{v}_{\theta_0} \tag{4.45}$$

and

$$\vec{\xi}_2 = \vec{\xi}_0 + l_2 \vec{v}_{\theta_0} \tag{4.46}$$

4.3 Estimating Arc Feature Parameters

4.3.1 Parameter Estimation

The parameter estimation process for the arc detection procedure can be implemented by performing the calculations given by equations (3.131) and (3.137) for the discrete array of coefficients. If the backwards phase differences for a spectrum coefficient $\hat{x}(u,v)$ are given by $\vec{\psi}(u,v)$,

$$\vec{\psi}(u,v) = \begin{bmatrix} \arg d_{11}(u,v) \\ \arg d_{21}(u,v) \end{bmatrix} \tag{4.47}$$

$$= \begin{bmatrix} \arg \hat{x}(u,v)\hat{x}^*(u-1,v) \\ \arg \hat{x}(u,v)\hat{x}^*(u,v-1) \end{bmatrix} \tag{4.48}$$

then the centre estimation is given by

$$\vec{\eta}_0 = \frac{1}{\Omega} T^{-1} \sum_{(u,v)} T(u,v)\vec{\psi}(u,v) \tag{4.49}$$

where

$$T = \sum_{(u,v)} T(u,v) \tag{4.50}$$

and $T(u,v)$ is a magnitude weighted tensor defined for each coefficient as

$$T(u,v) = |\hat{x}(u,v)|\vec{g}(u,v)\vec{g}^T(u,v) \tag{4.51}$$

where

$$\vec{g}(u,v) = \begin{bmatrix} \|\vec{\omega}_{(u,v)} - \vec{\Omega}_2\| - \|\vec{\omega}_{(u,v)}\| \\ \|\vec{\omega}_{(u,v)}\| - \|\vec{\omega}_{(u,v)} - \vec{\Omega}_1\| \end{bmatrix} \tag{4.52}$$

These vectors can be generated for appropriate values of N and simply "looked up" when required. The vector components can be rewritten in terms of u and v by substituting from equations (2.31), to give

$$\begin{aligned} \|\vec{\omega}_{(u,v)}\| &= \Omega \left\| \begin{bmatrix} u - N + 0.5 \\ v - N + 0.5 \end{bmatrix} \right\| \\ &= \Omega \sqrt{(u - N + 0.5)^2 + (v - N + 0.5)^2} \end{aligned} \tag{4.53}$$

$$\begin{aligned} \|\vec{\omega}_{(u,v)} - \vec{\Omega}_1\| &= \Omega \left\| \begin{bmatrix} u - N + 0.5 \\ v - N + 0.5 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\| \\ &= \Omega \sqrt{(u - N - 0.5)^2 + (v - N + 0.5)^2} \end{aligned} \tag{4.54}$$

$$\begin{aligned} \|\vec{\omega}_{(u,v)} - \vec{\Omega}_2\| &= \Omega \left\| \begin{bmatrix} u - N + 0.5 \\ v - N + 0.5 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\| \\ &= \Omega \sqrt{(u - N + 0.5)^2 + (v - N - 0.5)^2} \end{aligned} \tag{4.55}$$

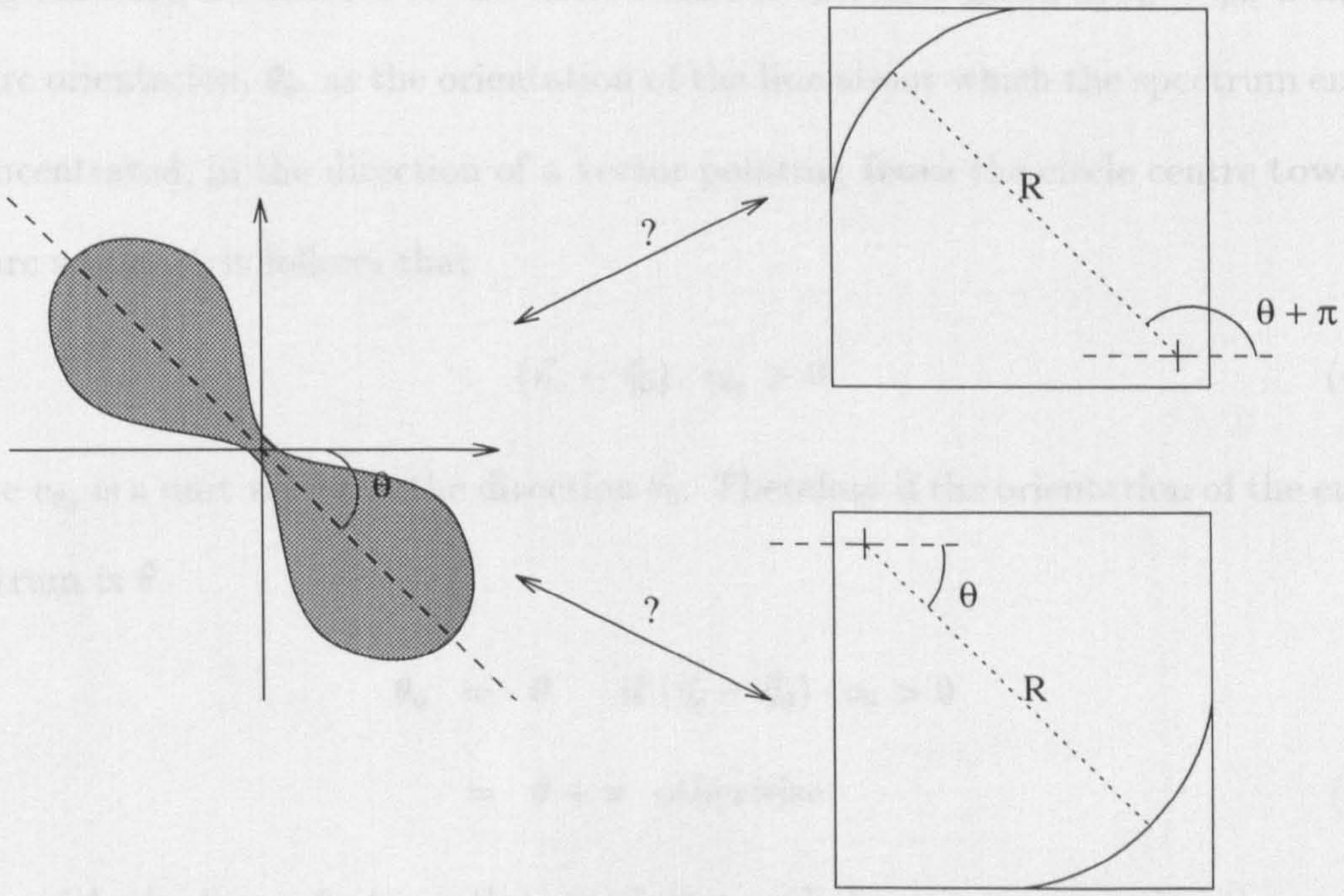


Figure 4.3: Two possible arcs with the same energy spectrum

The estimation of the radius is calculated by evaluating the expression in equation (3.137) over the array, ie.

$$\tilde{R} = \frac{\sum_{(u,v)} |\hat{x}(u,v)|^2 g_2(u,v)(\psi_1(u,v) - \Omega\eta_{01})}{\sum_{(u,v)} |\hat{x}(u,v)|^2 g_2^2(u,v)} \tag{4.56}$$

The orientation of the arc segment can be calculated using the moment of inertia tensor, as used for linear features. However, as this is based only upon the energy spectrum, it only gives an orientation in the interval $[-\pi/2, \pi/2]$ and will make no distinction between the arcs shown in figure 4.3, which have the same energy spectrum. In one case, the orientation is θ and in the other $\theta + \pi$. However, the centroid estimate, $\vec{\eta}_c$, generated when testing for a single linear feature, equation (4.5), indicates approximately the position of the arc relative to the region centre. A vector

giving the centroid relative to the circle centre is therefore given by $\vec{\eta}_c - \vec{\eta}_0$. Defining the arc orientation, θ_0, as the orientation of the line about which the spectrum energy is concentrated, in the direction of a vector pointing **from** the circle centre **towards** the arc segment, it follows that

$$(\vec{\eta}_c - \vec{\eta}_0) \cdot v_{\theta_0} > 0 \tag{4.57}$$

where v_{θ_0} is a unit vector in the direction θ_0. Therefore if the orientation of the energy spectrum is θ

$$\begin{aligned} \theta_0 &= \theta \quad \text{if } (\vec{\eta}_c - \vec{\eta}_0) \cdot v_\theta > 0 \\ &= \theta + \pi \quad \text{otherwise} \end{aligned} \tag{4.58}$$

As with the linear features the correlation and the parameter estimation process can be combined into a single step. Both can be obtained from a single statistic γ , the inner product between the actual spectral coefficients $\hat{x}(u,v)$ and a set of coefficients synthesised from the estimated parameters $\tilde{x}(u,v)$. This is given by

$$\gamma = \sum_{(u,v)} \hat{x}(u,v)\tilde{x}^*(u,v) \tag{4.59}$$

$$= \sum_{(u,v)} |\hat{x}(u,v)| \exp -\jmath[(\tilde{R} - R)\|\vec{\omega}_{(u,v)}\| + (\vec{\xi}_0 - \vec{\eta}_0) \cdot \vec{\omega}_{(u,v)} + \varepsilon] \tag{4.60}$$

Substituting with r as given by equation (3.122) and rearranging gives

$$\gamma = \exp[-\jmath\varepsilon]\frac{r}{2} \tag{4.61}$$

The correlation r is therefore given by

$$|r| = 2|\gamma| \tag{4.62}$$

and the phase constant estimate $\tilde{\varepsilon}$ by

$$\tilde{\varepsilon} = \arg\gamma \tag{4.63}$$

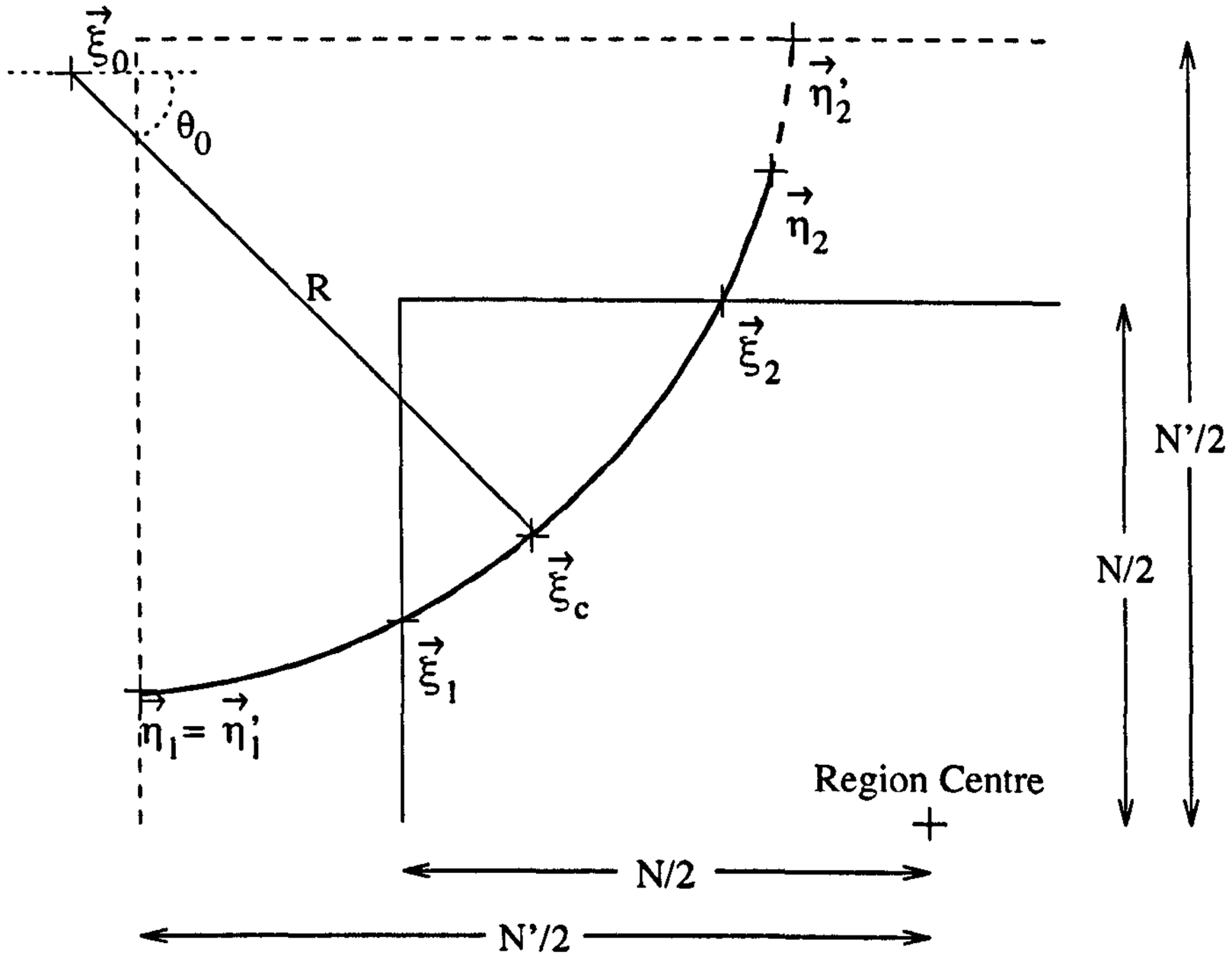


Figure 4.4: Calculation of arc end points

4.3.2 Feature Extent

The next step is to compute the end points for the arc segment. Again it is assumed that the local spectrum corresponds to an $N' \times N'$ region, in which an arc segment has been detected, and that at least one end of the arc continues out of this region.

A three stage process can be used to estimate these end points, as illustrated in figure (4.4):

1 Assuming that the arc continues out of the larger $N' \times N'$ region clip it to the region by using an algorithm similar to the Liang-Barsky line clipping algorithm used for linear features, this gives a single section of arc with end points $\vec{\eta}_1'$ and $\vec{\eta}_2'$, and containing the centre point $\vec{\xi}_c$.

2 The arc centre is midway between the endpoints of the feature. The feature should,

therefore, be truncated so that the endpoints are equidistant from $\vec{\xi}_c$, giving end points $\vec{\eta}_1$ and $\vec{\eta}_2$.

3 Since the spatial region of interest is the $N \times N$ inner region, the feature resulting from step 2 is clipped at the boundary of this region, by a further application of the clipping algorithm, to give the arc $(\vec{\xi}_1, \vec{\xi}_2)$.

A parametric expression for the arc is given by

$$\vec{\xi} = \vec{\xi}_0 + R\vec{v}_{(\theta_0+\theta)} \tag{4.64}$$

where θ_0 is the parameter that gives the centre of the arc, θ is a parameter that sweeps out the arc as it increases from $-\Theta/2$ to $\Theta/2$. For any point on the arc within the region the following conditions must hold

$$-\frac{N'}{2} \leq \quad \xi_{01} + R\cos(\theta_0 + \theta) \quad \leq \frac{N'}{2} \tag{4.65}$$

$$-\frac{N'}{2} \leq \quad \xi_{02} + R\sin(\theta_0 + \theta) \quad \leq \frac{N'}{2} \tag{4.66}$$

The inequalities can be re-expressed as

$$p_k(\theta_0 + \theta)R \leq q_k, \quad k = 1, 2, 3, 4 \tag{4.67}$$

where

$$p_1(\theta) = \quad -\cos\theta, \quad q_1 = (\frac{N'}{2} + \xi_{01}) \tag{4.68}$$

$$p_2(\theta) = \quad \cos\theta, \quad q_2 = (\frac{N'}{2} - \xi_{01}) \tag{4.69}$$

$$p_3(\theta) = \quad -\sin\theta, \quad q_3 = (\frac{N'}{2} + \xi_{02}) \tag{4.70}$$

$$p_4(\theta) = \quad \sin\theta, \quad q_4 = (\frac{N'}{2} - \xi_{02}) \tag{4.71}$$

Each value of k corresponds to one edge of the region, the left, right, top and bottom respectively. The value of θ corresponding to each of the points where the arc intersects with the region boundaries is given by

$$p_k(\theta_0 + \theta) = \frac{1}{R} q_k \tag{4.72}$$

Depending on the position and size of the arc, for each k this equation will either be invalid (q_k will be outside of the interval $[-R, R]$) or else have two solutions corresponding to the two intersection points of the arc and the edge. There are, therefore, up to eight intersection points. For each solution, the derivative of $p_k(\theta_0+\theta)$, $p'_k(\theta_0 + \theta)$, indicates whether the feature is entering or leaving the region as θ is increased. If $p'_k(\theta_0 + \theta) > 0$ then the arc crosses the boundary from inside to outside the region as θ increases.

The two end parameters, θ'_1 and θ'_2, correspond to the two intersections which are closest to the centre point, given by $\theta = 0$. The start parameter, θ'_1, is the maximum value from the set of solutions for which $p'_k(\theta_0 + \theta) < 0$. Similarly θ'_2 is the minimum from those for which $p'_k(\theta_0 + \theta) > 0$. The parameters correspond to the points $\vec{\eta}'_1$ and $\vec{\eta}'_2$ in figure 4.4.

Since the two endpoints should be equidistant from the centre point $\vec{\eta}_c$, the two parameters corresponding to the endpoints should have equal magnitude, ie.

$$\theta''_1 = -\theta''_2 \tag{4.73}$$

These can then be derived from the intersection parameters by

$$\theta''_1 = \mathrm{sgn}(\theta'_1) \min(|\theta'_1|, |\theta'_2|) \tag{4.74}$$

and

$$\theta_2'' = \mathrm{sgn}(\theta_2') \min(|\theta_1'|, |\theta_2'|) \tag{4.75}$$

The third step is to clip the arc to the inner $N \times N$ region. This can be achieved by reapplying the arc clipping algorithm described above. This gives the end parameters θ_1 and θ_2, which can be used to give the end points

$$\vec{\xi}_1 = \vec{\xi}_0 + R\vec{v}_{(\theta_0+\theta_1)} \tag{4.76}$$

and

$$\vec{\xi}_2 = \vec{\xi}_0 + R\vec{v}_{(\theta_0+\theta_2)} \tag{4.77}$$

4.4 Computational Requirements

Assuming that a local spectrum estimate has already been calculated, using the MFT, this section considers the additional computational demand of applying the local feature detection algorithm. The analysis is in terms of floating point multiplication operations. For the sake of simplicity, it is assumed that complex values are held in a cartesian form and that each complex multiplication is therefore equivalent to 4 floating point multiplications.

The number of multiplications required to perform the single linear feature extraction process is summarised in table 4.1. The centroid estimation is implemented by equations (4.1) and (4.3), and therefore requires 4 complex multiplications per coefficient. For the tensor calculation the energy for each coefficient need only be calculated once and then, multiplied by the frequency index terms for each of equations (4.19), (4.20) and (4.21). This requires 4 multiplications. The synthesis of each coefficient requires 5 floating point multiplications. The final step is the calculation

Procedure		Muliplications per coefficient
Linear Feature Detection	Centroid Estimation	16
	Orientation Tensor	5
	Spectrum Synthesis	5
	Calculation of γ	4
	Total	30
Arc Detection	Centre Estimation	6
	Radius Estimation	4
	Spectrum Synthesis	5
	Calculation of γ	4
	Total	19
Total for both arc and Linear Features		49

Table 4.1: Muliplications per coefficient required for feature detection

of γ using equation (4.22). Again this is 1 complex multiply per coefficient. The total computational requirement for the application of the single linear feature detector, including a measure of its accuracy in modelling the spatial region, is therefore 30 complex multiplications per coefficient.

The arc detection is performed after the linear feature process, and therefore some calculations, such as the backward conjugate products required for implementation of equation (4.47), have already been calculated and require no further computation. Additionally, it may be assumed that the values of the vector $\vec{g}(u,v)$, and the tensor formed by its outer product with itself, have been precalculated and may be looked up. Therefore calculation of each term in the summation for estimation of the arc centre, given by equation (4.49), only requires 6 multiplications per coefficient. As the energy for each coefficient has also been previously calculated, the evaluation of equation (4.56), in order to estimate the radius of the arc, requires 4 multiplications per coefficient. The spectrum synthesis, as for the linear feature case requires 5 multiplications per sample. The computation of γ, in order to estimate the phase constant

and correlation coefficient, requires 1 complex multiply (ie. 4 floating point multiplications) per coefficient. Therefore if the linear feature detector has previously been applied to a local spectrum, the additional application of the arc detector requires an extra 19 multiplications. This is summarised in table 4.1.

If the arc detection process were performed in isolation, ie. the linear feature detector had not previously been applied, then they have similar computational requirements. This can be compared to the Hough transform approach, where the computational requirement is proportional to n^N, N being the dimension of the Hough space, ie. the number of feature parameters, and n the number of quantising bins in each dimension. Thus the change of Hough transform detection process from line to circle detection gives an increase in computational complexity from $O(n^2)$ to $O(n^3)$.

The segmentation algorithm may be similarly analysed. The distance calculation of equation (4.29) must be applied for each coefficient and each cluster. Assuming that there are K clusters, this operation is therefore equivalent to $2K$ multiplications. The recalculation of the centroid is performed using an energy weighted average as given by equation (4.30), and therefore requires 2 multiplications per coefficient. Thus for each iteration of the K-means algorithm, $2 + 2K$ multiplications are required. If the algorithm converges after $\mathcal{I}$ iterations then K-means segmentation requires $\mathcal{I}(2 + 2K)$ multiplications. If there are actually $\mathcal{K}$ features in the region, and the K-means routine is applied for $K = 2, .., \mathcal{K}$, the number of multiplications per coefficient required for the segmentation is

$$\mathcal{N}_{\mathcal{K}} = \sum_{K=2}^{\mathcal{K}} \mathcal{I}_K(2 + 2K) \tag{4.78}$$

where $\mathcal{I}_K$ is the number of iterations for a given K. The actual values for this

Level	MFT calculation	Total feature detection
2	80	238
3	96	254
4	112	270

Table 4.2: Muliplications per coefficient for MFT generation and feature detection (256 × 256 image)

are obviously dependent on the image. Experimental results for regions containing two dominant features, such as the corners in the shapes image (figure 2.22), show convergence in approximately 5 iterations. This gives $\mathcal{N}_2 = 30$. Since the single linear feature, and arc feature, detectors have already been applied gives a total of approximately 79 multiplications per coefficient.

Due to oversampling each $N \times N$ local spectrum, corresponds to an exclusive $\frac{1}{2}N \times \frac{1}{2}N$ region of the image. Additionally, due to hermitian symmetry, it is only necessary to apply the above calculations to half of the local spectrum, ie $\frac{1}{2}N^2$ coefficients. Therefore this operation involves 158 multiplications per pixel, approximately equivalent to filtering the image using an 13 × 13 filter kernel.

From Calway [16] the number of complex multiplications required to generate a single level, n, of the MFT for an $M \times M$ image is given by

$$\mathcal{N}_n = 2^k M^2(1+n) \tag{4.79}$$

where k is the relaxation coefficient from equation (2.32). It is assumed that $k = 1$, for the relaxed MFT used in the current work. An initial DFT of an image has also to be generated and if generated using an FFT routine this requires an additional $M^2 \log_2 M$ operations[78]. Table 4.2 summarises the number of calculations required per pixel to generate a single MFT level (including initial DFT calculation) for a

number of levels of a 256 image. Additionally the requirement for generating the MFT and applying the feature detection processes is included. Even at the most computationally demanding level ($n = 4$) this is still approximately equivalent to convolution with a 17×17 filter.

4.5 Experimental Results

The feature detection processes were implemented as described above. The methods were applied to the three test images shown in section 2.4 - the "Shapes" image (figure 2.22), "Lena" (figure 2.26) and the "Vine leaf" image (figure 2.29).

Figures 4.5 and 4.6 show the level 2 MFT coefficients phase differenced along the two frequences axes, $d_1(\vec{\omega}_i)$ and $d_2(\vec{\omega}_i)$, as given by equation (3.48). Colour is used to indicate the phase of these coefficients and it is clear that for regions that fit the linear feature model the coefficients that correspond to a single linear feature have a constant phase, ie. are of a constant colour. This is obvious for the edges of both the square and triangle. The regions corresponding to parts of the circle have spectra that are more spread out, however the phase is approximately constant along radial lines from the origins of the local spectra, as would be expected from the phase model given by equation (3.115).

Figures 4.7 to 4.9 show the results of applying the feature detection algorithm to each of three MFT levels (2-4) for the "Shapes" image. A grid is overlaid on the result to indicate the spatial regions in which the features have been detected. The features drawn in green have been detected as arcs and those drawn in red as linear features. The straight lines and arcs were drawn using versions of the Bresenham line and circle algorithms[43] to join the estimated feature endpoints.

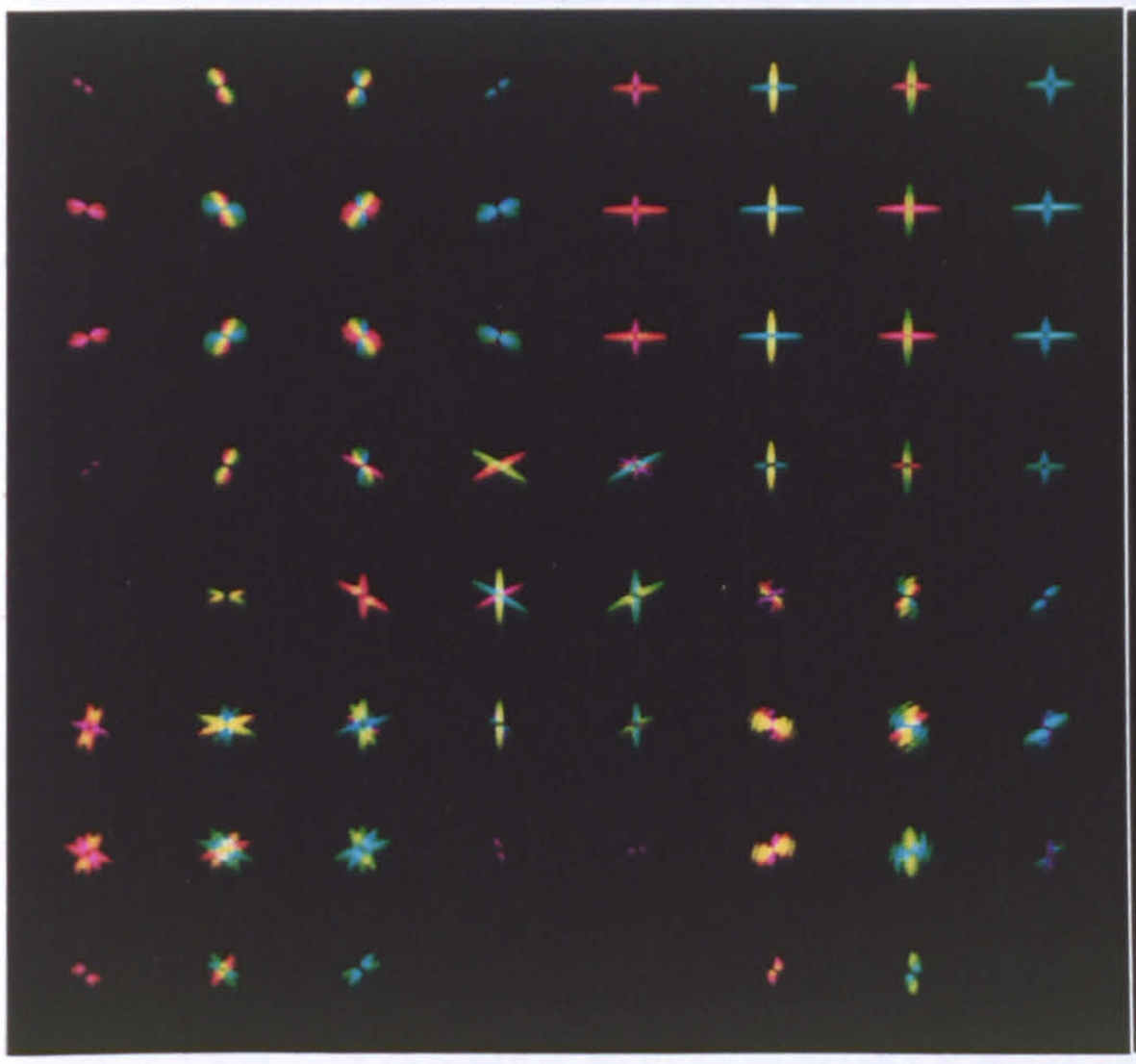


Figure 4.5: Shapes: Level 2
Horizontal phase differences

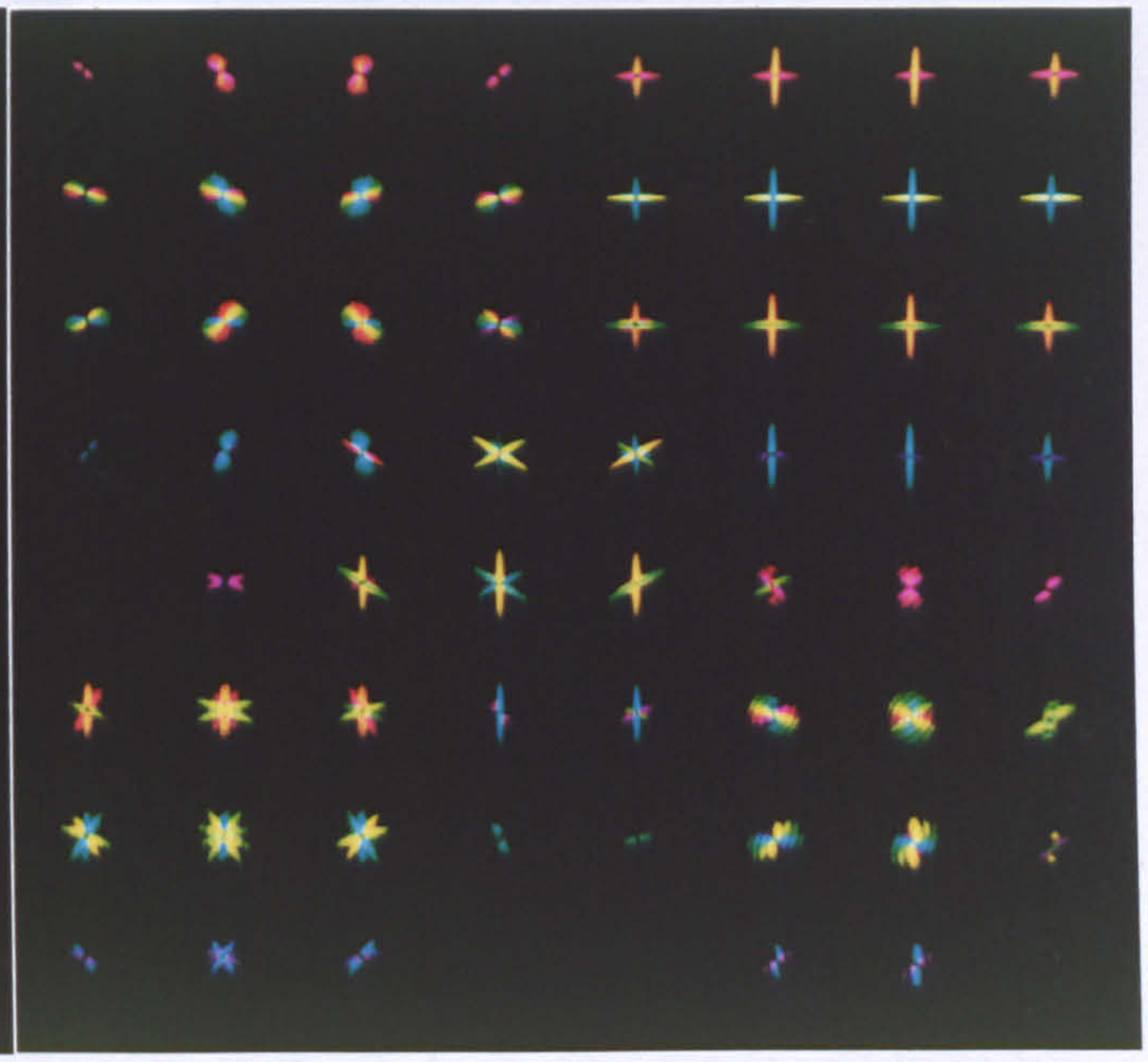


Figure 4.6: Shapes: Level 2
Vertical phase differences

At level 2 (figure 4.7), it can be seen that all of the square and triangle edges have been successfully detected in the correct orientation and position. The left hand edge of the square is positioned at the very edge of the blocks in which it has been detected and is hidden by the grid line. Most of the circle has been detected at this level. Part of the top of the circle has been detected as a linear feature since only a small part of it passes through the region and therefore there is only a small change in orientation across the region. The part of the circle that is not detected at this level is in a region that is close to the top of the triangle. Since the window used to generate the MFT is not spatially truncated the local spectrum would be influenced both by part of the circle and the edge of the triangle that is in the same orientation as part of the arc. It was noted in chapter 3 that a region can only be modelled if it contains no more

than one feature in any given orientation. It is also for this reason that no parts of the star or crescent are detected at this level.

The results of applying the algorithm to level 3 are shown in figure 4.8. At this level the square and triangle edges are once again completely detected. The circle is also completely detected at this level, however some parts are detected as linear features as opposed to arcs. Again this occurs when the size of the region leads to it containing only a small arc section, with only a small range of orientations. Parts of the crescent are detected (both as arcs and linear features) also. The edges of the star are completely detected at this level. On both the crescent and the star corners which are very acute, ie. are between two edges of similar orientation, are detected as a single line feature because this is how they appear locally within the regions where they are detected.

Figure 4.9 shows the results for level 4. In this image, although most of the features are detected, they are mainly detected as linear features due to the small region size.

Figures 4.10 to 4.12 show the results for the "Lena" image. Once again the algorithm has been applied to each of MFT levels 3 to 5. The larger scale features, such as parts of the hat and the mirror, are detected at level 3, but the more complex regions, corresponding to the feathers etc, do not fit the local models at this scale. At higher scales, with smaller spatial regions, more of the fine detail is detected, but as for the "Shapes", less of the curved features are detected as arcs when smaller image regions are considered.

It is also clear that, even at the highest level the very top of the edge of the hat is not detected. This part of the hat is a very low energy feature and when considered locally is very difficult to distinguish from the background noise in this part of the

image. In order to detect it more global consideration must be given to the edge of the hat. This is considered in chapter 5.

It was noted in section 2.4 that because of the method of computing the MFT an artificial edge was placed around the image boundary. When generating these results for "Lena" (and for the "Vine leaf") a simple routine has deleted all the features detected as running along the image boundary as these are assumed to be such artifacts.

The results obtained by applying the algorithm to the "Vine leaf" image are shown in figures 4.13 to 4.15. Few of the image features are detected at level 3, but most of the outline and much of the inner detail is detected at the higher scales.

Comparing the results with those obtained by Calway[16] indicates that the use of the extended models gives better curves, especially in regions containing boundaries of high curvature. This is most notable for the circle boundary in the "Shapes" image, compared to Calways circle result.

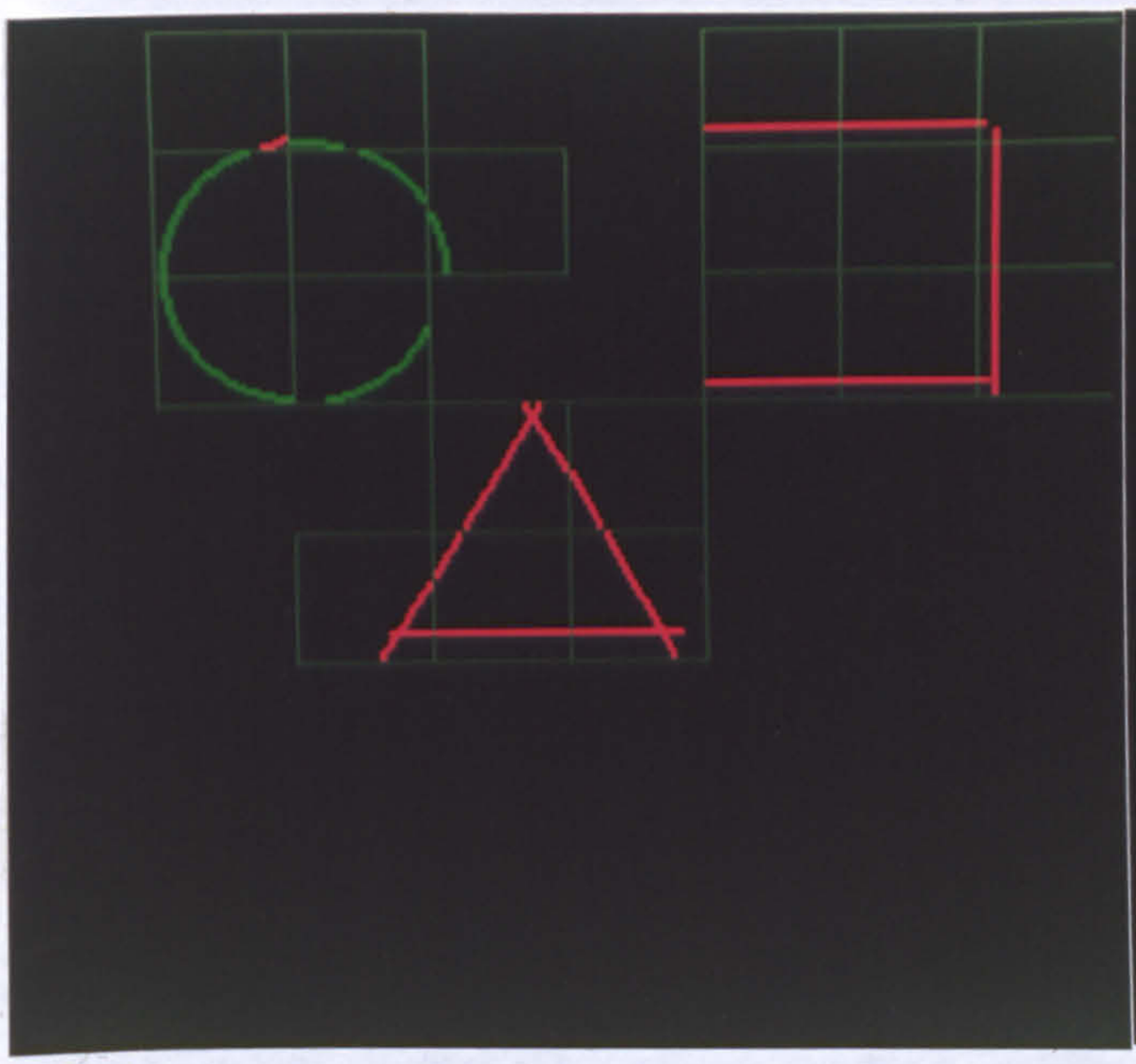


Figure 4.7: Shapes: Feature detection - Level 2

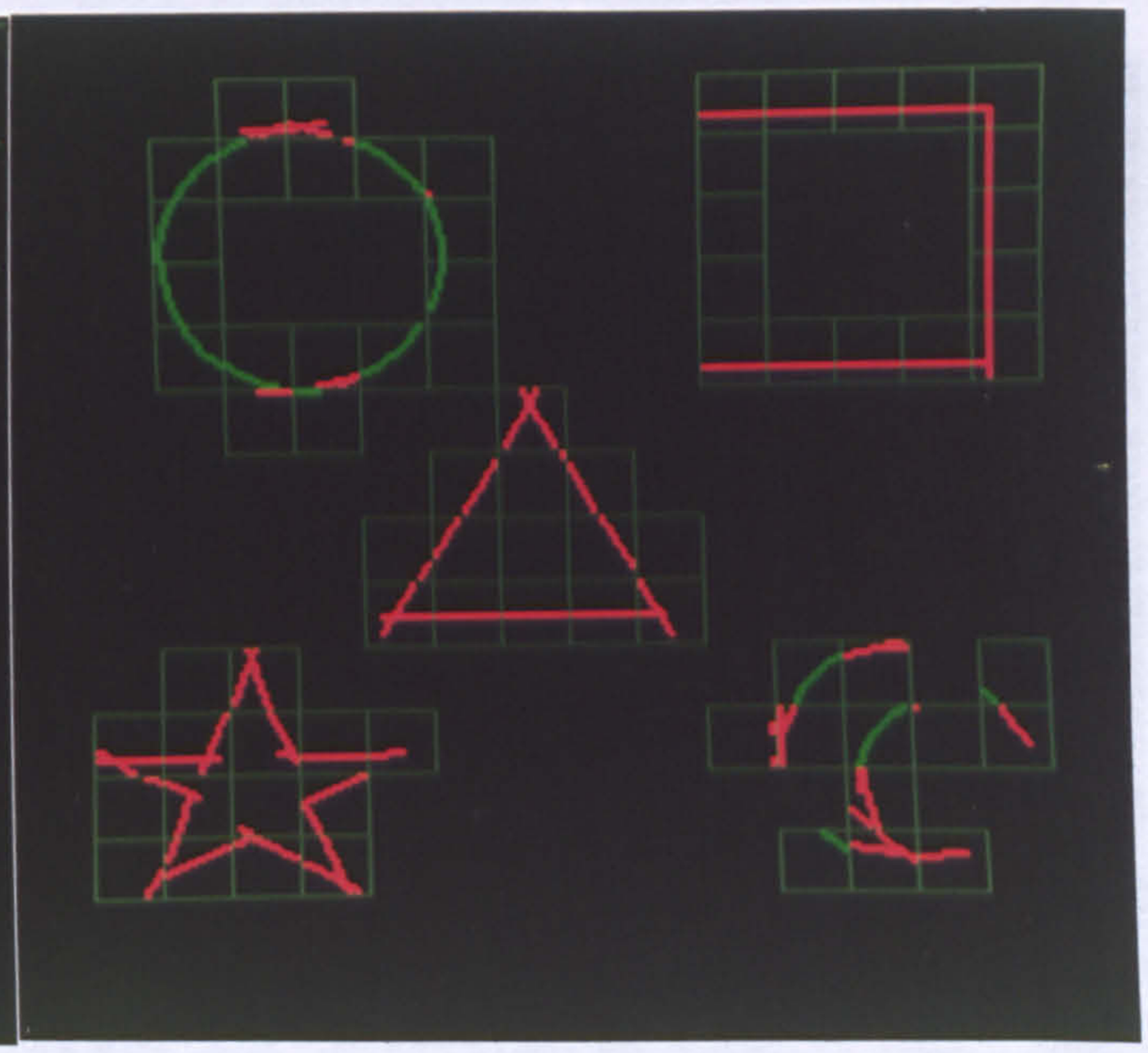


Figure 4.8: Shapes: Feature detection - Level 3

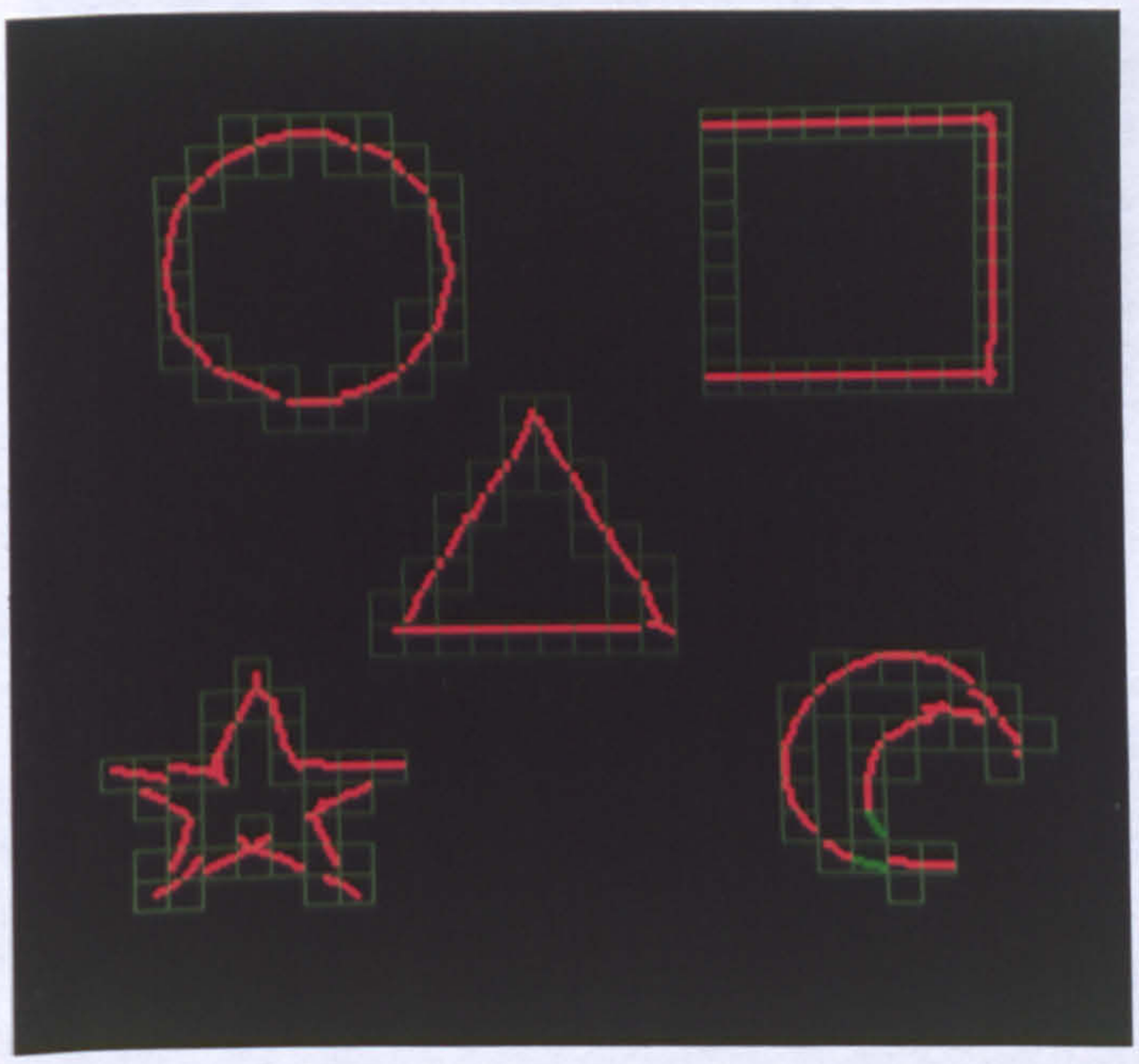


Figure 4.9: Shapes: Feature detection - Level 4

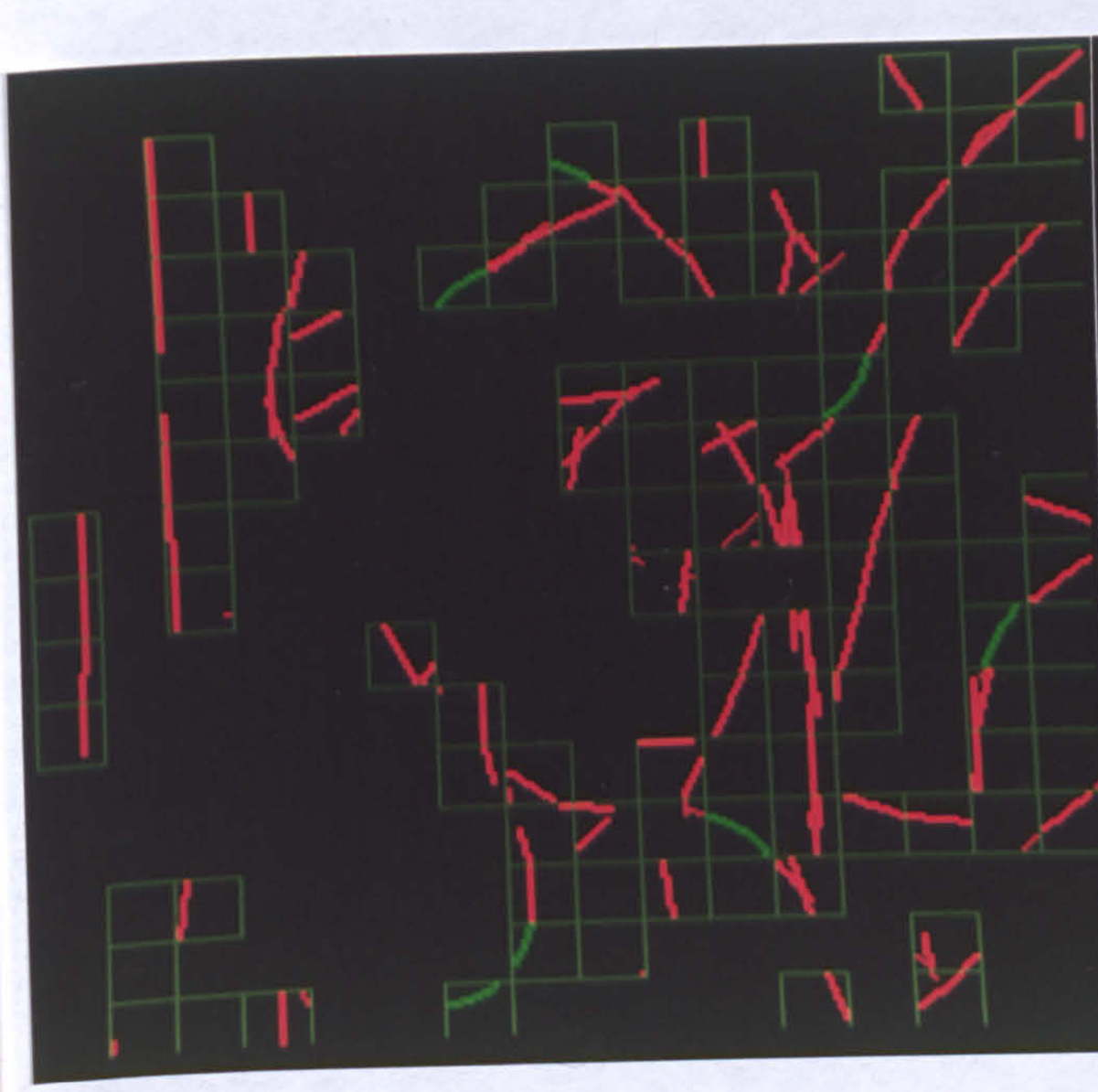


Figure 4.10: Lena: Feature detection - Level 3

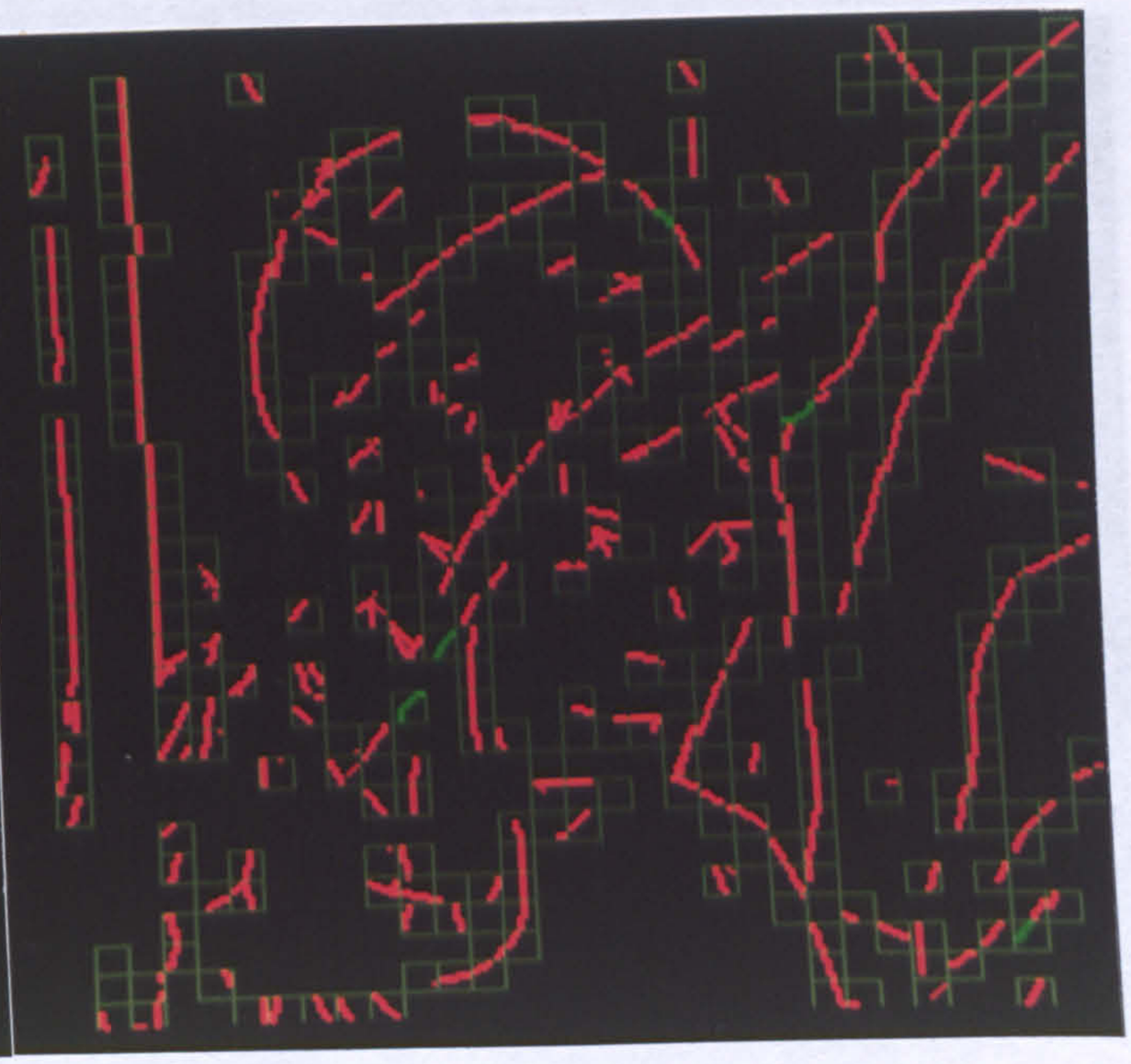


Figure 4.11: Lena: Feature detection - Level 4

Figure 4.12: Lena: Feature detection - Level 5

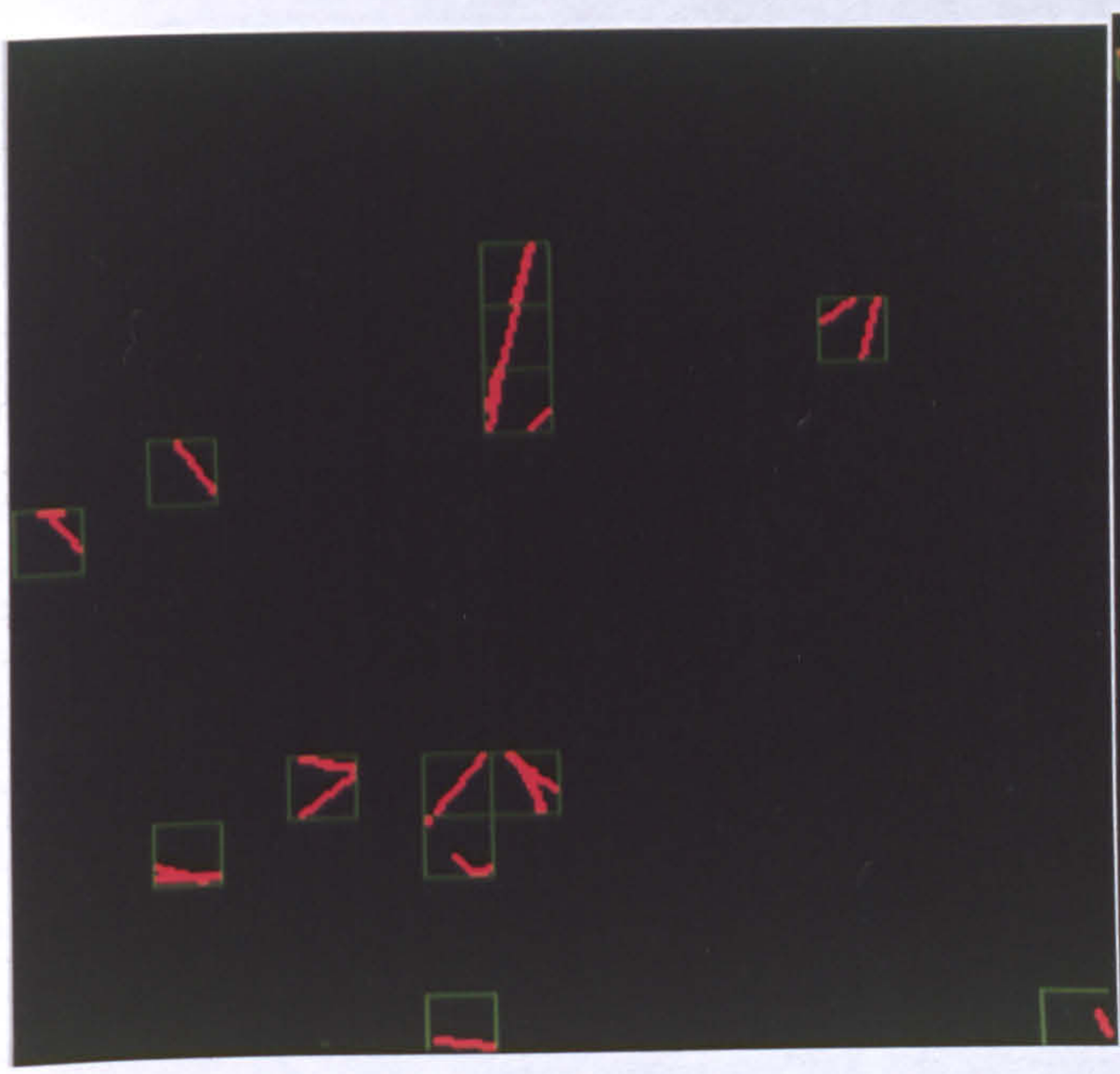


Figure 4.13: Vine leaf: Feature detection - Level 3

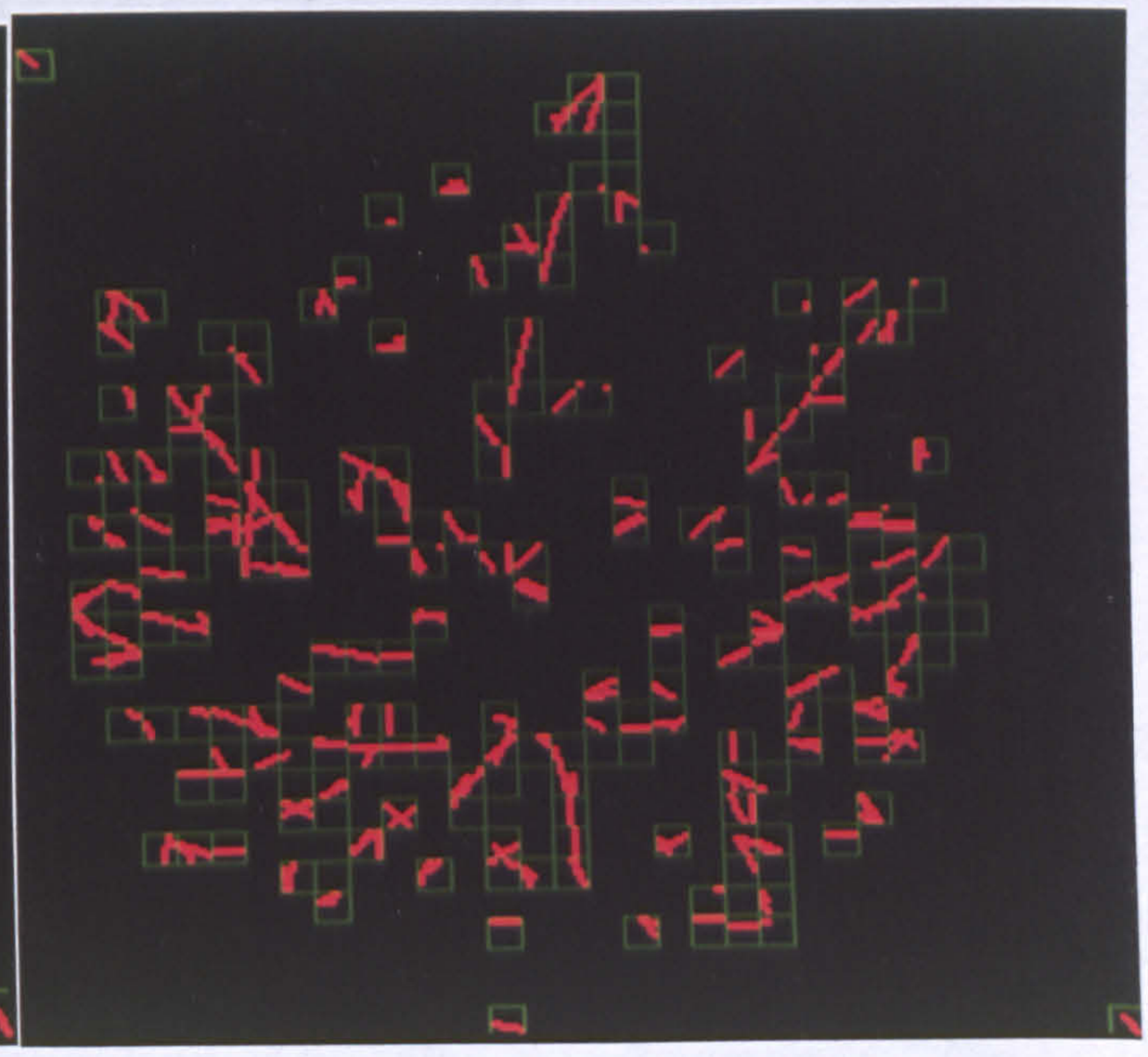


Figure 4.14: Vine leaf: Feature detection - Level 4

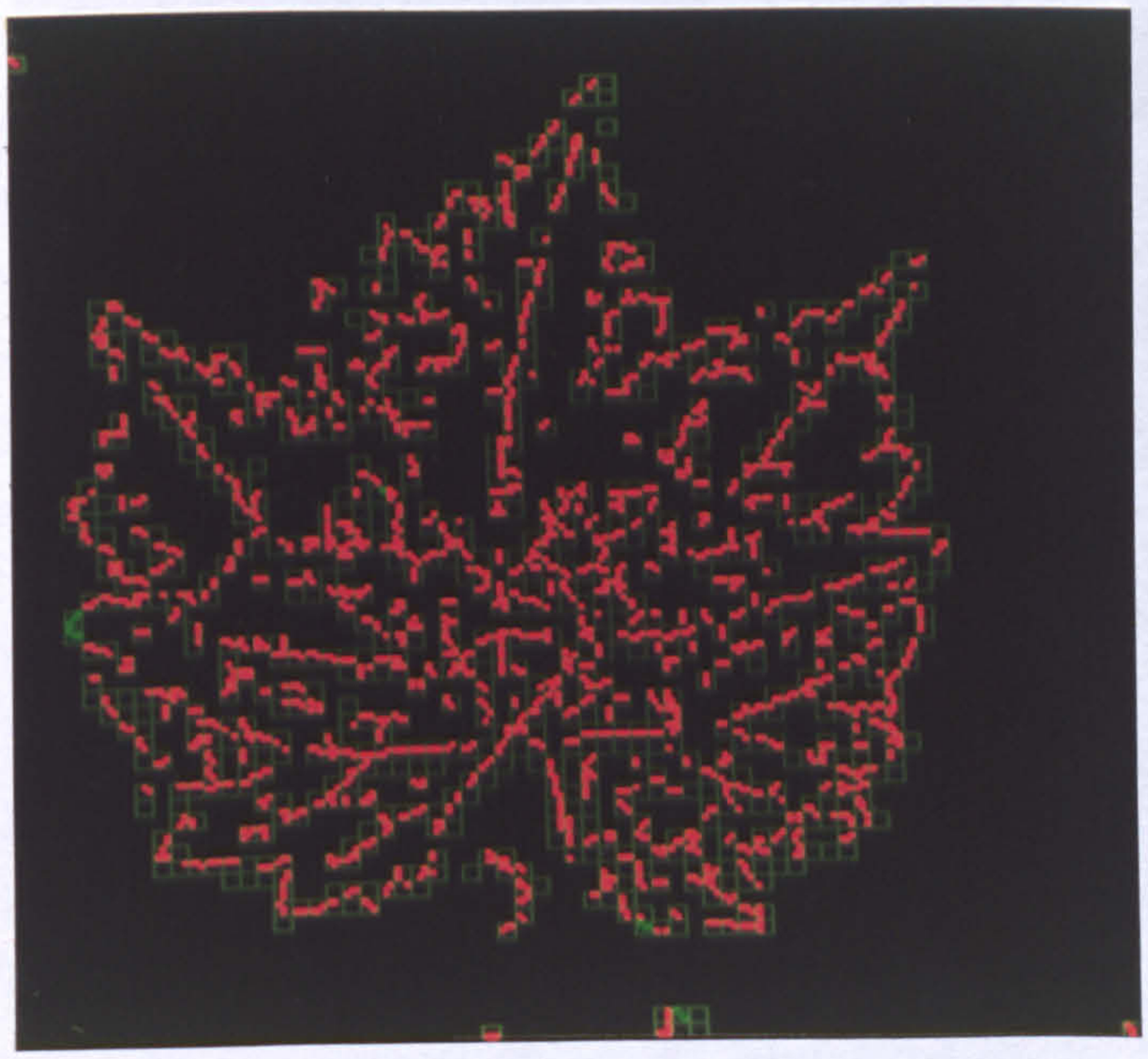


Figure 4.15: Vine leaf: Feature detection - Level 5

Chapter 5

Segmentation and Linking

5.1 Introduction

In the last two chapters, the detection of local image features using local spectrum estimates, has been discussed. These estimates have taken place at a single scale and each image region has either been successfully modelled or left as empty. It is clear from the discussion in chapter 2 and results in chapter 4 that images contain features at different scales. This chapter is concerned with combining features detected over a range of scales. In its simplest form, this consists of starting at a large scale and attempting to model the image regions. Each region where this fails is repeatedly sub-divided until it can be successfully modelled. This approach has been commonly used in earlier work, eg. [9, 25, 88].

Calway [16] included a scale check to ensure that features detected at one scale were consistent with those at neighbouring scales. In part, this was necessitated by the inability of the linear phase model to distinguish a region containing multiple parallel features, a problem solved by using spectrum synthesis to check that a region has been successfully modelled (chapter 3). There are other cases, however, where it is desirable to check that significant image features are not missed. This consistency

check is based on the correlation between parent and child - using the features detected in the parent to synthesise the child spectrum and using a correlation to check that the child features are accounted for in the parent.

The segmentation process splits the image into regions containing image boundary segments. These features will usually be segments of more complex large scale features, and therefore it is desirable to link the detected primitive features into the more complex features that model the edge/boundary structure of the image. The linking process described in this chapter is based on a correlation to test the similarity between a local spectrum and the transformed spectrum of a neighbouring region.

Having linked the local features it may be clear that there are gaps in the larger scale boundaries. Features may continue through a region but not appear in the appropriate neighbour. This may imply that a local feature exists in the neighbour, but has been missed by the feature detection process. Since there will be some indication of where such a feature lies, the detection algorithm may be reapplied using this information to guide it.

5.2 Segmentation Algorithm

Based on the model described in section 2.3.3, a multiresolution segmentation of the image may be regarded as the selection of a set of regions from different levels which partition the image, so that each region is represented at the highest level for which

1. The region is successfully modelled by primitive features, and

2. The features in the region are consistent with the features in its four children.

If such a partition of the image is denoted as Λ_0, and region $\Lambda_{i'}$ is the parent of region Λ_i, then the above conditions can be expressed in the form

$$\Lambda_0 = \{(i,n)|\Phi_i(n) \neq \emptyset \textbf{ AND } (\Phi_{i'}(n-1) = \emptyset \textbf{ OR } (\bigvee_{l=0}^{3} c(\Phi_{i'}(n-1), \Phi_{i'_l}(n)) < t_s)\} \quad (5.1)$$

where $\Phi_i(n)$ is the set of parameters representing the features detected in region Λ_i and $c(\Phi_i(n), \Phi_{i_l}(n+1))$ is a measure of the consistency of the features in a region, with the features in one of its children, and gives a value between 0 (totally inconsistent) and 1 (completely consistent). The choice of function $c(\Phi_i(n), \Phi_{i_l}(n+1))$ and threshold t_s, determine to what extent scale consistency is used to determine the segmentation process.

The simplest case is to assume that if the region $\Lambda_i(n)$ is well modelled by the set of features $\Phi_i(n)$, then these features must be consistent with any features detected in the children, ie.

$$\begin{aligned} c(\Phi_i(n), \Phi_{i_l}(n+1)) &= 1 \quad \text{if } \Phi_i(n) = \emptyset \\ &= 0 \quad \text{otherwise} \end{aligned} \quad (5.2)$$

and

$$t_s = 0.5 \quad (5.3)$$

This reduces equation 5.1 to

$$\Lambda_0 = \{(i,n)|\Phi_i(n) \neq \emptyset \textbf{ AND } \Phi_{i'}(n-1) = \emptyset\} \quad (5.4)$$

This suggests a top-down algorithm for generating Λ_0, as shown in figure 5.1. Starting at an upper level n_u, each region is tested; if it can be adequately modelled using the primitive features, the region indices, (i,n), are added to the set Λ_0. This is

$\Lambda_0 = \emptyset$

For each n, $n_l \leq n \leq n_u$

For each i,$0 \leq i \leq 4\Omega(n)$

If $\mathcal{A}(i,n) \cap \Lambda_0 = \emptyset$ **AND** $\Phi_i(n) \neq \emptyset$

$\Lambda_0 = \Lambda_0 \cup \{(i,n)\}$

Figure 5.1: Segmentation Algorithm

repeated for each region down to level n_l, except that indices are only added if those of an ancestor are not already in the set. The set $\mathcal{A}(i,n)$ is the set of the indices of the ancestor regions for region $\Lambda_i(n)$, and can be written,

$$\mathcal{A}(i,n) = \{(i',n') || \xi_{i'l}(n') - \xi_{il}(n)| < \Xi(n)/2, n' < n\} \tag{5.5}$$

In using the consistency criterion of equation (5.2), it is assumed that if a feature can be modelled with one of the feature primitives, using the detection methods of chapter 3, then it must be consistent with any features that could be detected in its children. Implementation of the feature detection process, however, uses a measure of how well the image data is modelled by an instance of the feature model. Since any image region will contain energy from features other than those of interest, eg. texture or noise, it is often necessary to accept a feature model that does not account for all of the local spectrum energy. Setting a percentage of the energy that must be accounted for in order that a feature is detected takes no account of whether the 'missing' energy is background texture or noise, and hence should be discounted, or due to another, low-level feature, which may be of interest.

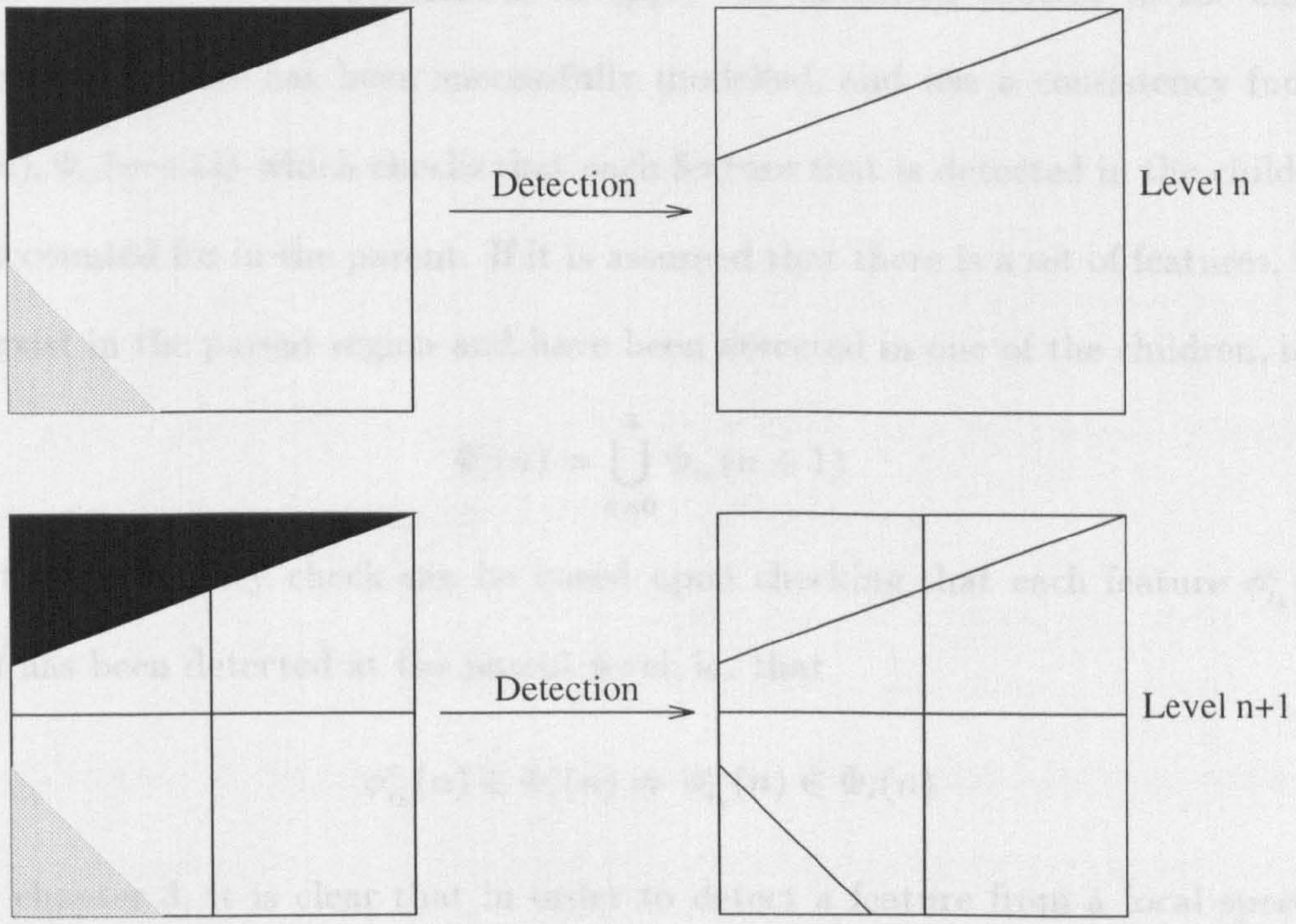


Figure 5.2: Feature missed at large scale

An example of this is shown of figure 5.2. When the region is considered at a large scale, n, the larger, high energy feature accounts for most of the region energy. In the feature detection process this edge is detected, and successfully modelled, and since this accounts for virtually all the energy it is accepted as modelling the region. Since the edge in the bottom left of the region is correspondingly shorter and of lower energy, it is missed. Simply increasing the threshold, would, however, make the detection process far more likely to fail in the presence of any background noise or texture. At the next scale, $n+1$, the region is split into four quadrants. Since each quadrant is treated individually, the second edge will be successfully detected in child 2.

One solution to this problem is to apply the detection process to the children of each region that has been successfully modelled, and use a consistency function $c(\Phi_i(n), \Phi_{i_l}(n+1))$ which checks that each feature that is detected in the children is also accounted for in the parent. If it is assumed that there is a set of features, $\Phi_i^c(n)$ that exist in the parent region and have been detected in one of the children, ie.

$$\Phi_i^c(n) = \bigcup_{k=0}^{3} \Phi_{i_k}(n+1) \tag{5.6}$$

then the consistency check can be based upon checking that each feature $\phi_{i_k}^c(n) \in \Phi_i^c(n)$ has been detected at the parent level, ie. that

$$\phi_{i_k}^c(n) \in \Phi_i^c(n) \Rightarrow \phi_{i_k}^c(n) \in \Phi_i(n) \tag{5.7}$$

From chapter 3, it is clear that in order to detect a feature from a local spectrum, the coefficients in the orientation of that feature should be due to that feature and nothing else. If every child feature has been detected in the parent, then it should be possible to synthesise the child spectrum using the parent features.

This approach is clearly an instance of the general model-based decision process described in section 3.2, ie.

Estimation: Features detected in the parent regions are truncated to one of its children in order to estimate the parameters of features in the child.

Synthesis: The estimated child parameters are used to synthesise the local spectrum corresponding to the child region.

Decision: By correlating the synthesised spectrum with the actual spectrum corresponding to features detected in the child region, a decision is made concerning whether the child features have been successfully accounted for in the parent.

Denoting a spectrum synthesised from a vector of feature parameters ϕ in the region Λ_i, by $\hat{x}(\vec{\xi}_i(n), \vec{\omega}_j(n), \sigma(n), \phi)$, then a synthesised spectrum for a child region $\Lambda_{i_l}(n+1)$, in terms of features found in $\Lambda_i(n)$, is given by

$$\hat{x}'(\vec{\xi}_{i_l}(n+1), \vec{\omega}_j(n+1), \sigma(n+1)) = \sum_{\phi \in \Phi_i(n)} \hat{x}(\vec{\xi}_{i_l}(n+1), \vec{\omega}_j(n+1), \sigma(n+1), \phi) \quad (5.8)$$

The scale consistency measure can then be given as a correlation between the actual child spectrum and the synthesised child spectrum, ie.

$$c(\Phi_i(n-1), \Phi_{i_l}(n)) = \frac{\sum_{j=0}^{4\Xi(n)} \hat{x}'(\vec{\xi}_{i_l}(n), \vec{\omega}_j(n), \sigma(n)) \hat{x}^*(\vec{\xi}_{i_l}(n), \vec{\omega}_j(n), \sigma(n))}{\sum_{j=0}^{4\Xi(n)} |\hat{x}'(\vec{\xi}_{i_l}(n), \vec{\omega}_j(n), \sigma(n))| |\hat{x}(\vec{\xi}_{i_l}(n), \vec{\omega}_j(n), \sigma(n))|} \quad (5.9)$$

This will clearly be equal to one, its maximum, if the child has been synthesised exactly from parent features. The higher the correlation, the better the synthesis and hence the greater the consistency between the scales.

5.3 Scale Consistency

The above scale consistency check is based upon synthesising the spectrum of each feature detected in a child region using only features detected in its parent. For each child, it is necessary to detect each parent feature that passes through it and calculate the parameters for that part of the feature 'seen' in the child region.

For a linear feature, a check can be made by first considering the displacement vector, perpendicular to the feature, from the child centre to the feature, denoted $\vec{r}_{i_k}$, which gives the point on the feature which is nearest the centre of the child region. This is shown in figure 5.3, where the centre of the parent region is $\vec{\xi}_i$ and that of the child region $\vec{\xi}_{i_k}$, and the feature orientation is θ. Since the centroid of the feature relative to the centre of the parent region is given by $\vec{\xi}_{i0}$, a vector giving this centroid

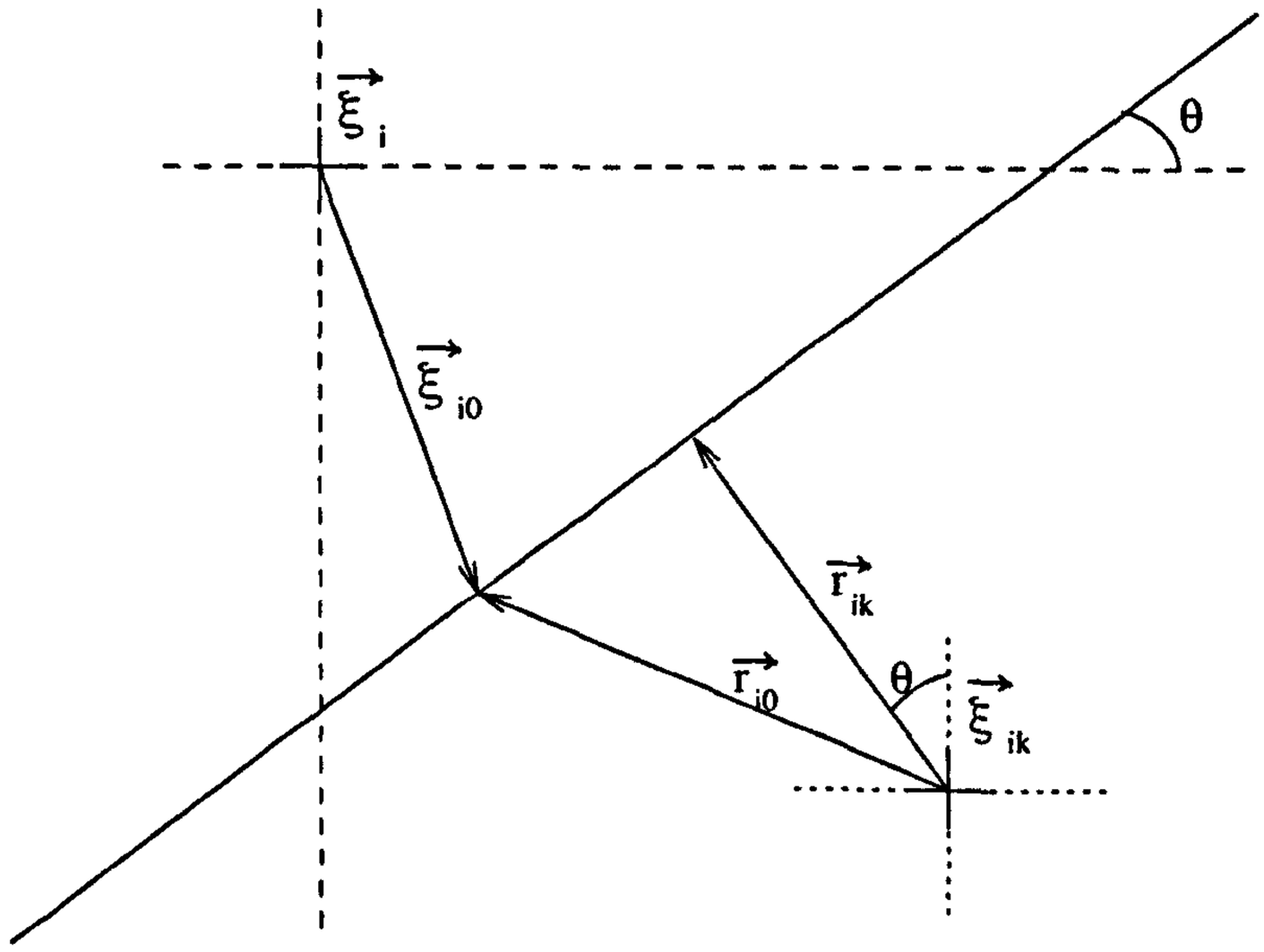


Figure 5.3: Feature in parent and child

relative to the child region centre is given by

$$\vec{r}_{i_0} = \vec{\xi}_{i0} + (\vec{\xi}_i - \vec{\xi}_{i_k}) \tag{5.10}$$

The unit vector in the direction of $\vec{r}_{i_k}$ is $\vec{v}_\theta$ ie.

$$\frac{\vec{r}_{i_k}}{\|\vec{r}_{i_k}\|} = \vec{v}_\theta \tag{5.11}$$

and since the length of $\vec{r}_{i_k}$ is given by

$$\|\vec{r}_{i_k}\| = \vec{r}_{i_0} \cdot \vec{v}_\theta \tag{5.12}$$

it follows that

$$\vec{r}_{i_k} = \|\vec{r}_{i_k}\|\vec{v}_\theta \tag{5.13}$$

$$= (\vec{r}_{i_0} \cdot \vec{v}_\theta)\vec{v}_\theta \tag{5.14}$$

Thus $\vec{r}_{i_k}$ gives an indication of how close the feature comes to the child region centre. If the edge of the child region is $\Xi(n+1)/2$ away from the centre, then for

$m = 1, 2$

$$|r_{i_k m}| \leq \Xi(n+1)/2 \quad \text{If the feature passes through the child region} \tag{5.15}$$

$$> \Xi(n+1)/2 \quad \text{otherwise} \tag{5.16}$$

$\vec{r}_{i_k}$ can therefore be used to test whether or not the feature passes through the region. If a feature does pass through the region, the next stage is to estimate its centroid within that region. Assuming that the feature is of constant magnitude along its length, the centroid can be regarded as the point halfway between the two feature endpoints. The end points, $\vec{\xi}_{i_k}\mathrm{s}$ and $\vec{\xi}_{i_k}\mathrm{e}$, can be calculated by clipping the parent feature to the child region using the algorithm described in section 4.2. The centroid relative to the centre of the child region is given by

$$\vec{\xi}_{i_k 0} = 0.5(\vec{\xi}_{i_k}\mathrm{s} + \vec{\xi}_{i_k}\mathrm{e}) \tag{5.17}$$

This centroid estimate can then be used to synthesise the component spectrum in the child that corresponds to the feature, using equation (3.42).

A similar process must be carried out when considering arc features detected in the parent region. In some cases, an arc at the parent level may be detected as an arc in its children, but in others the size of the arc segment seen by the child may be small enough to be modelled as a linear feature.

Consider a piece of circular arc, such as in figure 5.4, with radius R and centre, relative to the centre of the parent region, $\vec{\xi}_{i0}$. In order to determine whether the feature passes through the child region Λ_{i_k}, the vector $\vec{r}_{ik}$ is calculated. This is the shortest vector from $\vec{\xi}_{i_k}$ to the arc and is therefore perpendicular to the arc tangent at the point of contact and is on a straight line between the arc centre and the child

$\vec{r}_{ik}$ is given by the expression

$$\vec{r}_{ik} = \vec{\xi}_{i_k 0} \frac{\|\vec{\xi}_{i_k 0}\| - R}{\|\vec{\xi}_{i_k 0}\|} \tag{5.21}$$

$\vec{r}_{ik}$ can then be used to determine whether the arc passes through the child region, since for $m = 1, 2$

$$|r_{i_k m}| \leq \Xi(n+1)/2 \quad \text{If the feature passes through the child region} \tag{5.22}$$

$$> \Xi(n+1)/2 \quad \text{otherwise} \tag{5.23}$$

If the arc has been found to pass through the child region, then a spectrum can be synthesised using the radius R and the arc centre $\vec{\xi}_{i_k 0}$ given by equation (5.18) and the consistency test applied to it.

5.4 Linking Algorithm

The segmentation process produces a set of local image features that represent the boundary structures of the image. The local features are usually segments of the large scale boundaries of objects in the image. To represent these more complex boundaries it is necessary to link together appropriate local features. This section presents a method of doing this based upon the pair-wise similarity of features detected in neighbouring regions.

If two primitives are part of the same compound feature then it may be assumed that

1. They meet at the boundary between the regions in which they have been detected, and

2. They have similar curvature at this point.

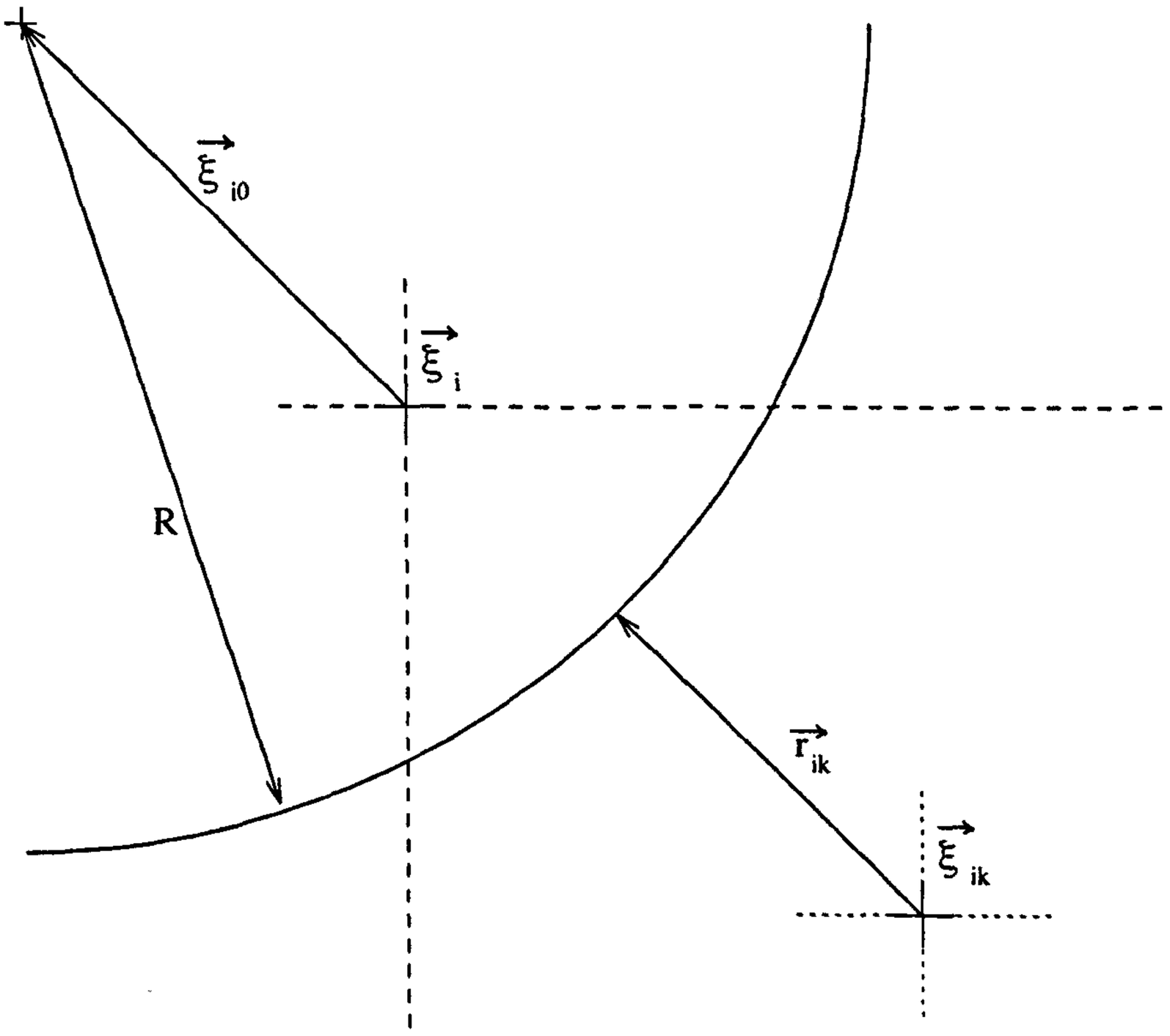


Figure 5.4: Arc feature in parent and child

region centre. If this point is not within the child region then the arc does not pass through it. The centre of the circle relative to the centre of the child region, $\vec{\xi}_{i_k 0}$, is given by

$$\vec{\xi}_{i_k 0} = \vec{\xi}_{i0} + (\vec{\xi}_i - \vec{\xi}_{i_k}) \tag{5.18}$$

Since the vector $\vec{\xi}_{i_k 0}$ is in the same orientation as $\vec{r}_{ik}$, $\vec{r}_{ik}$ can be calculated by scaling $\vec{\xi}_{i_k 0}$ so that it has length $\|\vec{r}_{ik}\|$, ie.

$$\vec{r}_{ik} = \frac{\|\vec{r}_{ik}\|}{\|\vec{\xi}_{i_k 0}\|} \vec{\xi}_{i_k 0} \tag{5.19}$$

Since it can be clearly seen that

$$\|\vec{r}_{ik}\| = \|\vec{\xi}_{i_k 0}\| - R \tag{5.20}$$

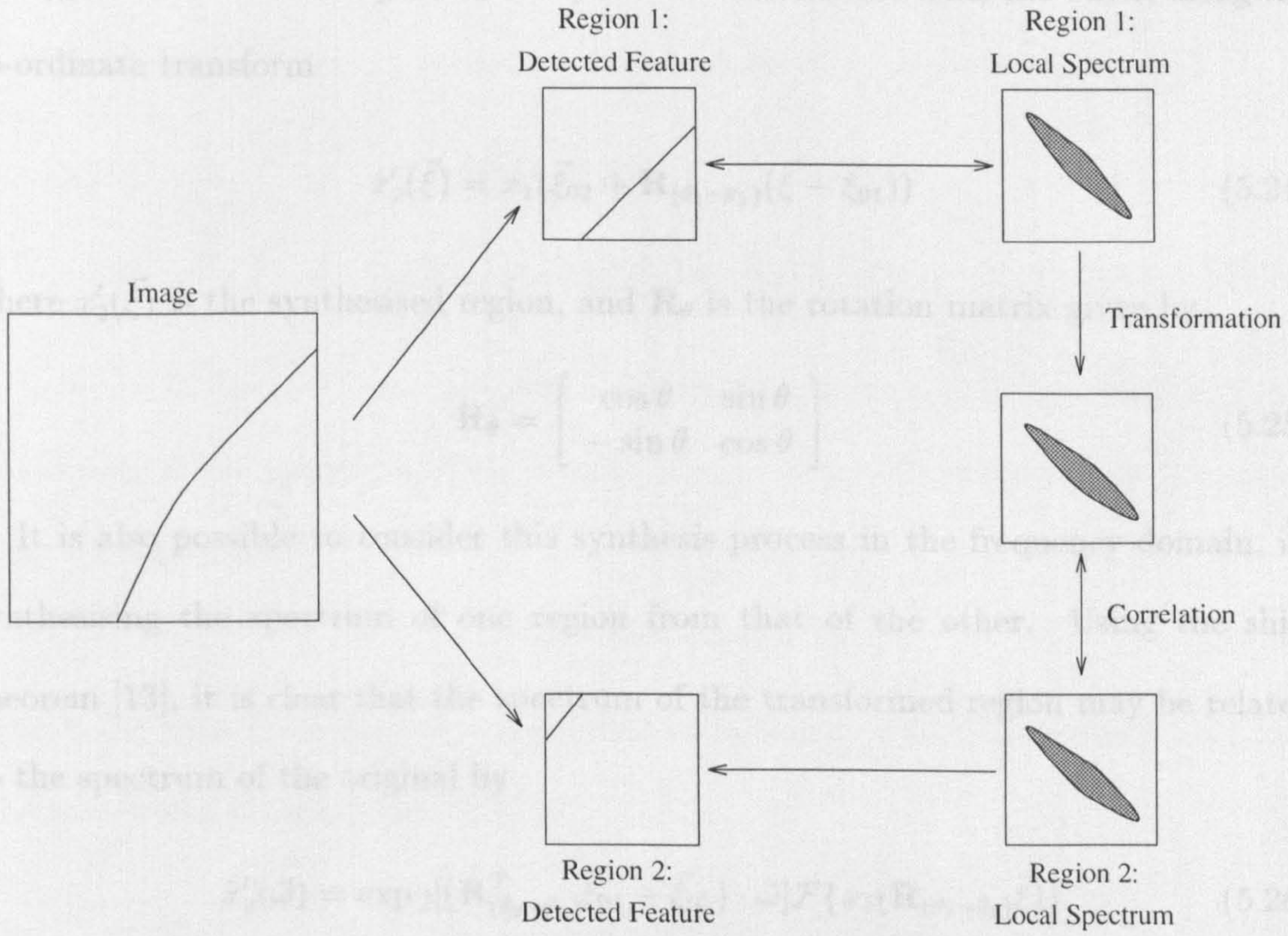


Figure 5.5: Linking Primitive Feature Segments

Consider part of a curve modelled by two linear features in adjacent regions $\Lambda_1(n)$ and $\Lambda_2(n)$ centered on sample points $\vec{\xi_1}(n)$ and $\vec{\xi_2}(n)$ respectively, as shown in figure 5.5. If the curve is continuous over the two regions, then the features should have similar orientation, the difference in orientation being dependent upon the curvature. Moreover, the positions of the two centroids relative to the region centres will differ according to their orientations. Thus it should be possible to transform one local feature region into the other using a simple affine co-ordinate transform consisting of a rotation and translation. In other words, suppose that the features can be represented by continuous 2-d functions they may be denoted $x_1(\vec{\xi})$ and $x_2(\vec{\xi})$. If the two features have orientations θ_1 and θ_2, and centroids relative to region centres $\vec{\xi}_{01}$

and $\vec{\xi}_{02}$ then it should be possible to synthesise one feature from the other, using the co-ordinate transform

$$x_2'(\vec{\xi}) = x_1(\vec{\xi}_{02} + \mathbf{R}_{(\theta_2-\theta_1)}(\vec{\xi} - \vec{\xi}_{01})) \tag{5.24}$$

where $x_2'(\vec{\xi})$ is the synthesised region, and $\mathbf{R}_\theta$ is the rotation matrix given by

$$\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{5.25}$$

It is also possible to consider this synthesis process in the frequency domain, ie. synthesising the spectrum of one region from that of the other. Using the shift theorem [13], it is clear that the spectrum of the transformed region may be related to the spectrum of the original by

$$\hat{x}_2'(\vec{\omega}) = \exp \jmath[(\mathbf{R}^T_{(\theta_2-\theta_1)}\vec{\xi}_{01} - \vec{\xi}_{02}) \cdot \vec{\omega}]\mathcal{F}\{x_1(\mathbf{R}_{(\theta_2-\theta_1)}\vec{\xi})\} \tag{5.26}$$

where $\mathcal{F}\{.\}$ is the Fourier operator. The transform of a rotated function is equivalent to the rotated transform of that function [60], ie. if

$$x(\vec{\xi}) \leftrightarrow \hat{x}(\vec{\omega}) \tag{5.27}$$

then

$$x(\mathbf{R}_\theta\vec{\xi}) \leftrightarrow \hat{x}(\mathbf{R}_\theta\vec{\omega}) \tag{5.28}$$

Hence

$$\hat{x}_2'(\vec{\omega}) = \exp \jmath[(\mathbf{R}^T_{(\theta_2-\theta_1)}\vec{\xi}_{01} - \vec{\xi}_{02}) \cdot \vec{\omega}]\hat{x}_1(\mathbf{R}_{(\theta_2-\theta_1)}\vec{\omega}) \tag{5.29}$$

The test of connectivity between two features can therefore be based upon the similarity between the actual region, $x_2(\vec{\xi})$, and the synthesised region, $x_2'(\vec{\xi})$. This

similarity can be measured by calculating a correlation coefficient between the real and synthesised spectra. In terms of continuous spectra, this is given by

$$r = \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\hat{x}_2'(\vec{\omega})\hat{x}_2^*(\vec{\omega})d\omega_1 d\omega_2 \tag{5.30}$$

$$= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\hat{x}_1(\mathbf{R}_{(\theta_2-\theta_1)}\vec{\omega})\hat{x}_2^*(\vec{\omega})\exp\jmath[(\mathbf{R}_{(\theta_2-\theta_1)}^T\vec{\xi}_{01}-\vec{\xi}_{02})\cdot\vec{\omega}]d\omega_1 d\omega_2 \tag{5.31}$$

In terms of discrete sampled spectra, a rotated spectrum may be considered as a resampling of the original spectrum using a rotated set of sample points. Denoting these points $\vec{\omega}_i'$, then

$$\vec{\omega}_i' = \mathbf{R}_{(\theta_2-\theta_1)}\vec{\omega}_i \tag{5.32}$$

In addition there is then a phase correction to take account of the centroid variation. The correlation is then given by

$$r = \sum_{i=0}^{N^2-1}\hat{x}_1(\vec{\omega}_i')\hat{x}_2^*(\vec{\omega}_i)\exp\jmath[\vec{\xi}_{01}\cdot\vec{\omega}_i'-\vec{\xi}_{02}\cdot\vec{\omega}_i] \tag{5.33}$$

The correlation will be greatest when the two features are most similar and hence most likely to be part of the same boundary contour. It can, therefore, be used to decide whether adjacent primitive features can be regarded as part of the same large scale feature. It should be noted that this linking process is entirely local, and based only on the relationship between features in adjacent regions, a notable difference from earlier work [16].

It is clear that this approach to feature linking follows the general model based decision procedure described in section 3.2. The model is that two adjacent features may be related by a simple coordinate transform and therefore the linking procedure may be summarised as follows:

Estimation: Using the estimated feature parameters, the transformation relating the two adjacent spectra is estimated.

Synthesis: The estimated transform is used to synthesise one spectrum from the other.

Decision: By correlating the synthesised spectrum with the actual spectrum, a measure of feature similarity is calculated which is used to decide whether the features are part of the same boundary.

The generality of this linking process gives a method of linking any two regions that can be related by a coordinate transform. The estimation of transforms between texture regions has been considered by Hsu[45].

5.5 "Missed" Features

There may be parts of any boundary that are missed by the detection procedure because they have low energy compared to other features or noise in their locality. Although the consistency criterion of section 5.2 may help sometimes, it is easy to imagine cases where this will not. These features may only be detected by the human visual system because they are part of larger scale features. One way of attempting to tackle this problem is based upon the fact that the feature may be part of a larger scale boundary.

In the linking process, detected features are tested against features in neighbouring regions and linked to them if they are similar. In some cases, however, a feature may appear to continue into the next region, since it is unlikely that a boundary feature will end precisely on a region boundary, but no corresponding feature has been detected

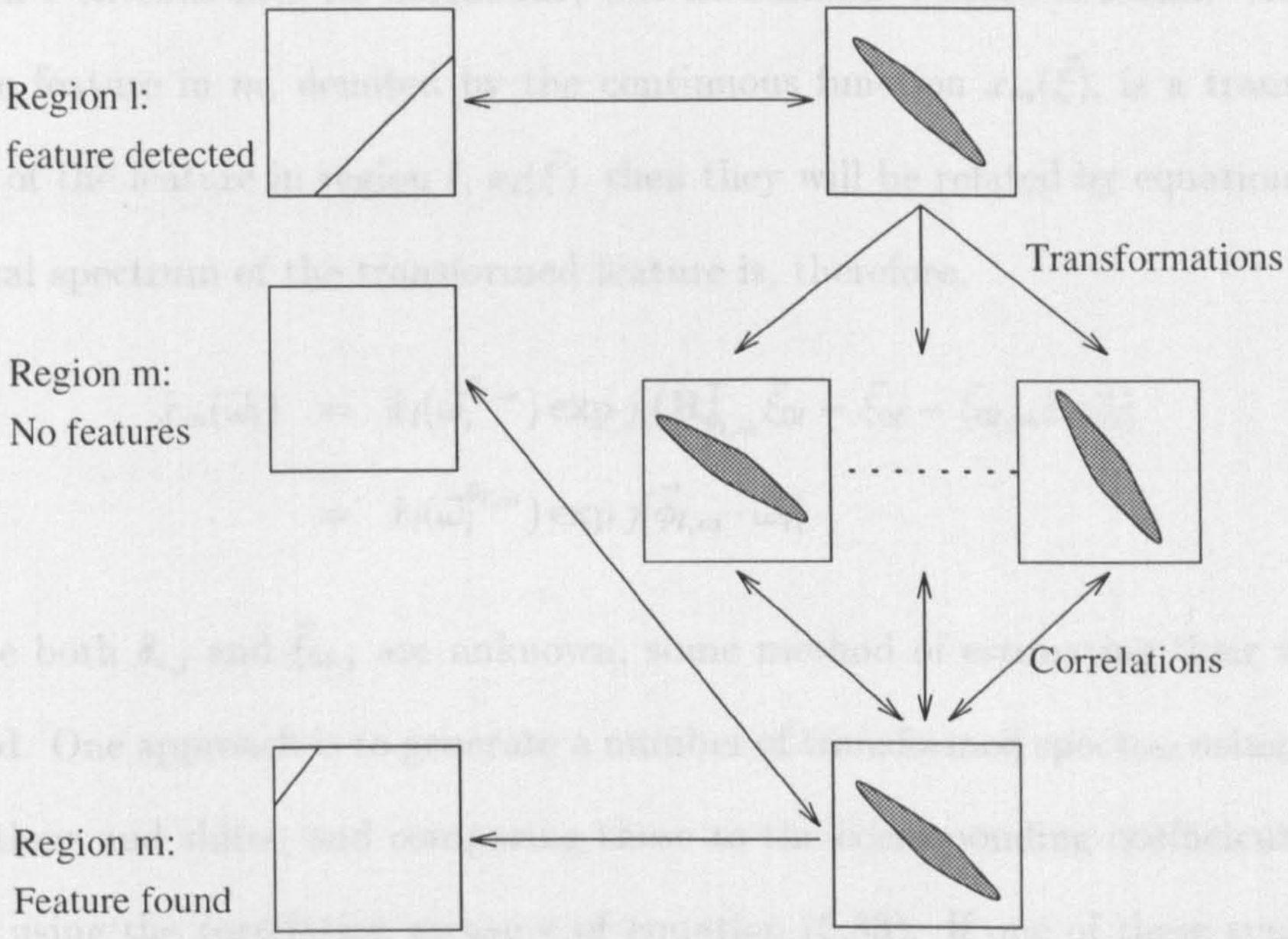


Figure 5.6: Detecting "Missed" Features

in the neighbouring region. This suggests that there may be a feature in the adjacent region, but that it has been missed by the detection process. Given that there is now some evidence that a feature may exist, and that, due to continuity, it will be in a given range of positions and orientations, a second detection process can be applied.

As in the linking process, if a feature exists in a neighbouring region, it should be possible to synthesise the feature coefficients by transforming the coefficients of the detected feature. Whereas, in the linking case, the transformation could be calculated from the estimated feature parameters, in this case the second feature is not known and may not even be present.

The approach taken is illustrated by figure 5.6. Regions l and m are subjected to the feature detection process, and part of a boundary is detected in l but nothing is detected in m. During the linking process, it is noted that it appears that the feature

in region i extends into its neighbour, but no suitable feature is found. Assuming that the feature in m, denoted by the continuous function $x_m(\vec{\xi})$, is a transformed version of the feature in region l, $x_l(\vec{\xi})$, then they will be related by equation (5.29). The local spectrum of the transformed feature is, therefore,

$$\hat{x}_m(\vec{\omega}_i) = \hat{x}_l(\vec{\omega}_i^{\theta_{l,m}}) \exp \jmath[(\mathbf{R}_{\theta_{l,m}}^T \vec{\xi}_{0l} - \vec{\xi}_{0l} - \vec{\xi}_{0l,m}) \cdot \vec{\omega}_i] \quad (5.34)$$

$$= \hat{x}_l(\vec{\omega}_i^{\theta_{l,m}}) \exp \jmath[\vec{\phi}_{l,m} \cdot \vec{\omega}_i] \quad (5.35)$$

Since both $\theta_{i,j}$ and $\vec{\xi}_{0i,j}$ are unknown, some method of estimating their values is required. One approach is to generate a number of transformed spectra, using a range of rotations and shifts, and comparing these to the corresponding coefficients in the region, using the correlation measure of equation (5.33). If one of these synthesised features correlates highly with a subset of the coefficients corresponding to region m then the feature detection process can be reapplied to them. Due to the continuity of boundary features, the rotation must be limited and it is assumed that $\theta_{i,j}$ is in the interval $[-\theta_{\max}, \theta_{\max}]$. Ignoring the centroid shift, a set of transformed spectra, rotated by discrete orientations in this interval, separated by θ_Δ can be generated. As there will be $K+1$ of these samples, where

$$K = 2\theta_{\max}/\theta_\Delta \quad (5.36)$$

then

$$\hat{x}_l^{(k)}(\vec{\omega}_i) = \hat{x}_l(\vec{\omega}_i^{k\theta_\Delta}) \text{ for } k = -K/2, -K/2+1, ..., K/2 \quad (5.37)$$

Assuming that $\hat{x}_m(\vec{\omega}_i)$ fits the model as given by equation (5.35) and that

$$k\theta_\Delta = \theta_{l,m} \quad (5.38)$$

then

$$\hat{x}_m(\vec{\omega}_i) = \hat{x}_l^k(\vec{\omega}_i) \exp \jmath[\vec{\phi}_{l,m} \cdot \vec{\omega}_i] \tag{5.39}$$

Defining the conjugate product $p^k(\vec{\omega}_i)$ by

$$p^k(\vec{\omega}_i) = \hat{x}_l^k(\vec{\omega}_i)\hat{x}_m^*(\vec{\omega}_i) \tag{5.40}$$

it can be clearly seen that

$$p^k(\vec{\omega}_i) = |\hat{x}_m(\vec{\omega}_i)|^2 \exp \jmath[\vec{\phi}_{l,m} \cdot \vec{\omega}_i] \tag{5.41}$$

ie. that $p^k(\vec{\omega}_i)$ has a linear phase characteristic with vector constant $\vec{\phi}_{l,m}$. An estimate of the $\vec{\phi}_{l,m}$ can be made by considering the phase differences of $p^k(\vec{\omega}_i)$, in a similar manner to the centroid estimation process, used for linear feature detection. Defining

$$r_p^k = \sum_{i=0}^{N^2-1} p^k(\vec{\omega}_i)p^k(\vec{\omega} - \vec{\Omega}_1) \tag{5.42}$$

then

$$r_p^k = \exp \jmath[\Omega\vec{\phi}_{l,m}] \sum_{i=0}^{N^2-1} |p^k(\vec{\omega}_i)p^k(\vec{\omega} - \vec{\Omega}_1)| \tag{5.43}$$

An estimate of $\vec{\phi}_{l,m}^k$, generated by assuming that the rotation is $k\theta_\Delta$ is therefore given by

$$\tilde{\phi}_{l,m}^k = \frac{1}{\Omega} \arg(r_p^k) \tag{5.44}$$

For a given orientation and phase, which accounts for the feature centroid shift, then a synthesised spectrum can be generated using

$$\hat{x}_m^k(\vec{\omega}_i) = \hat{x}_l^k(\vec{\omega}_i) \exp \jmath[\tilde{\phi}_{l,m}^k \cdot \vec{\omega}_i] \tag{5.45}$$

A measure of how well the synthesis for a given transformation matches the actual coefficients is a correlation function, similar to the correlation used to check parameter

estimates used in the feature detection process described in chapters 3 and 4. The correlation coefficient γ_k is defined as

$$\gamma_k = \sum_{i=0}^{N^2-1} \hat{x}_m^k(\vec{\omega}_i)\hat{x}_m^*(\vec{\omega}_i) \tag{5.46}$$

It is clear that this will be maximum if

$$\hat{x}_m^k(\vec{\omega}_i) = \hat{x}_m(\vec{\omega}_i) \tag{5.47}$$

in which case

$$\gamma_k = \sum_{i=0}^{N^2-1} |\hat{x}_m(\vec{\omega}_i)|^2 \tag{5.48}$$

The transformation which gives the highest value for γ_k is the one which best matches the coefficients in region m. The set of coefficients that correspond to this synthesised feature can then be passed to the detection process for parameter estimation, and confirmation that such a feature exists.

5.6 Discrete Spectrum Transformation

The basis of both the linking and "missing feature" algorithms described in section 5.4 is the transformation of the spectrum corresponding to a local feature modelling a boundary segment into a feature that can model an adjoining boundary segment. When linking detected features this transformation can be calculated from the estimated parameters of the known features. The transformation can be split into two parts. First, a rotation of the spectrum to account for change in orientation, and secondly multiplication by a complex exponential, with a linear phase characteristic, to account for the change in feature centroid.

In section 5.4, the rotation of the spectrum was presented as a resampling of the continuous spectrum using a rotated set of sample frequencies. Because implementation of the algorithm uses the discrete MFT to provide the local spectrum estimates, there is no continuous spectrum to re-sample. An estimate of the rotated spectrum has to be derived using the sampled spectrum obtained from the MFT. If it is assumed that the coefficients vary slowly enough, a bilinear interpolator can be used to approximate the rotated coefficients. Calway [16] has shown that this is indeed the case and has used it to calculate MFT sample values on a polar separable point set.

As given in section 2.3.1, the spectrum may be represented as a two dimensional array, $\hat{x}(u,v)$. The location in the array of a rotated sample point $\vec{\omega}_i^{\theta}$, is denoted by (u^{θ},v^{θ}), where in most cases the values will be non-integer. This can be calculated by rotating the sample point around the origin, ie.

$$\begin{bmatrix} u^{\theta} \\ v^{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u-\Xi(n)+0.5 \\ v-\Xi(n)+0.5 \end{bmatrix} + \begin{bmatrix} \Xi(n)-0.5 \\ \Xi(n)-0.5 \end{bmatrix} \tag{5.49}$$

A rotated coefficient $\hat{x}(\vec{\omega}_i^{\theta})$ may therefore be represented by the value $\hat{x}(u^{\theta},v^{\theta})$, which lies between four neighbouring array points. Using the floor function, where $\lfloor x \rfloor$ gives the integer part of the real number x, the four surrounding coefficients are $\hat{x}(\lfloor u^{\theta} \rfloor, \lfloor v^{\theta} \rfloor)$, $\hat{x}(\lfloor u^{\theta} \rfloor + 1, \lfloor v^{\theta} \rfloor)$, $\hat{x}(\lfloor u^{\theta} \rfloor, \lfloor v^{\theta} \rfloor + 1)$ and $\hat{x}(\lfloor u^{\theta} \rfloor + 1, \lfloor v^{\theta} \rfloor + 1)$, as shown in figure 5.7. A bilinear estimate of $\hat{x}(\vec{\omega}_i^{\theta})$ is given by

$$\hat{x}(u^{\theta},v^{\theta}) = \hat{x}(u^{\theta},\lfloor v^{\theta} \rfloor) + (\hat{x}(u^{\theta},\lfloor v^{\theta} \rfloor + 1) - \hat{x}(u^{\theta},\lfloor v^{\theta} \rfloor))(v^{\theta} - \lfloor v^{\theta} \rfloor) \tag{5.50}$$

where

$$\hat{x}(u^{\theta},\lfloor v^{\theta} \rfloor) = \hat{x}(\lfloor u^{\theta} \rfloor,\lfloor v^{\theta} \rfloor) + (\hat{x}(\lfloor u^{\theta} \rfloor + 1,\lfloor v^{\theta} \rfloor) - \hat{x}(\lfloor u^{\theta} \rfloor,\lfloor v^{\theta} \rfloor))(u^{\theta} - \lfloor u^{\theta} \rfloor) \tag{5.51}$$

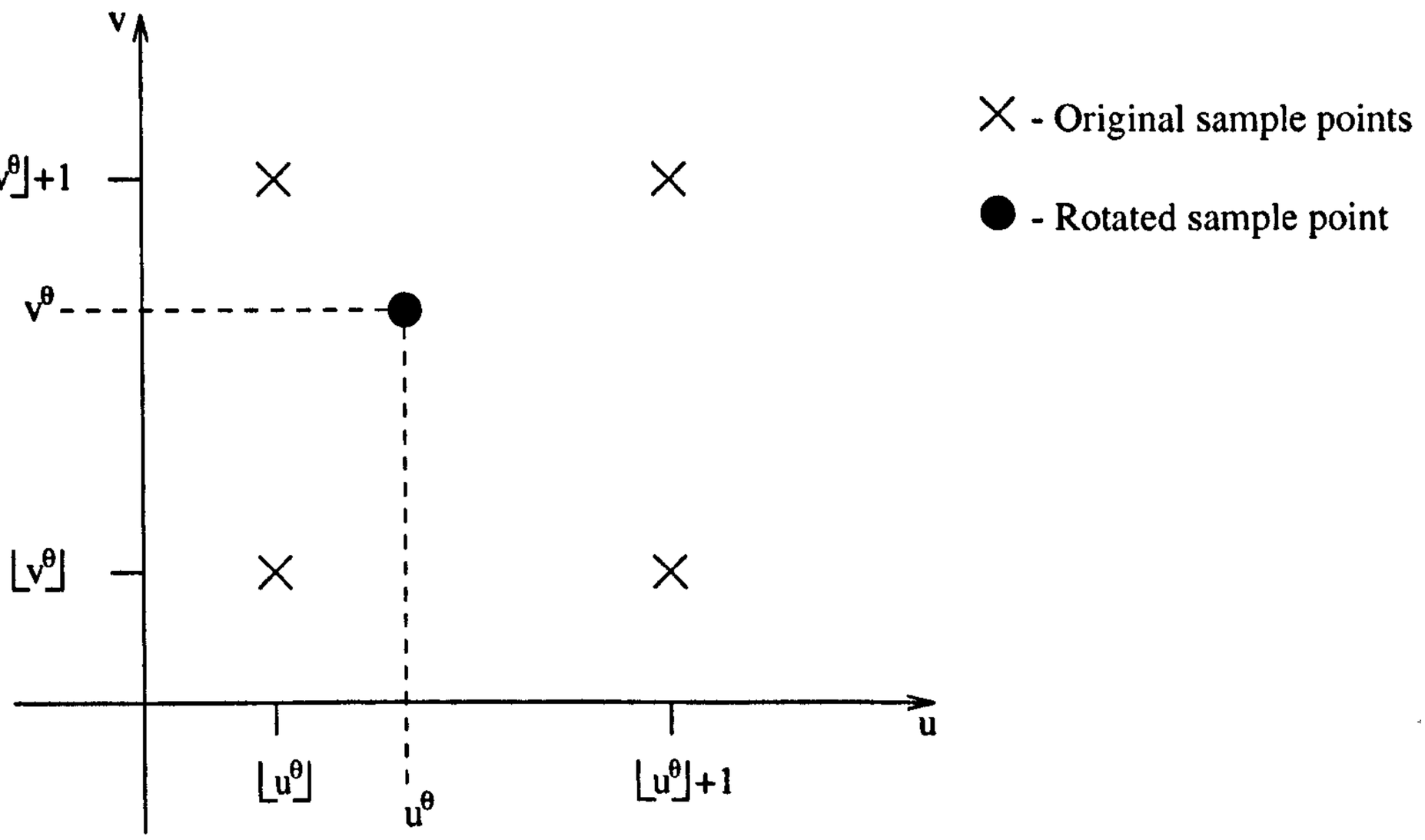


Figure 5.7: Bilinear Interpolation

and

$$\hat{x}(u^\theta, \lfloor v^\theta \rfloor + 1) = \hat{x}(\lfloor u^\theta \rfloor, \lfloor v^\theta \rfloor + 1) + (\hat{x}(\lfloor u^\theta \rfloor + 1, \lfloor v^\theta \rfloor + 1) - \hat{x}(\lfloor u^\theta \rfloor, \lfloor v^\theta \rfloor + 1))(u^\theta - \lfloor u^\theta \rfloor) \tag{5.52}$$

In some regions there may be more than one linear feature and for these regions the feature detection process segments the spectrum into component spectra - one for each feature. For each feature there is a set of the sample frequency indices that correspond to its local spectrum, given by equation (3.106), as

$$i \in I_k \iff \hat{x}_k(\vec{\omega}_i) \neq 0 \tag{5.53}$$

for the k^{th} feature.

Consider synthesising a spectrum $\hat{x}_t(\vec{\omega}_i)$, the target spectrum, representing a feature in one region, by transforming a source spectrum, $\hat{x}_s(\vec{\omega}_i)$, which represents a feature in an adjacent region. The set of sample points corresponding to these fea-

tures are I_s and I_t. To decide whether a rotated spectrum sample can be estimated from the desired spectrum, $\hat{x}_s(\vec{\omega}_i)$, a check can be made on whether it is adjacent to at least one of the original sample points associated with the source spectrum. The sample frequencies adjacent to the rotated frequency $\hat{x}(u^\theta, v^\theta)$ can be denoted $\hat{x}(\vec{\omega}_{i_{ab}})$ where

$$i_{ab} = 2(\lfloor v^\theta \rfloor + a)\Xi(n) + \lfloor u^\theta \rfloor + b, \quad \text{for a,b} = 0,1 \tag{5.54}$$

A selection function $s(\vec{\omega}_i^\theta, I_s)$, defined by

$$s(\vec{\omega}_i^\theta, I_s) = 1, \quad \text{if } (i_{00} \in I_s) \vee (i_{01} \in I_s) \vee (i_{10} \in I_s) \vee (i_{11} \in I_s) \tag{5.55}$$

$$= 0, \quad \text{otherwise} \tag{5.56}$$

indicates whether any of the four neighbouring samples, shown in figure 5.7, are within the source spectrum. An estimate of the rotated coefficient is thus given by

$$\hat{x}(\vec{\omega}_i^\theta) = s(\vec{\omega}_i^\theta, I_s)\hat{x}(u^\theta, v^\theta) \tag{5.57}$$

where $\hat{x}(u^\theta, v^\theta)$ is as given by equation (5.50).

Another part of the transformation may be the "shrinking" of the region, when attempting to link adjacent feature segments detected at different scales, ie. in regions of different size. The two local discrete spectra represent the same range of frequencies but sampled at a different density. In order to compare the spectra, a sample must be computed from one spectrum that corresponds to each actual sample in the other.

Consider two adjacent levels, n and $n+1$. Since the frequency sampling interval doubles with level (section 2.3.1), each sample at level $n+1$ is positioned centrally between four at level n, as shown in figure 5.8. Indeed, it can be clearly seen that the four level n samples surrounding (u_{n+1}, v_{n+1}) at level $n+1$ are $(2u_n, 2v_n)$, $(2u_n+1, 2v_n)$,

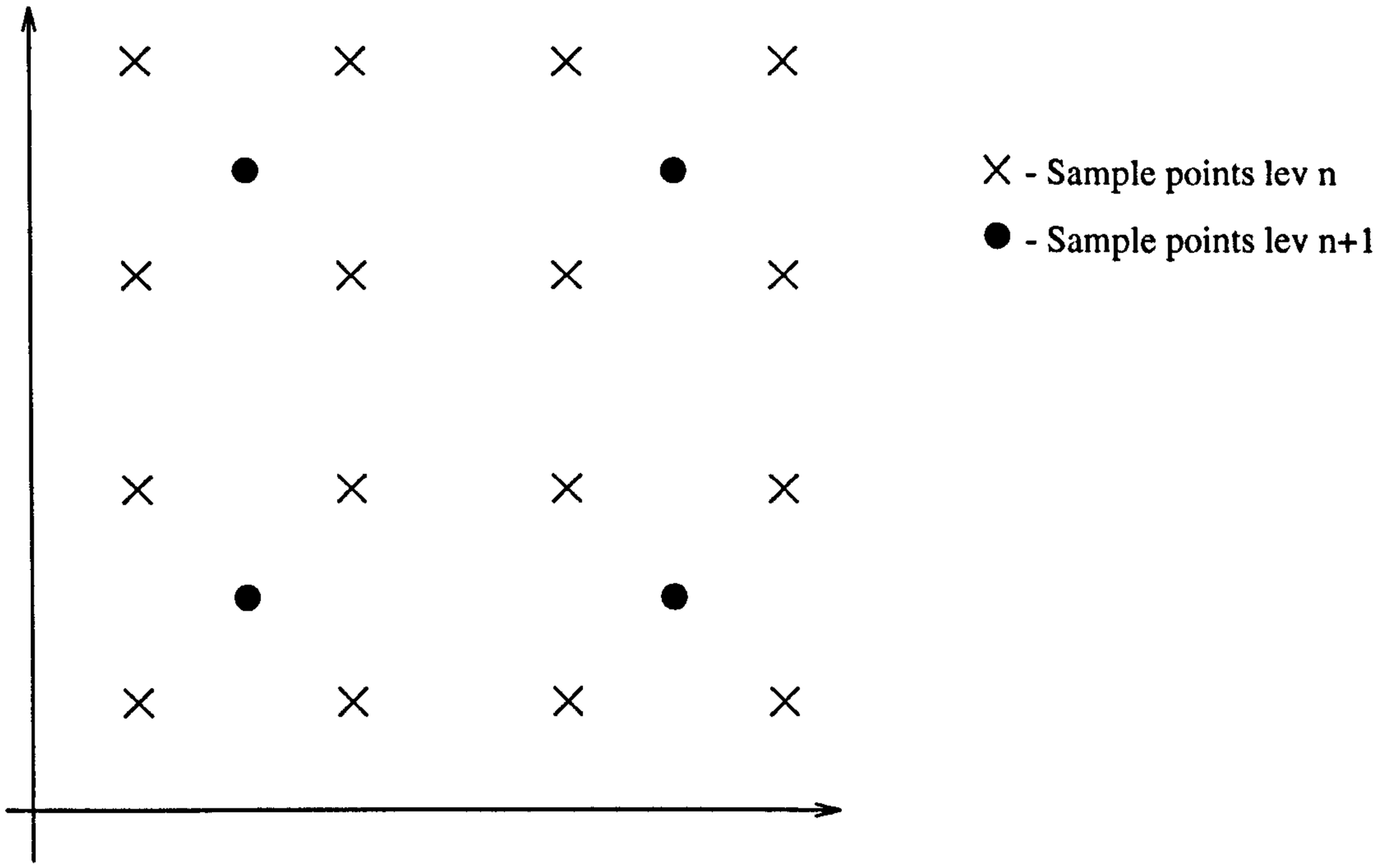


Figure 5.8: Sample points at levels n and $n+1$

$(2u_n, 2v_n+1)$ and $(2u_n+1, 2v_n+1)$. The real (non-integer) indices at level n, (u'_n, v'_n), representing the sample point (u_{n+1}, v_{n+1}), are therefore given by

$$(u'_n, v'_n) = (2u_{n+1} + 0.5, 2v_{n+1} + 0.5) \tag{5.58}$$

Generalising this to two arbitrary levels, n and $n+m$, gives

$$(u'_n, v'_n) = (2^m u_{n+m} + 2^{m-1} - 0.5, 2^m v_{n+m} + 2^{m-1} - 0.5) \tag{5.59}$$

The indices at level n corresponding to the indices at level $n+m$ will therefore be centrally placed between four level n samples. The corresponding coefficient can therefore be estimated using a simple bilinear interpolator. Since the desired sample frequency in this instance is always in the centre of its four neighbours, the bilinear interpolator, as defined in equations (5.50), (5.51) and (5.52) can be simplified to give

$$\hat{x}(u'_n, v'_n, n) = \frac{1}{4}(\hat{x}(\lfloor u'_n \rfloor, \lfloor v'_n \rfloor, n) + \hat{x}(\lfloor u'_n \rfloor + 1, \lfloor v'_n \rfloor, n)$$

$$+\hat{x}(\lfloor u'_n \rfloor, \lfloor v'_n \rfloor + 1, n) + \hat{x}(\lfloor u'_n \rfloor + 1, \lfloor v'_n \rfloor + 1, n)) \quad (5.60)$$

In other words, it is simply the arithmetic mean of its neighbours.

5.7 Spectrum Correlation

In chapter 3, it was shown that the spectrum of a continuous 1-d feature is highly concentrated. It was also shown that if local features are considered as windowed infinite features, then there is some spreading of the spectrum, perpendicular to the spectrum orientation, due to the windowing function. From equation (3.13) the sampled spectrum of a local line feature is given by

$$\hat{x}(\vec{\omega}_i) = \hat{x}(\vec{\omega}_i \cdot \vec{v}_{\theta_0})\hat{w}(\vec{\omega}_i \cdot \vec{v}'_{\theta_0}) \quad (5.61)$$

where $\hat{x}(\omega)$ is the 1-d function along the spectrum orientation, θ_0, due to the ideal feature, $\hat{w}(\omega)$ represents the spreading parallel to θ_0, due to the windowing function and $\vec{v}_\theta$ and $\vec{v}'_{\theta_0}$ are perpendicular, unit vectors given by

$$\vec{v}_\theta = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad (5.62)$$

$$\vec{v}'_\theta = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} \quad (5.63)$$

In this instance $\hat{w}(\omega)$ is the Fourier transform of the 1-d windowing function $w(\xi)$. If it is assumed that the window function is zero outside of the interval $[-\xi_{\max}, \xi_{\max}]$ and non-zero within this interval, then the length of the feature may be regarded as being the length of the non-zero portion of the window, ie. $2\xi_{\max}$. Providing $w(\xi)$ is sufficiently smooth, the "spreading" function, $\hat{w}(\omega)$ may be considered as having most of its energy within a given interval $[-\Omega_{\max}, \Omega_{\max}]$. This bandwidth can be used

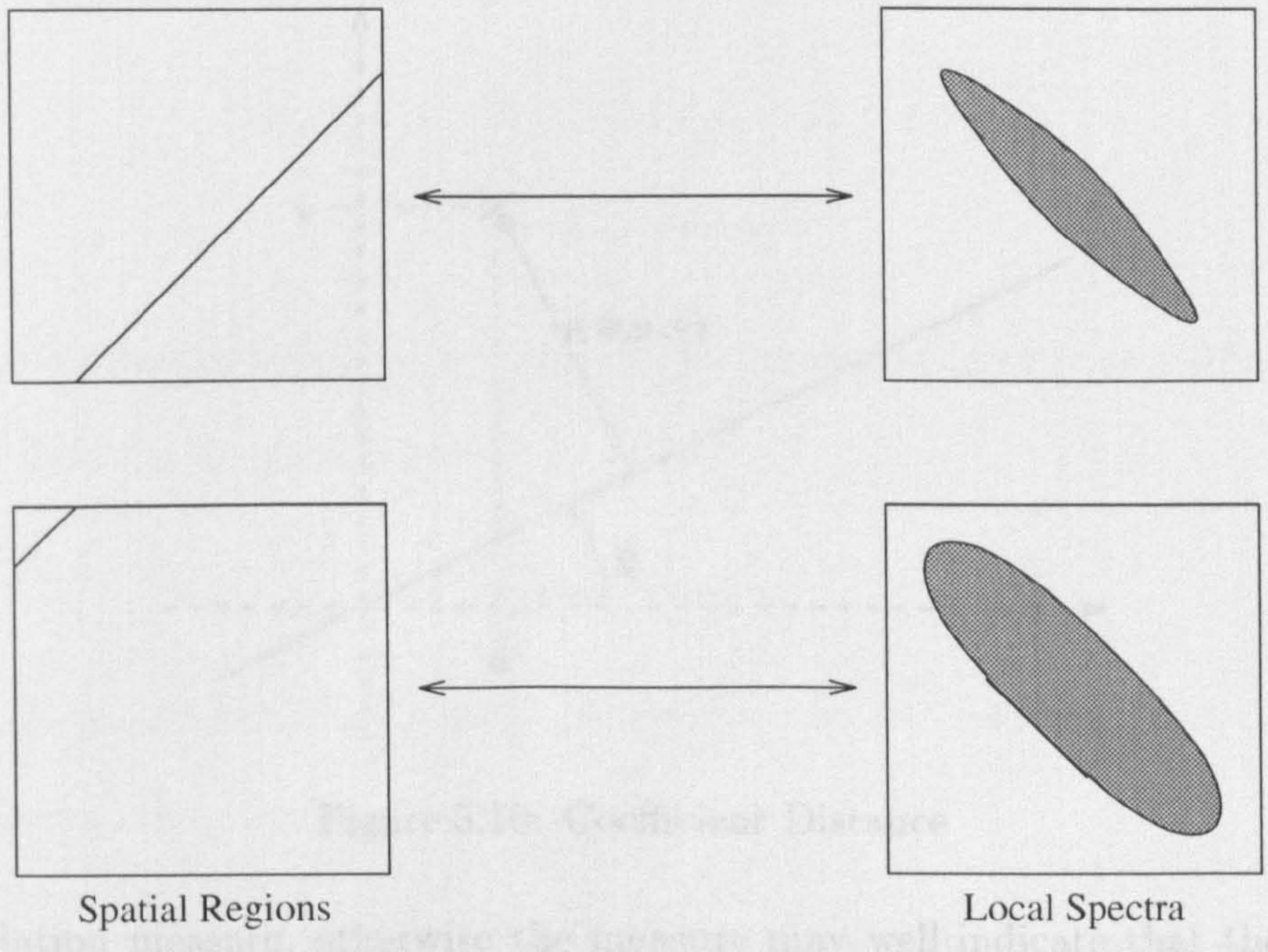


Figure 5.9: Spectrum Spreading

to give a measure of the amount of spreading. The similarity theorem [13] gives

$$w(a\xi) \leftrightarrow \frac{1}{|a|}\hat{w}\left(\frac{\omega}{a}\right) \tag{5.64}$$

This shows that shortening a feature, equivalent to contracting the windowing function ($a > 1$), causes the spreading function $\hat{w}(\omega)$ to dilate, increasing the smearing of the feature spectrum. Similarly, lengthening the feature reduces the smearing. Thus, although the phase of the local spectrum is only dependent upon the nature and centroid of the feature, the spread of the spectrum is dependent upon its length.

Consider the two local features detected in the regions shown in figure 5.9. It can be seen that, due to its orientation and position within the region, the feature in the lower region is shorter than the other. As such, although both have an oriented and concentrated spectrum, the spectrum corresponding to the shorter feature is more spread out. This suggests that compensation for the spreading must be built into

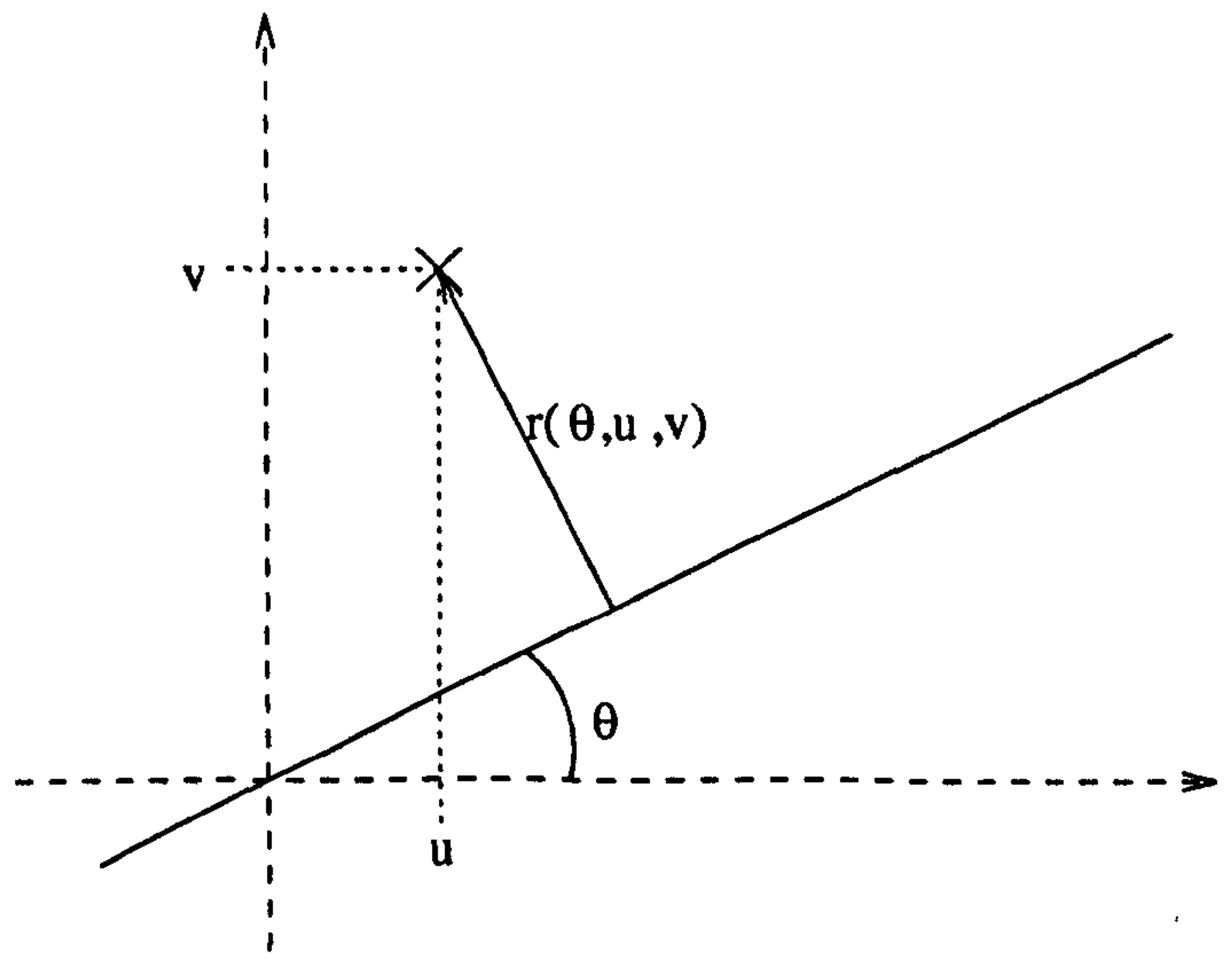


Figure 5.10: Coefficient Distance

the correlation measure, otherwise the measure may well indicate that the spectra relate to different features, whereas the difference is simply due to the position of the feature, relative to the centre of the window.

A simple approach to this is to assume that all the spectra are at least two samples wide and to compare a narrow band of coefficients along the spectrum orientation. A function $r(\theta, \vec{\omega}_i)$ is defined to give the distance of a sample point from a line passing through the origin of the spectrum with an orientation θ by

$$r(\theta, \vec{\omega}_i) = |\vec{\omega}_i \cdot \vec{v}'_\theta| \tag{5.65}$$

This can be expressed in terms of the two dimensional array and the indices u and v, as shown in figure 5.10, by

$$r(\theta, u, v) = |(u - \Xi(n) + 0.5)\sin\theta - (v - \Xi(n) + 0.5)\cos\theta| \tag{5.66}$$

A simple function $d_w(\theta, u, v)$ can then be defined to indicate whether a coefficient

is within a band w samples wide and with an orientation θ, ie.

$$d_w(\theta,u,v) = 1, \quad \text{if } r(\theta,u,v) \leq 0.5w \tag{5.67}$$

$$= 0, \quad \text{otherwise} \tag{5.68}$$

The correlation coefficient used to determine whether or not two primitive features should be linked is by given equation (5.33),

$$r = \sum_{i=0}^{N^2-1} \hat{x}_1(\vec{\omega}_i^{\theta_{12}})\hat{x}_2^*(\vec{\omega}_i)\exp \jmath[\vec{\xi}_{01}\cdot\vec{\omega}_i^{\theta_{12}} - \vec{\xi}_{02}\cdot\vec{\omega}_i] \tag{5.69}$$

where θ_{12} is the difference in orientation between the two features. In terms of the array representation, this may be rewritten as

$$r = \sum_{u=0}^{2\Xi(n)-1}\sum_{v=0}^{2\Xi(n)-1} \hat{x}_1(u^{\theta_{12}},v^{\theta_{12}})\hat{x}_2^*(u,v)d_2(\theta_2,u,v)\exp[\jmath\Omega(n)\phi(u,v,\vec{\xi}_{01},\vec{\xi}_{02},\theta_{12})] \tag{5.70}$$

where

$$\phi(u,v,\vec{\xi}_{01},\vec{\xi}_{02},\theta) = u^\theta\xi_{011} + v^\theta\xi_{012} - u\xi_{021} - v\xi_{022} \tag{5.71}$$

and u^θ and v^θ can be calculated from equation (5.49). The inclusion of the function $d_2(\theta_2,u,v)$ within the summation means that only the coefficients in a band two samples wide along the orientation of the spectrum $\hat{x}_2(u,v)$, θ_2, contribute to the correlation.

5.8 Computational Requirements

The computational requirements for applying the feature detection algorithms to a single level of the MFT were discussed in chapter 4. This section is concerned with the computation required in order to generate the multilevel image segmentation, and to link the features detected.

Initially the MFT of the image must be computed. For the relaxed ($k = 1$) MFT of an $M \times M$ image this requires [16]

$$\mathcal{N}_{\mathrm{MFT}} = 2M^2[(\log_2 M)^2 + \log_2 M - 2] + M^2 \log_2 M \tag{5.72}$$

complex multiplications, ie 592 multiplications per pixel for a 256×256 image. For this size of image it is assumed that there will be $\log_2 M + 1$ levels, including the original image.

For the segmentation, a worst case figure is given assuming that the feature detection process must be applied to all levels. Assuming that regions are all tested for up to two features then this requires $158(\log_2 M + 1)$ multiplications per pixel, eg. 1442 for a 256×256 image. This gives a total, including initial computation of the entire MFT, of 2034 multiplications per pixel, approximately equivalent to convolution by thirteen, 13×13 filter kernels. This figure is comparable to the requirements for applying the thirteen basis filters used by the steerable filter approach of Andersson[1], but the current method has the additional property of including an arc model, a determination of local orientation and curvature consistency built into the feature model and it gives global structure.

A more realistic figure can be obtained by considering the results presented in chapter 4, which indicate that for the images in question most of the features are detected at levels 2,3 and 4. Ignoring the scale consistency check, and assuming the that a third of the image is modelled at each of these levels, then the detection process is applied to all pixels at levels 0 to 2, two thirds at level 3 and one third at level 4. The computation requirement for the detection is therefore 632 multiplications per pixel. The inclusion of the MFT generation gives 1224 multiplications, approximately

equivalent to convolution by twelve, 10×10 filter kernels. The inclusion of scale consistency checking causes only a minor increase in computation, since the synthesis and correlation are directly comparable to those used for the feature detection stage, and therefore, as noted in table 4.1, only requires an extra 9 multiplications per pixel.

Additional computation is required for the linking process. The linking process for a given feature is clearly dependent upon the size of the spectra that represents it, ie. the scale at which it was detected. The correlation requires 4 floating point multiplies per pixel. The transformation stage, in order to synthesise one spectrum from another, requires 4 multiplications per pixel for the rotation (possibly including a scaling factor) and 1 complex multiply (4 floating point operations) for the centroid shift. Therefore a total of 12 floating point multiplications per pixel are required for the linking stage.

The total computational requirement for the boundary detection, based upon the assumptions made above, is approximately 1250 complex multiplications per pixel, ie. approximately equivalent to filtering with thirteen 10×10 kernels. The method presented has been shown to be computationally efficient and compares to the requirement of the steerable filter methods adopted by some authors [1, 34].

5.9 Experimental Results

Applying the hierarchical feature selection process described in this chapter to levels 2 to 4 of the "Shapes" MFT gives the results shown in figures 5.11 and 5.12. The result shown in figure 5.11 was obtained by ignoring the scale consistency and simply modelling the image features at the lowest possible level. Starting at level 2, an attempt was made to model each region of the image using the algorithm described

in chapters 3 and 4. Regions for which this process was unsuccessful were tried at level 3, and if this failed the detection was applied at level 4. It can be seen that on the whole this approach has produced an acceptable result. The bottom of the crescent is however slightly distorted by two almost parallel edges affecting the spectrum. Applying the child consistency check (shown in figure 5.12) to adjacent levels, the features detected at level 3 for this part of the image do not match those detected at level 4. The features detected at level 3 are then discarded in favour of those detected at the higher level. This also causes the top right corner of the square to be detected at level 3 as opposed to level 2, since this gives a better estimate of the positions of the edges that make up the corner. Figure 5.13 shows the results of applying the linking algorithm to the "Shapes" segmentation shown in figure 5.12. The colours this time represent features that have been linked. Clearly the square has been detected as four straight edges and the triangle as three. All the arc segments detected at the edge of the circle have been linked and the circle boundary is represented as a single feature. The star is represented by ten features and the crescent by two, as expected.

Figure 5.14 shows the result for the "Vine leaf" MFT. Most of the leaf boundary and inner detail have been detected, but since most of the features are small scale they have been mainly detected at levels 4 and 5. Figure 5.15 shows the results of applying the linking algorithm to these features. It is clear that many of the local features detected have been correctly linked with appropriate neighbouring features.

Figure 5.16 shows the result of applying the multilevel segmentation algorithm (with the scale consistency check) over levels 3 - 5 of the "Lena" MFT. It is clear that most of the large scale features are detected at level 3, but that smaller scale features, such as the edges of the eyes and nose are detected at higher levels. The results for

both the "Shapes" and the "Vine leaf" were generated using the linking process, and allowing additional features to be detected to fill in the gaps, as described in section 6.3. However, in both cases very few extra features were added in this final step. It is instructive however to consider the effects of allowing extra features to be added, by looking at the results of applying the algorithm to "Lena" both with and without this process. Figure 5.17 shows the result of applying the linking algorithm to the set of local features detected for "Lena" and shown in figure 5.16. It is clear that most of the features have been linked into larger scale boundaries. However, there is a gap in top of the curve representing the edge of the hat. Allowing additional features to be added, the result of which is shown in figure 5.18, the hole is filled in and the top of the hat is linked to form a single feature.

In some places, in each of the results, the boundaries look jagged - but it should be noted that all the features have been detected locally and are not considered, during the detection/estimation process as being part of some larger curve. It is only at the linking stage that attention is paid to the relationship between features detected in neighbouring regions. No smooth interpolation, such as the splines used by Calway[16], has been employed in drawing the output.

The use of arc and multiple feature in modelling an image region has allowed a segmentation into larger spatial regions than would be achievable using a single linear feature model (cf. [16]). In earlier work, eg. Calway[16], Parent and Zucker[72], features were linked together based upon the similarity of orientation and position of the set of feature centroids. Such an approach takes no account of the nature of the features, whether both lines or edges, for instance. By basing the comparison upon the set of frequency domain coefficients all the information about the feature

is present, since the transform is invertible, and hence the correlation takes account of more than just the position and orientation of the feature. For this reason, the result for the "Lena" image in the present work avoids some of the erroneous links in Calway's result [16], where edges of the mirror, hat and hair are linked together because of their similar orientation.

A further advance on previous work is the generality of the methods used - the general framework of parameter estimation followed by synthesis and correlation to test hypotheses has been used at all stages of the inference process: from local features of various types to global curves.

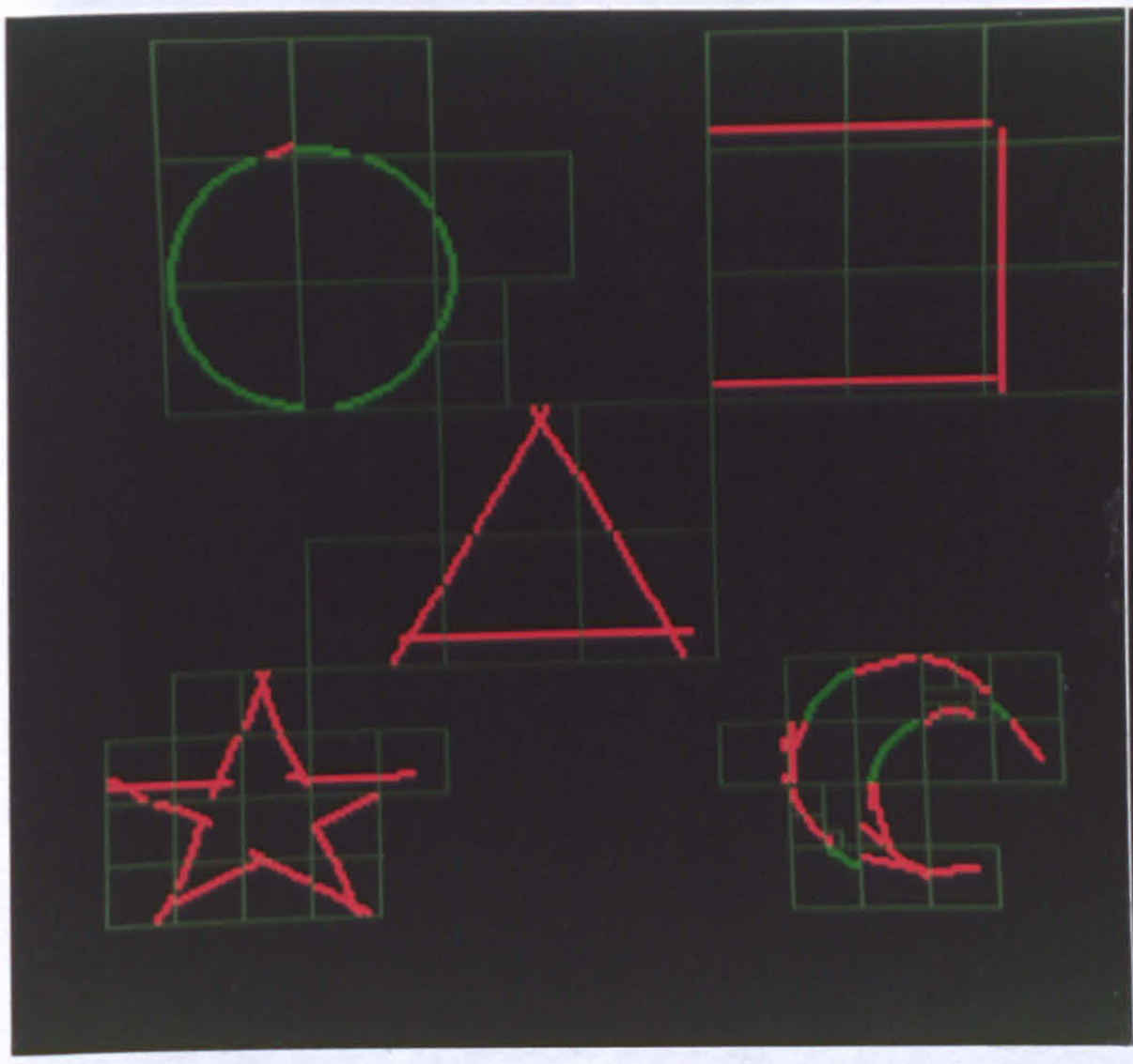


Figure 5.11: Shapes: Feature detection - over levels 2 - 4

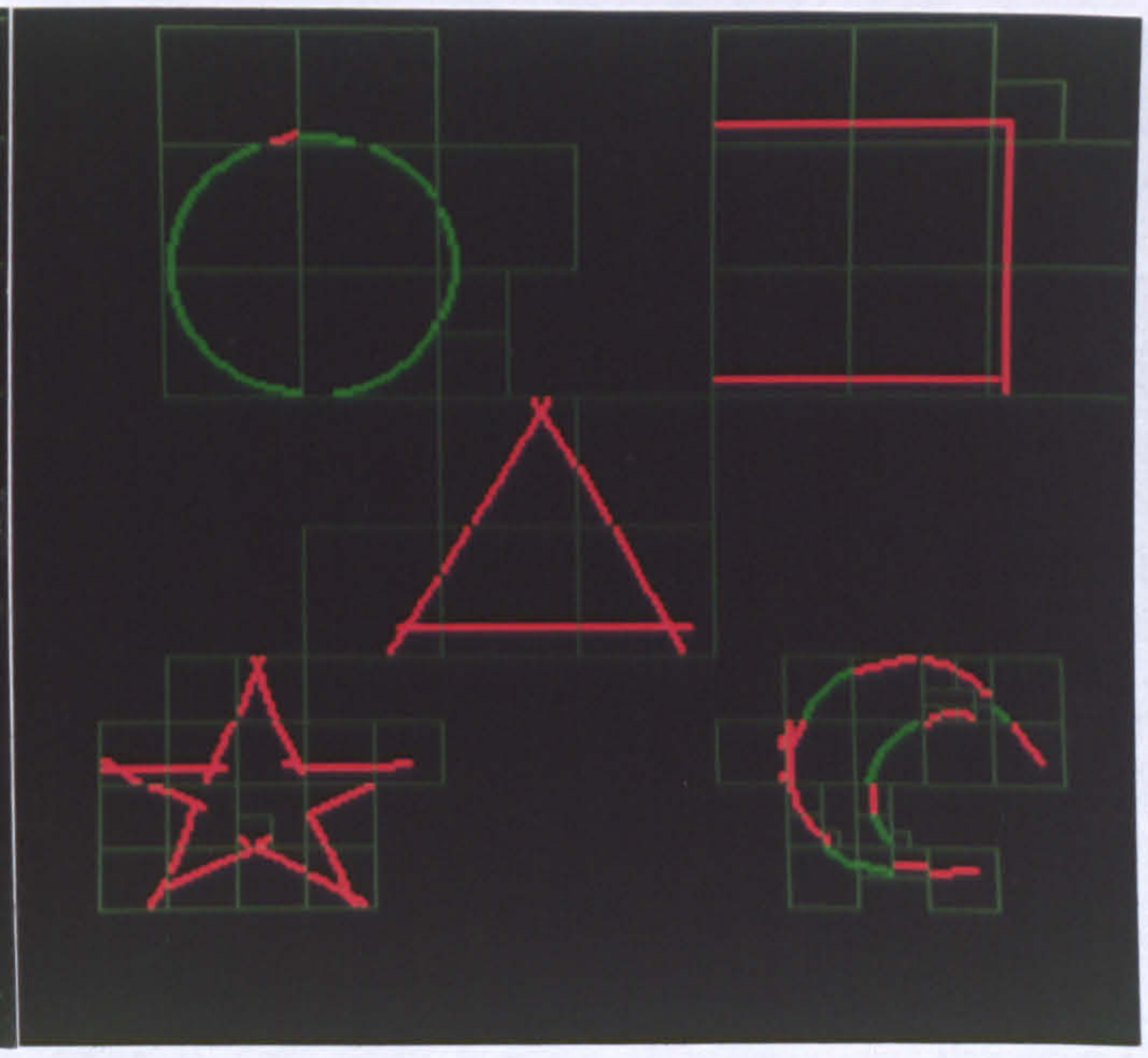


Figure 5.12: Shapes: Feature detection - with scale consistency

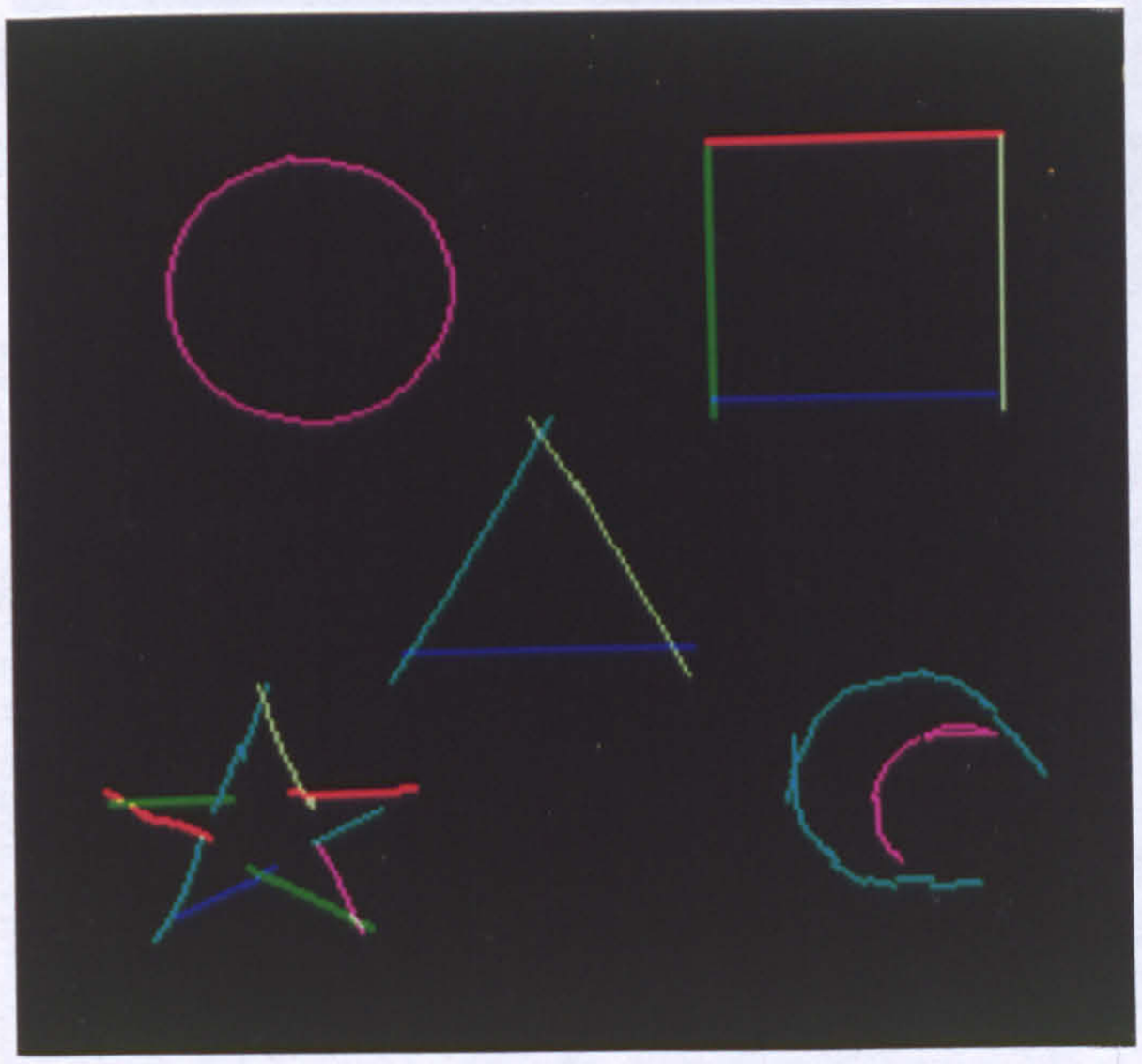


Figure 5.13: Shapes: Feature linking

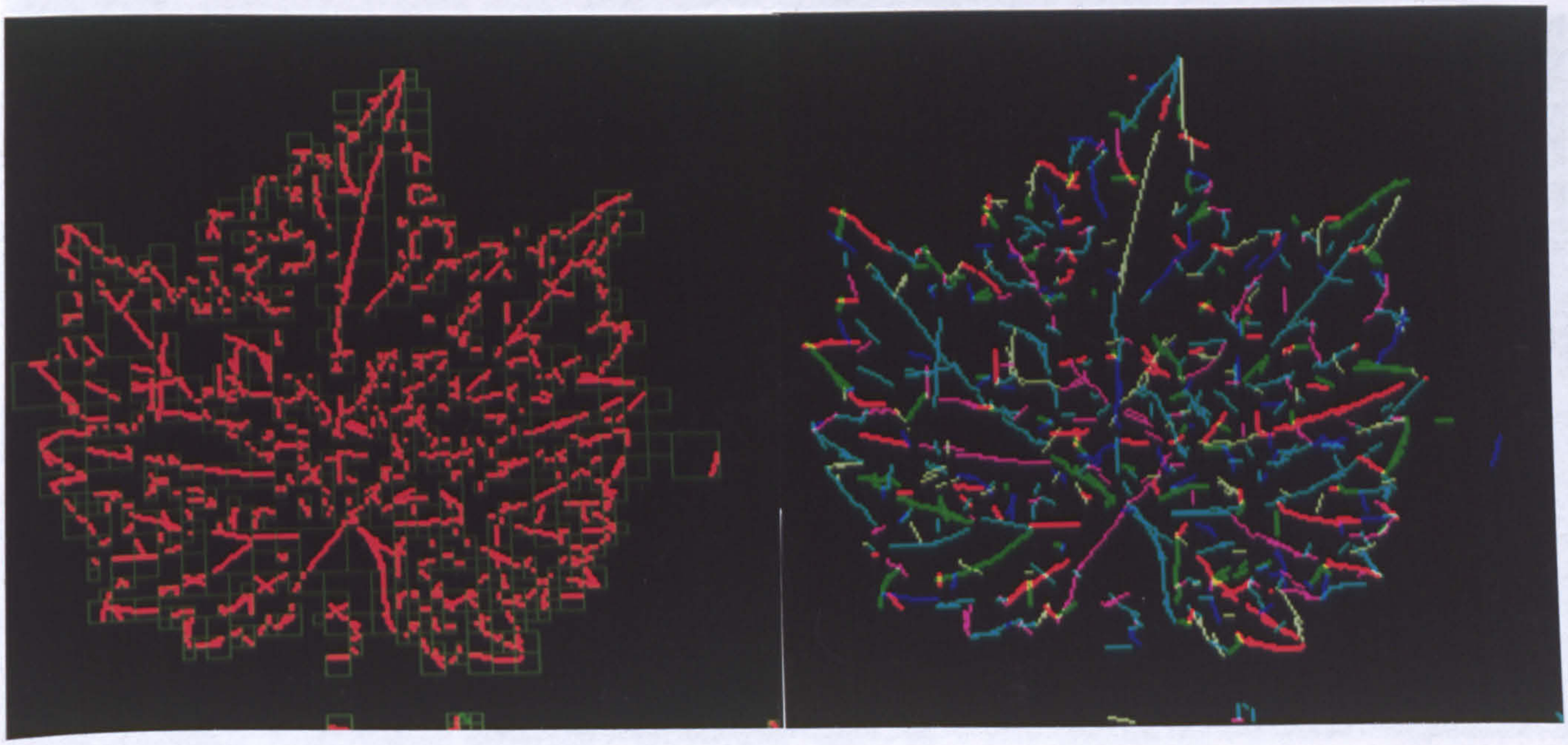


Figure 5.14: Vine leaf : Feature detection - over levels 3 - 5

Figure 5.15: Vine leaf: Feature linking

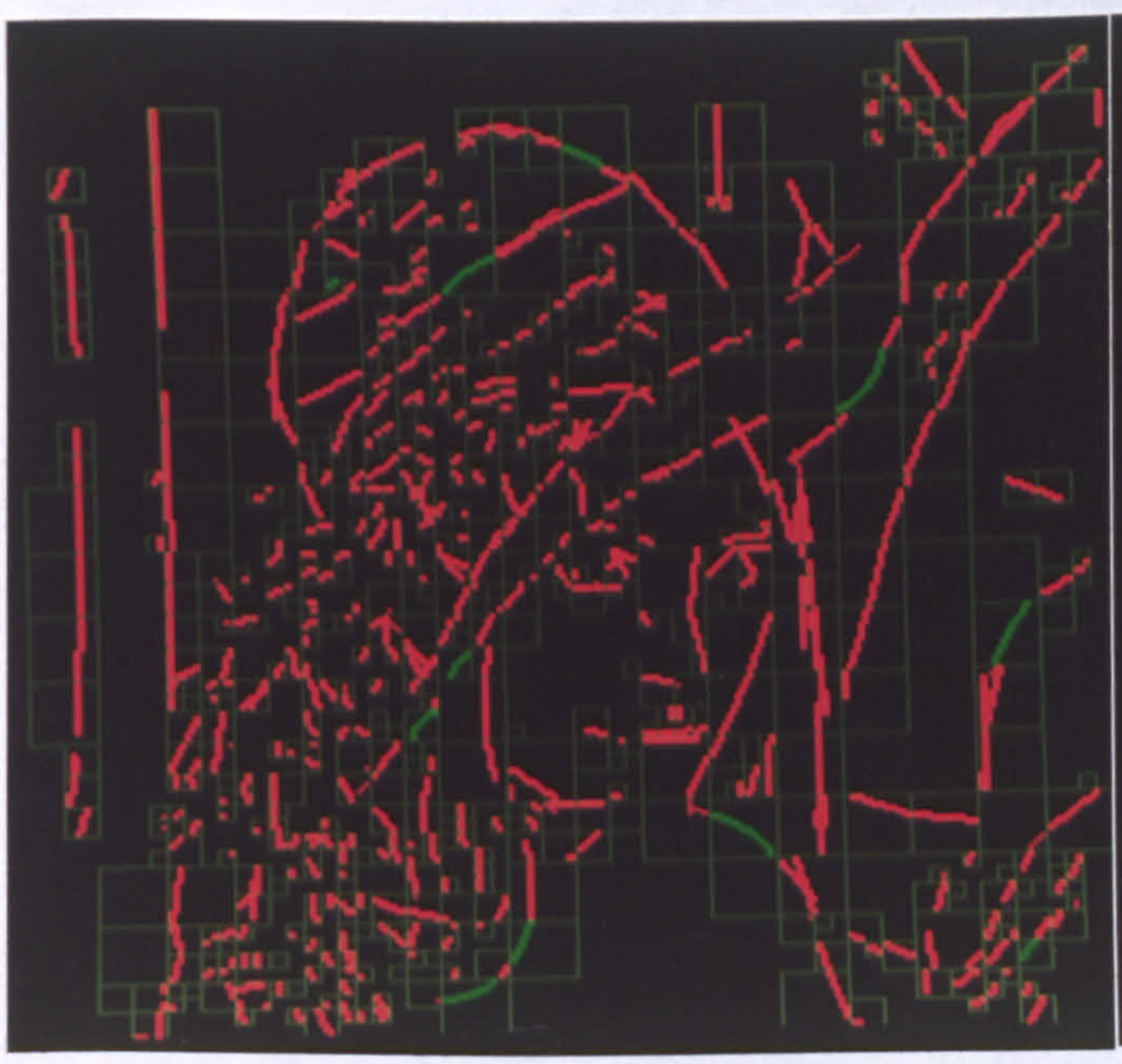


Figure 5.16: Lena: Feature detection - over levels 3 - 5

Figure 5.17: Lena: Feature linking

Figure 5.18: Lena: Feature linking - with backtracking for missed features.

Chapter 6

Conclusions

6.1 Thesis Summary

The work described in this thesis is concerned with the detection of image boundaries - the edges between homogeneous image regions. The approach used is based upon a multiresolution frequency domain representation - the MFT. The detection process uses a number of distinct phases - local feature detection, scale consistency based image segmentation and local feature linking. Each of these phases is based upon a general, model based decision process, which is illustrated in figure 3.4, and is similar to composite hypothesis testing[30].

For each phase, a model is presented of some image property of interest, be it a model of an image region or the relationship between regions, and a decision is made as to whether it can be used to describe a particular instance. This is achieved in three stages. First, a set of parameters is estimated, based upon some set of local spectra and/or previously estimated parameters. These parameters describe local image properties - such as local feature positions and orientations, or the transforms that can represent the relationship between two such features. Secondly, the parameter estimates are used to estimate the local spectrum of some region of the image. This

estimation, or spectrum synthesis, is based upon the model, in such a way that if the model holds, the synthesised spectrum should be similar to the corresponding data. The third and final step is to correlate the two spectra and use the result to decide whether the synthesis has been successful and hence whether the image property in question has been accurately modelled.

The rest of this section provides a review of the work described in this thesis. While reviewing this work, it is instructive to demonstrate how each part of the algorithm - local feature detection, scale consistent segmentation and linking - follows this approach.

Chapter 1 considered the problems of image processing, and compared it with the Human Visual System(HVS). A camera and some sampling process are used to give the artificial equivalent of the discrete image produced by the retina in the human eye. In both systems, this forms the input to the higher level processing necessary to describe the scene represented by the image in terms of some set of important properties or features. Marr's observation that it is often not possible to translate the raw iconic image directly into the desired representation, and that it is necessary to proceed "to the desirable via the possible", was noted[64]. It is often necessary, therefore, to perform a chain of transformations in order to produce the required result. However, since each transformation makes certain information explicit at the expense of other information, the intermediate representations and the order in which they are computed needs to be carefully considered.

Next, the problems of uncertainty were considered. The transformation of an image into a particular representation may often be regarded as giving a description of an image in terms of "what" is "where" in the image. This is equivalent to posing

the following question: can any given region of the image be represented by a feature from one of a set of feature classes? It was noted that there is a trade-off between resolution in the position and the property-space for the features of interest. This implies that achieving high accuracy in the feature space must be at the expense of high positional accuracy. A number of approaches to overcoming this using multiple resolution representations were briefly considered.

The importance of representation, and especially scale, was considered in more detail in chapter 2. First, the use of local processing was examined. It is clear that in order to analyse an image in terms of some model, that model must be sufficiently complex to describe all possible instances of the class of images of interest. A model for an entire image is, therefore, going to be extremely complex and cumbersome to deal with. By splitting the image into a set of regions and analysing these separately, the models can be chosen to be much simpler, such as the average orientation or grey level in the region. This leads to the question of what size the regions should be. If they are too small, very little is gained over the initial grey level representation. If too large then the model needs to be very complex. Therefore if the image is to be analysed at a single scale, chosen before analysis starts, it is necessary to use a sufficiently general model to account for all the possibilities. It was proposed in chapter 2 that an alternative approach is to choose a small number of simple feature classes and to independently vary the scale at which different regions of the image are analysed so that they may be modelled by these classes.

Next the use of *phase-space* representations as an appropriate framework in which to perform local analysis was considered. These representations conjoin some property coordinate to the spatial coordinate. Although there are many such representations

they fall into two main classes. The first of these is the class of *space-frequency* representations, in which a local spectrum estimate is used to represent each region of the image. In order to demonstrate the properties of this class of representations a specific example, the windowed Fourier transform (WFT), was considered in more detail. Although this provides a good general, local representation of the image, and is useful if the image properties of interest can be conveniently modelled in the frequency domain, the scale is fixed before analysis starts and is constant over the entire image. The selection of scale can therefore only be heuristically made, using knowledge of the class of images being considered. The scale may not be appropriate for the particular image under consideration, and even if it is a fair compromise for the image as a whole, it will not necessarily be the right choice for every region of it.

The *scale-space* representations form the second class of phase-space representations. These represent some property of the image over a range of scales. There are many such representations (most following the work of Witkin [98]) but recently much attention has been given to wavelet methods[27], such as the wavelet transform(WT). Although the image is represented over a range of scales the "features" represented are tied into the representation. Mallat's work, for example, uses a lowpass image plus a set of images that represent the detail of the image over a range of scales. Since the feature model is intrinsic to the representation, different WTs would be required to represent different image properties. It would be impossible, for example, to analyse an image in terms of edges and textures using a single wavelet representation. When considered as a space-frequency representation, the WT exhibits a "constant Q" characteristic - analysis of high frequencies leads to low frequency resolution [79].

In order to overcome the limitations and combine the advantages of both of these

types of representation, the multiresolution Fourier transform (MFT) was presented. This may be considered as a stack of WFTs, computed over a range of scales. Each level is a complete representation of the image and is invertible. The pyramidal nature of this representation means that each region represented at a given level is represented by its four quadrant regions at the level below. Since the image is represented over a range of scales, decision about the choice of scale that best represents each region of the image can be based upon the local properties of the image. Following the work of Calway[16], a model-based segmentation process can be used to choose the appropriate scale at which to represent image localities.

An important aspect of any local representation is the choice of window function and discussion of the factors involved in this choice was presented, along with a description of the function chosen - a relaxed form of the finite prolate spheroidal sequences (FPSS).

A form of multiresolution image modelling, based upon that developed by Clippingdale[25] and applied in the MFT framework, was then presented. This allows a quadtree-like representation of the image to be constructed, where each leaf represents a region of the image in terms of features from some class of image models.

In chapter 3, the modelling of local image features using local spectrum estimates was discussed. The approach taken is to model an image region directly in terms of an invertible representation of it, its spectrum, and as such avoids splitting the process into an edge detection followed by some linking or grouping process. A novel general approach to detecting such local features was described, which may be summarised as a three stage process:

Estimation: A parameter estimator based upon a model of some local feature class is applied to a local spectrum.

Synthesis: The parameter estimates are then used to synthesise the spectrum that would be expected if such a feature were present at that location.

Decision: The synthesised and actual spectra are correlated to measure their similarity. This can be used to decide if the parameter estimates model the region sufficiently closely.

As examples of this approach, models were derived for two classes of image feature: straight line or edge segments, and pieces of circular arc. The first class closely follows the linear feature models used by Calway and is based upon using the linear phase characteristic of a single feature, in which the phase gradient is dependent upon the feature centroid. The energy spectrum of such a feature may be regarded as a concentrated ellipse, with its major axis perpendicular to the feature orientation. Estimators for both feature centroid and orientation were derived, based upon these models.

In an extension of Calway's work, the case of regions containing more than one linear feature was considered. Such regions correspond to corners or junctions within the image and extending the feature model in this way allows such regions to be detected. These regions may be modelled by the sum of the component spectra due to the individual features. Under the condition that these spectra are sufficiently concentrated and differ sufficiently in orientation, such that at most frequencies only the spectrum of one feature has significant energy, it was noted that it is possible to segment the region spectrum into its components. A new algorithm to perform

this segmentation, using centroid estimates at each frequency, and based upon the ISODATA[36] algorithm, was described.

Another extension of the feature model was based upon allowing boundaries to be modelled as curves. This allows boundary segments with high curvature to be more directly modelled. Although the model derived in chapter 3 was restricted to segments of circular arc, given the condition that a feature only has one point in any orientation, a similar approach could be used to derive models of more complex curve segments, eg. elliptical or parabolic arcs. It seems doubtful whether the increase in complexity which this would entail would be justified by the gain in representational power.

By directly modelling regions in terms of local features, such as line segments or arcs, orientation and curvature constraints, such as those imposed by Parent and Zucker's relaxation labelling process[72], are intrinsic to the model - not an ad hoc addition. It is clear that if the boundary in an image region can be modelled by a linear feature, then the edge points in the region must have a consistent orientation. For arc segments the local curvature must be constant over the region.

Hough transform (HT) [50] based feature detection is based upon the co-linearity (or co-circularity) of a set of edge points. Such an approach ignores the connectedness of this set of points and as such may result in a number of disjoint image points being detected as a single feature. An HT based method has three distinct phases - edge detection, accumulation and peak detection - analysis of the method is complicated by the need to consider the interaction between these phases. The computational requirement, mainly in the peak detection stage, is dependent on the size of the accumulator array. Increasing the accuracy of the estimation by increasing the number of

bins for a given parameter, leads to a corresponding increase in the size of the search space. Increasing the number of parameters used to represent a feature, increases the number of dimensions in the Hough space and this also serves to complicate the peak detection.

The implementation of the feature detection process, using the discrete MFT to provide the local spectrum estimates, was considered in chapter 4. The detection process estimates the centroid of the feature and its orientation. For regions where the feature ends within the region (cf. Andersson[1]), it is desirable to provide an estimate of the feature endpoints. A method for performing this estimation, based upon computer graphic line clipping techniques, was developed in chapter 4. The chapter was concluded with a set of results obtained by applying the detection processes to a number of real and synthetic images. Each result was obtained at a single resolution and demonstrates the effectiveness of the approach developed in chapter 3.

In chapter 5 the linking together of these local features into larger scale boundary structures was discussed. Based upon the multiresolution feature model described in chapter 2, a new method of segmenting an image into a number of regions was presented. This approach, in its simplest form, is close to the process used to generate either a grey level or line quadtree. The image is repeatedly subdivided until each region can be represented, either by a number of linear features, or a single section of circular arc, using the feature detection procedures described in chapter 3. The issue of scale consistency was also addressed. Features detected at one level should be similar to those detected in neighbouring levels. When applied to a single MFT level, the detection process uses the correlation measure to decide whether a region is modelled sufficiently well. However, the choice of threshold determines what proportion

of the region energy may be ignored by this process. In a given region a large feature may dominate the spectrum, so that it passes the threshold for single linear feature detection. In this case, no attempt is made to detect multiple features. If another feature, with lower energy, also exists within the region then this will be missed. In an attempt to overcome this, features detected at one level can be compared to those detected in the region's children, since features at adjacent scales should be similar. If the parent region has been successfully modelled, every feature detected in a child region should also have been detected in its parent. If not, then the region should be modelled at the higher, child level. A novel method for detecting any inconsistency between adjacent scales was described. In terms of the estimation-synthesis-decision process described above, this may be summarised:

Estimation: Features detected in the parent regions are truncated to one of its children in order to estimate the parameters of features in the child.

Synthesis: The estimated child parameters are used to synthesise the local spectrum corresponding to the child region.

Decision: By correlating the synthesised spectrum with the actual spectrum corresponding to features detected in the child region, a decision is made concerning whether the child features have been successfully accounted for in the parent.

Note the similarity between this form of hypothesis testing and that used for the initial feature detection.

The local features are then linked together to give the boundaries in the image. The linking is performed by considering the similarity between local features in adjacent regions that represent part of the same boundary. If a curve passes through

two regions, then the linear features used to model it in those regions should be similar - allowing for a shift of position and, assuming continuity of the curve, a small change in orientation. If this shift and rotation are known, then it should be possible to transform one local feature into the other. Since each feature is represented in the MFT domain by the set of coefficients that represent its local spectrum, the transformation can be performed in the Fourier domain. This was also considered in chapter 5, where the relationship between the transformations in the two domains was described. In summary, the linking algorithm may be described as follows:

Estimation: Using the estimated feature parameters, the transformation relating the two adjacent spectra is estimated.

Synthesis: The estimated transform is used to synthesise one spectrum from the other.

Decision: By correlating the synthesised spectrum with the actual spectrum, a measure of feature similarity is calculated which is used to decide whether the features are part of the same boundary.

Linking is performed as a local process, but can result in global structure as the results of chapter 5 showed. Features are correlated with other features detected in neighbouring image regions and linked to those which pass the similarity threshold. This approach may be compared to that of Zucker [101], amongst others, ie. linking trace points to their neighbours, in that only neighbouring features are considered. This leads to global structures evolving as local features are joined to their neighbours.

In chapter 5, consideration was also given to features that may be missed by the local detection process. In addition to a feature being "masked" by a higher energy

feature in the same region, there are other reasons for a feature being missed, such as noise or the existence of background texture, where the features will be missed because they don't account for a high enough proportion of the region energy. When linking features, it may appear that a boundary continues into a region in which no appropriate features have been detected. A method was described for backtracking to search for such features, using an assumption about the continuity of the boundary to attempt a synthesis of the spectrum corresponding to the potential feature using a feature detected in a neighbouring region. The approach used followed closely the linking procedure, except that since the parameters are obviously unknown for the "missing" feature, the transformation between the two spectra cannot be calculated. Because of continuity, it can be assumed that if this feature exists it will have a "similar" orientation to the known feature, and therefore it will have a similar spectrum, allowing for a small rotation and a spatial shift. A method was described for estimating the spatial shift between the two features, based upon calculating the inner product of the two spectra, assuming that the difference in orientation is known. Given a set of likely rotations, Θ, then for each $\theta_i \in \Theta$ the following algorithm is applied:

Estimation: Assuming a feature exists, and is in an orientation θ_i relative to the known feature, the shift in position is estimated. Using this and the given rotation, the transformation required to transform the known feature spectrum into that corresponding to the "missed" feature is calculated.

Synthesis: The estimated transform is used to synthesise the potential feature spectrum from the known feature spectrum.

Decision: By correlating the synthesised spectrum with the actual spectrum, a measure is given of how well the transformed feature models the region. If the correlation is sufficiently high the set of coefficients corresponding to the synthesised spectrum can be passed to the feature detection algorithm to estimate position and orientation.

This approach may allow gaps in the detected boundary to be filled in, as was demonstrated in a result presented for the "Lena" image. A gap in the top of the hat edge, which was initially missed by the detection algorithm due to low contrast, is picked up using this backtracking method.

6.2 Overview of Boundary Detection Algorithm

The segmentation and linking algorithms are the two parts of a boundary detection algorithm, an overview of which is shown in figure 6.1. An MFT is generated for the image, providing a set of local spectrum estimates over a range of scales. The segmentation algorithm uses a top down approach and starting at a low scale, ie. large regions, attempts to fit the feature models to each of the local spectra. In each region where this is successful, and the scale consistency criterion is met, a feature is detected. If this fails for a region, then the process is applied to each of its four children. This continues until the region has been successfully modelled. The next step is to attempt to link each feature to any features detected in neighbouring regions. In the case where a feature appears to continue outside the region, the "missing features" process is applied to take a second look, this time using knowledge of where and what the feature may be.

Each stage of this process uses the same general framework of estimation of param-

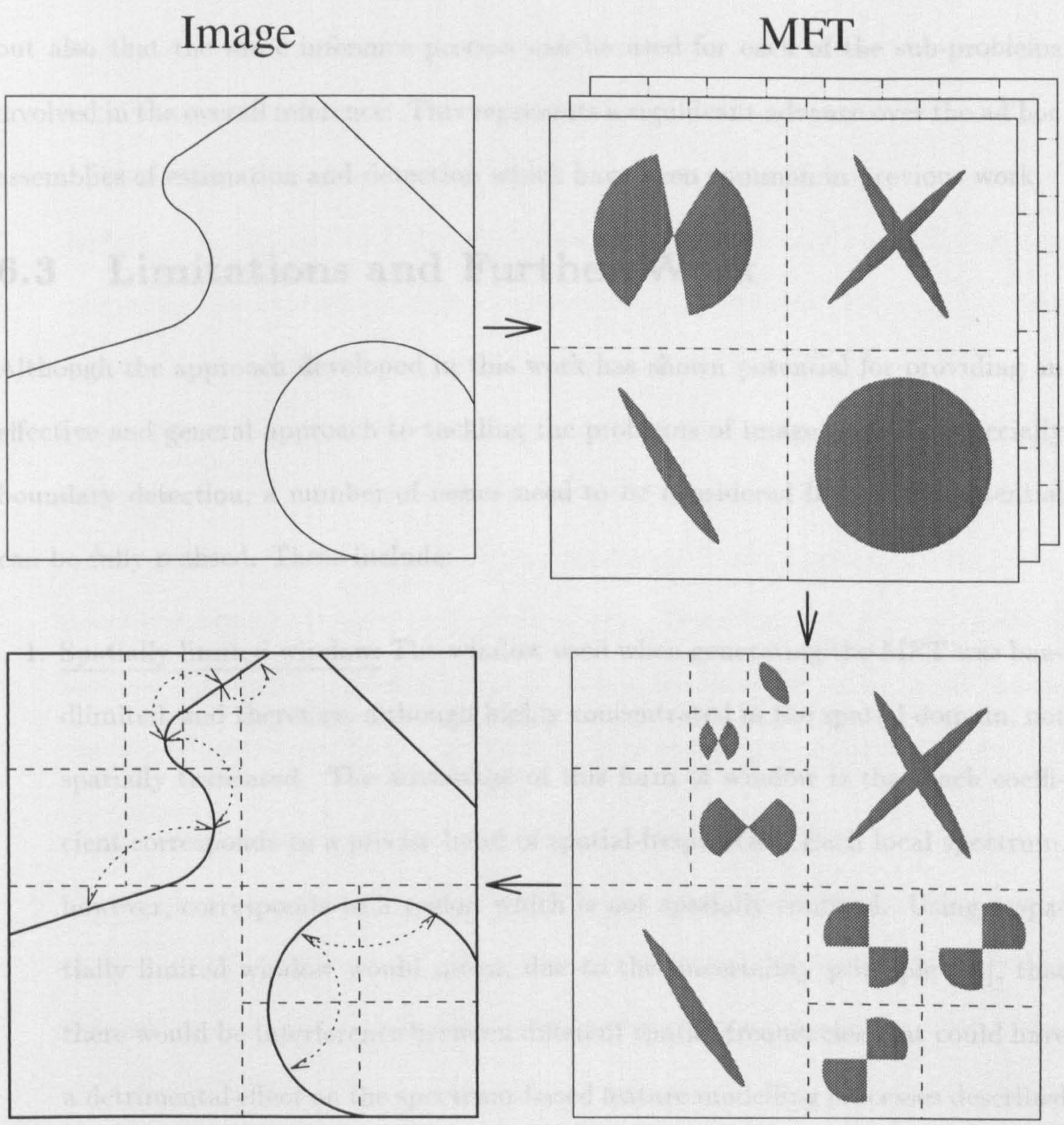


Figure 6.1: Segmentation and Linking Algorithm

eters, synthesis and hypothesis testing. The major contribution of the present work is perhaps to show not only that the MFT can be used as a vehicle for such inference, but also that the same inference process can be used for each of the sub-problems involved in the overall inference. This represents a significant advance over the ad hoc assemblies of estimation and detection which have been common in previous work.

6.3 Limitations and Further Work

Although the approach developed in this work has shown potential for providing an effective and general approach to tackling the problems of image analysis, especially boundary detection, a number of issues need to be considered before this potential can be fully realised. These include:

1. Spatially limited window: The window used when generating the MFT was bandlimited, and therefore, although highly concentrated in the spatial domain, not spatially truncated. The advantage of this form of window is that each coefficient corresponds to a precise band of spatial-frequencies. Each local spectrum, however, corresponds to a region which is not spatially confined. Using a spatially limited window would mean, due to the uncertainty principle [70], that there would be interference between different spatial-frequencies that could have a detrimental effect on the spectrum-based feature modelling processes described in chapters 3 and 4. Although it is clear from the results included in chapter 4 that the feature detection process produces good results using bandlimited windows, further experimentation could be performed to compare these results with results obtained using a spatially truncated window.

2. Extension of local feature models to other boundary features: The detection processes derived and implemented in the current work have considered two types of local image feature: straight line and edge segments and circular arc segments. From the results in chapter 4, it is clear that these can be used to model the boundary structure of natural and synthetic images and that the expansion of the feature model used by Calway[16] from single linear feature regions to regions containing arcs or multiple linear features increases the size of the regions that can be directly modelled from their local spectra, as well as allowing corners to be modelled. Expanding the classes of boundary features that are detected may lead to further improvements in this area and allow more complicated curves to be modelled directly. However this expansion would increase the computational complexity of the detection algorithm. There appears to be a trade-off between the number of feature classes, and hence the scale at which the image can be segmented and the complexity of the processing required to perform the segmentation.

3. Extension of local feature model to region properties: The work has been based on the detection of local edge features that represent the boundaries between different spatial regions. These regions are differentiated by regional properties such as grey level or texture. Since it has been reported that the use of boundary information can be used to aid region segmentation processes (eg. [9]), it is clear that inclusion of some form of region-based information could help the boundary detection process. The use of the local spectral information to analyse and segment textures has been considered in [45]. Combining region and boundary

models could lead to a segmentation of the image in terms of both its regional and boundary information.

4. Effects of noise: A linear feature is modelled in terms of its local spectrum which, as well as its linear phase characteristic, has its energy concentrated about a specific orientation. It has been shown [31] that since the feature energy is concentrated, while energy due to white noise remains evenly spread over the local spectrum, the signal to noise ratio at the frequencies corresponding to the feature is increased. Since the concentration of the feature spectrum is dependent upon the feature position within the region (chapter 5), this will affect the degree of noise reduction obtained by transforming that region to the frequency domain. Moreover, the phase differencing used to estimate the linear phase characteristic (chapter 3) performs some local averaging that further reduces the noise contribution. Regions containing multiple linear and circular arc features have their feature energy less highly concentrated than single linear feature regions, and this would suggest that the noise reduction effects in these regions would be less. However, these models allow regions to be considered at larger scales and this may have a positive effect on the noise immunity of the algorithm. Further work should consider these issues.

5. Boundary determination: The boundary linking process is based on local similarity between features detected in neighbouring regions, measured by correlating their local spectra. This indicates whether or not features should be linked. No account is taken, however, of the overall structure of the boundary. A feature may link to more than one neighbouring feature and the boundary may there-

fore appear to branch at this point, as is clear from the boundaries detected in the "Lena" image (chapter 5). Further consideration should be given to deciding whether such linked features really represent some boundary structure within the image or are due to uncertainty in the feature detection and linking processes. Inclusion of region information (above) should also help this process.

6.4 Concluding Remarks

This work has presented a general method for detecting boundary structures in images. It is clear that such structures are perceptually important to the human visual system - the most powerful image processing system we know. This suggests that boundary detection will form an important part of any artificial vision system that aims to compete in performance and generality with the HVS. It has been shown by many authors that the use of boundary information can be effectively employed in a large number of image processing tasks, such as restoration, coding and segmentation. Additionally such boundaries may be detected as the first step in a system that uses an 2-d image as a basis for inferring a model of the 3-d scene it conveys.

The detection methods presented have been designed to be of general use, and as such do not rely on any domain specific knowledge. The local feature detection is based on estimating a "best fit" feature of a particular class, and then deciding whether it models a region accurately enough. The method of linking these local features was based upon their similarity, and apart from assuming continuity, no assumptions were made about the form of the boundaries.

Although the work has been concerned with two specific local image models, the framework for their detection and linking was developed so as to be extendible both

to other types of boundary feature and to region based features, such as textures. The work has extended the work of Calway, and it is hoped that it has increased the use of such an approach as a framework in which to perform higher level processing.

Appendix

Conference Paper [32]

Published papers not copied on instruction from the university

Bibliography

[1] M.T. Andersson. *Controllable Multidimensional Filters and Models in Low Level Computer Vision.* PhD thesis, University of Linköping, Sweden, 1992.

[2] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image Coding Using Wavelet Transform. *IEEE Trans. I.P.*, 1(2):205–220, 1992.

[3] D. Ballard. Generalising the Hough Transform. *Pattern Recognition*, 13:111–22, 1981.

[4] D. H. Ballard and C. M. Brown. *Computer Vision.* Prentice Hall, New Jersey, 1982.

[5] S.T. Barnard and M.A. Fischler. Computational Stereo. *ACM Computing Surveys*, 14(4):553 – 572, 1982.

[6] J. Besag. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society*, 48(3):259–302, 1986.

[7] A. Bhalerao and R. Wilson. Multiresolution Image Segmentation Combining Region and Boundary Information. In *Proc. 7th Scandinavian Conf. Im. Analysis*, pages 1162–1169, Aalborg, Denmark, 1991.

[8] A. Bhalerao and R. Wilson. Multiresolution Image Segmentation Combining Region and Boundary Information. In P. Johansen and S. Olsen, editors, *Theory and Applications of Image Analysis*, pages 148–161. World Scientific, 1992.

[9] A. H. Bhalerao. *Multiresolution Image Segmentation.* PhD thesis, Department of Computer Science, The University of Warwick, UK, October 1991.

[10] J. Bigün. Local symmetry Features in Image Processing. Linköping Studies in Science and Technology Disertation No. 179, Computer Vision Laboratory, Linköping University, Sweden, 1988.

[11] C. Blakemore and F. W. Campbell. On the Existence in the Human Visual System of Neurons Selectively Sensitive to the Orientation and Size of Retinal Images. *Journal of Physiology*, 203:237–60, 1969.

[12] A. I. Borisenko and I. E. Tarapov. *Vector and Tensor Analysis.* Dover Publications, Inc., New York, 1979.

[13] R. N. Bracewell. *The Fourier Transform and Its Applications.* McGraw Hill Book Company, second edition, 1986.

[14] P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Trans. Comp.*, COM-31:532–540, 1983.

[15] P. J. Burt, T. H. Hong, and A. Rosenfeld. Segmentation and Estimation of Image Region Properties Through Cooperative Hierarchical Computation. *IEEE Trans. Sys, Man Cyber.*, 11(12):802–809, 1981.

[16] A. Calway. *The Multiresolution Fourier Transform: A general Purpose Tool for Image Analysis.* PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1989.

[17] A. Calway, H. Knutsson, and R. Wilson. Multiresolution Estimation of 2-D Disparity Using a Frequency Domain Approach. Research Report RR196, Department of Computer Science, University of Warwick, UK, October 1991.

[18] A. Calway and R. G. Wilson. A Unified Approach to Feature Extraction Based on an Invertible Image Transform. In *Proc. 3rd IEE Int. Conf. Image Processing*, pages 651–655, Warwick, U.K, 1989.

[19] F. W. Campbell and J. Robson. Application of Fourier analysis to the visibility of gratings. *Journal of Physiology*, 197:551–566, 1968.

[20] J. Canny. A Computational Approach to Edge Detection. *IEEE Trans. PAMI.*, 8:679–698, 1986.

[21] H. Choset, R. Bajcsy, and H. Anderson. Active Vision With Multiresolution Wavelets. Research Report MS-CIS-89-73, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, October 1989.

[22] T. A. C. M. Classen and W. F. G. Mecklenbräuker. The Wigner Distribution - A Tool for Time-Frequency Analysis. Part I: Continuous-Time Signals. *Phillips J. Research*, 35:217–250, 1980.

[23] T. A. C. M. Classen and W. F. G. Mecklenbräuker. The Wigner Distribution - A Tool for Time-Frequency Analysis. Part II: Discrete-Time Signals. *Phillips J. Research*, 35:276–300, 1980.

[24] T. A. C. M. Classen and W. F. G. Mecklenbräuker. The Wigner Distribution - A Tool for Time-Frequency Analysis. Part III: Relations with other Time-Frequency Signal Transformations. *Phillips J. Research*, 35:372–389, 1980.

[25] S. Clippingdale. *Multiresolution Image Modelling and Estimation.* PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1988.

[26] S. Connelly and A. Rosenfeld. A Pyramid Algorithm for Fast Curve Extraction. *Computer Vison, Graphics and Image Processing*, 49:322 – 345, 1990.

[27] I. Daubechies. The Wavelet Transform, Time-Frequency Localisation and Signal Analysis. *IEEE Trans. Information Theory*, 36:961–1005, 1990.

[28] I. Daubechies. *Ten Lectures on Wavelets.* SIAM Press, Philadelphia, 1992.

[29] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency and orientation optimized by two-dimensional vision cortical filters. *J. Opt. Soc. Amer. A*, 2(7):1160–1169, July 1985.

[30] W. B. Davenport and W. L. Root. *An Introduction to the Theory of Random Signals and Noise.* IEEE Press, New York, 1987.

[31] A. Davies and R. Wilson. Linear Feature Extraction Using the Multiresolution Fourier Transform. Technical Report RR170, Department of Computer Science, University of Warwick, 1990.

[32] A. R. Davies and R. G. Wilson. Curve and Corner Extraction using the Multiresolution Fourier Transform. In *Proc. of 4th IEE Conf. Image Processing and its Applications, Maastricht, The Netherlands*, pages 282 – 285, 1992.

[33] R. O. Duda and P. E. Hart. Use of the Hough Transform to Detect Lines and Curves in Pictures. *Communications of the ACM*, 15:11–15, 1972.

[34] W.T. Freeman. *Steerable Filters and Local Analysis of Image Structure.* PhD thesis, Vision and Modelling Group, Media Lab, MIT., June 1992.

[35] W.T. Freeman and E. H. Adelson. The Design and Use of Steerable Filters. *IEEE Trans. P.A.M.I*, 13(9):891–906, September 1991.

[36] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1972.

[37] D. Gabor. Theory of Communications. *Proc. IEE*, 93(26):429–441, November 1946.

[38] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans. PAMI.*, 6(6):721–741, 1984.

[39] J. J. Gibson. *The Senses Considered as Perceptual Systems.* Houghton Mifflin, Boston, 1966.

[40] R. C. Gonzalez and P. Wintz. *Digital Image Processing (2nd. Ed).* Addison-Wesley, New York, 1987.

[41] A. Grossman and J. Mortlet. Decomposition of Hardy functions into square integrable wavelengths of constant shape. *SIAM J. Math. Anal.*, 15:723–736, 1984.

[42] J. F. Haddon and J. F. Boyce. Simultaneous Image Segmentation and Edge Detection. In *Proc. IEE 3rd Int. Conf. on Image Proc. and its Applications*, pages 411–415, University of Warwick, U. K., 1989.

[43] D. Hearn and M. P. Baker. *Computer Graphics.* Prentice-Hall International, 1986.

[44] F. Hlawatsch and G. F. Boudreaux-Bartels. Linear and Quadratic Time-Frequency Signal Representations. *IEEE Signal Processing Magazine*, pages 21 – 67, April 1992.

[45] Tao-I. Hsu, A.D. Calway, and R. Wilson. Texture Analysis using the Multiresolution Fourier Transform. In *Proc. 8th Scandinavian Conf. Im. Analysis*, Tromso, Norway, 1993.

[46] D. H. Hubel. *Eye, Brain and Vision.* New York, Scientific American Library, 1988.

[47] D. H. Hubel and T. N. Wiesel. Recpetive fields and functional architecture of monkey striate cortex. *Journal of Physiology*, 198:215 – 243, 1968.

[48] R. A. Hummel and S. W. Zucker. On the Foundations of Relaxation Labeling Processes. *IEEE Trans. PAMI.*, 5(36):267–287, 1983.

[49] G. W. Humphreys and V. Bruce. *Visual Cognition: Computational, Experimental and Neuropsychological Perspectives.* Lawrence Erlbaum Associates, 1989.

[50] J. Illingworth and J. Kittler. A Survey of the Hough Transform. *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.

[51] A. K. Jain. *Fundamentals of Digital Image Processing.* New York, Addison-Wesley, 1989.

[52] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. *Int. J. Computer Vision*, 1:321–332, 1988.

[53] R. Kirsch. Computer Determination of the Constituent Structure of Biological Images. *Computational Biomedical Research*, 4:315 – 328, 1971.

[54] H. Knutsson. *Filtering and Reconstruction in Image Processing.* PhD thesis, University of Linköping, Sweden, 1982.

[55] H. Knutsson. Representing Local Structure Using Tensor. In *Proc. of Sixth Scandinavian Conference on Image Analysis*, pages 244 – 251, 1989.

[56] H. Knutsson, R. Wilson, and G. Granlund. Anisotropic Nonstationary Image Estimation and Its Applications: Part 1 - Restoration of Noisy Images. *IEEE Transactions on Communications*, 31:388 – 397, 1983.

[57] H. E. Knutsson, R. Wilson, and G. H. Granlund. Estimation the Local Orientation of Anisotropic 2-d Signals. In *Proc. IEEE ASSP Workshop on Spect. Est.*, pages 234–9, Tampa, 1983.

[58] S. Levialdi. Edge Extraction Techniques. In O. D. Faugeras, editor, *Fundamentals in Computer Vision*, pages 118–144. Cambridge University Press, 1983.

[59] Y. D. Liang and B.A. Barsky. A New Concept and Method for Line Clipping. *ACM Transactions on Graphics*, 3:1–32, 1984.

[60] J. S. Lim. *Two-dimensional Signal and Image Processing.* Prentice-Hall, 1990.

[61] S. G. Mallat. A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Trans. P.A.M.I*, 11(7):674–693, July 1989.

[62] S. G. Mallat. Multifrequency Channel Decompositions of Images and Wavelet Models. *IEEE Trans. A.S.S.P*, 37(12):2091–2110, December 1989.

[63] S. Mann and S. Haykin. 'Chirplets' and 'Warblets': Novel Time-Frequency Methods. *Electronics Letters*, 28(2):114 – 116, January 1992.

[64] D. Marr. *Vision.* Freeman Press, San Fransisco, C.A., 1982.

[65] D. Marr and E. Hildreth. Theory of Edge Detection. *Proceedings of the Royal Society London*, B207:187–217, 1980.

[66] J. L. Melsa and D. L. Cohn. *Decision and estimation theory.* McGraw-Hill, New York, 1978.

[67] D. Mihovilovic and R.N. Bracewell. Adaptive Chirplet Representation of Signals on Time-Frequency Plane. *Electronics Letters*, 27(13):1159 – 1161, June 1991.

[68] R. Nevatia and K. Ramesh Babu. Linear Feature Extraction and Description. *Computer Graphics and Image Processing*, 13:257 – 269, 1980.

[69] A.V. Oppenheim and J.S. Lim. The Importance of Phase in Signals. *Proc. of the IEEE.*, 69:529 – 541, 1981.

[70] A. Papoulis. *Signal Analysis.* McGraw-Hill, New York, 1977.

[71] A. Papoulis. *Probability, Random Variables and Stochastic Processes (2nd Ed.).* McGraw-Hill, New York, 1984.

[72] P. Parent and S. W. Zucker. Trace Inference, Curvature Consistency, and Curve Detection. *IEEE Trans. PAMI.*, 11(8):823–839, 1989.

[73] Edward R.S. Pearson. *The Multiresolution Fourier Transform and its apllication to Polyphonic Audio Analysis.* PhD thesis, Warwick University, 1991.

[74] P. Perona. Steerable-Scalable Kernals for Edge Detection and Junction Analysis. In *Proc. of ECCV '92, Second European Conference on Computer Vision*, pages 3 – 18, 1992.

[75] L. Peytavin. Multi-Oriented Multi-Resolution Edge Detection. Research Report MS-CIS-90-13, Department of Computer and Information Science, Universit y of Pennsylvania, Philadelphia, May 1990.

[76] M. R. Portnoff. Time-frequency Representation of Digital Signals and Systems Based on Short-Time Fourier Analysis. *IEEE Trans. A.S.S.P*, 28(1):55–69, February 1980.

[77] 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, October 1990.

[78] L. R. Rabiner and B. Gold. *Theroy and Applications of Digital Signal Processing.* Prentice Hall, Englewood-Cliffs, 1975.

[79] O. Rioul and M. Vetterli. Wavelets and Signal Processing. *IEEE Signal Processing Magazine*, pages 14 –38, October 1991.

[80] L. Roberts. Machine Perception of 3-Dimensional Solids. In J. Tippett, editor, *Optical and Electrooptical Information Processing.* MIT Press, Cambridge MA, 1965.

[81] J. G. Robson. Receptive Fields: Neural Representation of the Spatial and Intensive attributes of the Visual Image. In E. C. Carterette and M. P. Friedman, editors, *Handbook of Perception*, volume 5. Academic Press, New York, 1975.

[82] A. Rosenfeld and M. Thurston. Edge and Curve Detection for Visual Scene Analysis. *IEEE Trans. Comp.*, 5(20):562–569, 1971.

[83] M. Shneier. Two Hierarchical Linear Feature Representations: Edge Pyramids and Edge Quadtrees. *Computer Graphics and Image Processing*, 17:211 - 224, 1981.

[84] David Slepian. On bandwidth. *Proceedings of The Institute of Electrical and Electronic Engineers*, 64:292–300, 1976.

[85] M. Spann. *Texture Description and Segmentation in Image Processing.* PhD thesis, Department of Electrical Engineering, The University of Aston in Birmingham, UK, September 1985.

[86] S. L. Tanimoto and T. Pavlidis. A Hierarchical Data Structure for Picture Processing. *Computer Vision, Graphics and Image Processing*, pages 104–119, 1975.

[87] C. W. Therrien. *Decision, Estimation and Classification.* J. Wiley and Sons, 1989.

[88] M. Todd. *Image Data Compression Based on a Multiresolution Signal Model.* PhD thesis, Department of Computer Science, The University of Warwick, UK, November 1989.

[89] R. Watt. *Understanding Vision.* Academic Press Ltd., London, 1991.

[90] 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. IT, Special Issue on Wavelet Representations*, 1992.

[91] R. Wilson and S. C. Clippingdale. A Class of Nonstationary Image Models and their Applications. In J. G. McWhirter, editor, *Mathematics In Signal Processing II.* Clarendon Press, Oxford, 1990.

[92] R. Wilson, S. C. Clippingdale, and A. H. Bhalerao. Robust Estimation of Local Orientations in Images Using a Multiresolution Approach. In *Proc. 5rd SPIE Conf. Vis. Comm. Image Proc.*, Lausanne, 1990.

[93] R. Wilson and H. Knutsson. Uncertainty and Visual Inference. *IEEE Trans. Sys, Man Cyber.*, 18:302–312, 1988.

[94] R. G. Wilson. Finite Prolate Spheroidal Sequences and Their Applications I: Generation and Properties. *IEEE Trans. P.A.M.I*, 9(6):787–795, November 1987.

[95] R. G. Wilson and G. H. Granlund. The Uncertainty Principle in Image Processing. *IEEE Trans. P.A.M.I*, 6(6):758–767, November 1984.

[96] R. G. Wilson and M. Spann. Finite Prolate Spheroidal Sequences and Their Applications 2: Image Feature Description and Segmentation. *IEEE Trans. P.A.M.I*, 10(2):193–203, March 1988.

[97] R. G. Wilson and M. Spann. *Image Segmentation and Uncertainty.* Pattern Recognition and Image Processing Series. Research Studies Press Ltd, 1988.

[98] A. P. Witkin. Scale Space Filtering. In *Proc. of the 8th Int. Joint Conf. on Artificial Intelligence*, pages 1019–1022, 1983.

[99] A. P. Witkin, D. Terzopoulos, and M. Kass. Signal Matching Through Scale Space. In *Proc. of the American Assoc. Artificial Intelligence*, pages 714–719, 1986.

[100] S.Y.K. Yuen, T.S.L. Lam, and N.K.D. Leung. Connective Hough Transform. *Image and Vision Computing*, 11(5), June 1993.

[101] S. W. Zucker, C. David, A. Dobbins, and L. Iverson. The Organization of Curve Detection: Coarse tangent fields and fine spline coverings. In *Proc. of Int. Conf. Computer Vision*, pages 568–577, 1988.