Second Order Derivative Shapes: Calculation, Visualisation, and Application

Michael Barrett-Lennard

*This report is submitted as partial fulfilment
of the requirements for the Honours Programme of the
Department of Computer Science
The University of Western Australia
1998*

Abstract

Feature detection is important in computer vision. Many methods have been proposed that attempt to detect features in images automatically. The method proposed by Per-Erik Danielsson is investigated in this thesis. It uses second derivative information to extract local shape and orientation from a grey-scale image. The algorithm returns a parameter that classifies the types of shape including blobs, ridges, lines, and saddles present in the image. One of the implementation issues addressed is the algorithm's many-to-one shape mapping, where different image features are mapped to the same shape parameter. The thesis explains how to determine correct results from the shape parameter. Another issue that is addressed is the choice of scale of the filters used. The proposed solution applies several filters at different sizes, then combines the responses using the "best" scale for each pixel. The thesis also shows how to visualise the information returned by the algorithm. This is done by colouring an image according to the shape parameter. Canonical shapes, blobs, ridges, lines, and saddles, being represented by red, green and blue. Finally, the method is applied to mammograph images to highlight important features of each image.

Keywords: Rotation Invariance, Second Derivatives, Segmentation, Orientation, Orthogonality, Feature Detection.
CR Categories: I.4.6, I.5.4

Acknowledgements

Most of all I would like to thank my supervisors Dr Peter Kovesi and Dr Chris Pudney for their help and support throughout the year. Without their suggestions the writing of this thesis would not have been possible. Thanks also go to Paul Taylor for supplying mammograph images, to Dr Des Hill for his mathematical advice, and my friends Phil Tate, Paul Wagland, Peter McFawn, and my brother David, who generously offered their free time to proof-read this thesis. Finally I would like to thank my fellow students who made the honours room a fun place to be.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Feature detectors are important in computer vision. The aim is to enable a computer to recognize objects in images by the application of a series of algorithms. I have extended an algorithm by Per-Erik Danielsson [2, 3, 4] that uses derivative information for feature detection. His approach returns a shape parameter that identifies symmetric shapes in an image. This shape information can be used to segment and process images.

The use of derivative operators to find features is not new. However, this algorithm is different in that it uses three orthonormal second-order filters rather than two first-order filters or a symmetric second-order filter. Other methods determine a derivative magnitude of some order, and a derivative orientation to locate a feature (normally a step edge); this includes Prewitt [11], Sobel [10], and Canny [1] edge detection schemes. Danielsson's algorithm not only calculates the second derivative magnitude but also returns the shape type and orientation of the shape. It accomplishes this by using second-order convolution kernels to extract information about features in an image. However, Danielsson's algorithm has some problems that need to be addressed before it can be applied.

There is a problem with the shape parameter returned by the algorithm. Danielsson's method uses only second derivative information, the result of which is that only a subset of symmetric shapes (a range of shapes possessing bilateral or rotational symmetry) can be detected. Any shape in the image that is not of this type will return a misleading shape parameter. This is the problem of false positives which affects many feature detectors. The solution proposed is to determine a confidence factor for the shape.

It is possible that Danielsson's method may be improved by incorporating first derivative information. First derivative methods detect only odd-symmetric intensity changes, while second derivative methods detect even-symmetric intensity changes. This means that first and second derivative methods can be considered complementary in nature. Applying both first and second derivative operators should allow most feature information to be extracted. With the introduction of first derivative information it might be possible to detect non-symmetric shapes in addition to symmetric shapes. The advantages of using both first and second order information can be seen in the local energy feature detector [9] by Morrone and Owens. It uses a pair of filters, one of which is for even-symmetric changes, and the other for odd-symmetric changes. Its advantage is that it can detect edge features without the problem of false positives that affects other edge detection schemes.

Another problem is that the types of shapes detected depend on the size of the filters used. It is possible to find the appropriate size of the filters through trial-and-error. However, an automated solution is preferable. The problem of scale plagues many areas in computer vision. A solution may come from the efforts of Koenderink [7, 6]. It is hoped the unique nature of Danielsson's algorithm may allow for an easy way to determine scales to use when processing an image.

One problem is finding an effective way of visualising the information calculated by this algorithm. Using colour to represent shape types is one way that the data could be represented. Colouring in this way may enhance important features in an image. One possible application for visualising this way is the enhancement of cancer features in mammographs. The method can enhance cancer features by using contrasting colours, allowing the radiologist to identify such features with greater ease. Another possibility is the removal of veins and arteries from the image to make the cancers easier to detect.

Chapter 2 introduces some of the basic methods involved in feature detection. This chapter also includes an overview of Danielsson's papers and a preliminary investigation into Koenderink's work in scale space. Chapter 3 gives a detailed introduction to Danielsson's method and describes the filters used. Chapters 4–6 address the problems of Danielsson's algorithm and propose solutions. Chapter 7 gives an application to mammographs of the methods shown in this thesis and Chapter 8 gives a conclusion and looks at how the methods may be modified to work on volumetric images.

CHAPTER 2

Literature review

Since the method used in this thesis is a difference method, this review discusses difference methods already in use. Next the issue of the scale of filters is discussed. This refers mainly to the work by Koenderink [7]. Then the work of Per-Erik Danielsson [2, 3, 4] is described. This shows how the ideas of Danielsson have developed into an implementable algorithm applicable to two- and three-dimensional images. In conclusion, Danielsson's method is shown to suffer from many of the problems that affect other difference methods.

2.1 Difference methods

A common approach to detecting features in images is the use of difference methods. Difference methods find features by calculating the derivative values of an image and then analysing them. They normally use a mask that is overlayed on the image at all points and calculates a weighted sum (a process called digital convolution). The simplest of these is Roberts' cross operator [12], which applies the following 2×2 masks.

λ_1

0	1
-1	0

λ_2

1	0
0	-1

Applying the masks gives two derivative magnitudes λ_1, λ_2 in the two diagonal directions. Given λ_1 and λ_2, the first derivative gradient and orientation are calculated as follows.

$$\text{magnitude} = \sqrt{\lambda_1^2 + \lambda_2^2}$$

$$\text{orientation} = \arctan\left(\lambda_2/\lambda_1\right)$$

Roberts' cross operator has the problem that it is of limited spatial extent, resulting in inaccurate λ values. More accurate but more computationally expensive filters for extracting the two λ values are as follows:

3×3 Prewitt operator [11]

λ_1

-1	0	1
-1	0	1
-1	0	1

λ_2

1	1	1
0	0	0
-1	-1	-1

4×4 Prewitt operator

λ_1

-3	-1	1	3
-3	-1	1	3
-3	-1	1	3
-3	-1	1	3

λ_2

3	3	3	3
1	1	1	1
-1	-1	-1	-1
-3	-3	-3	-3

Sobel operator [10]

λ_1

-1	0	1
-2	0	2
-1	0	1

λ_2

1	2	1
0	0	0
-1	-2	-1

High values of the first derivative gradient correspond to step edge features in the image. A step edge is a steep change in the intensity in the image over position. This is an example of an odd-symmetric change. Odd-symmetric changes occur where intensities on one side of a position in the image correspond to equally great but opposite intensities on the other side. Even-symmetric changes occur where intensities in opposing positions are equal. Both odd- and even-symmetric features have an orientation except for the special case of even symmetric features that are circularly symmetric. First derivative methods can detect only odd-symmetric changes; to detect even-symmetric changes a second-order method is required. Examples of odd- and even-symmetric features are shown in Figure 2.1.

One problem with difference methods is that they accentuate image noise. To overcome this problem an averaging mask is often applied to the image to suppress the effect. Unlike the difference filters, whose weights sum to zero, an averaging mask's weights sum to one. The reason being so that applying an averaging mask has no effect on the total brightness of the image. Averaging masks are often produced by sampling values of the two-dimensional Gaussian function. One well known method that uses Gaussians to suppress image noise is the Marr-Hildreth operator [8]. This second order method applies a discretised

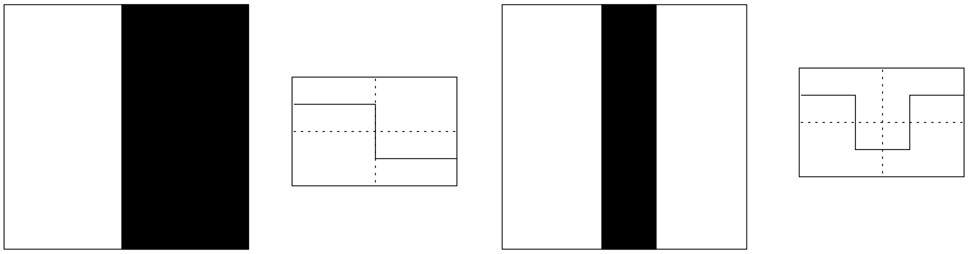


Figure 2.1: Examples of odd- and even-symmetric features. Odd-symmetric feature on the left. Even-symmetric feature on the right.

version of the Laplacian of the Gaussian to the image. Let $G(x, y)$ be the two dimensional Gaussian:

$$G(x, y) = \frac{1}{2\pi} \exp\left(\frac{-x^2 - y^2}{2}\right).$$

The Laplacian of the Gaussian is:

$$\nabla^2(G) = \frac{\partial^2}{\partial x^2}G + \frac{\partial^2}{\partial y^2}G.$$

This function is then sampled to give a discretised version that is applied to the image. At step edges the operator produces a zero crossing. The Marr-Hildreth operator has the shape of an inverted Mexican hat. A main justification of the use of the Marr-Hildreth operator is that this shape is similar to receptors in the retina of the eye. Another filter that uses the Gaussian is the Canny operator [1]. The main difference between the two methods is that the Canny operator uses the direction of the gradient. The first-order directional derivative of the Gaussian is applied to the image and maximal derivative values correspond to edges.

All the methods mentioned above produce false positives, that is, they detect edges where none exist. This occurs where there is sufficiently steep smooth shading. Second derivative detectors produce false positives at points of inflection of the intensity function.

The methods shown here also suffer from the problem of scale. Roberts' cross, Prewitt, and Sobel operators are small masks of fixed size. They fail to detect edges that cover a large area. This can occur where there is blurring. For both the Marr-Hildreth and Canny operators, the size of the masks used affects the edges they detect.

2.2 Filter scale

Feature detectors often require a scale factor to be chosen. For many of these methods there is no way to automate the selection of scale. In many situations a data

noise. The result is an ideal operator that finds the exact gradient of the luminance function and removes any high frequency components (such as noise). In the two-dimensional case a pillar box is used instead of a rectangular low pass filter, resulting in two operators g_x and g_y, which are are rotation invariant.

The first-order two-dimensional operators g_x and g_y can be used to generate three second-order operators g_{xx}, g_{yy}, and g_{xy}. These are not rotationally invariant, but by combining the filters into simple linear combinations they can be made into rotationally invariant operators called f_{20}, f_{21}, and f_{22}. Operators f_{20}, f_{21}, and f_{22} are ideal second-order operators; one of these operators is the Laplacian.

2.3.2 The use of ideal operators

The ideal second order operators described in Section 2.3.1 are used in Danielsson's [2, 3] method to detect shape. When converted to the spatial domain the pillar box filter has infinite extent, which is too computationally expensive to implement. Instead a Gaussian function is used to remove high frequency components. The three operators are applied to the image and the three resultant numbers for each pixel are treated as a vector in three dimensions. Next this vector's orientation angles are extracted using a process called derotation. Derotation rotates the vector onto the coordinate planes to extract the angles. The extracted angles represent a shape type, the orientation of that shape, and the magnitude is the second derivative magnitude.

The method for the second-order case in three dimensions is now examined. This is required to apply the algorithm to volumetric data. An application for this could be in medical imagery, where it would be possible to segment out organs and tissue in volumetric data. A 6×6 matrix is multiplied by the six calculated second-order derivatives to make them orthonormal. The first operator response is the Laplacian and is independent of the orientation of the shape, the other five operators contain orientation and shape information. Together they form a six dimensional vector. This vector is derotated to give the three Euler angles, two angles representing shape type, and the magnitude response.

2.4 Problems addressed in this thesis

The problems that occur in other difference methods carry over to Danielsson's method of shape detection.

- Difference methods tend to accentuate noise in the image and require the use of a Gaussian to smooth the image. This is true of Danielsson's method and so a Gaussian is applied to the operators.

- Difference methods have the problem of false positives. Danielsson's algorithm also has this problem as it responds to shapes that are not bilaterally symmetric.

- The Marr-Hildreth and Canny methods have the problem of choosing the appropriate scale. In Danielsson's method this is even more of a problem; the scale chosen determines the sizes of the shapes detected.
- Most difference methods have the deficiency that they only use derivative information of a single order. In Danielsson's method this is also true; only the second-order derivative information is used and first-order information is ignored.

The main aims of this thesis are the following:

- Solve the problem of the algorithm returning a shape parameter for shapes it cannot detect. The solution to this problem will be covered in Chapter 4.
- Find a way to incorporate first derivative information. This problem will be partially addressed in Chapter 4.
- Visualise the information returned by the algorithm. A solution is described in Chapter 5.
- Find a way of selecting the scale to use. A solution is described in Chapter 6.
- Apply the technique to mammographs. The results are displayed in Chapter 7.

CHAPTER 3

Second-order derivatives of two-dimensional images

This chapter provides details of Danielsson's algorithm for two-dimensional images. The description begins with a theoretical discussion treating the intensity function as continuous. Next, implementation issues are discussed for a discretised image. A Gaussian is used as a regularising function. Finally the resultant filters are examined to show how they work.

3.1 Theory

The aim of this discussion shows how the second derivative operators g_{xx}, g_{yy}, and g_{xy} can be transformed to an orthonormal basis that is rotationally invariant and can be used to extract shape, orientation, and magnitude information. Let $f(x, y)$ be a continuous two-dimensional scalar field with (x, y) being a point in the image. To determine the derivatives an operator is applied to the intensity function $f(x, y)$. For this algorithm the three second derivatives f_{xx}, f_{yy} and f_{xy} are required.

Assume that there exists a regularising function $h(r)$ ($r^2 = x^2 + y^2$) that when applied to the image reduces noise by taking a weighted average of all values local to a point. The usual function used for this is a two-dimensional Gaussian. Whatever function is used it should be circularly symmetric. The operators used to extract second-derivative information are described as:

$$\begin{aligned} g_{xx} &= \frac{\partial^2}{\partial x^2} h(r), \\ g_{yy} &= \frac{\partial^2}{\partial y^2} h(r), \text{and} \\ g_{xy} &= \frac{\partial^2}{\partial x \partial y} h(r). \end{aligned}$$

A two-dimensional Fourier transform is applied to the above equations with the

Fourier transform of $h(r)$ being $H(\varrho)$:

$$\begin{aligned} G_{uu} &= -4\pi^2 u^2 H(\varrho), \\ G_{vv} &= -4\pi^2 v^2 H(\varrho), \text{and} \\ G_{uv} &= -4\pi^2 uv H(\varrho). \end{aligned}$$

Then convert to polar coordinates $u = \varrho \cos\varphi, v = \varrho \sin\varphi$.

$$\begin{aligned} G_{uu} &= -4\pi^2 \varrho^2 H(\varrho) \cos^2\varphi, \\ G_{vv} &= -4\pi^2 \varrho^2 H(\varrho) \sin^2\varphi, \text{and} \\ G_{uv} &= -4\pi^2 \varrho^2 H(\varrho) \cos\varphi \sin\varphi. \end{aligned}$$

For the following, let it be assumed that the inner product is a double integral over ϱ and φ. The operators G_{uu}, G_{vv}, and G_{uv} are not orthonormal, as shown in Appendix B.2. To generate a set of orthonormal basis functions, specific linear combinations of the above operators are used. This operation is indicated below:

$$\begin{bmatrix} B_{20} \\ B_{21} \\ B_{22} \end{bmatrix} = \sqrt{\frac{1}{3}} \begin{bmatrix} 1 & 1 & 0 \\ \sqrt{2} & -\sqrt{2} & 0 \\ 0 & 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} G_{uu} \\ G_{vv} \\ 2G_{uv} \end{bmatrix} \tag{3.1}$$

Expanding equation(3.1) the following basis functions are obtained.

$$\begin{aligned} B_{20} &= -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{1}{3}} (\cos^2\varphi + \sin^2\varphi) = -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{1}{3}}, & (3.2) \\ B_{21} &= -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{2}{3}} (\cos^2\varphi - \sin^2\varphi) = -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{2}{3}} \cos 2\varphi, & (3.3) \\ B_{22} &= -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{2}{3}} (2\cos\varphi \sin\varphi) = -4\pi^2 \varrho^2 H(\varrho) \sqrt{\frac{2}{3}} \sin 2\varphi. & (3.4) \end{aligned}$$

The functions B_{20}, B_{21}, B_{22} are orthonormal (this is shown in Appendices B.3, B.4). As the inverse Fourier transform is linear, the operators are also orthonormal in the spatial domain. Let $\mathbf{f_2} = (f_{20}, f_{21}, f_{22})$ be the values returned by the orthonormal operators in the spatial domain. It is the properties of the vector $\mathbf{f_2}$ that determine shape, shape-orientation and second-derivative magnitude using the derotation scheme mentioned in section 2.3.2. The two orientation angles of $\mathbf{f_2}$ represent the shape orientation and shape type. Rotating $\mathbf{f_2}$ onto a coordinate plane in 3D produces the shape orientation φ. This also gives a vector $\hat{\mathbf{f}}_\mathbf{2}$ with two non-zero values. Rotating $\hat{\mathbf{f}}_\mathbf{2}$ onto another coordinate plane gives the general shape type λ. This leaves a single non-zero value a, the magnitude of $\mathbf{f_2}$; this is the second derivative magnitude. Figure 3.1 shows how λ and φ relate to the operators' responses.

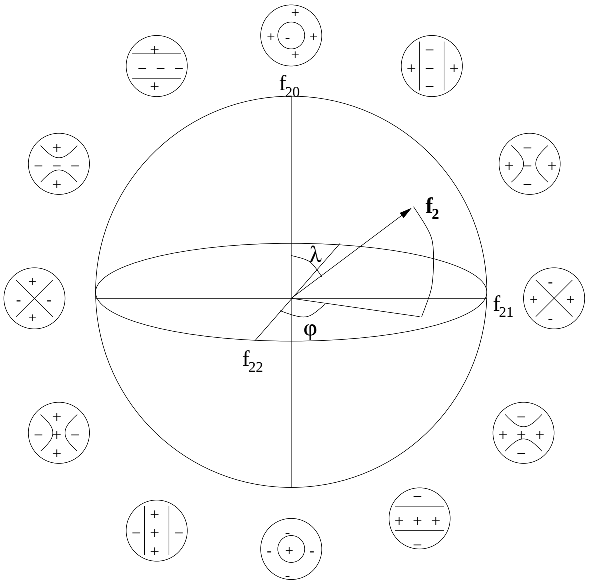


Figure 3.1: Diagram indicating how the vector $\mathbf{f_2}$ relates to shape and orientation. The shapes shown around the outside of the sphere show how shape changes with changing λ and constant φ.

3.2 Implementation

To implement Danielsson's algorithm, filters need to be generated that can be used to extract second derivative values. Firstly, the second derivatives of the Gaussian function are calculated. Let $g(x, y)$ be the two-dimensional Gaussian with a standard deviation $\sqrt{\sigma}$ value of one, the second derivative operators g_{xx}, g_{yy}, and g_{xy} are calculated as follows.

$$\begin{aligned} g_{xx} &= \frac{x^2-1}{2\pi}\exp\left(\frac{-x^2-y^2}{2}\right), && (3.5) \\ g_{yy} &= \frac{y^2-1}{2\pi}\exp\left(\frac{-x^2-y^2}{2}\right), \text{ and} && (3.6) \\ g_{xy} &= \frac{xy-1}{2\pi}\exp\left(\frac{-x^2-y^2}{2}\right). && (3.7) \end{aligned}$$

The three filter masks are generated by sampling these functions in a local area around the origin. The results in this thesis are produced by sampling to a radius of three standard deviations as a local area. The sampling density determines the size of the filter, which in turn determines the size of the shapes detected. If the filter values do not sum to zero (because of discretisation) then all the values of the filters are offset so that they do.

The filters are convolved with the image to produce the second derivatives of the image f_{xx}, f_{yy}, and f_{xy}. These are orthonormalised via the matrix in equation(3.1) to give $\mathbf{f_2}$.

$$\mathbf{f_2} = \begin{bmatrix} f_{20} \\ f_{21} \\ f_{22} \end{bmatrix} = \sqrt{\frac{1}{3}} \begin{bmatrix} 1 & 1 & 0 \\ \sqrt{2} & -\sqrt{2} & 0 \\ 0 & 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} f_{xx} \\ f_{yy} \\ 2f_{xy} \end{bmatrix}$$

The shape factor λ is:

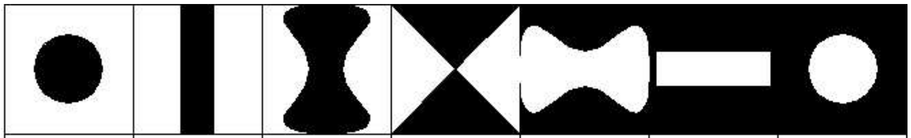
$$\lambda = \arctan\left(\frac{\sqrt{(f_{21}^2 + f_{22}^2)}}{f_{20}}\right), \tag{3.8}$$

the shape orientation φ is:

$$\varphi = -\frac{1}{2}\operatorname{atan}(f_{22}, f_{21}), \tag{3.9}$$

and the second derivative magnitude a is:

$$a = \sqrt{f_{20}^2 + f_{21}^2 + f_{22}^2}. \tag{3.10}$$



	-ve blob	-ve line	-ve ridge	saddle	+ve ridge	+ve line	+ve blob
λ	0	$\arctan \frac{1}{\sqrt{2}}$	$\arctan \sqrt{2}$	$\pi/2$	$\arctan -\sqrt{2}$	$\arctan -\frac{1}{\sqrt{2}}$	π

Figure 3.2: Images of the different canonical shapes.

Figure 3.2 shows how the values of λ correspond to different shape type. The λ variable is continuous so there is a continuous change in shape as λ changes from 0 to π. In Figure 3.2, starting with a circular negative blob ($\lambda = 0$), as λ increases the blob elongates, becoming elliptical and starts resembling a negative line. Then the ends of the line start flaring out to form a negative ridge. As λ approaches $\pi/2$ the negative ridge becomes a saddle. Increases to λ after this produce positive shape types, with orientation changed by $\pi/2$.

3.3 Properties of the filters

Filter f_{20} (see Figure 3.3) is a scaled Laplacian. It is circularly symmetric and so cannot detect the orientation of the shape in question; rather it detects the degree of "blobbyness". The other two filters f_{21} and f_{22} shown in Figures 3.4 and 3.5 have a saddle shape, and are offset at 45 degrees to each other. These two filters return the orientation of the current shape; they also give the degree of "saddleness".

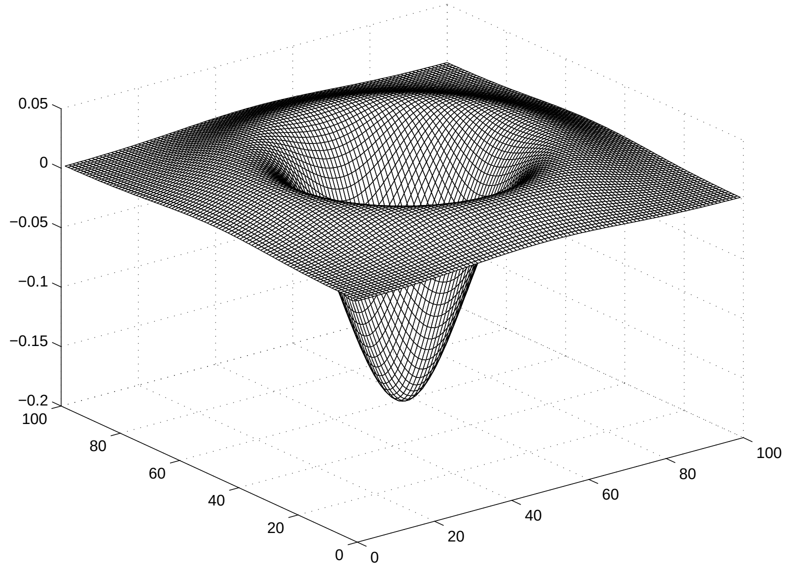


Figure 3.3: Surface plot of filter f_{20}.

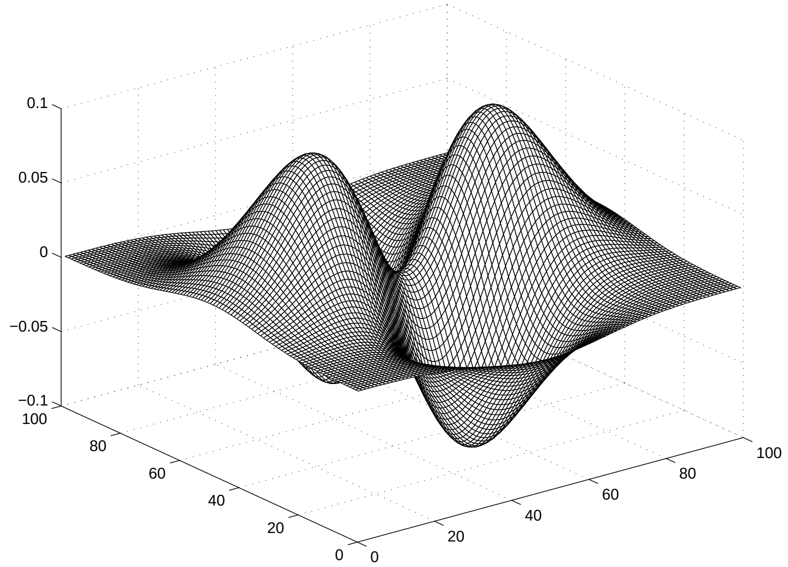


Figure 3.4: Surface plot of filter f_{21}.

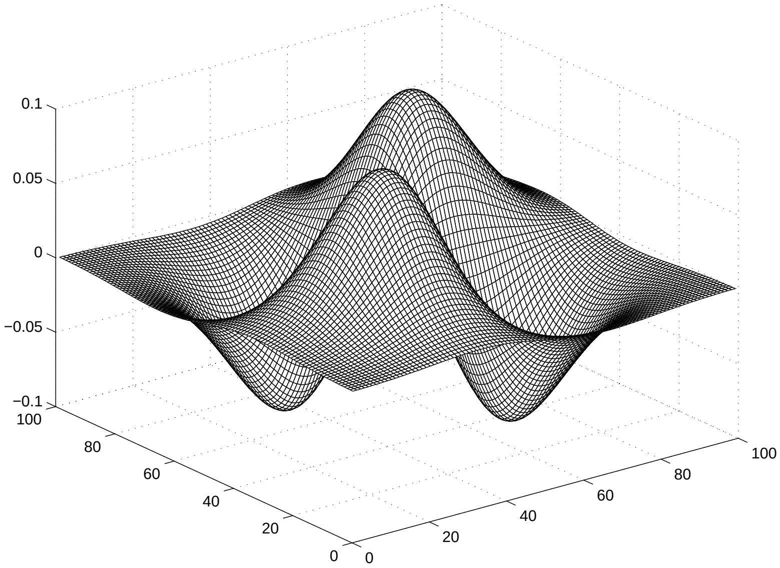


Figure 3.5: Surface plot of filter f_{22}.

CHAPTER 4

Suppressing false positives

Danielsson's algorithm can produce false positives; it returns a shape parameter for shapes it cannot detect (shapes not in the range of negative blob to saddle to positive blob). For example at points where a step edge exists the algorithm will detect this as a line feature. To solve this problem a confidence factor is calculated for each pixel in the image. The confidence factor is the match between the shape detected and the actual shape in the image. Two ways of generating a confidence factor were tried: the first worked poorly, but the second gave impressive results.

The first attempt used the second derivative magnitude as a confidence factor. This value was believed to confirm shape because the second derivative magnitude is a good detector of even symmetry. The shapes detected by Danielssons's algorithm are all even-symmetric because second derivative filters have been used. So a strong second derivative magnitude should suggest being near one of the canonical shapes in Table 3.2. Also in areas without features, that is in blank areas, the algorithm will return a shape value. In such areas the second derivative magnitude tends to zero, indicating no shape in that area. Unfortunately, it was found that there were many features that had high second derivative magnitude, but had no resemblance to the canonical shapes. Danielsson's algorithm cannot detect all even-symmetric shapes, only a subset of them.

The second approach tried was to generate a spatial correlation value as a confidence factor between a local area in the image and a reconstructed image of the shape indicated by the algorithm. The reconstruction is generated using the values returned by the basis filters f_{20}, f_{21}, and f_{22}. When these values are multiplied by the basis filters and summed a reconstructed image is produced approximating the feature indicated by the algorithm. Figure 4.2 shows an example of the reconstructed images of all the local areas of the saddle image shown in Figure 4.1. Note that the algorithm gives a false indication of shape where there is a large first derivative, that is, along step edges.

Once the local image is reconstructed it is compared with the original local image. This is done using the following normalised correlation formula by Faugeras [5]:

$$\tau = \frac{1}{K} \sum_{u=0}^{h} \sum_{v=0}^{w} [I_1(u,v) - \overline{I_1}][I_2(u,v) - \overline{I_2}],$$

where

τ is the correlation value,

h is the height the correlation window,

w is the width the correlation window,

I_1,I_2 are the original and reconstructed window areas of the images respectively,

$\overline{I_1}, \overline{I_2}$ are the means within the window areas of the original and reconstructed images respectively,

K is the normalization factor $K = hw\sigma_1\sigma_2$, where σ_1 and σ_2 are the standard deviation of I_1 and I_2 respectively.

This formula gives high correlation only in localised areas of the canonical shapes. The results are shown in Chapter 5.

4.1 Incorporating the first derivatives

Danielsson's algorithm can detect only a subset of symmetric shapes, which causes the problem of false positives. If first derivative information could be incorporated into the process then a greater variety of shapes would be detectable. Including the first derivative operators f_x and f_y would allow for two new shape parameters to be derived. Where currently there is a single λ value controlling shape there now would be three values. This would give a three-dimensional shape space that would need to be mapped to colour.

If third-, fourth- or even higher-order derivative information could be included then even more shape parameters would be generated. Each added parameter would add to the dimensions of the shape space and may result in a greater range of detectable shapes. Just as the Taylor series approximates continuous functions by using derivative information of different orders, in the same way the image intensity function could be approximated. If all orders of derivatives could be included then all possible shapes could be classified and detected.

The problem then is finding a way of including the other derivative orders of information. We start by considering adding first-order information. The following is the derivation of the first-order operators:
Using $h(r)$ as the Gaussian regularising function, let

$$\begin{aligned} g_x &= \frac{\partial}{\partial x} h(r), \text{and} \\ g_y &= \frac{\partial}{\partial y} h(r). \end{aligned}$$

Applying the Fourier transform, gives

$$\begin{aligned} G_u &= 2\pi i u H(\rho), \text{and} \\ G_v &= 2\pi i v H(\rho). \end{aligned}$$

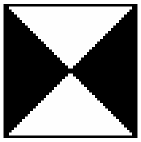


Figure 4.1: Original 21x21 image of a saddle.

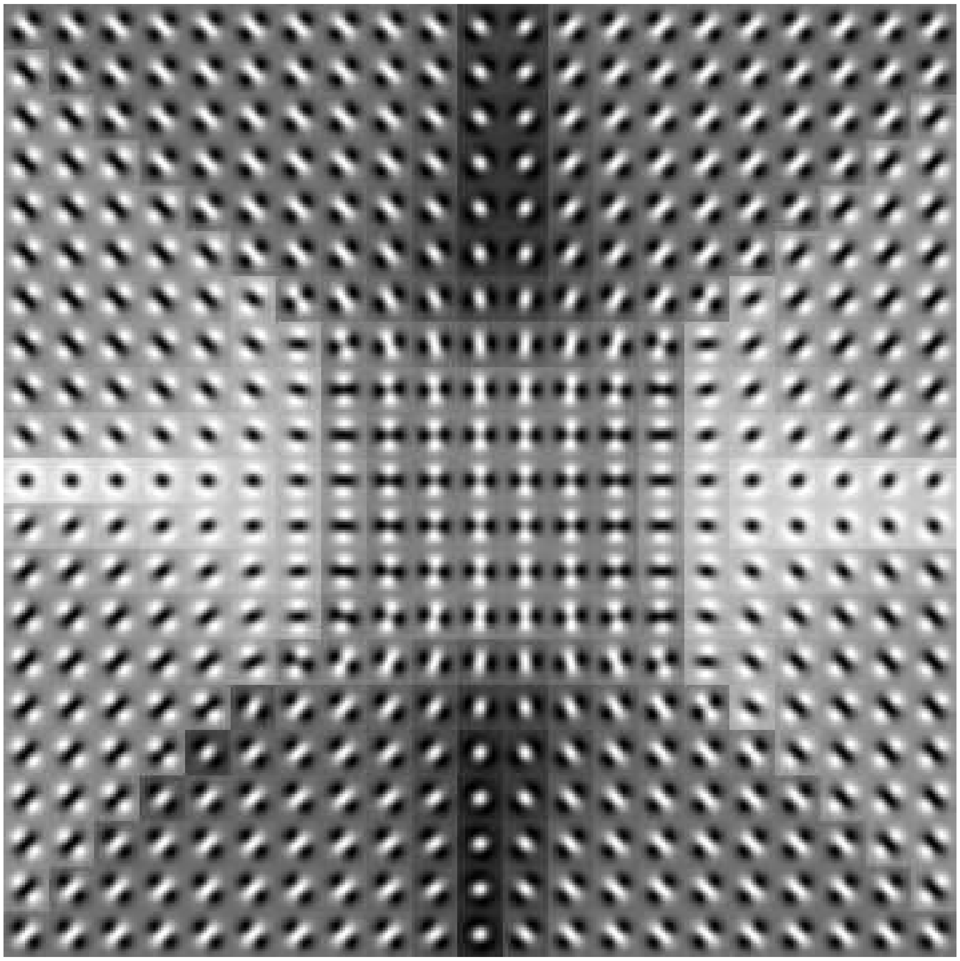


Figure 4.2: Reconstructed 17x17 images from summed filter responses to a saddle shape (Image size 357x357).

Converting to polar coordinates $u = \rho \cos(\varphi)$ and $v = \rho \sin(\varphi)$, gives

$$\begin{aligned} B_{10} &= 2\pi i \rho \cos(\varphi) H(\rho), \text{and} \\ B_{11} &= 2\pi i \rho \sin(\varphi) H(\rho). \end{aligned}$$

In Appendices B.5 and B.6 it is shown that the two first-order operators and three second-order operators are orthonormal. The results of these operators when applied to the image can be treated as a five-dimensional vector. If a scheme to derotate this vector could be developed a meaning could be assigned to the rotation angles. This step is not obvious. Unlike the method in this thesis which has just three responses, the derotation scheme for five responses has a degree-of-freedom. The degree-of-freedom allows for five possible solutions after derotation. In Danielsson's report [3] this situation is handled for the algorithm applied to three-dimensional images. In the 3D case there are six responses allowing for six possible solutions. The response vector is rotated prior to derotation depending on the position the vector starts in. Unfortunately I can find no way of implementing this for the five-dimensional situation.

Even if a derotation scheme were to be found it is not known what some of the angles generated from this process would mean. The angles between our original second-order axes would mean the same as before, but how would this information relate to the two new angles that are introduced by having two first-order axes? The main problem with these angles is that there is no way to get an intuitive feel for what they represent.

CHAPTER 5

Visualisation of shape

Danielsson's algorithm described in the previous chapter returns a shape parameter λ for each pixel in the image. How can λ be used for feature detection? I propose to use the shape parameter to colour images according to shape type to enhance important features in the image.

One obvious way of colouring the image would be to assign a colour to each of the base shapes in Table 3.2. A pixel would be coloured based on the shape its λ value is closest to. Unfortunately this gives misleading results, as pixels can be assigned a shape colour when really the shape is somewhere between two classifications. A continuous allocation of colour is required to represent the range of shapes that are detectable. The solution is to use a hue, saturation, and value (HSV) colour scheme, and map the λ value to hue.

5.1 Assigning hue, saturation, and value

A simple way to assign hue is to map λ linearly to the entire hue scale. Unfortunately this gives poor results because the hue scale wraps around to the same colour. This means that two of the canonical shape types are given the same colour resulting in poor contrast. A better solution is to assign the primary colours red, green, and blue to three of the shapes. Then intermediate hues can be chosen to cover other shape types.

The canonical shapes in Figure 3.2 all have "perfect mathematical" representations as quadratic surfaces as shown in Table 5.1. Figure 5.1 illustrates the shape reconstructions from the filters f_{20}, f_{21}, f_{22}. Examining the line and ridge shapes in this diagram shows that the human eye does not recognise these shapes as being "perfect mathematical" representations. However the blob and saddle shapes are easily distinguishable from other shapes in the range because of their symmetry. Therefore, I use the primary colours blue, green, and red to represent positive blob, saddle, and negative blob respectively. Intermediate hues are inserted between blue and green, and red and green colours to give a continuous change in colour. The line and ridge features are not assigned any particular hue value.

Shape type	Name	Quadratic Representation
-ve blob	elliptic paraboloid	$z = \frac{x^2}{k^2} + \frac{y^2}{k^2}; k > 0$
-ve line	elliptic paraboloid	$\lim_{m \to 0} z = \frac{x^2}{l^2} + \frac{y^2}{m^2}; l > 0, m > 0$
-ve ridge	-ve hyperbolic paraboloid	$z = -\frac{y^2}{k^2} + \frac{x^2}{k^2}; k > 0$
saddle	hyperbolic paraboloid	$\lim_{l,m \to 0} z = \frac{y^2}{m^2} - \frac{x^2}{l^2}; l > 0, m > 0$
-ve ridge	hyperbolic paraboloid	$z = \frac{y^2}{k^2} - \frac{x^2}{k^2}; k > 0$
-ve line	-ve elliptic paraboloid	$\lim_{m \to 0} z = -\frac{x^2}{l^2} - \frac{y^2}{m^2}; l > 0, m > 0$
-ve blob	-ve elliptic paraboloid	$z = -\frac{x^2}{k^2} - \frac{y^2}{k^2}; k > 0$

Table 5.1: Mathematical representations of canonical shapes

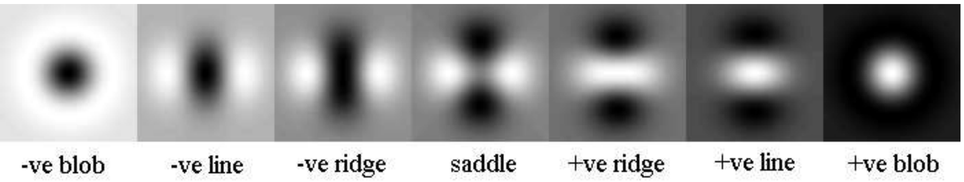


Figure 5.1: Reconstructed canonical shapes from summed filters

The saturation of each pixel is set according to the confidence factor such that shapes that exist in the original image are brightly coloured, and pixels with lower confidence factors have washed out colours, typically in areas where the first derivative is large. The "value" of the HSV colourmap is set to one for all pixels. The colour map shown in figure 5.2 shows how this is done.

5.2 Results

The method was tested on the image shown in Figure 5.3. The results are shown for an 11x11 size filter in Figure 5.4 along with a diagram of the colour map used in Figure 5.2. The resulting colour image shows bright colours where canonical shapes exist. The dark ridges in the face of the baboon, correspond to yellow lines in the coloured image. Yellow represents the mid-point in the transition from negative blob to saddle, that is, line-/ridge-like features. The specularity in the centre of the eyes corresponds to a bright blue dot in the coloured image, indicating a positive blob. All the shapes detected in the coloured image are small. If we were to use a larger filter then larger features would be detected.

Unfortunately, the proposed algorithm is not efficient. Let n be the number of pixels along one side of a square image and f be the number of pixels along one side of the square filter.

For each pixel:

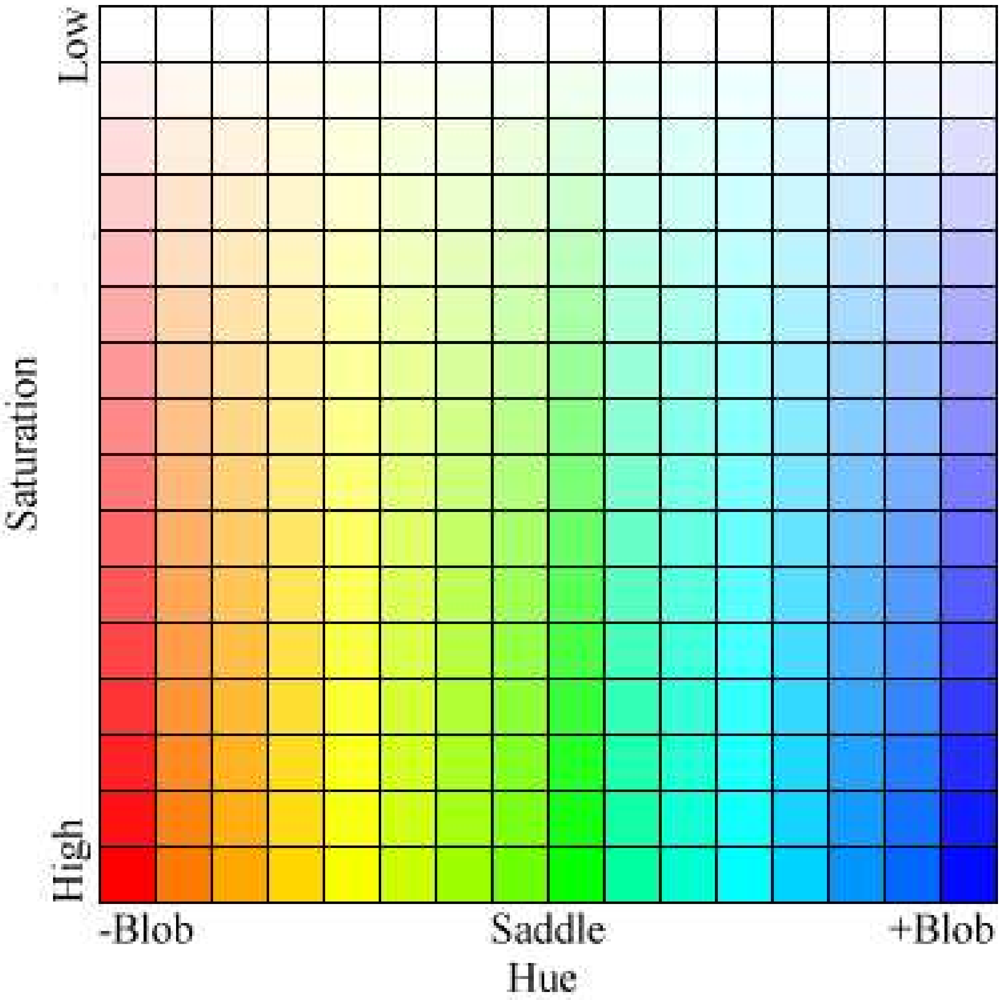


Figure 5.2: The colour map used showing saturation and hue.

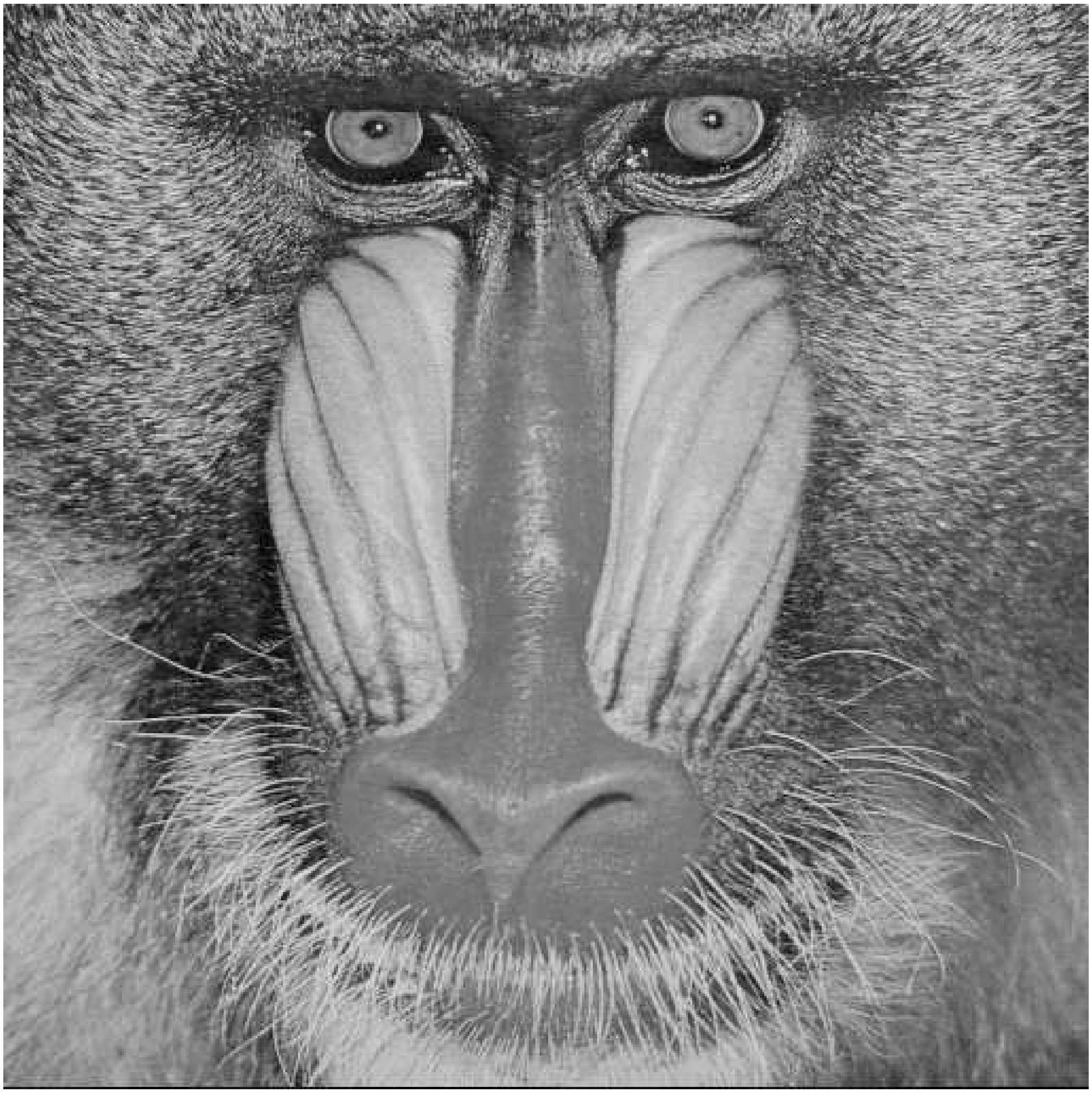


Figure 5.3: The original baboon image used to test the algorithm (Image size 512x512).

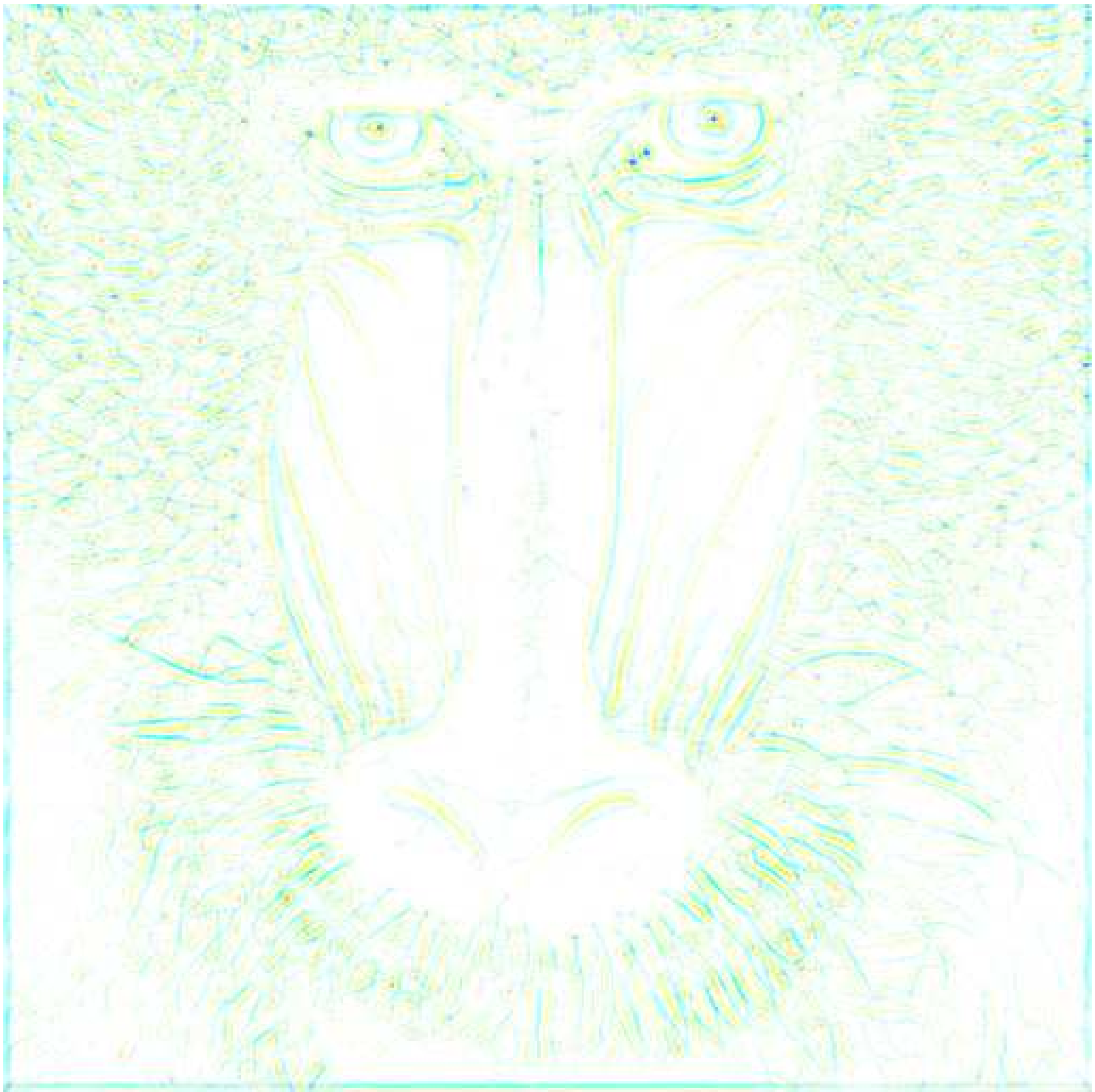


Figure 5.4: The resulting baboon image after applying 11x11 filters and colouring using the HSV scheme.

calculate shape parameter	$3f^2$
create filter image	$3f^2$
calculate std dev	f^2
calculate correlation	f^2
Total	$8f^2$

For n^2 pixels in the image:
Total complexity $O(n^2 f^2)$.

If the filter size is large we would expect slow performance from the algorithm.

CHAPTER 6

Scale

An unresolved issue in the study of computer vision is the problem of scale, that is, choosing the correct size filter to apply to an image. Often the size of the filter determines what features will be detected. This is easier to resolve when the size of features to detect is known. An algorithm can have parameters set to make it a good detector of certain sized features. This approach is easily applied to edge detectors since sharp edges are by definition very localized. However, without initial information about the size of the feature there is no way of determining what size filters to use. What is normally done is to apply many filters of different sizes and embed them in a structure called a *scale space pyramid*.

6.1 Scale space animation

Before describing the solution it is instructive to examine a series of images produced by the HSV colouring scheme at different scales. This series of images can be animated showing the image changing with increasing scale. The animation shows how shapes can be embedded inside other shapes; as the scale increases the interpretation of what is seen completely changes. The frames of a animation generated from Figure 6.1 are shown in Figure 6.2. They show how the detection of new features reduces as scale increases. This agrees with Koenderink's use of an exponentially increasing scale [7].

A possible use for scale space animation could be to detect cancers in X-ray images. The radiologist could play the animation and stop it whenever a bright shape response occurs in an unexpected place. The time in the movie where the shape is brightest could indicate the size of the cancer. If a cancer has a particular shape then the movie could automatically stop whenever such a shape appears above a certain confidence level.

6.2 Combining different scales

To combine results from different scales I propose to assign pixel color based the "best" scale to use. The confidence factor introduced in Section 5.1 can be used

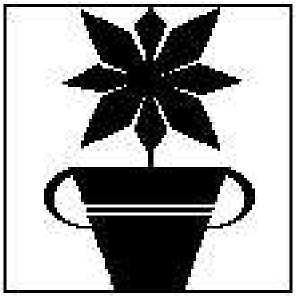


Figure 6.1: Image of flower pot used to produce a scale space animation (Image size 100x100).

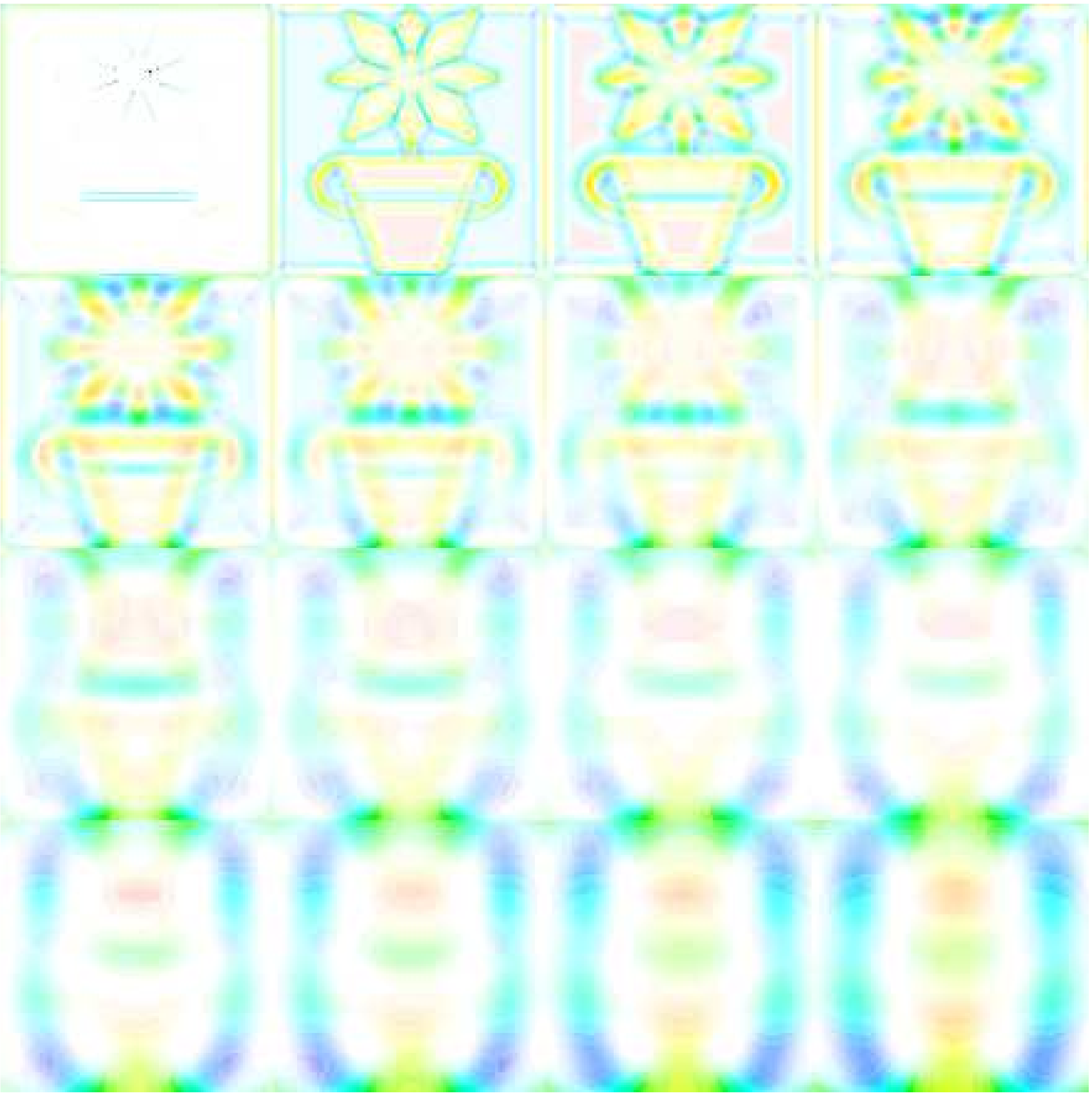


Figure 6.2: Frames of the scale space animation. Filter sizes range from 5 to 101.

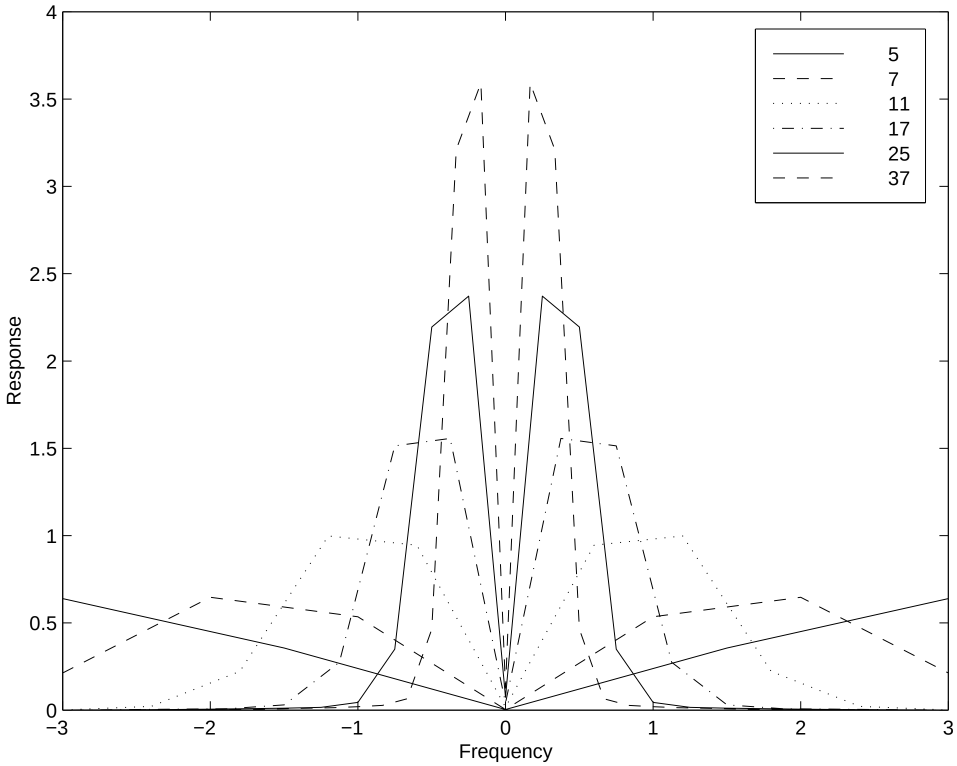


Figure 6.3: Plot of different sized filters in the Fourier domain.

as a way to choose the "best" scale. As different sized filters are applied the shape value with the highest confidence factor is stored ready for a final output image.

A series of different sized filters that can detect features at all scales needs to be determined. Empirically a 5x5 filter will detect features as small as one pixel in size. We need to find an efficient way to increase the size of the filters beyond 5x5. An exponential increase in scale is proposed by Koenderink [7]. In this thesis a factor of 1.5 has been used to increase filter sizes. This generates the series of filter sizes: 5, 7, 11, 17, 25, 37, 57, 85, 128 that is used to generate my resulting images. To show that the series will detect features at most scales the filters have been plotted in the Fourier domain. Figure 6.3 shows that the series covers the Fourier spectrum almost completely.

The result of applying this process to the baboon image is shown in Figure 6.4. When compared to Figure 5.4 there is considerable improvement over using a single scale. In Figure 5.4 the resulting image does not show much of the detail of the original image. In Figure 6.4 the image is much fuller, and shapes of a greater diversity are brightly coloured indicating the images structure.

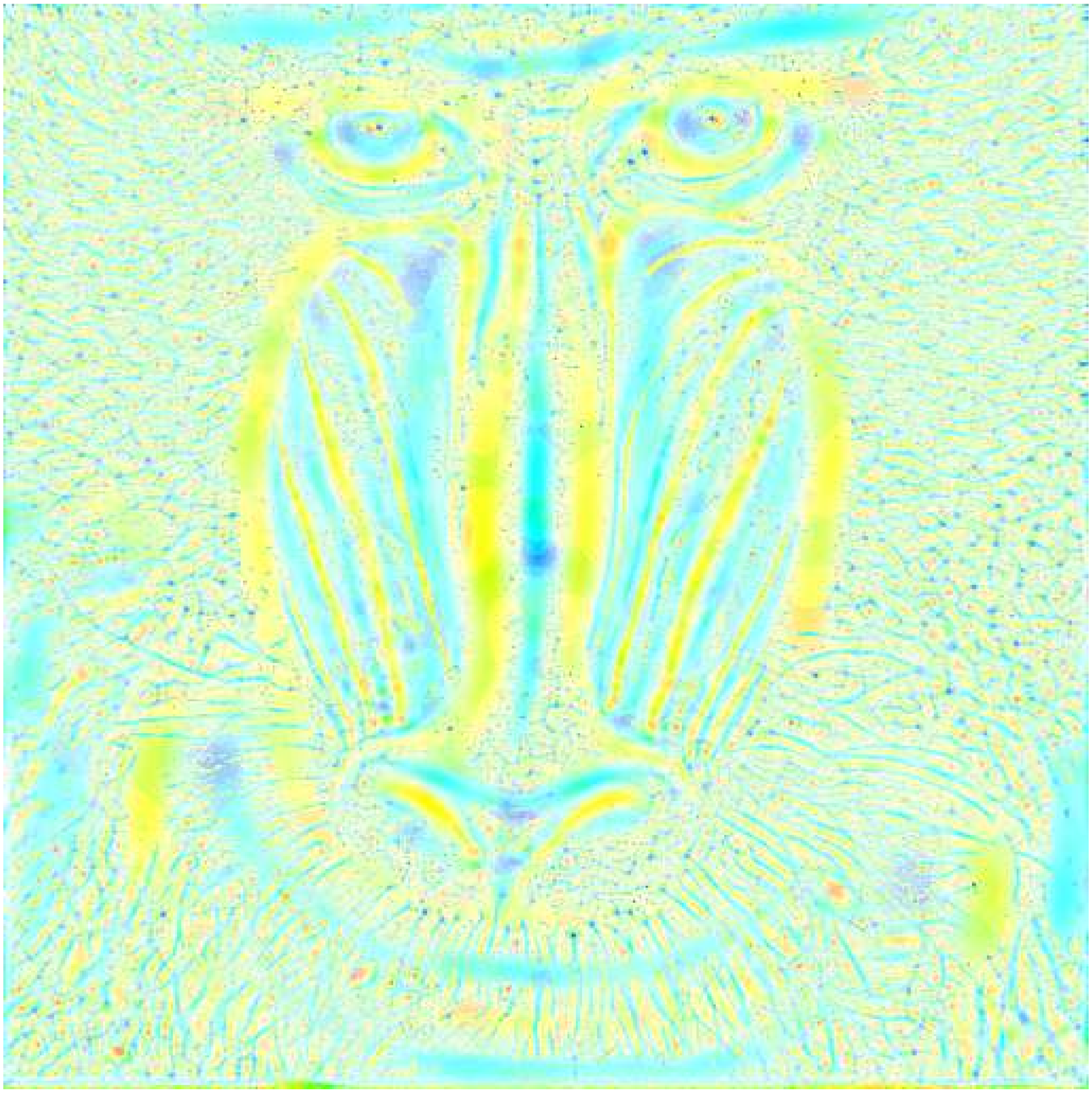


Figure 6.4: Baboon result processed at multiple scales.

CHAPTER 7

Application to mammography

Radiologists have a difficult task identifying cancers in X-ray images of breasts; it is possible to miss the cancer amongst the other features in the image. To overcome this problem much effort has gone into enhancing X-ray images to make the radiologist's job easier. Often this involves removing small features from the image such as veins and arteries in an effort to clean up the image. However, if it were possible to simply enhance the cancers in the image or at least contrast them with other features then that would also be helpful.

The method developed in this thesis can contrast different features using colour. It has been applied to a mammograph of a normal breast in Figure 7.1 and the resulting image is Figure 7.2. It has also been applied to a mammograph of a cancerous breast in Figure 7.3 and the resulting image is Figure 7.4. Veins and arteries are all shown in bright cyan allowing the viewer to easily distinguish them from other features in the image. Cancers show up as white patches in mammographs and as bright blue responses in the shape processed image. Unfortunately, other non-cancerous features in the image also result in bright blue colours. Perhaps the coloured images should be used as a starting point for the radiologist before looking at the original image.

Another way to enhance the image is to remove the veins and arteries. Recording where there are pixels with shape parameter values indicating a line-/ridge-like features with a high correlation values. The intensity of the central pixel of the reconstructed image for the local area around that pixel is recorded. This is the reconstructed image used for spatial correlation mentioned in Section 5.1. The recorded intensities are subtracted from the original image to remove veins and arteries. This idea has been applied to Figure 7.5 resulting in Figure 7.6. The resulting image shows a reduction in the veins and arteries in the image. However, a closer examination shows that there are dark outlines around positions where veins and arteries have been removed. This problem arises because there is no easy way to determine acceptable correlation values. For this reason not all pixels that make up lines/ridges in the image are detected. Lowering the acceptable threshold for shape correlation would fail as the method would remove veins where no veins exist.

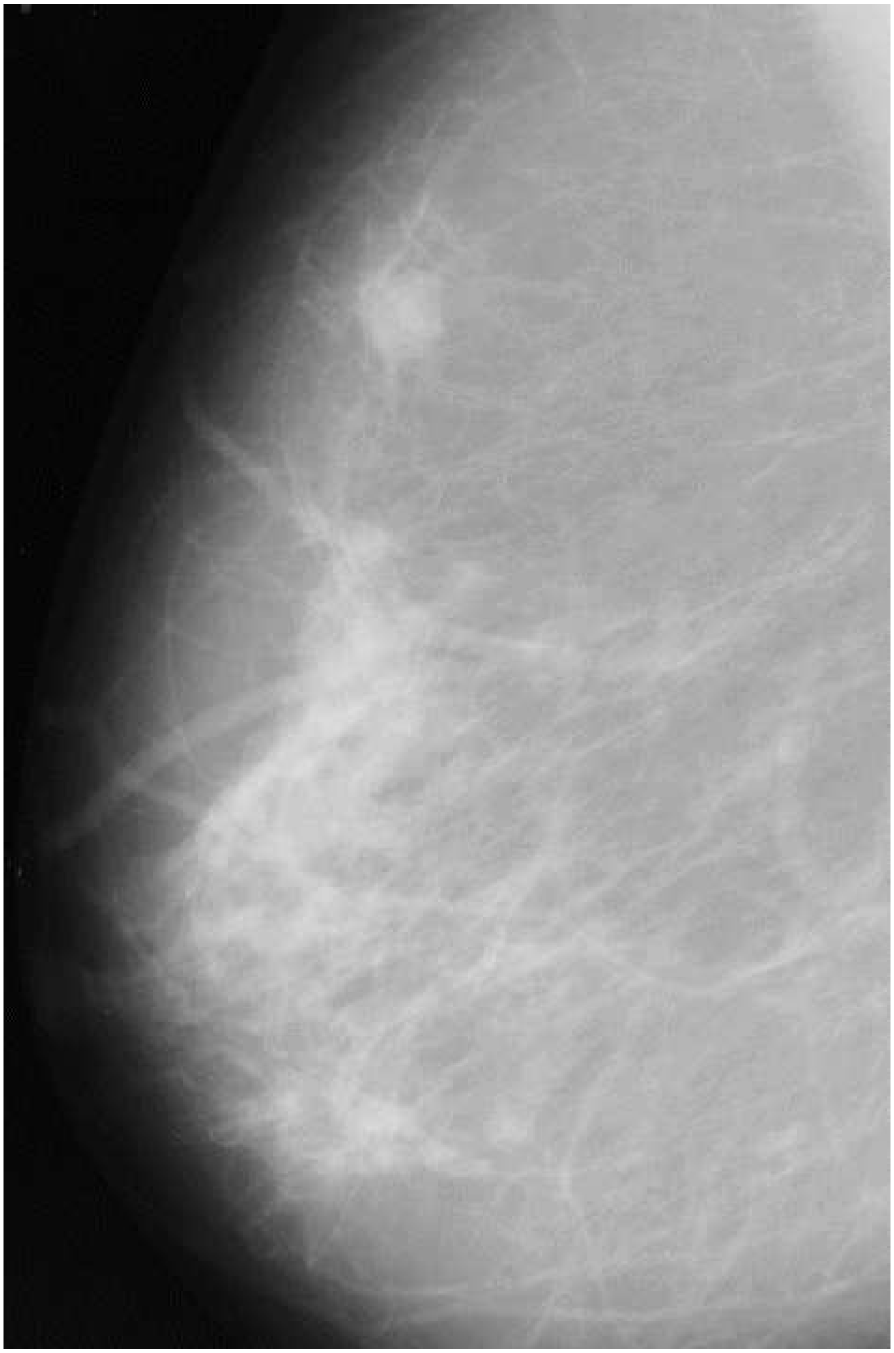


Figure 7.1: Mammograph of normal breast. (Image size 474×719)

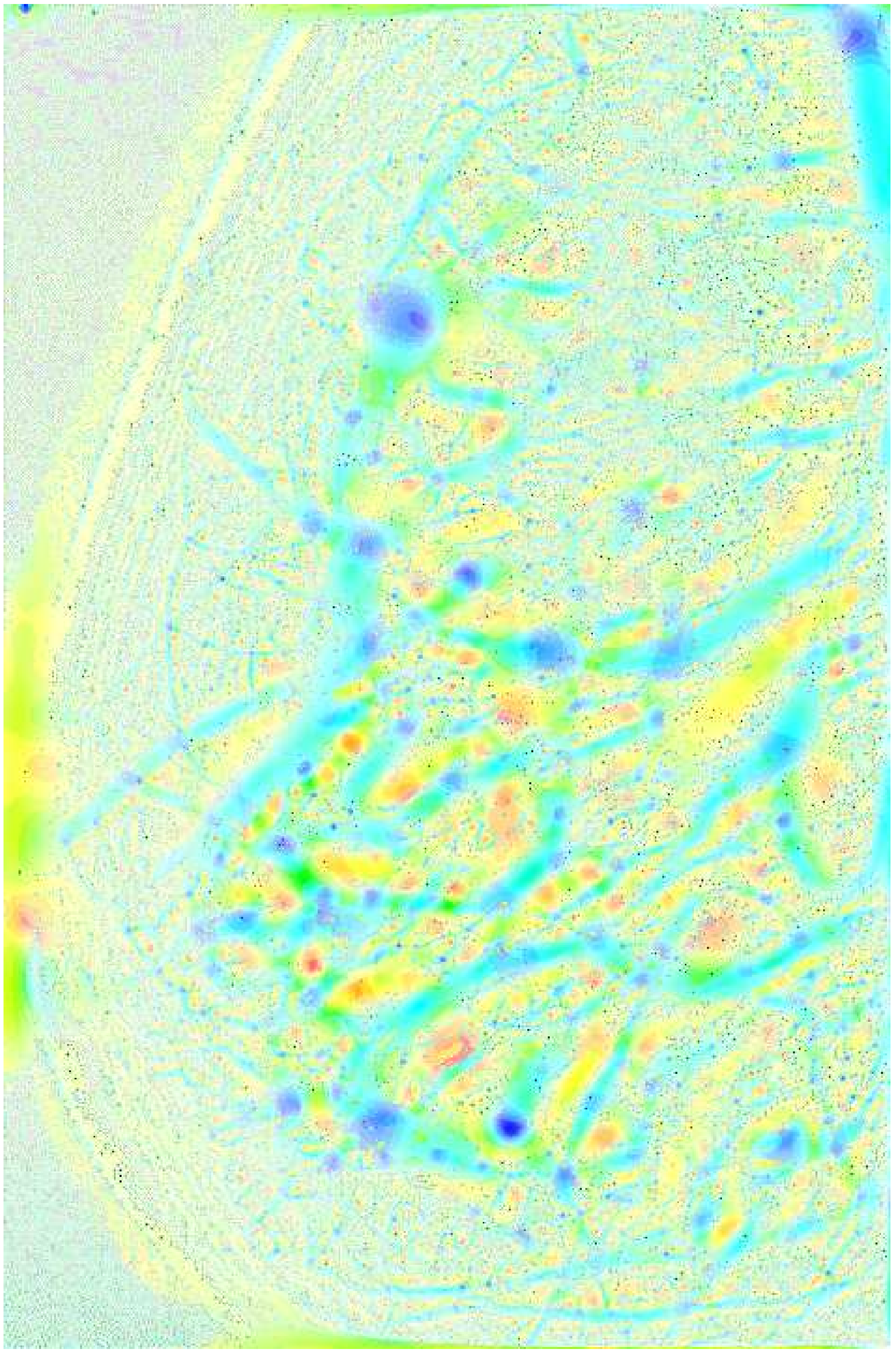


Figure 7.2: Coloured result for the mammograph.

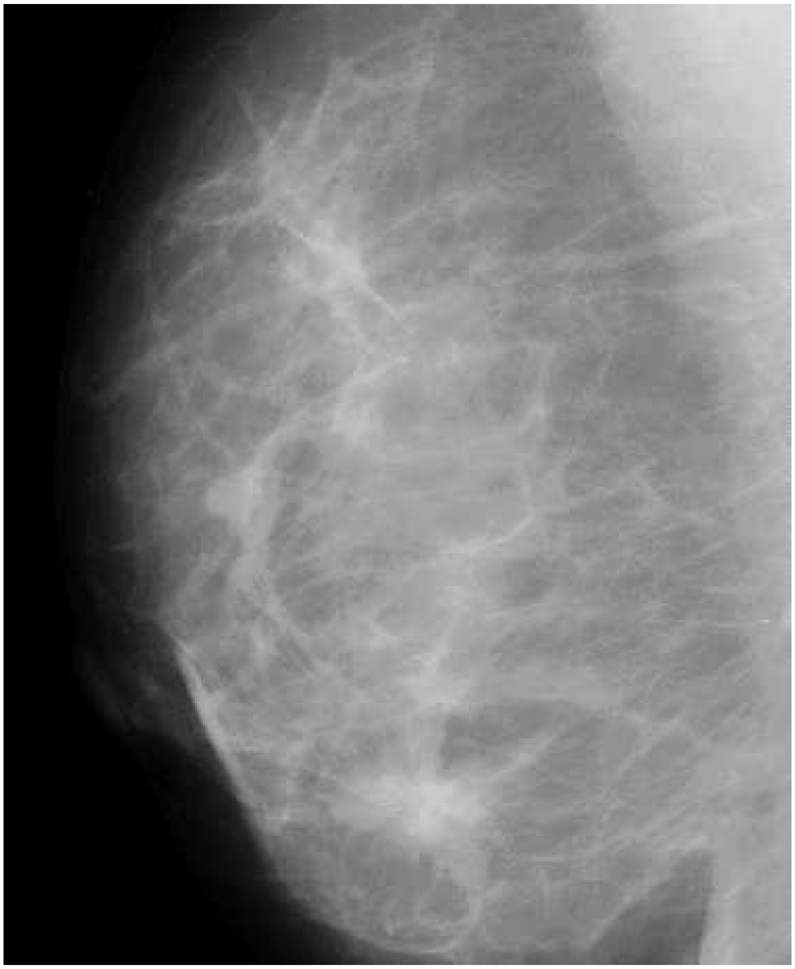


Figure 7.3: Mammograph of cancerous breast. The cancer is in the bottom centre. (Image size 379×462)

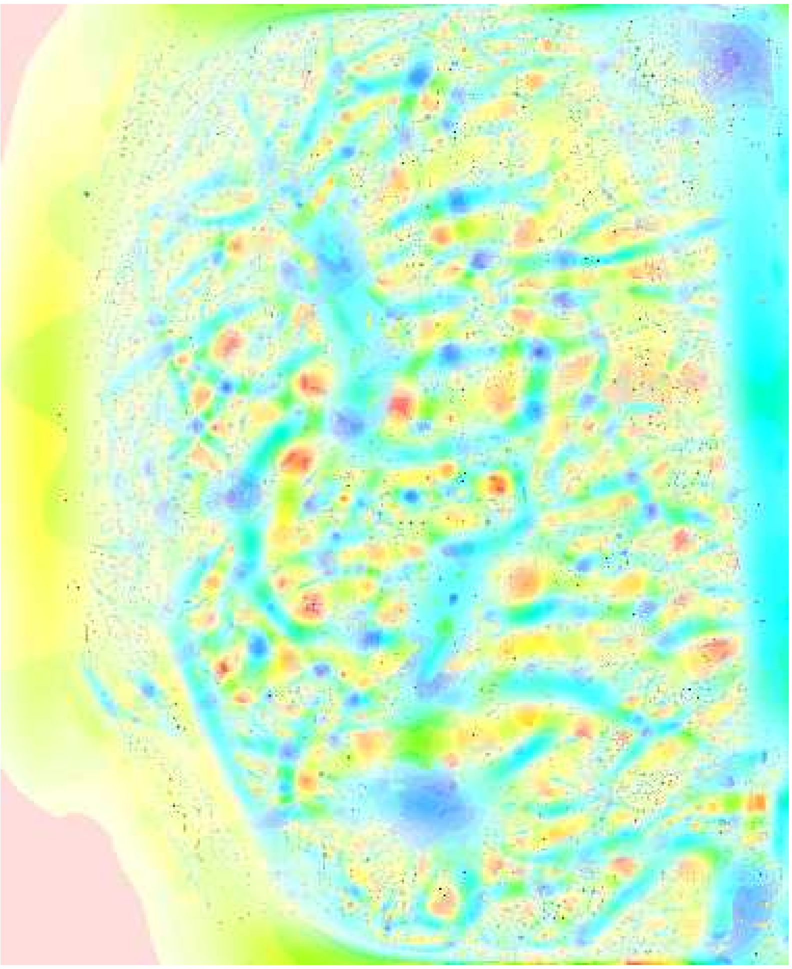


Figure 7.4: Coloured result for the mammograph. The cancer is a bright blue patch.

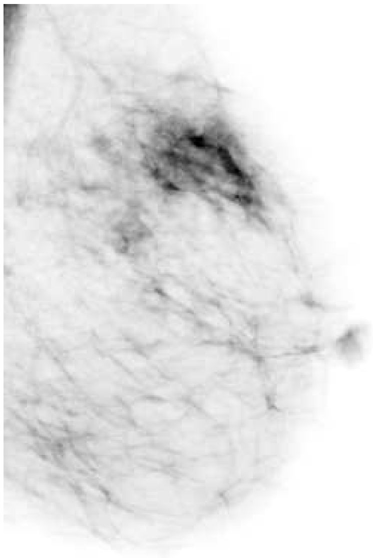


Figure 7.5: Original mammograph. (Image size 260×389)

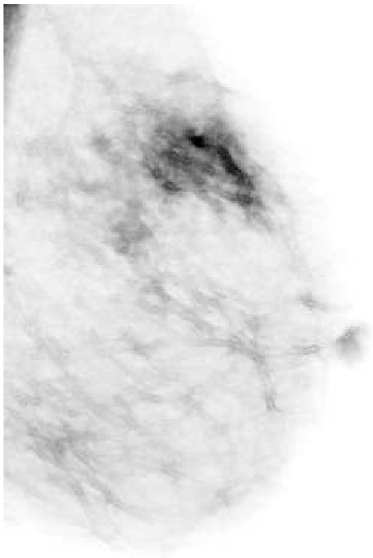


Figure 7.6: Mammograph with veins and arteries suppressed.

CHAPTER 8

Conclusion

Of the five main problems stated at the end of Chapter 2, four have been successfully resolved.

- The problem with the many-to-one shape classification has been solved by using a spatial correlation as a confidence factor.
- I have proposed a partial solution for incorporating first derivative information into the technique. Success of this rests with the development of a five-dimensional derotation algorithm and should be the subject of further research.
- By mapping the shape parameter to hue and the confidence to saturation we can visualize the shape information.
- The problem of scale has been resolved by using multiple filters that increase in size exponentially. The "best" scale for any pixel is the one with the highest confidence value. This method has been applied to some grey scale images and the results highlight important features from these images.
- The method has been applied to mammograph images and the results contrast veins, arteries, and cancers.

The problems stated are not the only areas that require investigation. Other enhancements could be made to improve the method.

Danielsson [3] shows in his report that the algorithm can be applied to volumetric images. Modifying the method outlined in this thesis to apply to three-dimensional images should not be difficult. The two-dimensional Gaussian would now be three-dimensional. The method for three dimensions described in Danielsson's report uses the six second-order derivative operators to calculate six orthonormal responses. After derotation the values extracted represent three Euler angles which are the shape's orientation, two shape parameters, and a second-order magnitude. Then a confidence factor could be calculated using a three-dimensional spatial correlation. This correlation would be calculated between a local volume of the original image and a reconstructed image generated from a weighted sum of the six three-dimensional filters. The two shape parameters and confidence factor could be mapped to a three-dimensional colour map (instead of the two-dimensional map used in this thesis).

No work into improving the efficiency of the algorithm was done. As it is, the algorithm has a high complexity and requires a great deal of computing power to process large images. Reducing the complexity would make the method much more useful for application in industry.

References

1. J Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8, No. 6:679–698, November 1986.

2. Per-Erik Danielsson. Analysis of 3D-volume data using 2nd derivatives. In *Proceedings DICTA-95*, pages 14–19, St John's College, University of Queensland, Brisbane, December 1995.

3. Per-Erik Danielsson. Second derivatives for detection of orientation and shape in 2D- and 3D-data. Unpublished results, email:`ped@isy.liu.se` for a copy, February 1997.

4. Per-Erik Danielsson and Olle Seger. Rotation invariance in gradient and higher order derivative operators. *Comp. Vision, Graphics and Image Processing*, 49:198–221, 1990.

5. Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Massachusetts, 1993.

6. Luc M J. Florack, Bart M. ter Haar Romeny, Jan J. Koenderink, and Max A. Viergever. Scale and the differential structure of images. *Image and Vision Computing*, pages 376–388, 1992.

7. Jan J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

8. D. Marr and E. Hildreth. Theory of edge detection. In *Proceedings of the Royal Society of London*, volume 207, pages 187–217, 1980.

9. M. C. Morrone and R. A. Owens. Feature detection from local energy. *Pattern Recognition Letters*, 6:303–313, 1987.

10. K. K. Pingle. Visual perception by a computer. In *Automatic Interpretation and Classification of Images*, pages 277–284. Academic Press, New York, 1969.

11. J. M. S. Prewitt. Object enhancement and extraction. In *Picture Processing and Psychopictorics*, pages 76–149. Academic Press, New York, 1970.

12. L.G. Roberts. Machine perception of three-dimensional solids. In *Optical and Electro-Optical Information Processing*, pages 159–197. MIT press, Cambridge, Massachusetts, 1965.

APPENDIX A

Research proposal

Title:	The Investigation of 2nd Derivative Operators on 3D/Volumetric Data
Author:	Michael Barrett-Lennard
Supervisors:	Peter Kovesi Chris Pudney

Keywords: Rotation invariance, Second Derivatives, 3D volumes, Segmentation, 3D Shapes, Orientation, Confocal Microscope, Spherical Harmonics, Orthogonality.

C. R. Classification : I.4.6, I.5.4

A.1 Background

This project will implement an image processing algorithm designed by Per-Erik Danielsson [2]. The algorithm finds second order derivative information in the image to detect features normally not found by first order methods. For each feature shape, orientation and magnitude information are returned by this algorithm.

Many phenomena in nature are described by second order differential equations. These include heat conduction, fluid mechanics and all forms of wave motion. In situations like these, it seems logical that image data to be analyzed, will have information that can only be extracted using second derivative methods. Second order methods have many characteristics that make them different from first order methods.

First derivative methods only detect odd symmetric intensity changes, while second derivative methods detect even symmetric intensity changes. This means that first and second derivative methods can be considered complementary in nature. For example, the local energy feature detector uses a pair of filters one of which is for even symmetric changes and the other for odd symmetric changes [9].

First derivative methods only give information about gradient strength, and orientation. The second order method in addition, gives information about the general shape of the feature around the point we are looking at. Some classifications in 3D include vortices, strings, blobs, and planes. Shape information is useful in segmenting and visualizing 3D images. For example, automatically identifying an organ inside a patient. We could only show regions having a specific type of local shape and use this to find boundary positions of the organ in the body.

A.2 Aims

The project will aim to achieve the following.

- An implementation of the algorithm for 2D images, and a comparison with first order methods. (5 weeks)
- Extend the implementation to work with 3D images and again make a comparison with first order methods. (11 weeks)
- Apply the program to 3D confocal microscope images. Supplied by Chris Pudney from his work in the Pharmacology Department. Use the second order information for visualization and segmentation. (5 weeks)
- The thesis will be written summarizing the results obtained. (6 weeks)

A.3 Methods

The algorithm first finds the six second derivatives f_{xx}, f_{yy}, f_{zz}, f_{xy}, f_{xz}, and f_{yz} using a filter for each point in the image. These quantities will change based on the orientation, magnitude, and shape of the feature. To identify the feature shape without regards to its orientation we convert these values to be rotationally invariant.

The six second derivatives of the function need to be viewed as a six dimensional space. Unfortunately the quantities are not orthogonal in general. To convert the quantities to be orthogonal they are transformed to spherically harmonic components called f_{20}, f_{21}, f_{22}, f_{23}, f_{24}, and f_{25}.

Next the orientation of the feature can be determined using a procedure named Derotation. Now that we have orthogonal axiss in our space, the vector f_{20}–f_{25} can now be rotated onto three of the coordinate planes and a magnitude extracted. The angles we use in this rotation are the Euler angles of the feature. After the three rotations there are three non zero quantities left. Then these quantities are combined to give a total derivative magnitude.

In first derivative methods only a single quantity indicating derivative strength is returned. The fact that we now have three values for second derivatives shows that more information is returned from this method. The extra information is used to identify the general shape of the feature.

The three magnitude values are used to derive two angles λ_1,λ_2 which are used to determine an orientation in the three dimensional space of the magnitudes. It is this orientation that indicates the feature shape. If you can imagine a sphere with an arrow pointing out of it from its centre, the arrow points at different shape type areas on the spheres surface.

The project will be attempted by following these steps.

- Filter the image to obtain the 2nd derivative information.
- Find the orientation of features that are revealed in the image.
- Extract the second derivative magnitude.
- Then determine the shape type for each feature.

The above steps will be implemented for both 2D and 3D images on Matlab.

APPENDIX B

Mathematical Proofs

B.1 Derivation of $H(\rho)$

Let our regularising function $h(x, y)$ be equal to the Gaussian.

$$h(x,y) = \frac{1}{2\pi} \exp\left(\frac{-x^2 - y^2}{2}\right)$$

Taking a Fourier transform with respect to x.

$$\hat{h}(u,y) = \frac{\sqrt{2\pi}}{2\pi} \exp\left(\frac{-u^2 - y^2}{2}\right)$$

Taking a Fourier transform with respect to y.

$$\begin{aligned} H(u,v) &= \frac{2\pi}{2\pi} \exp\left(\frac{-u^2 - v^2}{2}\right) \\ &= \exp\left(\frac{-u^2 - v^2}{2}\right) \end{aligned}$$

Converting to polar coordinates let $\rho^2 = u^2 + v^2$

$$H(\rho) = \exp\left(\frac{-\rho^2}{2}\right)$$

B.2 Proof that G_{uu}, G_{vv}, and G_{uv} are not orthonormal

Proof by counterexample G_{uu},G_{vv}:

$$\begin{aligned} &\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\cos{(2\varphi)^2})(-4\pi^2\rho^2 H(\rho)\sin{(2\varphi)^2}) d\varphi d\rho \\ &= \int_0^\infty \int_{-\pi}^{\pi} 16\pi^4\rho^4 H(\rho)^2 \cos{(2\varphi)}^2 \sin{(2\varphi)}^2 d\varphi d\rho \\ &= \frac{3}{2}\pi^{\frac{11}{2}} \end{aligned}$$

Therefore G_{uu}, G_{vv}, and G_{uv} are not orthogonal and therefore not orthonormal.

B.3 Proof that B_{20}, B_{21}, and B_{22} are orthogonal.

B_{20},B_{21}:

$$\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{1}{3}})(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\cos{(2\varphi)})d\varphi d\rho$$
$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{16\sqrt{2}\pi^4}{3}\rho^4 H(\rho)^2 \cos{(2\varphi)} d\varphi d\rho$$
$$= 0$$

B_{20},B_{22}:

$$\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{1}{3}})(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\sin{(2\varphi)})d\varphi d\rho$$
$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{16\sqrt{2}\pi^4}{3}\rho^4 H(\rho)^2 \sin{(2\varphi)} d\varphi d\rho$$
$$= 0$$

B_{21},B_{22}:

$$\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\cos{(2\varphi)})(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\sin{(2\varphi)})d\varphi d\rho$$
$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{32\pi^4}{3}\rho^4 H(\rho)^2 \cos{(2\varphi)}\sin{(2\varphi)} d\varphi d\rho$$
$$= 0$$

B.4 Proof that B_{20}, B_{21}, and B_{22} are normalised.

B_{20}:

$$\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{1}{3}})^2 d\varphi d\rho$$
$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{16\pi^4}{3}\rho^4 H(\rho)^2 d\varphi d\rho$$
$$= 1$$

B_{21}:

$$\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\cos{(2\varphi)})^2 d\varphi d\rho$$
$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{32\pi^4}{3}\rho^4 H(\rho)^2 \cos{(2\varphi)}^2 d\varphi d\rho$$
$$= 1$$

B_{22}:

$$\begin{aligned}
&\int_0^\infty \int_{-\pi}^{\pi} (-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\sin{(2\varphi)})^2 d\varphi d\rho \\
&= \int_0^\infty \int_{-\pi}^{\pi} \frac{32\pi^4}{3}\rho^4 H(\rho)^2 \sin{(2\varphi)}^2 d\varphi d\rho \\
&= 1
\end{aligned}$$

All the above used the same scaling factor for the regularising function $H(\rho)$.

B.5 Proof that the first and second derivative operators are orthogonal

B_{10},B_{20}:

$$\begin{aligned}
&\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho\cos{(\varphi)}H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{1}{3}})d\varphi d\rho \\
&= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8i\pi^3}{\sqrt{3}}\rho^3 H(\rho)^2 \cos{(\varphi)} d\varphi d\rho \\
&= 0
\end{aligned}$$

B_{10},B_{21}:

$$\begin{aligned}
&\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho\cos{(\varphi)}H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\cos{(2\varphi)})d\varphi d\rho \\
&= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8\sqrt{2}i\pi^3}{\sqrt{3}}\rho^3 H(\rho)^2 \cos{(\varphi)}\cos{(2\varphi)} d\varphi d\rho \\
&= 0
\end{aligned}$$

B_{10},B_{22}:

$$\begin{aligned}
&\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho\cos{(\varphi)}H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\sin{(2\varphi)})d\varphi d\rho \\
&= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8\sqrt{2}i\pi^3}{\sqrt{3}}\rho^3 H(\rho)^2 \cos{(\varphi)}\sin{(2\varphi)} d\varphi d\rho \\
&= 0
\end{aligned}$$

B_{11},B_{20}:

$$\begin{aligned}
&\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho\sin{(\varphi)}H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{1}{3}})d\varphi d\rho \\
&= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8i\pi^3}{\sqrt{3}}\rho^3 H(\rho)^2 \sin{(\varphi)} d\varphi d\rho \\
&= 0
\end{aligned}$$

B_{11},B_{21}:

$$\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho \sin{(\varphi)} H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\cos{(2\varphi)}) d\varphi d\rho$$

$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8\sqrt{2}i\pi^3}{\sqrt{3}} \rho^3 H(\rho)^2 \sin{(\varphi)} \cos{(2\varphi)} d\varphi d\rho$$

$$= 0$$

B_{11},B_{22}:

$$\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho \sin{(\varphi)} H(\rho))(-4\pi^2\rho^2 H(\rho)\sqrt{\frac{2}{3}}\sin{(2\varphi)}) d\varphi d\rho$$

$$= \int_0^\infty \int_{-\pi}^{\pi} \frac{-8\sqrt{2}i\pi^3}{\sqrt{3}} \rho^3 H(\rho)^2 \sin{(\varphi)} \sin{(2\varphi)} d\varphi d\rho$$

$$= 0$$

B.6 Proof the first derivative operators are normalised

B_{10}:

$$\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho \cos{(\varphi)} H(\rho))^2 d\varphi d\rho$$

$$= \int_0^\infty \int_{-\pi}^{\pi} -4\pi^2\rho^2 \cos{(\varphi)}^2 H(\rho)^2 d\varphi d\rho$$

$$= 1$$

B_{11}:

$$\int_0^\infty \int_{-\pi}^{\pi} (2\pi i\rho \sin{(\varphi)} H(\rho))^2 d\varphi d\rho$$

$$= \int_0^\infty \int_{-\pi}^{\pi} -4\pi^2\rho^2 \sin{(\varphi)}^2 H(\rho)^2 d\varphi d\rho$$

$$= 1$$

All the above used the same scaling factor for the regularising function $H(\rho)$.