DOCTOR OF PHILOSOPHY

Boundary and shape recognition for automated skin tumour diagnosis

Denton, William Edward

Award date:
1998

Awarding institution:
University of Wales, Bangor

Boundary and Shape Recognition for Automated Skin Tumour Diagnosis

William Edward Denton

Thesis Submitted in Candidature for the Degree of
Doctor of Philosophy

August 1998

School of Electronic Engineering and Computer Systems
University of Wales, Bangor
United Kingdom

Summary

This thesis examines the analysis of digitised optical skin cancer images. Its aims are to develop a new method of skin lesion boundary detection that meets the need for reliable automated boundary detection; to develop and assess shape analysis methods that are suitable for skin lesion diagnosis and can be related to human shape perception to allow application of expert knowledge; and to develop methods of creating simulated skin lesion images to allow study of the behaviour of boundary detection methods.

Methods for analysing lesion shape are presented and tested on real and synthesised shapes. The effect of boundary noise on shape analysis is investigated and shown to increase the bulkiness and textural and structural fractal dimensions. It is shown that the necessity of using high resolution shapes and the effect of noise on fractal dimension measurement indicates that it may not be suitable for assessing lesion shape.

A new method for locating and isolating a lesion within a skin image is presented, which provides an image containing the lesion and surrounding skin, but excluding background objects, with an indication of the lesion's size. This method correctly identified all of the lesions in a test set. Its extension to images containing multiple lesions is also discussed.

An edge focusing algorithm for skin lesion boundary detection, which uses either Laplacian of Gaussian (LoG) or Canny edge detection, is presented and tested on real images. This is a new application of edge focusing, which uses new methods to control the boundary during focusing and to select the output boundary, using image contrast. By combining this algorithm with the process of isolating lesions, a system that can find lesion boundaries in a range of images is produced. Testing using real and verification images indicate that it is capable of working on a wide range of images.

A new method for synthesising simulated skin lesion images is presented and used to assess the edge focusing algorithm using an area based comparison method. Using parameters, derived from real images, the lesion location and isolation method correctly identified all the simulated lesions. The performance of the LoG and Canny edge focusing algorithms is shown to decrease as the lesion boundary becomes less distinct.

The development of a computer based tool to perform or assist in the diagnosis of skin cancer and how the research presented in this thesis would be incorporated into such a tool is discussed. Further research into the measurement techniques, required to obtain diagnostic and prognostic information, is outlined and the use of this information in providing diagnosis and prognosis is examined.

Acknowledgements

I would like to express my sincere thanks to Dr Andrew Duller and Peter Fish for their invaluable advice, guidance and motivation throughout the course of this research. I also would like to acknowledge and give thanks to the following for providing the images used in this research: Prof. S. Leinster of the Medical School, University of Liverpool and Dr D. Crawlford and V. Wallace of the Department of Physics, Institute of Cancer Research, Royal Marsden Hospital.

My especial thanks goes to my parents, Judith and Ron Denton, and to Bibi Khan for their patience, support and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Skin cancer is potentially fatal and the incidence of its deadliest form, melanoma[1], is increasing across the world [63, 9, 60, 13, 59, 34, 40]. Early diagnosis and treatment are essential as it is easiest to cure when it is caught in its early stages, which results in the best prognosis. However, early accurate diagnosis of skin cancer, particularly melanoma, is difficult and requires specialists with considerable training and experience, making it expensive [40, 60, 63]. Skin cancer is often inaccurately diagnosed by human experts, who find it difficult to agree on the presence of particular diagnostic and prognostic features [40, 60]. Automated or assisted diagnosis has the potential to reduce the mortality rate by allowing more reliable and earlier assessment.

Although the safest treatment procedure would be to excise all lesions, this is not practical because of the time and expense involved [21]. The high cost of excision combined with the effects of inaccurate diagnosis makes it essential that diagnosis is highly sensitive (very few false negatives, i.e. very few dangerous lesions are missed) and highly specific (very few false positives, i.e. very few benign lesions are misclassified as dangerous). Mass screening requires high sensitivity, but high specificity is not as important as suspicious lesions can be considered in more detail [61]. However, it is important, in mass screening, to ensure that the number that require expert diagnosis is kept small to minimise costs.

The effects of incorrect or late diagnosis on the prognosis combined with the cost of diagnosis and unnecessary treatment makes the automated diagnosis of skin cancer attractive. Computerised diagnosis could bring high accuracy and consistency combined with relatively low cost. Even if the performance of a computerised system is not good enough to perform automated diagnosis, it could still assist in diagnosis and mass screening [53].

---

[1] The terms malignant melanoma and melanoma are used interchangeably.

In the development of a computer based diagnostic system, a number of factors must be considered to ensure that it can usefully and practically perform or assist diagnosis. The system must be reliable and consistent, which is achievable by ensuring that it is tolerant of the conditions in which it is used and that it can cope with the wide range of skin features. To be useful in a clinical environment it must be easy to use, have low equipment and usage costs and able to operate reasonably quickly.

Even when a computerised system is suitable for use in a clinical environment, i.e. when the performance, reliability, usability and costs are acceptable, there may be resistance to its use by doctors and patients. Whilst doctors may be familiar with using computerised tools, such as ultrasound scanners, to assist in diagnosis, currently they are not accustomed to accepting a diagnosis from a computerised system. In addition, doctors may find it difficult to interpret new information, from an automated system, and to relate this to their perception of the disease [11]. The diagnosis task may be de-skilled by automation, hence this may be resisted by doctors and patients may also find it difficult to accept. To increase the acceptability, an automated system should be extensively tested and be able to explain its actions.

Human diagnosis is mainly performed by visual assessment of the lesion, and is the easiest, quickest and cheapest method [59, 40]. Consequently, knowledge exists of human methods and the diagnostic and prognostic factors which are considered important. Optical skin images can easily be obtained and digitised either indirectly by digitisation of photographic slides or directly from video cameras and digital photographic cameras. These factors make it attractive to develop image processing techniques to aid diagnosis.

To analyse skin lesion images, segmentation is necessary to identify the different parts of the image. The most important form of this is the accurate and reliable identification of the lesion's boundary [36, 30, 77] enabling the division of the image into lesion, skin and other background features, thereby ensuring that colour and texture measurements are carried out only on the lesion. In addition, given the boundary, the important diagnostic factor of lesion shape can be analysed [59, 40, 63, 34]. Boundary detection is made difficult by the highly variable nature of lesion images (e.g. lesion size and nature, skin texture, presence of hair, etc.) and image capture conditions (e.g. lighting conditions [21], background objects, etc.). The difficulty of automated boundary detection has

been noted by those attempting to construct automated diagnostic methods for skin lesions, who have found it necessary to use manual boundary detection [55, 29, 53] to obtain the necessary performance. Many of the diagnostic factors which they used (e.g. irregularity, asymmetry and area) had to be either manually estimated or calculated from the manually obtained boundaries. Consequently, their automated diagnostic methods rely on information provided by dermatologists. To increase the ease of use and repeatability whilst reducing usage costs, it is desirable that diagnostic methods are fully automated and so do not rely on the presence of suitably trained dermatologists to provide information which could be obtained automatically. The inadequacy of current boundary detection methods, which has been noted by those constructing automated diagnostic methods [55, 29, 53] and by those developing the boundary detection methods [30, 36], shows the need to develop new boundary detection methods.

Whilst developing a system, testing is necessary to ensure the acceptability of the methods used. Testing can ensure the system is built correctly and the correct system is built, i.e. the system performs as expected and the system does what is required [88]. Data (images) with known characteristics are required for this purpose [88]. Real images, being the most realistic, can be used as test data, however, when using real images it is difficult to find a "gold standard" with which to compare the test results. For the overall diagnosis histology provides a "gold standard", but it cannot provide measurements of particular image features. By synthesising images with known features, the performance of individual parts of the system can be assessed.

Analysis of skin images uses image processing, which is a rapidly expanding field with a huge range of applications including image compression for TV transmission [38], recognition of car number plates [38], enhancement of X-ray images [58] and human face recognition [15]. The growth in image processing has been assisted by the availability of cheap high performance computers and the decreasing costs of image capture (e.g. video cameras, digital cameras and image digitisation boards) and storage (e.g. hard disks and PhotoCDs). Digital image processing techniques, readily performed on computers, can be used in many tasks including the detection of edges, the analysis of texture and shape, the compensation of geometric deformation and the identification of regions by thresholding or region growing methods [38, 80, 85, 58].

1.1 Aims of this Research

The broad aims of this research can be summarised by:

- **automatic boundary detection and shape analysis**: to develop a method of skin lesion boundary detection and shape analysis algorithms which overcome the limitations of existing methods with a view to improving the performance of automatic skin lesion diagnosis.
- **assessment methods**: to develop an improved method of assessing the performance of boundary detection and shape analysis algorithms using simulated images with known characteristics.
- **comparison of methods of boundary detection and shape analysis**: to use the above to compare methods of boundary detection and shape analysis with a view to selection of optimum methods and to gain understanding of the behaviour of the algorithms.

Fulfilling these aims is likely to improve the performance of automatic skin lesion diagnostic systems based on computerised analysis of optical skin images and facilitate the improvement of boundary detection and shape analysis algorithms through greater understanding of algorithm behaviour under known conditions. These aims must be pursued within the constraints of the equipment (e.g. computing and image acquisition) which is reasonably available, affordable and suitable for use in an automated diagnostic system. At the start of this research, it was considered that the possible improvement in performance from colour did not justify the increased cost, complexity and computation associated with colour image acquisition and analysis (processing and storage). In addition, research in this area had already shown the potential for lesion boundary detection through the use of luminance [36, 77]. It was, also, decided to use 256 (8 bits) greylevels as suitable capture and processing equipment is readily available, affordable and reasonably fast, and has already used by many researchers in this area (such as [12, 36, 77]) and other areas (such as [38, 48, 52]).

1.2 Overview of the Thesis

Chapter 2 provides the background for the research presented in the thesis and illustrates the difficulty of analysing skin images. It starts with an outline of the occurrence and causes of skin cancer. The skin's structure is described as background for an explanation

of the types of skin cancer and benign lesions. Current diagnostic methods are described together with their limitations and the necessity of accurate and early diagnosis along with treatment. The chapter concludes by describing the need for and requirements of a computer based diagnostic system.

The digital image processing techniques, which are used for analysing optical skin cancer images, are briefly described in chapter 3. This is the background for the literature review, in chapter 4, and the research reported in this thesis. The topics briefly covered include the following: segmentation, contour tracing, thresholding, region growing, edge detection, texture, Fourier and inverse Fourier transforms, spatial filtering, frequency filtering, image histograms, morphology, principal component transform, colour, graphics operations, image processing software, data formats and image file formats.

A review of published work on the computerised analysis of skin cancer images is presented in chapter 4. It begins by reviewing methods for the important and difficult task of identifying lesion boundaries. Methods for characterising lesion shape from these detected lesion boundaries are reviewed. Publications which use colour and texture to provide diagnostic information are briefly reviewed. A brief survey of methods of measuring lesion 3-dimensional shape and thickness by image processing is given. The chapter finishes with a review of computerised diagnosis methods.

Methods for analysing lesion shape to determine "roundness" and "roughness" are presented in chapter 5, where they are illustrated and tested on real and test shapes. A method of assessing lesion "roundness" by a bulkiness factor is explained. The measurement of fractal dimension, which assesses "roughness", is examined and an automated method for assessing the large and small scale structure of a shape through fractal dimension is presented. In an automated diagnostic system, these techniques would be used after the lesion boundary had been found, but are presented here as the bulkiness measurement is used in chapter 6.

Chapter 6 examines the problem of approximately locating and isolating a lesion within an image and presents a process which provides a cropped image containing the lesion and sufficient surrounding skin, but excluding background objects and skin features. This process starts with pre-processing and filtering, which aims to improve the quality

of the binary image thresholding used to separate the lesion from the background. A method for simplifying the thresholded image to make the identification, using shape, of the lesion easier and more reliable is explained. After creating a simplified thresholded image, the lesion is identified by size and shape. The chapter finishes with a description of the information generated to allow further analysis of the image.

The detection of luminance edges in images, which forms the background for the research presented in chapter 8, is examined chapter 7. It begins by outlining the limitations of simple edge detectors, which make them unsuitable for lesion boundary detection. The implementation of the convolution and zero crossing detection used in the Laplacian of Gaussian (LoG) edge detector is described in detail. A description of the implementation of the Canny edge detector and associated non-maxima suppression is given. Two methods for converting a map of the edges in an image (generated by either the LoG or Canny edge detectors) into the co-ordinates of the isolated edges are presented. The effects of image border erosion on the output of edge detectors are discussed and a method for reducing its impact is reported. The processing of irregularly shaped image regions is discussed and the use of a mask image for this task is explained.

Chapter 8 presents a new edge focusing algorithm for skin lesion boundary detection, which uses the information provided by the process described in chapter 6. The algorithm is outlined and then the process by which the algorithm generates a series of boundaries is explained using LoG edge detection. A method for selecting a suitable boundary from this series is given. The modifications necessary to change from using LoG to Canny edge detection are explained. The chapter ends by describing the testing of the algorithm on real images.

A method for synthesising simulated skin and lesion images is presented in chapter 9. This method is divided into three parts: shape generation, boundary transition modelling and texture generation. The creation of shapes similar to those of lesions based on ellipses with random large and small scale irregularities is explained. A method for creating an indistinct transition between the synthesised lesion and synthesised skin is presented. The synthesis of skin and lesion textures from measurements of real skin and lesion textures is discussed and a method presented for performing this task. The combining of the synthesised lesion shape, boundary transition and texture is described. A method for

comparing a true boundary with an estimated boundary, which can be used to test boundary detection methods, is presented. The chapter concludes by describing the results of testing the edge focusing algorithm, presented in chapter 8, on simulated images.

The development of a computer based tool to perform or assist in diagnosis of skin cancer is outlined in chapter 10. The discussion shows how the work presented in this thesis and other skin cancer image processing research would be incorporated into such a system. It also indicates areas of future research and other research which may be applied to this area. The chapter begins by discussing the factors that should be considered whilst developing techniques for analysing skin cancer images. With these factors in mind, the development and use of these techniques is discussed. The use of real and simulated skin cancer images for assessing and testing a computerised diagnostic system is examined. The chapter concludes by considering the implicit and explicit acquisition and use of knowledge for the early detection and diagnosis of skin cancer.

Chapter 11 focuses on the principal findings of this thesis and begins with the conclusions of the literature review, which indicated where new research was required. It then outlines the main findings of this work, which include the results of the development and assessment of an edge focusing algorithm for skin lesion boundary detection and the development and use of simulated skin lesions in the assessment of this algorithm. The chapter concludes by outlining ideas for future research and development which could lead to a computerised diagnostic system for skin cancer.

1.3 Contributions of this Research Work

The contributions which this thesis makes to the computerised analysis of skin cancer may be summarised as:

- **Shape analysis**: The automation of fractal dimension measurement to analyse large and small scale irregularities in lesion shape. A new method of creating shapes with known fractal dimensions was developed, which allows more reliable assessment of fractal dimension measurement methods. Using this method the effect of boundary noise on shape analysis was investigated and shown to increase the bulkiness and textural and structural fractal dimensions. It is shown that the necessity of using high

resolution shapes and the effect of noise on fractal dimension measurement indicates that it may not be suitable for assessing lesion shape.

- **Skin lesion locating and isolating**: A new method for approximately locating and isolating a lesion within a digitised optical skin image. This method, which allows lesion analysis to be tolerant of image capture conditions and contents, correctly identified and isolated all of the lesions in a test set.
- **Skin lesion boundary detection**: The development of a new edge focusing algorithm for detecting lesion boundaries in monochrome images. This algorithm creates a series of boundaries of increasing detail from which a suitable boundary for the particular image is selected. The boundary selection criteria is based on the image contrast between just inside and just outside the boundary. These developments, combined with the lesion locating and isolating method, have made edge focusing suitable for lesion boundary detection on a wide range of images and may have made it suitable for other applications where automatic boundary detection is required for objects with boundaries which are not clearly defined.
- **Synthesis of skin lesion images**: The synthesis of simulated images containing a lesion and surrounding skin using textures measured from real images. These simulated images can be used to assess boundary detection algorithms using an area based measurement. These simulated images were used to compare two edge focusing algorithms based on the Laplacian of Gaussian (LoG) and Canny edge detectors [45, 10]. This comparison showed that as the boundary became less distinct the performance of both algorithms decreased. In addition, it showed that the Canny based edge focusing algorithm performed better than the LoG algorithm on small lesion shapes. However, on simulated lesion images with boundaries similar to those observed in real images both algorithms had good performance. The simulated lesion images and boundary comparison method could be used to compare the relative performance of different boundary detection algorithms and other algorithms such as shape analysis. This would enable the comparative testing, refinement and improvement of boundary detection and shape analysis algorithms, through gaining a better understanding of the behaviour of the algorithms under known conditions.

To date three refereed publications have resulted from this work [17, 18, 19], which are reproduced in Appendix D.

Chapter 2

Skin Cancer, its Diagnosis and Computerised Diagnosis

2.1 Introduction

This chapter introduces human and computerised diagnosis of skin cancer (in particular malignant melanoma[1]) and provides the background which guided the research presented in this thesis. By describing the characteristics of skin, benign lesions and skin cancer it illustrates the difficulty of analysing skin images. The chapter is divided into the following seven sections:

- **skin cancer**: The increasing occurrence of skin cancer together with its causes, including UV radiation, is described in this section.
- **skin structure**: This section describes the skin's many functions and its structure.
- **non-melanoma skin cancer**: The most common and less dangerous non-melanoma skin cancers are described in this section.
- **melanoma skin cancer**: The more dangerous melanoma skin cancers are introduced in this section.
- **benign lesions**: This section lists some benign lesions which require distinguishing from skin cancer.
- **diagnosis, prognosis and management**: Human diagnostic methods, including checklists, and the prognostic factors are described in this section together with the main treatment method (excision).
- **computer based diagnosis**: This section outlines the need for and requirements of a computer based diagnostic system.

---

[1] The terms malignant melanoma and melanoma are used interchangeably. Melanoma can be known as cutaneous melanoma.

2.2 Skin Cancer

Skin cancer, such as melanoma, is the most common form of cancer in the US [9] and across the world its incidence is increasing. Between 1979 and 1989 there was an 82% increase in melanomas in Scotland [13]. In Australia, the lethal mix of fair skins, from people of North European ancestry, and sunshine has made it 10 times as common as in northern Europe [13]. Most melanomas occur in fair-skinned, blue-eyed, red or blond-haired individuals; the incidence in black people is much lower. A relationship exists between sun exposure and the number of people who develop melanoma, and there seems to be a correlation between melanomas and brief intense exposure to UV which occurs long before the cancer appears [63]. There is also evidence of a hereditary component in some melanomas [9].

Some of the blame for the increasing occurrence of melanomas may lie with the thinning of the ozone layer [74], but a large part of the blame lies with beach holidays and the myth of the healthy tan [13]. As melanoma takes 10-20 years to develop there has been insufficient time for the thinning of the ozone layer to have had a significant impact [74]. The increasing number of reports may reflect better reporting or the popularity of suntans in the 1970's and 1980's [74].

Ultraviolet (UV) radiation (wavelength: 250 to 400nm) is biologically very important as it causes most of the changes in the skin associated with ageing, sunburn, skin cancer and other skin diseases [63, 34]. It is divided into three types: UVA (320-400nm), UVB (280-320nm) and UVC (250-280nm). UVC is not currently important as most it is filtered out by the ozone layer, but it will become more important if the ozone layer becomes thinner [63]. Sunburn, suntan and ageing are affected by UVB and UVA, which can be blocked by sunscreens. All forms of skin cancer are becoming more frequent due to increasing exposure to UV and carcinogenic substances, and to an increasingly 'elderly' population [63].

2.3 Skin Structure

The skin, with a typical surface area of 1.8 m^2, is one of the largest organs, making up about 16% of body weight [34], and is frequently damaged as it forms the first layer of defence against disease and mechanical injury [63]. It contains many cell types which can malfunction, giving rise to many different types of skin diseases [63]. Skin disease

is very common, making 10-15% of the general practitioner's work, in the UK, and is the second commonest cause of loss of work [63]. The skin's functions are [63, 34]:

- protection against water loss and gain, mechanical injury, toxic substances and micro-organisms.
- reduction of UV penetration.
- thermal regulation through heat lost by sweat evaporation and dermal blood supply control.
- heat conservation by body hair and insulation by body fat.
- assistance in the synthesis of vitamin D using UVB.
- assistance in immune surveillance.
- provision of sensory input for touch, pressure, vibration, pain, itch and temperature by a high density of nerves.
- provision of a surface for grip.

The skin's structure and thickness vary with site; it can be divided into three main layers: epidermis, dermis and subcutis, which are illustrated in Figure 2-1 and described below [34, 63]:

- **Epidermis**

  The epidermis provides mechanical, micro-organism and water protection; it can be divided into four layers (from the inside outwards): basal cell layer, prickle cell layer, granular layer and horny layer (stratum corneum). Most of the cells in the basal layer divide and move outwards to provide a protective layer of strong, but flexible dead cells (the horny layer or stratum corneum). The basal layer also contains melanocytes, which synthesise melanin (a dark protective pigment which is a natural sunscreen and gives the skin its colour), and cells concerned with touch. The prickle cell layer contains cells which have migrated upwards from the basal layer and immune system cells. In the granular layer, the cells moving outward are strengthened and cemented together to form the horny layer. The stratum corneum is composed of polyhedral sheets of overlapping dead cells, which give the skin its small scale texture (skin creases), and are eventually shed from the skin surface. In some diseases, the formation of this structure may be disturbed preventing the shedding of cells and changing the skin creases. UVB is mostly absorbed by the stratum corneum, but

some of it photo-oxidises melanin and stimulates melanocytes to produce more melanin.

- **Dermis**

  The dermis, below the epidermis, provides mechanical protection and contains many specialised structures. These structures include hair follicles, nerves, sweat glands, immune cells, blood and lymphatic vessels. Cells in this layer synthesise collagen and elastin fibres which provide mechanical strength and elasticity respectively.

- **Subcutis**

  The subcutaneous layer contains loose connective tissue and fat, which provides insulation.

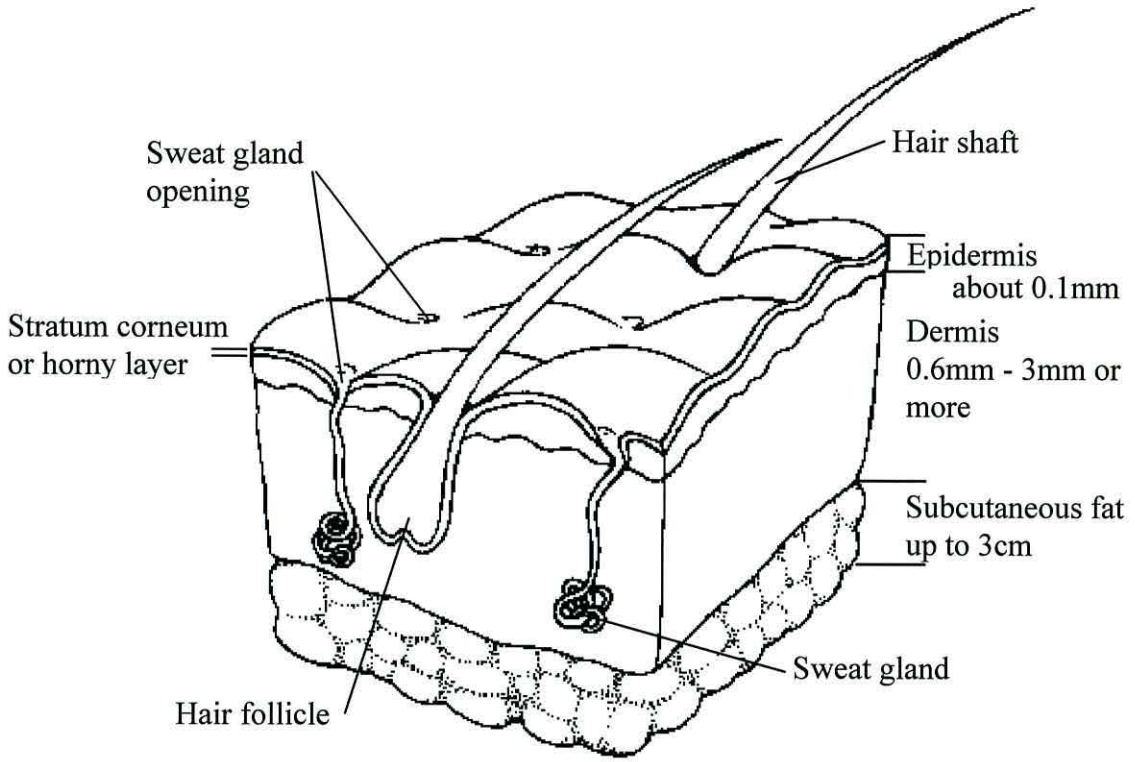


Figure 2-1: Labelled skin cross-section, adapted from Marks [63].

(a) **(b)**

Figure 2-2: Non-melanoma skin cancers [69]. (a) Basal cell cancer. (b) Squamous cell cancer.

2.4 Non-melanoma Skin Cancer

Skin cancer can be divided into non-melanoma (basal cell cancer and squamous cell cancer, which are the most and second most common) and melanomas which are less common but more dangerous, as they spread quite rapidly [9, 63]. Most non-melanoma skin cancers can be cured.

- **Basal cell cancer**[1] [34, 9, 63] Figure 2-2(a)

  This slow growing painless skin cancer usually occurs in areas where the skin has been exposed to the sun and commonly on the face in elderly and middle-age people. It is often pigmented and may later be pink, flesh, red, blue, or ivory coloured. In an early phase, it may appear as a small raised bump with a smooth shiny appearance. It may spread to tissues around the cancer, but usually does not spread to other parts of the body.

- **Squamous cell cancer**[2] [34, 63, 9] Figure 2-2(b)

  This usually occurs in areas where the skin has been exposed to the sun, often on the top of the nose, forehead, lower lip, and hands. It is more common in men than in women and mainly occurs in people over 55 years old. The tumour is a firm irregular fleshy growth which may feel scaly, bleed or develop a crust. It can rapidly increase in size creating a large lump which may form an ulcer and, if untreated, may spread to the surrounding lymph glands.

2.5 Melanoma Skin Cancer

Malignant melanoma arises in the melanocyte cells (in the basal layer of the epidermis), which it is named after, and can occur anywhere on the body, but most commonly on the back, chest, abdomen, and lower legs [9, 34, 63]. It can occur in pre-existing cells, most often a mole, which may begin to enlarge, become mottled, and develop an irregular surface or borders, or a variety of colours (variegated colouring). The melanoma may also itch, burn, or bleed easily. Solar UV radiation is believed to the single most important cause, but as up to half of lesions do not occur on sun-exposed sites other

---

[1] Basal cell cancer is also known as basal cell carcinoma[3], basalioma and basal cell epithelioma[4].

[2] Squamous cell cancer is also known as squamous cell carcinoma[3] and squamous cell epithelioma[4].

[3] carcinoma: malignant tumour derived from skin tissue.

[4] epithelioma: skin cancer.

factors play a role [63]. The effects of a brief intense sun exposure, causing sunburn, may account for occurrence of melanoma on areas of skin that are only occasionally exposed to the sun [63]. When melanoma is at an early stage it is easily cured by prompt surgical excision, hence its early diagnosis is very important [59, 63, 40, 34]. Melanomas have been divided into groups, with different biological behaviour and therefore prognosis, depending on their growth patterns and appearance.

- **Superficial spreading malignant melanoma** [40, 34] Figure 2-3(a)

  This form primarily affects people in their forties and fifties, and in the UK over 50% of melanomas are of this type. It commonly occurs on the legs and backs in women, who are twice as likely to be affected as men, and on the trunk, particularly the back, in men. It is macular[1] with an irregular outline, may be notched and may have variable colour with hues of tan brown, black, blue, grey and may be white or pink in patches where there is tumour regression. Thin curable lesions do not distort skin creases or cause loss of hair, but these features will occur where there is significant tumour activity within the dermis.

- **Nodular melanoma** [40] Figure 2-3(b)

  These represent about 25% of melanomas, affect middle aged people and are more common in men. They commonly occur on the head, neck, and torso and are difficult to diagnose early. In later stages, when they are thick and have a poor prognosis, they are termed "nodular" with spherical "blue-berry" like nodules[2] with a relatively smooth surface, a relatively uniform blue-black colour and may bleed and ulcerate[3]. In earlier stages, they are small (diameters of around 0.5 cm or less), dark and appear regular, but closer examination of the border frequently reveals notching or streaking of the pigment on one edge. Sometimes they can be almost completely de-pigmented, but even in these a rim of pigment is usually left. It is important that these melanomas are detected in the earlier stage, as they are curable by excision.

- **Lentigo malignant melanoma** [40] Figure 2-3(c)

  These account for about 10% to 20% of all melanomas and usually occur in people in their seventies on sun-damaged skin, mainly, on the face. They are slowly evolving, initially tan-coloured, flat lesions which can resemble stains. As they develop they

[1] macule: a localised area of colour change without elevation or infiltration [34].
[2] nodule: a solid elevation of the skin with a diameter greater than 5mm [34].
[3] ulcer: an area of skin loss extending through the epidermis into the dermis [34].

increase in size and may change colour with dark brown or black areas appearing and the pigmentation may regress to leave blue-grey or white patches. The border can be very irregular and a fine pattern of black lines may cross it. For many months or years it can grow in the epidermis, but should be detected before any increase in thickness when it may have metastasised[1].

- **In-situ melanoma** [40]

  This type can occur on any part of the body in younger people and is confined solely to the epidermis. They are a flat poorly bounded asymmetrical lesions with, notched, scalloped or jagged borders and mottled brown pigmentation which can be tinged with blue, black or pink. Diagnosis and excision of these lesions can lead to a complete cure.

- **Acral lentiginous melanoma** [40, 34, 59] Figure 2-3(d)

  These are the least common type (approx. 5%) of melanoma in the UK, but are the commonest type in Black and Oriental people. They occur on the palms (palmar), soles (plantar), and under the nails on fingers and toes (subungual). The tumour can appear as dark brown or black spreading patch and have features of the previously described types. It is more biologically aggressive than the other types, is often diagnosed late and has poor survival figures.

[1] metastasis: the spreading of disease, especially cancer cells, from one part of the body to another.

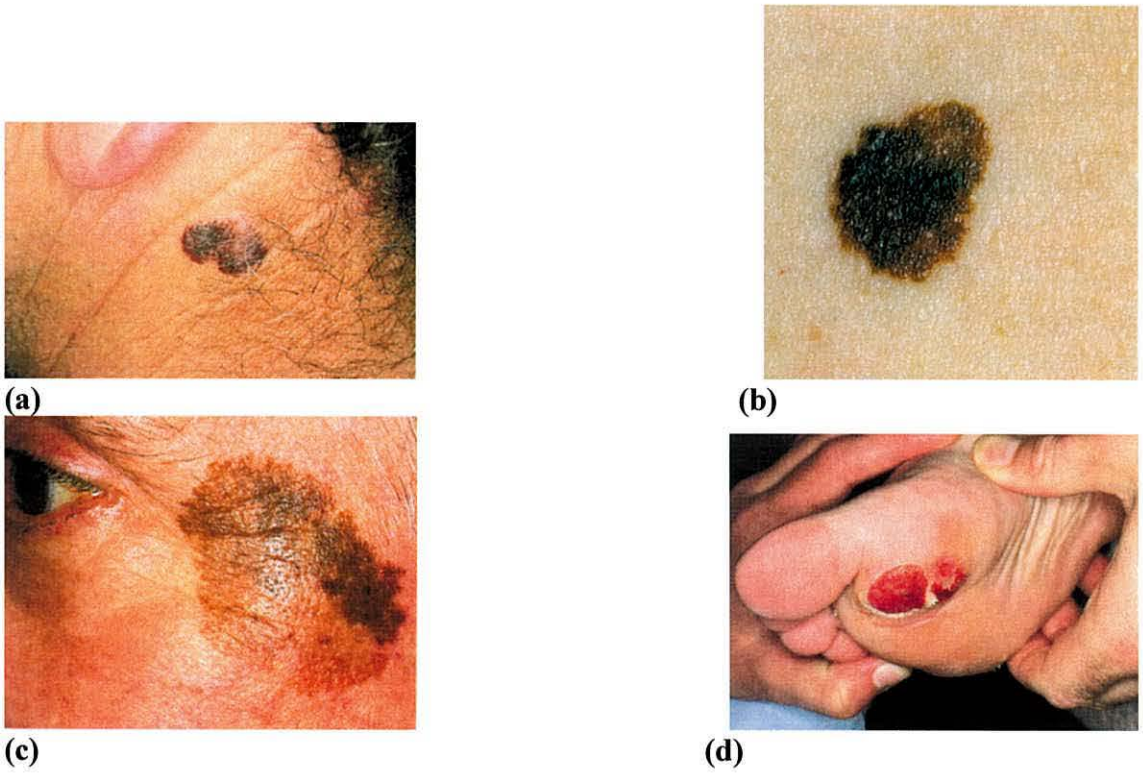


Figure 2-3: Malignant melanoma. (a) Superficial spreading malignant melanoma [34]. (b) Nodular melanoma [63]. (c) Lentigo malignant melanoma [34]. (d) Acral lentiginous melanoma [34].

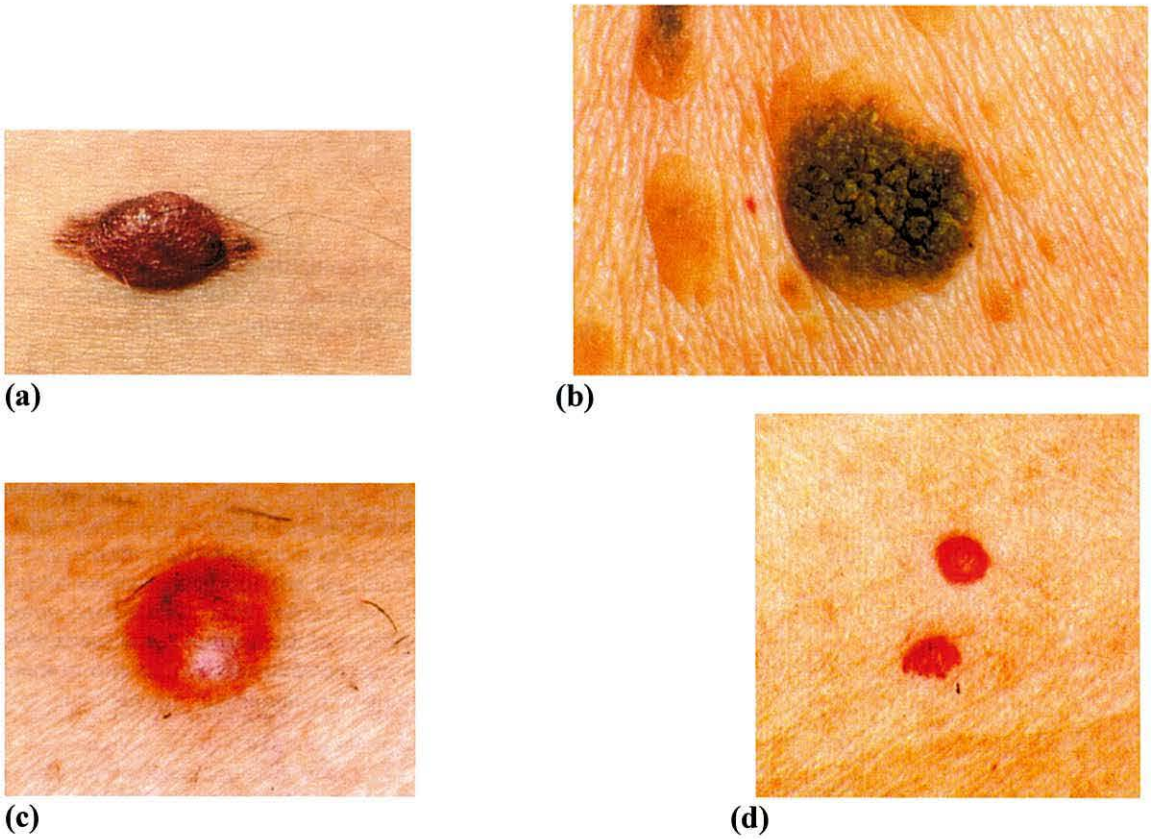


Figure 2-4: Benign lesions. (a) Benign melanocytic naevus [34]. (b) Seborrhoeic wart [63]. (c) Dermatofibroma [34]. (d) Vascular malformation [63].

2.6 Benign Lesions

Diagnosis of pigmented skin lesions is difficult [63], since other skin diseases, including the following, may have similar physical characteristics to melanoma. The differentiation of melanoma and the following conditions and basal cell cancer is important as these diseases are benign and basal cell cancer, although locally malignant, is virtually never fatal [59]. However, melanoma is a potentially fatal disease, which in an early stage is easily cured [63].

- **Benign melanocytic naevi[1] (benign pigmented moles)** [63] Figure 2-4(a)

  Moles, which exist in range of shapes and sizes, are very common in white people and are the result of clustering of melanocytes. Generally, they are less variegated in colour and have smoother outlines than melanoma. It is unusual for benign pigmented moles to transform into malignant melanoma, but if any the diagnostic signs are present it may be a melanoma [59]. The dysplastic naevi type of mole indicates a greatly increased risk of melanoma, especially when a relative has had this disease [63, 40]. Dysplastic naevi may be confused with melanoma as they may be over 7mm in diameter with an irregular edge and variable pigmentation [34].

- **Seborrhoeic warts[2]** Figure 2-4(b)

  These occur very often in people over 50 and are frequently located on trunk [59]. They have a "stuck on" appearance with usually a sandy brown colour, but may be a darker greyish brown or black [59]. Their shape is generally round or oval with a well-defined edge and they may be papular or nodular [34]. As their diameter may be greater than 1cm (with a history of increasing size) and they may have some colour variation [59], they can be confused with nodular melanoma, but may be distinguished by the number of lesions [34] and their warty appearance [63].

- **Dermatofibroma[3]** Figure 2-4(c)

  These occur in young adults, most commonly women on the lower legs, and have a brownish colour with diameters of 5-10mm [63, 34]. They are of no serious clinical significance, but are sometimes mistaken for melanomas [63].

---

[1] Types of benign melanocytic naevi include congenital naevi, dysplastic naevi, acquired naevi and compound naevi.

[2] Seborrhoeic warts are also known as seborrhoeic keratosis or basal cell papillomas[4].

[3] Dermatofibroma is also known as histiocytoma or sclerosing haemangioma.

[4] papilloma: a nipple-like projection from the skin surface [34].

- **Vascular malformations**[1] [59] Figure 2-4(d)

  These deeply pigmented dome-shaped lesions, which are sharply demarcated from the surrounding skin, may be confused with nodular melanoma. They may occur anywhere on the body at any age and may be distinguished by having red rather than brown pigment.

2.7 Diagnosis, Prognosis and Management

The early diagnosis of melanoma is important as in the early stages it is relatively easy to treat and patients have survival rates of near 100% [59]. Checklists have been proposed as an aide-memoire to assist in its diagnosis. MacKie [59] proposed the original seven-point checklist, which was adopted by the Cancer Research Campaign:

1. Itch: Minor itch or other change in sensation.
2. Size: A lesion greater than 1 cm in largest diameter.
3. Increasing size: A history of growth or other change in a pigmented lesion in an adult.
4. Shape: An irregular outline.
5. Colour variation[2]: Irregular and varied colours.
6. Inflammation: Inflammation in or at the edge of the lesion.
7. Crusting or bleeding.

A lesion with three or more of the points in the checklist is considered suspicious and those with four or more points highly likely to be a melanoma. This list was revised [60] to emphasise the importance of change in size, shape, or colour which makes the checklist more memorable and useful:

Major signs	Minor signs
Change in size	Inflammation
Change in shape	Crusting or bleeding
Change in colour	Sensory change
	Diameter 7 mm or more

[1] Vascular malformations are also known as angiomas.
[2] Colour variation can be referred to as variegated colouring (VC).

Lesions with one or more of the major signs warrant referral and those with one or more of the minor signs warrant further consideration. The American Cancer Society's ABCDE checklist is very similar to the seven-point checklist [40]:

- Asymmetry: One half of the tumour does not match the other half.
- Border Irregularity: The edges are ragged, notched, blurred.
- Colour: Pigmentation is not uniform. Shades of tan, brown and black are present. Dashes of red, white and blue add to the mottled appearance.
- Diameter: greater than 6 mm and growing.
- Elevation: elevated by 2 mm or more over the surrounding skin.

These checklists provide a guide to assist diagnosis, but are not sufficiently sensitive to never miss any melanomas [40]. Their primary purpose is to provide the general practitioner with guidance as to when to refer a suspected melanoma to a specialist. They also help in public education, which is important as patients frequently delay consulting a general practitioner about new or changing pigmented lesions, which can lead to treatment being delayed until the melanoma has a poor prognosis [59]. Both checklists are biased towards detecting superficial spreading melanomas, which are slow-growing and consequently their detection is less important than faster growing nodular melanoma [40].

After excision, the tumour thickness (Breslow thickness) between the deepest identifiable melanoma cell and the overlying skin surface can be measured by staining sections through the centre of the lesion. It can also be measured without excision by an experimental instrument called the 'nevoscope', which is described in chapter 4, and by high frequency ultrasound, which only gives accurate measurements for some lesions [87]. Breslow [4] found the thickness to be the single most important guide to prognosis, as is shown in the following table [9]:

thickness: 0.75 mm or less	5-year cure rate: almost 100% 10-year survival rate: 95%
thickness: 0.76 mm to 1.50 mm	5-year cure rate: 85% 10-year survival rate: 75%
thickness: 1.51 mm to 4.0 mm	5-year cure rate: 65% 10-year survival rate: 30%
thickness: 4.0 mm or greater	5-year cure rate: 20% 10-year survival rate: no greater than 15%

The prevention of death from malignant melanoma, or any disease, can be brought about by either primary or secondary prevention [61]:

- **Primary prevention** is the prevention of the development of the disease, which in melanoma is public education in the avoidance of excessive sun exposure. This is a very long term activity as the interval between the start of a tumour and the development of melanoma may be longer than 20 years.
- **Secondary prevention** is the prevention of deaths from the disease by either improved therapy or by earlier diagnosis.

Melanoma is a curable cancer if it is diagnosed and treated early, before it has spread beyond the initial site. Surgery, in which the lesion and a margin of surrounding skin are excised, is the primary treatment for melanoma. Other methods such as radiotherapy and chemotherapy have been less successful [34]. This absence of significant advances in the non-surgical treatment of advanced melanoma mean that efforts to reduce mortality must rely on early diagnosis of thinner lesions and also on primary prevention [61].

2.8 Computer Based Diagnosis

It is important that melanomas and other pigmented lesions are diagnosed accurately and early, as melanoma is curable if treated early. Accuracy is also important as although the safest route would be to remove all pigmented lesions, it is not practical to do this because of the time and expense [21]. Current human diagnosis is relatively inaccurate; even specialists are only 50% to 85% accurate when compared with histopatholgists[1] [40, 60, 61]. Small flat lesions, which are likely to be curable, are particularly hard to diagnose; one study concluded that clinicians were only 50% accurate when compared with histopatholgists [40]. Clinicians also find it difficult to agree on the presence of signs in pigmented lesions and, in particular, whether lesions are irregular or not [40]. Family doctors, in the UK, are unlikely to see more than one melanoma in 10 years, and hence cannot be expected to accurately diagnose melanoma [40]. These factors make the idea of computerised assistance attractive and important. Even if the accuracy is not

[1] histopathology: a branch of pathology[2] concerned with the tissue changes characteristic of disease.

[2] pathology: the branch of medicine that studies the causes and nature and effects of diseases.

high enough to perform automated diagnosis, it could assist in diagnosis and mass screening. Even the basic provision of image storage will probably assist in assessing changes in a lesion [60]. The high cost of excision, combined with the effects of inaccurate diagnosis, make it essential that diagnosis is highly sensitive (very few false negatives, i.e. very few dangerous lesions are missed) and highly specific (very few false positives, i.e. very few benign lesions are misclassified as dangerous).

Computerised diagnosis may be useful in screening or surveillance:

- **Screening** is the systematic examination of a population, which may be confined to one geographic area [61]. Individuals in this population may be selected for screening by age, sex, family history of melanoma or other factors.
- **Surveillance** is the examination at regular intervals of individuals for the development of new pigmented lesions, which may be malignant melanoma [61]. The labour intensive nature of surveillance means that it is currently confined to research centres and individuals with a known high risk of developing melanoma [61].

Both of these activities would be assisted by image storage, and computerised diagnosis or assistance could bring decreased costs through increased speed and accuracy. However, it is important that any screening or surveillance operation is fully supported with treatment and that it brings proven survival benefit [61]. The increasing occurrence of melanoma and the effect of early treatment on mortality make it an ideal candidate for screening [40, 9]. Because of its seriousness no early melanoma should be screened out and thus sensitivity is more important than specificity [60] as suspicious lesions can be considered in more detail.

For a computer based system to be useful and practical in assisting or performing diagnosis it is important, as well as being accurate and reliable, that it meets a number of other requirements:

- **ease of use**: To be useful in a clinical environment it must be easy to use, so that the operator does not require specialist knowledge in the areas of computing and image processing.

- **tolerance of image capture conditions**: To be usable in a clinical environment it must be tolerant of lighting variations (such as flash highlights), camera set-up and position, image background and equipment noise.
- **reasonable equipment and operating costs**: This ensures that it can be widely used.
- **speed**: Data (image) acquisition must be rapid and results must be produced in a reasonable time, so that the operating cost is minimised.
- **repeatability**: The same results should be generated for different images of the same lesion.
- **robust**: Robustness increases its usability by reducing the necessity for re-capturing images and reducing the need to control tightly the image contents and capture conditions. This is important as the nature of the problem (the wide range of skin images and capture conditions) creates images that contain a wide range of lesion size and nature, skin and lesion texture, hair, etc.

The checklists, described earlier, provide guidance for the measurements which should be performed. However, they do not show the complete picture, as specialists use their experience to perform diagnosis, which makes obtaining and using their methods very difficult.

Computerised diagnosis (e.g. expert systems or neural networks) requires accurate and consistent assessment of lesion features. This assessment cannot be reliably carried out by humans. Hence, computerised analysis of the lesion is required to obtain the necessary diagnostic measurements. Optical skin images provide the best method of assessing lesions as they can provide the necessary measurements quickly, easily and cheaply, and allow human diagnostic knowledge to be used.

An important first step in automatic diagnosis is to accurately and reliably find the boundary of a lesion and this thesis concentrates on this problem. An accurate boundary allows the image to be segmented into lesion, skin and other background, which ensures that colour and texture measurements are carried out only on the lesion image. In addition, given an accurate outline, the important diagnostic factor of lesion shape can be analysed to provide quantitative measurements of size, asymmetry [90] and border irregularity [36, 12].

Chapter 3

Image Processing

3.1 Introduction

Digital image processing enhances or extracts information from spatially sampled and intensity quantised images for either human or automatic interpretation. This chapter provides the background in image processing for the literature review and research presented in this thesis. The background covered includes common image processing terminology, methods and the choice of programming languages and file formats. It briefly describes the following areas:

- **notation and terminology.**
- **image connectivity.**
- **segmentation.**
- **contour tracing.**
- **image thresholding.**
- **region growing.**
- **edge detection.**
- **texture.**
- **Fourier and inverse Fourier transforms.**
- **spatial filtering.**
- **frequency filtering.**
- **image histograms.**
- **morphology.**
- **principal component transform.**
- **colour.**
- **graphics operations.**
- **image processing software.**
- **data formats.**
- **image file formats.**

3.1 Notation and Terminology

The basic notation and terminology for image processing, used in this thesis, are described in this section and the notation is listed in appendix A.

To capture a *digital image* the output from a camera must be digitised both spatially and in amplitude, by the processes of *image sampling* and *grey-level quantisation* respectively. Images are usually sampled on a regular rectangular grid which may vary from 640 by 480 for VGA graphics on an IBM PC to 3072 by 2048 for PhotoCD images. Images are usually quantised into an integer power of two number of grey-levels, and in this research monochrome images with 256 (2^8) grey-levels were used. 256 grey-levels are commonly used as this facilitates the easy, cheap and quick capture, storage and processing of images with standard hardware. If G = number of possible grey-levels in the image, then the maximum range of grey-levels in an image is 0 to L = G-1. For a particular image, the maximum and minimum intensity are denoted by I_{min} and I_{max} respectively. Each image is considered as an array of numbers, with a width of M and a height of N, where each number represents a single picture element or *pixel* which represents the *brightness* (or *grey-level*) of a single point (the intensity of the point x, y is denoted by $i(x, y)$). The orientation and range of the image co-ordinates is illustrated in Figure 3-1. The Euclidean distance between any two points $((x_1, y_1), (x_2, y_2))$ in an image is given by:

$$D_e(x_1, y_1, x_2, y_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{3.1}$$

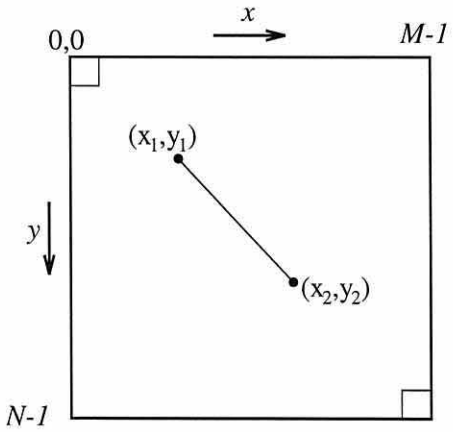


Figure 3-1: Image notation.

3.2 Image Connectivity

Image connectivity describes the spatial relationship between adjacent pixels and is important in establishing the boundaries of regions and the extent of edges. The neighbours of a pixel are classified as *direct neighbours* (*d-neighbours* or 4-*way* neighbours) if the neighbouring pixels share a side and as *indirect neighbours* (*i-neighbours* or 8-*way* neighbours) if they only touch at a corner [38, 76, 80]. The notation for the positions of neighbouring pixels, d-neighbours and i-neighbours is shown in Figure 3-2.

An *i-path* is a sequence of pixels in which each is an i-neighbour of the previous one. A *d-path* is a similar sequence in which each is restricted to be a d-neighbour of its predecessor. A *closed path* occurs when the first and last pixels are the same. A region is *i-connected* if for every pair of pixels in the region there is an i-path between them which uses pixels in the region only. Similarly the region is *d-connected* if there is a d-path between its pixels. The *contour* or *i-contour* of a region is the set of pixels in the region which has at least one neighbour not in the region. A *full region* has more than four pixels with a simple i-contour and the difference between the region and its i-contour is d-connected. This definition excludes regions with contours which are 8-way connected to themselves at any point.

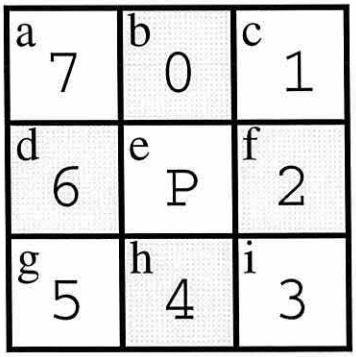


Figure 3-2: Image connectivity. P = central pixel. Direct neighbours are shaded. a-i = pixel names. 0-7 = order in which pixels are stored for operations involving the neighbours of a pixel.

3.3 Segmentation

Segmentation separates the constituent parts in an image, which can identify the areas of interest for a particular application and consequently is dependent on that application [38]. Generally, automatic segmentation is one of the most important and difficult tasks in image processing and is usually the process which leads to the eventual success or failure of an application. Segmentation is based on the recognition of discontinuities and similarities in image features. This may be the grey-levels themselves in the case of edge detection or thresholding (discontinuity) and region growing (similarity). Alternatively, it may be based on texture analysis or a combination of features. Once segmentation has been performed, the image will be divided into *regions* of connected pixels. Each region has an *area* given by the number of pixels contained within the region's *boundary* and a *perimeter* given by the length of its boundary.

3.4 Contour Tracing

Contour tracing (or *contour following*) finds the co-ordinates of the external and internal contours of a region and is used in finding the locations of isolated regions in a thresholded image [76, 80]. The brief description given here is based on Pavlidis' method [76] and uses the relative neighbour positions shown in Figure 3-3.

The first stage of contour tracing is to find an initial pixel on the contour which has at least one direct neighbour not in the region. This initial pixel can be found by any method, but is often found by a top-to-bottom, left-to-right scan of the image. A search direction (S) determines where to search for the next contour point and is initially set to 6 (vertically downwards) (Figure 3-3). The neighbouring pixels are examined in the order S-1, S and S+1, which ensures that the leftmost available pixel is selected. If the

3	2	1
4	P	0
5	6	7

Figure 3-3: Relative neighbour positions used in contour tracing. P = central pixel. [76]

pixel at a position corresponding to one of these search directions is in the region then contour tracing moves onto that pixel and changes the search direction. The change in search direction depends on which pixel it moves to: S-1, S and S+1 change to S-2, S, S+1 respectively. If none of the three neighbouring pixels are in the region then the search direction, S, is changed to S+2. The searching of neighbouring pixels and changing of the search direction is performed at most three times, which ensures that the search does not continue indefinitely for regions with only one pixel.

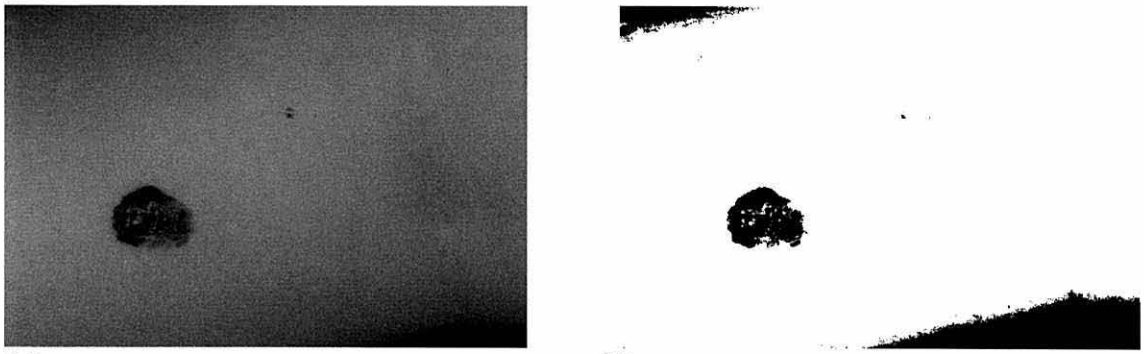
This method produces a closed i-path and follows external contours anti-clockwise and internal contours clockwise, which allows them to be distinguished. It can trace external and internal contours when combined with an algorithm which, after finding the external contour, searches for the internal contours [76]. This is not described here as it is not used in this thesis.

3.5 Image Thresholding

Thresholding is a simple and extensively used method of image segmentation, which can separate objects from a different intensity background by reducing the number of intensity levels in an image and can be applied to grey-scale and colour images [38, 80]. *Binary thresholding* converts an image into a binary image and *multi-level thresholding* isolates different intensity ranges. Thresholding labels all of an intensity range with the same value, which for binary thresholding results in all values less than or equal to the threshold being set to black and all values greater than the threshold being set to white. Multi-level thresholding assigns labels to more than two intensity ranges. *Fixed level thresholding*, in which the threshold is not dependent on the image, is the simplest form of thresholding, which can only be applied when the image capture conditions and image contents are tightly controlled. In most applications this is not possible as the illumination varies, hence the threshold must be adapted for every image. *Adaptive thresholding* methods can be divided into two classes:

- *global* - threshold is dependent on the whole image and fixed across the image.
- *local* - threshold is dependent on local image properties and may also be dependent on the whole image.

When performing local thresholding, the image may be divided into rectangular sub-images and a threshold computed for each sub-image. These sub-image thresholds may be interpolated to give a continuous threshold across the whole image. Threshold selection may be based on the whole image or a subset of the image, such as those points with high edge gradients, hence may be less affected by the relative sizes of the objects and background. The relative sizes of the objects and background can affect the threshold calculation. Thresholding is used in section 6.4 for separating a lesion from the surrounding skin and binary global thresholding is illustrated in Figure 3-4.



(a) **(b)**

Figure 3-4: Binary image thresholding. (a) Original image (744 by 487). (b) Thresholded image.

3.6 Region Growing

Region growing groups connected similar pixels into homogeneous regions [38] and is explained here as background for the literature review. The similarity between pixels may be determined by grey-level, texture, colour, etc. Each region grown starts from a *seed point* and is progressively grown outwards from this point to include neighbouring pixels until no more connected pixels similar to those in the region are found. To use region growing, a method of manually or automatically selecting the seed point is required. In addition, a method of determining the similarity between new points and those in the region is required. The simplest method of stopping region growing is to stop when no more similar pixels are found. However, this does not take account of a region's size and shape, which can be used when a model of expected region is available.

3.7 Edge Detection

Luminance or colour discontinuities in an image are important as they can indicate the boundaries of objects or regions [38, 80]. An *edge* indicates where there is a boundary

between two regions with relatively distinct grey-level or colour properties. Edge detection is important in image processing and is considered in chapter 7, which describes some different types of edge detection techniques and their application.

There are two basic approaches to edge detection: model fitting and differential detection [80]. In model based edge detection, a small region of the image is fitted to a model of an edge and if the fit is sufficiently close, an edge is considered to exist at that point. In a differential approach, derivatives of the original image are used to accentuate the spatial grey-level changes. Differential edge detectors can be divided into the following major categories:

1. **First Derivative Edge Detectors**

   A spatial first differentiation of the image is performed and an edge is judged to be present if the local maximum gradient exceeds a threshold value. An example of this is the Canny detector [10] described in section 7.5.

2. **Second Derivative Edge Detectors**

   The spatial second differential of the image is found and edges are indicated by the zero-crossings. An example of this is the Laplacian of Gaussian (LoG) detector [65, 64] described in section 7.3.

First and second derivative edge detectors differ in their recognition of extended ramp edges [80]. A first derivative edge detector, would mark an edge wherever the gradient (slope) exceeds a threshold; raising this threshold results in low amplitude edges being missed. With a second derivative edge detector, such as LoG, the edge (zero-crossing) will be marked at the mid-point of the ramp edge provided that the size of the LoG exceeds the slope width; when the LoG size is smaller than the slope width, an edge is marked at each end of the slope.

After obtaining a map of the edges in the image (an *edge map*), it may be necessary to obtain the co-ordinates of all of the edges in the image so that higher level processing can be performed. Two methods for performing *edge tracing* or *following*, which is the process of converting an *edge map* into the co-ordinates of the connected edges, are presented in chapter 7.

In a discrete image, an edge may lie between two pixels, hence a convention which marks the edges on the higher amplitude pixels is used [80]. The edge orientation nomenclature used in this thesis is illustrated in Figure 3-5.

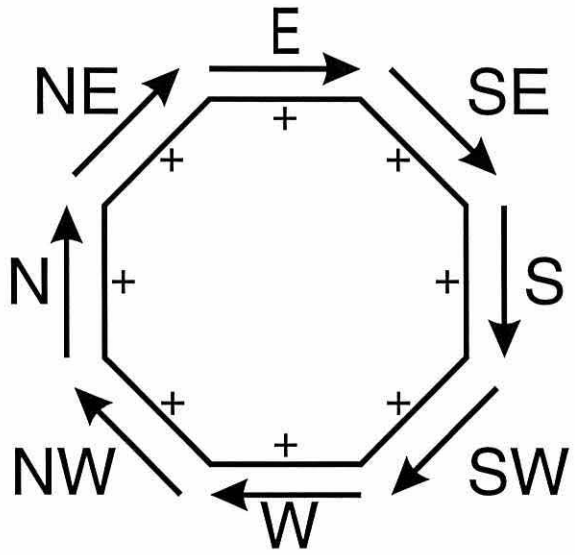


Figure 3-5: Edge direction. 8-way edge direction nomenclature. + = higher intensity pixels.

3.8 Texture

Texture is a property of the local spatial variation of image intensity, which is difficult to define quantitatively and qualitatively [38, 80, 58]. It is an area based property, hence its analysis depends on the size of the area and should be restricted to areas of relative uniformity. Qualitatively, texture measures properties such as smoothness and regularity. Analysis of texture may be used to segment an image into areas of uniform texture and identify image features. Three principal methods of texture analysis exist:

- **Statistical** texture measures can analyse coarseness. *Co-occurrence matrices* (or *joint amplitude histograms* of pairs of pixels) are one method of statistically analysing texture and examine the distribution of grey-levels and the relative positions of pixels with given grey-levels. Statistical analysis of these co-occurrences matrices yields the texture coefficients.
- **Structural** texture analysis considers the arrangement of image primitives, such as lines, and uses rules to build the texture from arrangements of the primitives.
- **Spectral** texture analysis considers the Fourier spectrum of an image. For example, the coarseness of a region is proportional to the spatial period of the texture, hence the shape of the Fourier spectrum should indicate the texture's coarseness. Although

this is true to a certain extent, difficulties can arise from changes in the spatial period and phase of texture pattern repetitions [80].

The analysis and synthesis of textures is used in chapter 9 to provide simulated skin and lesion textures.

3.9 Fourier Transform

The *Fourier transform* converts data from the spatial (or time) domain into the frequency domain and the *inverse Fourier transform* performs the opposite task of converting the frequency domain into the spatial domain [38]. The Fourier and inverse Fourier transforms for continuous functions are defined as:

$$\text{Fourier transform, } \mathcal{F}\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x)\exp[-j2\pi ux]\,dx \tag{3.2}$$

$$\text{Inverse Fourier transform, } \mathcal{F}^{-1}\{F(u)\} = f(x) = \int_{-\infty}^{\infty} F(u)\exp[j2\pi ux]\,dx \tag{3.3}$$

To apply a Fourier transform to discrete data (e.g. a digitised image) the *discrete Fourier transform* (DFT) is used:

$$\text{DFT, } F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)\exp\left[\frac{-j2\pi ux}{N}\right] \tag{3.4}$$

$$\text{Inverse DFT (IDFT), } f(x) = \sum_{u=0}^{N-1} F(u)\exp\left[\frac{j2\pi ux}{N}\right] \tag{3.5}$$

where N = the number of samples.

The number of multiplications and additions required to directly implement these equations is considerable and hence the transform is decomposed to give the fast Fourier transform (FFT), which is only of order $N\log_2 N$ compared with order N^2. This gives a considerable saving in computation over direct implementation, particularly when N is large.

The equations presented above were for the one-dimensional Fourier transform, but to apply it to an image the two-dimensional transform is required. The two-dimensional Fourier transform is separable, hence it may be computed by two applications of the

one-dimensional transform, i.e. by first taking transforms along the columns of the image and then along the rows of that result.

In image processing, the function $f(x)$ (the image) is real, but the Fourier transform is generally complex, i.e.:

$$F(u) = R(u) + jI(u) \tag{3.6}$$

which can be expressed in exponential form:

$$F(u) = |F(u)|e^{j\phi(u)} \tag{3.7}$$

where: $|F(u)| = \sqrt{R^2(u) + I^2(u)}$ and $\phi(u) = \arctan\left[\frac{I(u)}{R(u)}\right]$.

The *Fourier spectrum* of $f(x)$ is given by the *magnitude function* $|F(u)|$, $\phi(x)$ is the *phase angle* and the *power spectrum* or *spectral density* is given by the square of the spectrum, i.e.:

$$P(u) = |F(u)|^2 = R^2(u) + I^2(u) \tag{3.8}$$

3.10 Spatial Filtering

Spatial filtering uses *spatial masks* (which are called *spatial filters* and are restricted to a spatially limited area) to operate directly on the image. Two forms of it exist: linear and non-linear filters [38]. The output of a linear filter is a linear combination of pixels under the mask with the following form, which is the convolution of the mask with the image:

$$i_F(x,y) = w_1 i(x_1,y_1) + w_2 i(x_2,y_2) + \cdots + w_n i(x_n,y_n) \tag{3.9}$$

where w_j = filter coefficients,
$i(x_j, y_j)$ = grey-levels of the pixels under the mask.

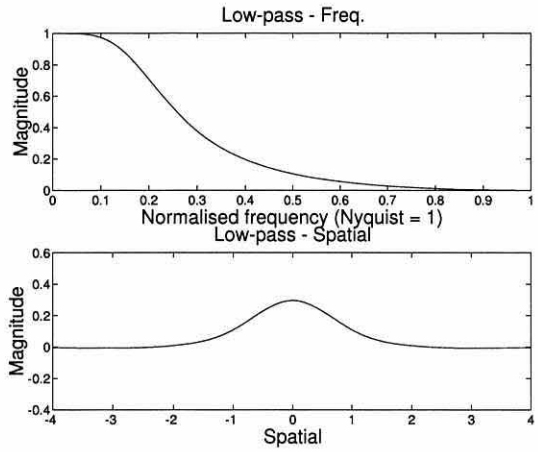
Non-linear filters also operate in a restricted neighbourhood, but generally directly use the grey-levels of the pixels in the neighbourhood and do not explicitly use coefficients. They do not use linear operations (i.e. addition and multiplication) to combine pixels in the neighbourhood and hence have outputs which are not linearly related to the input.

Three main types of linear filters exist:

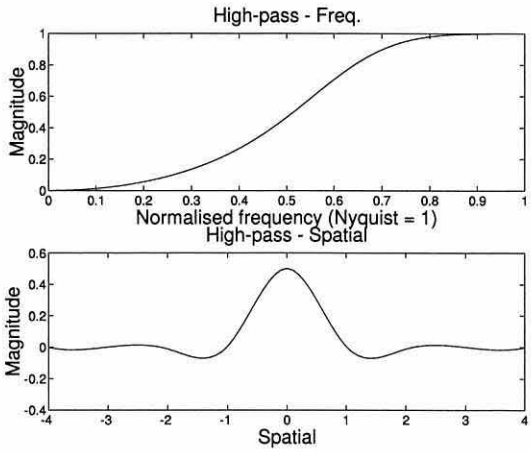
- **low-pass filters**: Attenuate high frequency components whilst leaving low frequencies untouched, which blurs the image.
- **high-pass filters**: Attenuate low frequency components, which emphasises edges and other sharp details but reduces the overall contrast.
- **band-pass filters**: Attenuate all but a selected band of frequencies.

The one-dimensional frequency and spatial domain shapes of examples of these filters are shown in Figure 3-6. Low-pass filtering, in the form of average filtering, is used and described in section 6.3.1. Average filtering replaces each pixel by the average of itself and the pixels within its neighbourhood. (Examples of low-pass and high-pass filtered images are shown in Figure 3-7 and the filters used are shown in Figure 3-8.) Another form of linear filtering that enhances the visual quality of edges is *unsharp masking*, which is described in chapter 6.

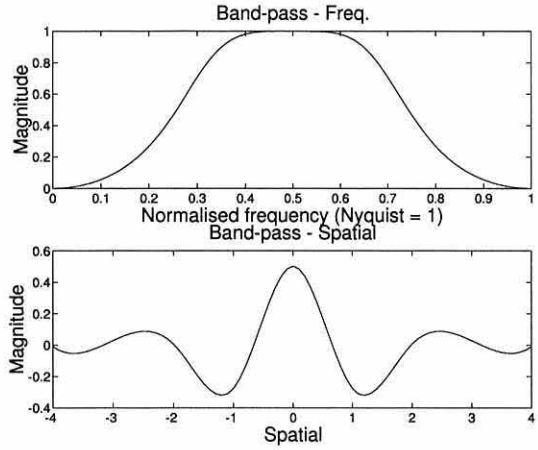
Non-linear *median filtering* can be used for noise reduction and other non-linear filters include the *max* filter, which finds the brightest points, and the *min* filter, which does the opposite. The median filter replaces each pixel by the median of the points in its neighbourhood (including the point itself), which removes isolated spike noise but preserves edges. The efficient application of median filtering and examples of its operation are given in section 6.3.2. *Iterative median* filtering is the repeated application of a median filter until either it has been applied a specified number of times or a specified condition has been met.



(a)

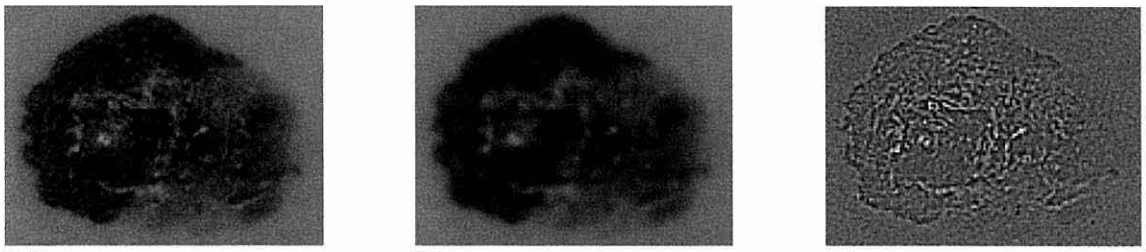


(b)



(c)

Figure 3-6: Example 1-D filter frequency and spatial shapes. (a) Low-pass filter. (b) High-pass filter. (c) Band-pass filter.



(a) (b) (c)

Figure 3-7: Image filtering examples on a 132 by 100 image. (a) Original image. (b) Low-pass filtered image. (c) High-pass filtered image.

$$\frac{1}{9}\times\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$$

(a)

$$\frac{1}{9}\times\begin{bmatrix}-1&-1&-1\\-1&8&-1\\-1&-1&-1\end{bmatrix}$$

(b)

Figure 3-8: Example spatial filtering masks, used in Figure 3-7. (a) Low-pass average filter. (b) High-pass filter.

3.11 Frequency Filtering

To apply a frequency filter to an image, the image is Fourier transformed, to convert it into the frequency domain, and the frequency domain image multiplied by the frequency domain filter transfer function and the inverse Fourier transform taken to yield the filtered image [80, 38]. The same types of linear filtering (low-pass, high-pass, etc.) can be achieved by frequency filtering, but one domain may be more efficient than the other depending on the size of the masks and the method used. Convolution of a spatial image by a spatial filter is equivalent to multiplication of the Fourier transformed image by a frequency filter. By Fourier transforming the filter a spatial filter can be converted into a frequency filter and similarly a frequency filter can be converted into a spatial filter by the inverse Fourier transform. For large filters, frequency multiplication may be quicker than spatial convolution, but it requires more space to store the intermediate complex results. For small filters, spatial convolution is quicker and simpler.

3.12 Image Histograms

The *histogram* of an image gives an estimate of the probability of occurrence of each grey-level and can be obtained by counting the number of pixels with each grey-level [38, 80, 76, 58]. The *normalised histogram* provides an estimate of the probability of occurrence of each grey-level:

$$h_N(k) = \frac{h(k)}{MN} \tag{3.10}$$

where k = grey-level,
$h(k)$ = number of pixels in the image with the grey-level k.

Manipulation of the histogram allows image enhancement including contrast and dynamic range improvement or thresholding. In a typical histogram some values of $h_N(k)$ may be zero, which means the full available intensity range is not being used, hence by reassigning the grey-levels the image may be enhanced. This histogram manipulation can be performed globally across the whole image, where a single histogram is used to control the modification of all the pixels, or locally, where a histogram is obtained for the neighbourhood of each pixel and that pixel's modification controlled by that local histogram.

An image may be thresholded using its histogram if assumptions are made about the shape of the histogram. When the image has two principal brightness regions the image histogram is the sum of the histograms of the two regions and may be bimodal, i.e. having two peaks corresponding to the two regions. In this case, the image may be segmented by placing a threshold between the two peaks. However in practice, it may be difficult to determine a suitable threshold from the unmodified histogram because it may have multiple peaks without clearly defined minima between the peaks.

Histogram equalisation (or *histogram linearisation*) is a widely used technique which automatically improves image contrast by reassigning the grey-levels to make the resulting histogram as flat as possible, i.e. with approximately the same number of pixels at each grey-level [38, 58]. Ideally, there would be $\frac{MN}{G}$ pixels at each grey-level, but in a quantised image the same number of pixels cannot be assigned to each grey-level as that would involve a one-to-many mapping of some pixels. However, the available grey-levels can be reassigned with the following transformation:

$$s_k = \text{round}\left(L \sum_{j=0}^{k} h_N(j) \right) \tag{3.11}$$

where $h_N(j)$ = normalised histogram for the original j^{th} grey-level,
s_k = new grey-level which is assigned to the original k^{th} grey-level,
L = maximum possible grey-level,
round(x) = round x to the nearest integer.

The summation in equation (3.11) is the normalised sum of number of pixels with grey-levels less than k, which when multiplied by the maximum possible grey-level (L) gives the number of pixels with original grey-level k or less divided by ideal number of pixels at each grey-level. This indicates the grey-level with which the original grey-level k is replaced. This spreads the original grey-levels across the whole of the grey-level range instead of just a small part of it. The effects of histogram equalisation on an image and its histogram are shown in Figure 3-9.

Histogram specification is an extension of histogram equalisation where the shape of the desired histogram is specified and may be used to highlight particular grey-level ranges [38]. Multi-dimensional histograms, such as bi-dimensional histograms, can be

created where more than one variable is being considered, e.g. RGB data [38]. Thresholding now becomes the problem of finding clusters of points in the histogram.

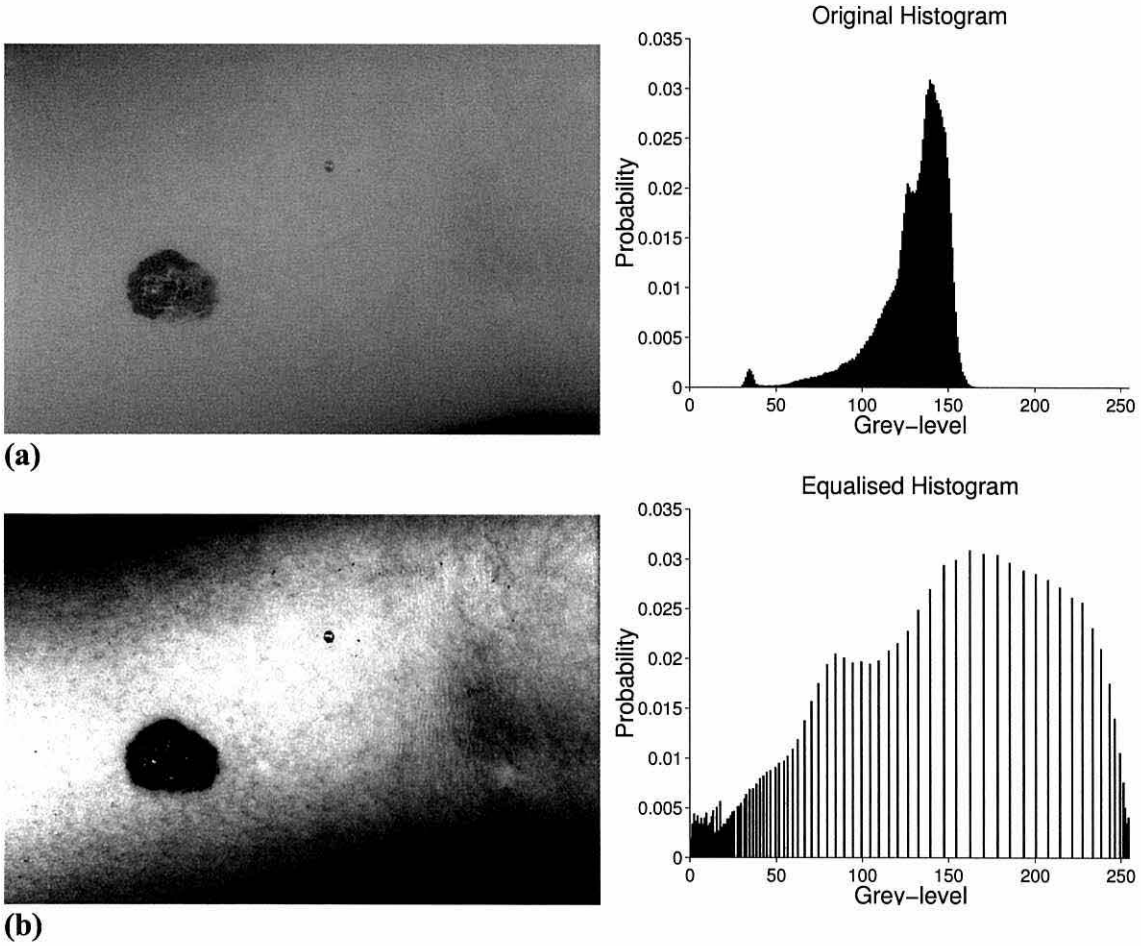


Figure 3-9: Image histogram equalisation. (a) Original image (744 by 487) and histogram. (b) Histogram equalised image and histogram.

3.13 Morphology

Mathematical morphology [38, 41, 78] is a useful tool for image analysis, filtering and enhancement which can be applied to binary images, grey-scale images and three or higher dimensional data. Morphology uses set theory to modify the spatial form or structure of regions in an image to, for example, perform operations such as non-linear smoothing of a region's outline, texture analysis and extracting boundaries or skeletons. Important binary morphological operations are described here and this binary morphology may be extended to process grey-scale images.

Dilation and *erosion* are the two basic operations which are the basis for most morphological operations. Binary dilation is defined as:

$$I_O = I_A \oplus I_{BS} = \left\{ x \left| \left[\left(\left(\hat{I}_{BS} \right)_x \cap I_A \right) \right] \subseteq I_A \right. \right\} \tag{3.12}$$

where I_A = input binary image,
I_{BS} = binary structuring element image,
I_O = output binary image,
$(I_{BS})_x$ = translation of I_{BS} by $x = (x_1, x_2)$,
$\hat{I}_{BS}$ = reflection of I_{BS} about the origin, i.e. rotation by 180° :

$$\hat{I}_{BS} = \left\{ x, y \middle| \, x = -x, y = -y \text{ for } x, y \in I_{BS} \right\}.$$

The set operations in dilation are similar to the arithmetic operations in convolution, in that the structuring element I_{BS} is rotated by 180° and moved over the input image. Binary dilation is used in chapter 8 for boundary quality measurement. Binary erosion is defined as:

$$I_A \Theta I_{BS} = \left\{ x \left| \left(I_{BS} \right)_x \subseteq I_A \right. \right\} \tag{3.13}$$

Dilation and erosion expand and shrink an image respectively and can be combined to provide *opening* and *closing*:

$$\text{Opening: } I_A \circ I_{BS} = (I_A \Theta I_{BS}) \oplus I_{BS} \tag{3.14}$$

$$\text{Closing: } I_A \bullet I_{BS} = (I_A \oplus I_{BS}) \Theta I_{BS} \tag{3.15}$$

Both opening and closing are *idempotent*, i.e. applying them more than once has no further effect on the results. Opening smoothes region outlines, removes thin protrusions and thin connections between regions. Closing smoothes region outlines, eliminates small holes inside regions and fuses small gaps between regions. The effects of dilation, erosion, opening and closing are shown in Figure 3-10. Binary closing is used in chapter 6 for simplifying a thresholded image.

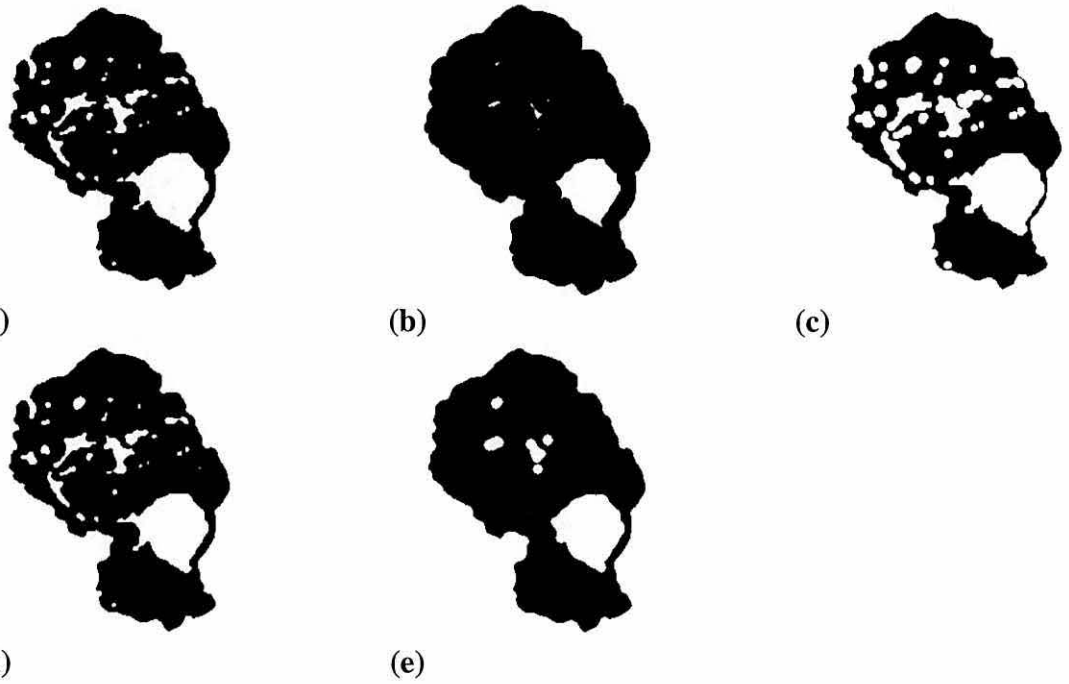


Figure 3-10: Binary morphological operations. (a) Original image (300 by 300). (b) Dilated with a 5 pixel disk. (c) Eroded with a 5 pixel disk. (d) Opened with a 5 pixel disk. (e) Closed with a 5 pixel disk.

3.14 Principal Component Transform

The principal component transform (PCT), which is also known as the Karhunen-Loève, eigenvector or Hotelling transform, is an adaptive statistical method of converting signals into the sum of orthogonal functions such that each signal can be represented by a minimal sequence of uncorrelated coefficients [38, 80]. The resulting coefficients are arranged in order of importance. It can align 2-D objects with their principal axis and hence remove the effects of rotation which can make their recognition easier. As the component which contains the greatest variance is the first one and is followed by components containing decreasing variance, data may be compressed by discarding the later components.

3.15 Colour

This section describes the common colour models that are mentioned in the literature review in chapter 4. *Colour models* represent colours as a point in a 3-dimensional *colour space* and can be generally divided into two categories: hardware orientated models and manipulation orientated models [38, 80, 56]. The hardware orientated models are typically used for image capture (e.g. by a video camera) and display (e.g. on a screen or printer), and include RGB (red, green, blue) space for colour monitors and video cameras; CMY (cyan, magenta, yellow) for colour printers; and YIQ (luminance,

inphase, quadrature) for colour TV broadcast. Colour spaces orientated towards colour manipulation are used in image processing and image creation (e.g. raytracing or animation) and include RGB, HSI (hue, saturation, intensity), YIQ and HSV (hue, saturation, value). A brief description of some common colour models is presented in Table 3-I.

Colour model	Comments
RGB (red, green, blue)	Used in colour monitors and cameras.
Normalised RGB (r, g, b)	These are also known as RGB chromaticity co-ordinates.
HSV (hue, saturation, value)	Hue = dominant wavelength or pure colour, such as red or orange. Saturation = amount of white mixed with the colour, e.g. white mixed with red becomes pink. Value = amount of black mixed with the colour, e.g. black mixed with orange becomes brown. HSV is a polar co-ordinate model of a rotation of the RGB model.
HSI/IHS (Hue, Saturation, Intensity) or HLS (Hue, Lightness, Saturation)	Hue and saturation represent the chromaticity of a colour and when combined with the brightness (intensity) characterise the colour. HSI is a deformation of HSV produced by moving the white point.
Lab system (L*, a*, b*)	Frequently used for colourimetric measurements. L*, a* and b* are correlated with brightness, redness-greenness and yellowness-blueness respectively.
SCT (Spherical Co-ordinate Transform)	Used by Umbaugh et al. [95] for automatic segmentation of colour skin for the detection of variegated colouring (discussed in section 4.4).
$\{I_1, I_2, I_3\}$	Used by Dhawan and Siscu [26] for segmentation of colour skin images (discussed in section 4.2).

Table 3-I: Colour models and transforms.

3.16 Graphics Operations

Graphical (or drawing) operations are useful in image processing and display, and this section describes the operations used in later chapters. Digitised images may be combined or masked by bitwise or logical operations such as AND, OR and XOR.

Breshnam's line drawing algorithm [32] uses only integer arithmetic to draw discrete lines and a modified form of it is used in chapter 8. It draws lines along their shortest axis, so a pixel is drawn for every step. Also, it always draws in the same direction, which is that of increasing values on the shortest axis, i.e. either left to right or top to bottom.

The interior of a closed contour (or region) may be filled by a *flood fill* algorithm (which is also known as *seed filling* or *contour filling by connectivity*) that starts from an interior *seed point* [76, 35]. The algorithm works outwards from the seed point filling all the 4-way connected points and uses a stack to allow filling of concave shapes. This algorithm will not fill any unconnected regions and requires a seed point. Neither of these constraints are present in the *parity check filling* algorithm [76], which fills regions line by line, using the fact that a line intersects any closed curve an even number of times. When the start of the line is known to be outside the region, counting the intersections with the contour can indicate which points are inside the region. For an odd number of intersections the point is inside the contour, otherwise it is outside. However, complications arise when the intersecting line is on a tangent to the contour and in this case each point of contact must be counted twice. Tangents and extrema (points which have only one 4-way connected neighbour on the contour) may be detected by examining the lines above and below the current line. This information allows filling of all full regions and the detection of the incorrect filling of regions which are not full. (The definition of a full region was given in section 3.2.)

3.17 Image Processing Software

There are a wide range of packages and toolboxes for software image processing and this section gives a brief description of uses, capabilities and limitations of the packages used in this research. The C programming language [8] was used to develop nearly all of the programs used in this research and is extensively used in image processing and in the packages described in this section. This language was selected because of familiarity and experience with it, which allowed easy and quick development of programs; its flexibility and speed, which make it suitable for image processing tasks and the availability of image processing packages written in it.

Some basic image processing operations were provided by the IPLIB 'C' library [84], which contains over 200 common image processing functions ranging from simple row, column and pixel manipulation to complex line extraction and component and texture recognition. The library is provided as source code, which required modification to make it suitable for the systems used. Whilst using this library, it was found to contain many bugs and the code was often difficult to follow. Consequently, many of the functions initially provided by this library were completely replaced with new 'C' code.

Mathematical morphology was described in section 3.13 and is used in later chapters. The initial development of programs involving morphology and some of the final programs used a morphology software package [78] written in 'C', which provides many morphological operations on both binary and grey-scale images. This software includes a 2-dimensional image morphology program, a 3-dimensional voxel image morphology program and a program for enhancement and noise reduction of 2-D images. It also provides script files which make it easier to use and a 'C' library for linking to new programs.

MATLAB (MATrix LABoratory) [67] is an interactive program which can perform numerical analysis, matrix computation, signal processing and graphics, and uses an automatically resizeable matrix as its basic data type. Algorithms can be quickly expressed in its own programming language and stored in M-files, which may form part of an application specific toolbox. (For example the signal processing toolbox [66].) 'C' code, in the form of MEX-files, may be integrated into MATLAB to provide support for new data formats (e.g. image files) and to implement algorithms which are slow or difficult to implement in MATLAB. Although it can be quicker to develop an algorithm in MATLAB than in a conventional programming language, such as 'C', the quantity of data and computation in image processing makes MATLAB inefficient for this application. This situation is due to MATLAB being an interpreted language and its use of single precision floats for all operations which reduces its speed, increases storage requirements and are not necessary for many image processing operations. Consequently, MATLAB was only used occasionally for the initial development of ideas.

Khoros [83, 82] is a software integration and development environment, which includes a visual programming language (Cantata), software development tools that extend the visual language and help in the creation of new applications, an interactive user interface editor, an interactive image display package, 2-D/3-D plotting, and an extensive suite of image processing, data manipulation, scientific visualisation, geometry and matrix operators. It is relatively quick and easy to set-up a sequence of image processing operations using the visual front-end (Cantata). However, although it is relatively easy to use existing Khoros tools, it is harder to develop new tools because of the complexity of the package. In addition, currently the front-end is unreliable and

some tools can be considerably slower than custom written software. It uses a specialised file format to store images and other data, which gives it considerable flexibility but does impose an overhead. As a consequence some of the individual image processing tools were used in this research and the front-end was used for small tasks.

3.18 Data Formats

Table 3-II shows the data formats commonly used in image processing, with their size and the number range which they can represent. The images used in this research are stored in unsigned bytes; intermediate data and results are stored in an appropriate format. Bytes provide the most efficient storage and calculation, but do not have the range and accuracy of single and double precision floats.

Name	Size (bytes/bits)	Range
unsigned byte	1/8	0 to 255
signed byte	1/8	-128 to 127
single precision float	4/32	$\pm 10^{-37}$ to $\pm 10^{38}$
double precision float	8/64	$\pm 10^{-307}$ to $\pm 10^{308}$

Table 3-II: Data formats.

3.19 Image File Formats

A wide range of file formats are available for storing images and these can be divided into two broad categories: *lossy* and *lossless* (or *information preserving*). The distinction between these categories is that the original image can be recovered exactly using a lossless format and approximately using a lossy format. However, images stored in a lossy format occupy considerably less storage space. The two main file formats used were chosen for their ability to store losslessly 256 grey-level images in the minimum space and for ease of data access. The simplest format used was the *Portable Grey Map* (PGM), which is used by the NetPBM toolbox [70]. This format stores images in an uncompressed format and is very easy to read and write. However, the raw images can occupy considerable disk space, hence the more efficient *GIF* format was used. The GIF format supports up to 256 colours, which may be monochrome or colour, and uses compression to reduce the file size. The GIFLIB 'C' source code library [28] was adapted to read and write these files. On a simple (binary) image, such as that shown in Figure 3-10(a), the GIF format can be considerably more efficient than PGM and this case only takes 1975 bytes compared with 90101. (Both of these figures include a small

overhead for storage of non-image information.) Even on more complex images a considerable saving can be achieved.

3.20 Conclusions

This chapter has provided the background needed to understand the literature review and the research presented in later chapters. It has, also, documented the selection of tools and programming languages. Some of the areas covered in this chapter will be expanded upon in later chapters where necessary.

Chapter 4

Image based Skin Cancer Analysis - Literature Review

4.1 Introduction

This chapter reviews previous work on the computerised analysis of skin cancer images. The publications covered are roughly classified into the following areas:

- **boundary detection and segmentation**: the segmentation of images to provide accurate lesion boundaries and regions within the lesion.
- **shape analysis**: the measurement of shape characteristics from the lesion boundary.
- **colour and texture analysis**: the use of colour and texture to provide directly diagnostic information and to improve the measurement of other factors.
- **three-dimensional analysis**: the determination of lesion three-dimensional shape and thickness.
- **diagnosis and prognosis**: the use of measurements to aid diagnosis and prognosis.

4.2 Boundary Detection and Segmentation

Determining a lesion's boundary or segmenting a skin image is very important as it is the starting point for many measurements such as shape, texture and colour. As many diagnostically important measurements directly or indirectly depend on its accuracy, reliability and consistency, it is important that it is insensitive to image rotation, capture conditions (e.g. lighting, camera set-up and position) and skin and lesion variations. To ensure consistency and allow a system to function without the presence of a human expert, boundary detection should be performed automatically. The number of different approaches devised shows the complexity and difficulty of consistent and accurate boundary detection.

The simplest method of segmentation, used by Cascinelli et al. [11] and White et al. [98], has the investigator drawing a line around the perceived boundary of the lesion. This approach takes time and the results may be operator-dependent, which make it unsuitable for inclusion in an automated system.

To overcome these problems, Claridge et al. [12] semi-automatically segmented monochrome lesion images by thresholding with the segmentations being confirmed by a clinician. They did not state the fate of images which the clinician considered to be incorrectly segmented. This simple thresholding approach is limited by requiring operator confirmation and is sensitive to variations in lesion and skin intensity, which may cause non-lesion regions to be detected or the lesion to be broken into separate regions.

A more sophisticated thresholding approach, which automatically identifies and separates the lesion from other thresholded regions, was published by Ercal et al. [30]. They used an adaptive colour transform to change RGB (Red-Green-Blue) colour space into one in which the lesion may be separated by thresholding from the background skin. They found the lesion and skin indistinguishable in histograms obtained from RGB data because these histograms did not exhibit bi-modality. An iterative median filter was used to reduce the effect of noise (caused by the presence of hair, flash reflections, etc. and variations in image quality due to lighting and film changes) and to improve lesion border visibility. By histogramming and approximate colour segmentation two windows inside and outside the lesion were identified, which provided the colour means and variances needed for segmentation. They tested a number of colour transforms, including the one used by Dhawan and Sicsu [26] which is described later, and found the following linear X transform allowed effective segmentation:

$$X = \omega_R R + \omega_G G + \omega_B B \tag{4.1}$$

where $\omega_R = \frac{v_R}{v_R + v_G + v_B}$, $\omega_G = \frac{v_G}{v_R + v_G + v_B}$, $\omega_B = \frac{v_B}{v_R + v_G + v_B}$

v_R, v_G, v_B = colour variances between tumour and non-tumour windows,

$$v_R = \sum_{i,j} \left(r_T(i,j) - r_{NT}(i,j)\right)^2, \quad v_G = \sum_{i,j} \left(g_T(i,j) - g_{NT}(i,j)\right)^2,$$
$$v_B = \sum_{i,j} \left(b_T(i,j) - b_{NT}(i,j)\right)^2$$

r_T, g_T, b_T = red, green and blue values in the tumour window,
r_{NT}, g_{NT}, b_{NT} = red, green and blue values in the non-tumour window, i.e. background skin.

This transform produced weights and results similar to the Principal Component Transform (PCT) [38] without the computational cost of the PCT. After applying this transform, the image was thresholded using:

$$T_0 = \frac{\mu_T + \mu_{NT}}{2} \tag{4.2}$$

where μ_T, μ_{NT} = mean of X within the tumour and non-tumour window.

The tumour boundary was found by contour tracing the tumour region formed by region growing from a known tumour point and removing all other areas. This prevented the inclusion of erroneous areas which were unconnected to the tumour region and any holes in the tumour region. They considered these boundaries to be generally rougher than hand drawn ones, hence each boundary was smoothed by creating a new boundary from a cubic spline with control points that were 10 points apart on the original boundary. The number of points between control points was heuristically obtained and hence did not change with image scale. The points between the control points were discarded and hence the smoothed boundary may not have been as accurate as it would have been had all the points been used in smoothing it. A dermatologist rated 82% of the boundaries from 61 colour images as "good" or "excellent". Melanoma images gave the worst performance of 77% and the best performance of 100% was for intradermal nevus, hence they considered the method to be good but not universal.

Another approach for finding lesion boundaries is to use the edges caused by intensity changes across the lesion boundary, which may be less sensitive to overall intensity

variations. Perednia et al. [77] used this approach by defining the boundary as the zero-crossing of the convolution of a fixed sized Laplacian of Gaussian (LoG) function with the image. Using monochrome images in which the lesions only occupied a small area, sub-pixel boundaries were obtained by quadratic interpolation of the zero-crossings of an LoG convolution on user selected regions of the image. The low resolution (of the order of 10 to 25 pixels per lesion diameter) was intended to overcome problems introduced by hair, wrinkles, shadows and other small-scale image "noise". The LoG function's scale was determined from the lesion size that was to be detected and the camera's point spread function (PSF). The camera's PSF was modelled by a Gaussian function whose size was set from the edges detected in the image of nine black squares. Measurements of camera PSF were intended to compensate for camera variations. Area and perimeter were calculated from the boundaries and the effects of changes in imaging conditions on these measurements determined. They found the errors due to repeated imaging, repeated focusing, and varying position to be small and the errors caused by image rotation to be negligible. This method was not fully automated as the area containing the lesion was user selected. The lesions were assumed to occupy only a small image area and consequently may not have contained sufficient pixels to make any other measurements, such as texture analysis. The LoG filter may remove noise, but at the same time its fixed size blurs the boundary reducing its accuracy. Although sub-pixel boundaries were obtained, the fixed scale of the LoG filter and the very small image area occupied by the lesions meant that the boundaries could not have been as accurate as those obtainable with a larger scale image.

An alternative method of searching for intensity changes, which can be applied to lesions occupying a larger proportion of the image, was published by Golston et al. [36]. They outlined an overall approach and described a radial search algorithm for finding lesion boundaries in monochrome images. Their overall approach combined boundaries found by a number of primitive boundary characteristics, such as colour, luminance and texture, using confidence levels. The basis for this approach was that each individual characteristic would not work reliably on all images, but a combined approach would be able to cope with all images. The four main steps of their radial search algorithm are:

1. **Determine flash threshold**. An adaptive flash threshold allows identification of areas that are bright due to flash reflections.

2. **Perform radial search**. Search for the first significant sustained jump in average luminance along a number of equally spaced radial lines. When searching for a sustained change, individual pixels with intensities greater than the flash threshold are ignored. This search uses two parameters: jump threshold (minimum increase in luminance considered significant) and sustained length (minimum number of consecutive pixels, excluding pixels classified as flash, that satisfy the jump threshold). These parameters are initially set high to minimise the chances of detecting false edges, but are relaxed and new searches performed if a boundary point is not found.
3. **Test boundary points for obvious errors**. The length of each radial line is compared with its neighbours and any lines which are significantly different from their neighbouring lines are ignored.
4. **Connect confirmed boundary points** into a closed curve using a quadratic B-spline curve.

The radial search starts from the lesion's approximate centroid, which is assumed to be the centre of the image. For the algorithm to function successfully, the lesion boundary must be radially connected, i.e. each radial line must cross the boundary only once, which they estimated to be violated in only 1% of images. In their twenty test images, seventeen were rated as "good" or "excellent" by a dermatologist. However, in a later publication [90] on asymmetry detection only a 54% success rate was achieved for the radial search algorithm.

Dhawan and Sicsu [26] published a segmentation algorithm which combines intensity, colour and texture information with the intention of providing better results than using each feature separately. Before attempting segmentation they applied the following transform to go from the captured RGB colour space to $\{I_1, I_2, I_3\}$ colour space (see Table 3-I):

Intensity, $$I_1 = \frac{R+G+B}{3} \tag{4.3}$$

Colour variation, $$I_2 = R-B \tag{4.4}$$

Deviation from the mean colour, $$I_3 = \frac{2G-R-B}{2} \tag{4.5}$$

The intensity segmentation on the I_1 component was performed using a modified multi-resolution pyramid [43] which combined inter-level and intra-level information. Regions with areas greater than a predetermined minimum area were extracted from this multi-resolution pyramid. The two I_2 and I_3 images were histogram equalised before their texture was examined using a new generalised co-occurrence matrix, which combined edge magnitude and orientation. Histogram analysis of a texture contrast measure, based on two sizes of overlapping windows, was used to label the pixels, with the intensity segmentation assisting in the labelling when the label could not be decided from the texture analysis alone. The two texture segmentations, obtained from the I_2 and I_3 images, were combined with an AND operation and very small regions merged with similar adjacent regions. The texture and intensity based segmentations were similarly combined to provide the final intensity, colour and texture based segmentation. Testing on simulated and real images showed that the extraction of all the regions with different colours and/or textures was possible on the simulated images. However, the region boundaries found on the simulated images were not as well defined as the input images. They considered the final segmentation of real images to be close to that of human perceived boundaries.

No "gold standard" exists for lesion boundary detection as there is no easily identifiable change between lesion and skin, which causes inter- and intra- observer variations. However, useful boundaries can be acquired provided that the methods are consistent. To test consistency repeated images of the same lesion under different conditions (lighting, rotation, scale, etc.) are required. These are not easily obtained in a clinical situation where taking repeated images may not be possible. Another approach to assessing performance is to use simulated images, which was used by Dhawan and Sicsu [26]. By specifying the boundaries during simulation, boundary detection methods can be tested. Dhawan and Sicsu's [26] simulated images represented colour shades and textures, but they did not specify how the images were produced.

The wide range of approaches taken illustrates the complexity and difficulty of consistent and accurate boundary detection. Manual boundary determination [11, 98] is time consuming and may not be consistent, making it unsuitable for an automated system. Semi-automated simple thresholding [12] is affected by lighting variations, which are difficult to control, and by lesion image intensity variations. More sophisticated colour

thresholding [30] should overcome the problem of identifying the lesion, but although giving good overall results gave worst performance on the most important type, melanoma. Another approach for finding boundaries is to use the edges caused by intensity changes across the lesion boundary, which was used by Perednia et al. [77] and Golston et al. [36]. Perednia et al. [77] used LoG edge detection, on small user selected parts of an image, to find the boundary. Their semi-automated method uses lesion images which may be too small for analysis of shape, colour and texture. However, they did determine that errors due to repeated imaging, repeated focusing, varying position and image rotation were small or negligible, which indicates that LoG edge detection may be suitable as part of a boundary detection method. Golston et al.'s [36] radial search algorithm initially gave an overall success rate of 85%, but in a larger test set only a 54% success rate [90]. Their method uses two assumptions which may not be valid: the lesion is centrally placed in the image and the boundary is radially connected. Dhawan and Sicsu's [26] complicated intensity, colour and texture segmentation algorithm gave boundaries in real images which were considered close to those perceived by humans. No detailed information on its performance was given, hence it is difficult to give an overall assessment.

4.3 Shape Analysis

Lesion shape is an important diagnostic factor which appears in the diagnostic checklists described in chapter 2. Two shape characteristics are considered to be particularly important: asymmetry and boundary irregularity, consequently research has concentrated on characterising these. It is known that dermatologists as well as patients find it difficult to agree on whether a lesion exhibits irregularity [40]. This makes computer based assessment important as the computer can provide quantitative and consistent measurements without being influenced by external factors such as the perceived diagnosis. To compare different sized lesions, possibly taken at different scales, it is necessary for shape measurements to be independent of size, rotation and position and be unaffected by the discrete nature of the boundaries being assessed.

White et al. [98] semi-automatically measured asymmetry by a bilateral symmetry index which varied between zero and one. A vertical line was drawn manually, by a physician, through the centre of the image and the absolute difference in the number of pixels on either side of the midpoint line summed for a series of horizontal lines one pixel apart

and then this total divided by the lesion's area. This method is not fully automated, as the boundary was manually obtained by the method described in section 4.2. The method does not attempt to find the axis of symmetry, hence is affected by image rotation and changes in the vertical line's position due to intra- and inter- person variations. They also assessed border irregularity using the following equation:

$$B_I = \frac{P}{2\sqrt{\pi A}} \tag{4.6}$$

where P = perimeter, i.e. length of the boundary,
A = area, i.e. number of pixels enclosed.

This produces a rotation and scale invariant ratio, which has a minimum of one for a circle, of the lesion's perimeter to the perimeter of a circle with the same area as the lesion. Tests using a very small number of malignant and benign lesions suggested that these indices, combined with colour measurements (described in section 4.4), would prove helpful in recognising in-situ melanoma.

Stoecker et al. [90] automated lesion asymmetry detection, by finding the axis of greatest symmetry from one of the principal axes through the centroid. (The principal axes are also used in calculating the bulkiness, which is discussed later in this section and in section 5.3.) The principal axis with the minimum moment of inertia was assumed to be the axis around which the shape was closest to being symmetrical. After finding this axis, the shape was filled and rotated to align this axis with the x-axis. This rotated shape was reflected in the x-axis and the absolute difference in area between the two sides found. The rotated shape was also separately reflected in the y-axis and another absolute area difference found. These two absolute area differences were used to produce an asymmetry index (A_I):

$$A_I = \frac{\Delta A_{\min}}{A} \tag{4.7}$$

where $\Delta A_{\min}$ = minimum of the two absolute area differences,
A = area.

The image rotation was performed by direct forward rotation (i.e. each pixel was mapped from the un-rotated image to the rotated image), which can cause omission of some pixels in the rotated image, hence the forward rotation was followed by a back-

ward rotation to correct this problem. This problem could have been overcome by rotating using a fast algorithm, described in Glassner [35], which uses shears together with anti-aliasing. Test lesion boundaries were obtained using either the radial search algorithm or manually. A threshold value of A_I was used to discriminate between symmetric and asymmetric lesions. The threshold was set to minimise the number of tumours which were misclassified. Using a test set of 85 skin tumour images, they compared this method with a dermatologist's assessment of asymmetry and found agreement in about 93% of the images.

Golston et al. [37] used the irregularity index:

$$B_I = \frac{P^2}{4\pi A} \tag{4.8}$$

where P = perimeter, i.e. length of the boundary,
A = area, i.e. number of pixels enclosed,

to classify lesion boundaries as irregular or regular. (Ercal et al. [29] also used this index in a neural network diagnosis method which is discussed in section 4.6.) This index is scale and rotation independent and theoretically has the lowest value of one for a circle. It is the inverse of the circularity index (or thinness ratio) described by Pratt [80] and is similar to the compactness ratio [38], which is defined as $C = \frac{P^2}{A}$. It is also the square of the irregularity index used by White et al. [98] (equation (4.6)), so should have similar performance to the earlier index. They chose an empirical threshold, which would depend on the implementation method, of 1.8 on the basis of initial test images; hence lesions with B_I greater than 1.8 were considered to be irregular and those with less to be regular. The method was tested on 60 boundaries found by the radial search algorithm (described in section 4.2) and an 87% agreement with classifications provided by a dermatologist obtained.

Golston et al.'s [37] and White et al.'s [98] irregularity indices only characterise overall irregularity. To obtain a better assessment of lesion irregularity Claridge et al. [12] measured it on two scales using the structural and textural fractal dimensions, which characterise the overall shape and detailed shape respectively. The fractal dimension was assessed by Flook's morphological dilation method [31] (described in chapter 5) and used two fixed points to determine where the textural dimension ended and the

structural dimension started. Claridge et al. [12] also assessed the overall shape by a 'bulkiness' measurement [68], which is a dimensionless shape factor that compares a shape's area with the area of an 'equivalent ellipse'. The 'equivalent ellipse' is an ellipse with the same moment of gyration (moment of inertia about the centre of gravity) as the original shape (described in section 5.3). The 'bulkiness' is smaller for benign lesions, as these tend to be more circular or elliptical in shape. Their test data consisted of monochrome digitised colour slides of 43 melanomas and 45 benign lesions, which were confirmed using histology. Using a test set of the silhouettes (produced by the method described in section 4.2) they achieved 91% sensitivity (malignant lesions classified as malignant or true positives) and 69% specificity (benign lesions classified as benign or true negatives) when the two fractal dimensions were combined with the bulkiness measurement. They found that bulkiness was more sensitive and more specific than the structural and textural fractal dimensions. In addition to the computer assessing the lesions, 119 dermatologists assessed the lesion silhouettes for irregularity and malignancy, and the dermatologists were found to be on average more specific but less sensitive than the computer assessment. The lesion silhouettes were normalised in size so the dermatologists could not use size to assist in determining malignancy. They concluded that irregularity alone was a strong discriminating factor for the studied melanomas and that the gross undulations (i.e. structural dimension) were more significant than small irregularities (i.e. textural dimension).

In order to automate the analysis of skin lesions, the shape assessment methods must not rely on human assistance. White et al.'s [98] bilateral symmetry index does not meet this criteria as the axis of symmetry was found manually. However, their bilateral symmetry index could be used with a method which finds the axis of symmetry automatically. Stoecker et al.'s [90] method automated the measurement of asymmetry and gave a good agreement with a dermatologist, hence is suitable for an automated system. White et al. [98] and Golston et al. [37] assessed overall lesion irregularity with Golston et al. [37] finding a good agreement with a dermatologist. Both of these methods compare lesions with circles, which results in lesions with elliptical, but smooth shapes, giving higher irregularity indexes than similar lesions with circular shapes. Claridge et al. [12] overcame this problem and distinguished between irregular shape and irregular border by using the 'bulkiness' [68], structural fractal dimension and textural fractal dimension. They concluded that the large scale irregularity (structural fractal dimension)

was more important in identifying melanomas than the bulkiness and small scale irregularity (textural fractal dimensions). Their assessment of large scale irregularity is more robust than the previous methods as it is unaffected by small errors in the lesion boundary position.

4.4 Colour and Texture Analysis

Colour and texture have been used to obtain diagnostic information (e.g. identification of variegated colouring) and have been suggested as methods of improving the measurement of other factors. Colour analysis would allow quantitative instead of qualitative statements, such as bluish and reddish, to be made concerning lesions. This section discusses how colour and texture analysis has been applied to skin lesion images; colour and texture are also used in some of the segmentation and diagnosis methods described in sections 4.2 and 4.6.

Texture analysis alone has not been used for skin lesion image analysis, but Dhawan and Sicsu [26] used texture in conjunction with intensity to segment the image. Dhawan [24] used texture and colour to identify pigmentation patterns. Texture may assist in segmentation, identification of variegated colouring and locating skin creases.

Cascinelli et al. [11] digitised colour slides (with standardised film and lighting) and examined the histograms, bi-dimensional histograms and Fourier spectra of the images in RGB (Red-Green-Blue) and HSV (Hue-Saturation-Value) colour spaces. By applying a threshold to the histograms the difference between melanoma and non-melanoma could be enhanced. The colour analysis was used by an expert system, which is described in section 4.6, to discriminate between melanoma and non-melanoma. White et al. [98] also examined the RGB histograms and obtained the means and standard deviations of the RGB data. Assessment using a very small test set suggested that these measurements would be helpful in recognising in-situ melanoma.

Herbin et al. [42] considered the quantitative assessment of colour in skin images. They digitised colour images to which global illumination correction (constant across the whole image) and local illumination correction (varying across the image) were applied by reference to a digitised image of a sheet of white paper. By using a chromameter, a comparison of colour assessment obtained from human observers, directly digitised

from a video camera and from a digitised colour slide was performed. They concluded that although the assessment from digitised colour slides was not as good as that obtained directly from the digitised scene, both were better than human observers. The colour quantisation from digitised images was found to be less precise than from a chromameter. However, the quantisation from digitised images was adequate and would permit other measurements such as area which cannot be performed with the pinpoint colour measurements of a chromameter.

Umbaugh et al. [95] described the automatic colour segmentation of skin lesion images and the detection of variegated colouring (VC) in the segmented images. The training and test images were labelled in blocks by a dermatologist to allow identification of specific features and masking of areas which were not of interest. The images were analysed in the non-linear spherical co-ordinate transform (SCT) colour space (Table 3-I), which represents colours by two angles and a brightness value (which was ignored). The two colour angles were quantised and the image resolution reduced to reduce the storage and processing requirements. The colour quantised and averaged image was filtered to remove small objects, fill in holes and smooth object outlines. After this, continuous blocks of the same colour were labelled to identify the different objects and a simple heuristic rule, based on the number of objects present and their colouring, used to identify variegated colouring. On a test set of 160 images this rule gave a 73% agreement with a dermatologist on the presence or absence of variegated colouring. The heuristic rule gave 47% sensitivity (VC correctly identified) and 85% specificity (absence of VC correctly identified). This method is not completely automated as it relies on the masking information provided by a dermatologist.

In a later publication [93], Umbaugh et al. replaced this simple rule by the use of automatic rule induction. An automatic rule induction tool using the ID3 algorithm was used to create a decision tree from the features extracted from a training set of images. The automatic induction rule using 18 measurements gave at best 60% sensitivity (VC correctly identified) and a specificity (absence of VC correctly identified) just under 100%. Using the heuristic rule on the test set used for the automatic rule induction only gave a 26% sensitivity and 78% specificity. Hence, the automatic induction method gave much better results than achieved by the simple heuristic rule used in [95].

The automatic colour segmentation algorithm was developed further in Umbaugh et al. [94] which compared the effectiveness of the spherical co-ordinate transform (SCT) with a two-dimensional split and a principal component transform (PCT) with a three-dimensional split (in six different colour spaces). Automatic rule induction (ID3) was used to create rules for identifying six features (tumour, crust, hair, scale, shiny and ulcer) which were considered important by a dermatologist. The PCT 3-D split was shown to be more effective than the SCT 2-D split for identifying the six features.

The effectiveness of neural network and rule induction methods in identifying VC were compared in a publication which used the SCT [27]. No overall conclusion of the most effective method was reached as the relative performance varied with the training and test set sizes. A peak accuracy of 80% was obtained using the neural network, which is not very impressive considering that 80% of the data set was non-variegated tumours.

Herbin et al. [42] showed that performance of colour identification from digitised images was sufficient to be used in an automated system. Identification of variegated colouring and six image lesion features (tumour, crust, hair, scale, shiny and ulcer) was shown to be possible by Umbaugh et al. in [95, 93, 94] and by Durg et al. [27].

4.5 Three-dimensional Analysis

Breslow thickness (tumour thickness) is a very important guide to prognosis and tumour volume may be even more significant [33]; both can only usually be determined after excision. To simplify tumour volume measurement, White et al. [98] semi-automated its measurement from a series of step-sectioned slides prepared from an excised lesion. The slides were digitised under a microscope and the tumour perimeter manually marked on a computer screen. A wire-frame model of the tumour was created from this, which allowed a three-dimensional view of the tumour and measurement of its volume.

This method only allowed volume assessment after excision, hence preventing its use in diagnosis. To allow diagnostic use of tumour volume Dhawan et al. [25, 23] constructed a 'nevoscope' which could capture multiple views of a lesion non-invasively without excision. Three images of the skin at 90° (vertical), 180° (glancing) and at 45° were digitised and used in a limited view computed tomography algorithm to reconstruct the three-dimensional shape of the lesion. The lesion was transilluminated by fibre optics

which pressed into the skin. In addition to recovering the thickness and volume, limited details of the internal structure were recovered by the computed tomography algorithm.

4.6 Diagnosis and Prognosis

After analysing a lesion image diagnosis and prognosis may be attempted. This analysis may assist human diagnosis by providing consistent, accurate and objective measurements [21], but this assumes that an expert is present and has been trained in the use of this additional information. Hence, research has attempted to construct systems which can provide diagnosis and prognosis. Even a simple system may provide assistance in estimating a patient's prognosis once a melanoma is diagnosed [98].

Cascinelli et al. [11] constructed the first prototype of a simple expert system, which seemed to be able to discriminate between melanoma and non-melanoma in most instances. They considered it difficult to split the reasoning behind the clinical evaluation into very simple rules which could be entered into the expert system. Their system was not sufficiently developed to allow quantitative assessment of its performance, but it gave "encouraging results".

Ercal et al. [29] used a multi-layer perceptron neural network, trained using back-propagation, to achieve above 80% correct classification of malignant and benign lesions in colour digitised skin lesion images. Fourteen features were used to discriminate between malignant melanoma and three types of benign lesion. The features assessed irregularity, asymmetry, RGB colour variances, relative RGB chromaticity of the tumour relative to the surrounding skin, the spherical colour space co-ordinates (SCT) and (L*, a*, b*) colour space co-ordinates (Table 3-I). All of these features required accurate detection of the lesion boundary, which was performed manually as they considered the accuracy of automatic boundary detection to be insufficient. The tumour size was not included in the feature set as the slides were obtained from various sources and photographed under varying magnifications. They found that including manually estimated approximate tumour sizes did not improve the diagnostic accuracy. Using training and test sets taken from 240 images they achieved a maximum sensitivity of 79.6% and corresponding specificity of 86.3%.

Kjoelen et al. [53] evaluated the performance of two artificial intelligence (AI) methods: rule induction by the ID3 algorithm and a polynomial network. The polynomial network combines the neural network concept with statistical regression techniques and consists of interconnected nodes that compute their outputs from a number of inputs. Sixteen features were extracted from digitised colour images and these included irregularity, asymmetry, variance of the RGB components, relative chromaticity of three colour components (relative to the colour of the surrounding skin), spherical colour co-ordinates (SCT), HSI colour co-ordinates (Table 3-I), elevation (greater than 2mm above surrounding skin) and area (tumour diameter exceeding 6mm). The last two features were estimated by a dermatologist. The overall performance of the rule induction method was found to be better than the polynomial network, but the absence of atypical moles in the training set dramatically improved the performance of both methods. Using 251 images a maximum sensitivity of 72% was achieved.

Dhawan [24] analysed the images obtained from the top view of lesions imaged by the 'nevoscope'. The prognostic and diagnostic features of lesion thickness, 3-D size, colour and margin, boundary and surface characteristics were combined with the patient's history (occurrence of melanoma or dysplastic nevi in the family, life style, skin type, etc.) to provide diagnosis and prognosis using knowledge databases, for early detection of melanoma, developed with the assistance of dermatologists and published case studies. The lesion was separated from the skin by a background subtraction algorithm, which used adaptive histogram-based region growing. After separation from the skin, the lesion colour was processed to obtain saturation and hue measurements used for spectral analysis in the high-level analysis system. After these pre-processing stages, the lesion image was segmented into regions by a modified multi-resolution pyramid and these regions merged and split to provide fewer sub-optimal but more "meaningful" regions. Rule-based low-level analysis was then used to label these regions with appropriate symbolic labels using features including area, average grey-level value and centroid. A pigment-pattern analyser considered the colour and texture of each region to create patches with consistent textures and colours. The knowledge database for the adequate classification of colour and texture (pigmentation patterns) commonly found in skin lesions, for the pigment-pattern analyser, was created from a training set. A rule-based expert system provided the high-level analysis by combining features from low-level segmentation, rule-based image analysis, historical patient data (such as the history of

melanoma incidence in the family) and current patient information (such as the lesion's site on the body) to provide probabilities of particular lesion types. Particular features, such as thickness, were assigned scores and these scores totalled and thresholded to detect melanoma. The final output contained the probabilities of particular lesion types, the reasoning and the feature measurements used. The performance was not assessed because the high-level diagnosis and prognosis expert system was not complete.

These publications show that the construction and use of explicit expert systems is difficult and their performance is poor [11, 24]. This is caused by the difficulty of obtaining and representing, in a form suitable for entry into a computer, the necessary complete dermatological knowledge. However, Ercal et al. [29] and Kjoelen et al. [53] showed that implicit knowledge acquisition and use by artificial neural networks, rule induction and polynomial network may be suitable for melanoma diagnosis. Explicit and implicit knowledge capture and use, in diagnosis, are discussed and compared in section 10.3.

4.7 Conclusions

The first stage in analysing a lesion image is segmentation to identify the lesion boundary accurately, consistently and reliably. This is important as diagnostically significant measurements of lesion shape, size, colour and texture depend on the boundary. The highly variable nature of skin images makes this problem difficult and hence many boundary detection methods [11, 98, 12, 30, 77, 36, 26] have been proposed. Assessing the performance of these methods is difficult as there is no "gold standard" for lesion boundaries. However, some of these methods [30, 36] have been visually assessed by dermatologists and their performance judged to be inadequate for an automated system, particularly on potentially fatal melanomas. In addition, researchers attempting to construct automated diagnostic methods for skin lesions [29, 53] have found it necessary to use manual boundary detection as they considered the performance of automatic boundary detection to be inadequate. The inadequate performance and the difficulty in assessing performance indicates two areas where research is required: new boundary detection methods and performance assessment methods. Consequently, a new boundary detection method using edge focusing was developed and is presented in chapter 8. To assess this boundary detection method, a method of producing simulated skin images and an area based measure were developed (chapter 9).

Lesion shape is diagnostically important [59, 60, 1, 63, 34, 40] and hence its automatic and reliable assessment is necessary for a computerised diagnostic system. Overall irregularity has been assessed by methods [98, 37] which compare the lesion with a circle. However, these methods assign higher measurements to lesions with elliptical, but smooth shapes, than similar lesions with circular shapes. This problem was addressed by Claridge et al. [12] using 'bulkiness' [68] and consequently this is extended upon in the next chapter. In addition to assessing overall irregularity, it is necessary to have measurements of large and small scale irregularity [11, 40, 63]. Claridge et al. [12] obtained this by fractal dimension analysis [62, 51]. However, they only assessed the accuracy of their measurements on two shapes and did not automatically separate the large and small scale irregularity. Both of these problems are addressed in the automatic method presented in the next chapter.

Chapter 5

Shape Analysis

5.1 Introduction

In chapter 2 the importance of lesion shape, in particular roundness, irregularity and asymmetry, as a diagnostic factor [59, 60, 1, 63, 34, 40] was described. In this chapter methods of assessing roundness (by bulkiness) and irregularity (by fractal dimension) are presented and illustrated with test shapes and real lesion shapes. The impact of noisy shapes on these shape analysis methods is assessed and conclusions drawn on the suitability of these methods for skin lesion diagnosis. In an automated diagnosis system these techniques would be used after the lesion boundary had been found, but are presented here as the bulkiness measurement is presented as part of a method of skin lesion location in chapter 6. The lesion boundaries used to illustrate this chapter were obtained using the edge focusing algorithm presented in chapter 8. However, in the initial development of these methods, boundaries were obtained by binary thresholding (using each image's average intensity) followed by removal of isolated pixels and boundary tracing.

5.2 Review of Shape Analysis Methods

There are many methods of analysing shape which lead to a range of shape descriptors [80, 38, 75]. To use existing human knowledge in the assessment of skin lesions, it is necessary that the shape measurements used can be related to human perception of shapes. For example, human diagnostic methods (section 2.7) identify irregularity and "roundness" as important diagnostic indicators. In addition, as the scale and rotation of the lesion shapes are variable the shape measurements must be scale (size) and rotation invariant.

A review of published methods of analysing skin lesion shapes was given in section 4.3. In this section, these methods and shape analysis methods used in other research areas are examined, and appropriate methods of assessing lesion shape chosen.

White et al. [98] and Golston et al. [37] assessed lesion irregularity using equations (4.6) and (4.8) respectively. Both of these assess overall irregularity by comparing the shape with an equivalent circle and are rotation and scale invariant. There are other methods of comparing a shape with a circle, such as the compactness ratio [80], but all of these methods have the disadvantage of assigning higher measurements to elliptical shapes than to similar circular shapes. This problem can be overcome by measuring bulkiness [68], as used by Claridge et al. [12], which gives a rotation and scale invariant index unaffected by the elliptical nature of some lesion shapes. A method for measuring bulkiness, which is an enhancement of Medalia's [68] method, is presented in the next section.

These methods of assessing shape give only an overall description of shape. In melanoma diagnosis, it is necessary to distinguish between large and small scale irregularity [11, 40, 63]. To assess both of these, it is necessary to use other methods. Fourier descriptors [38, 75] provide a method of assessing shape at a range of scales. In this method, the Cartesian co-ordinates of a boundary are considered as complex numbers, with the x and y co-ordinates represented as the real and imaginary parts respectively and these complex numbers Fourier transformed to give a frequency space representation of the shape. The original shape or an approximation can be reconstructed with the inverse Fourier transform. By examining particular frequencies, the shape can be assessed. To measure large and small scale irregularity using Fourier descriptors would require selection of an arbitrary frequency threshold to separate the two. Although Fourier descriptors can capture a great deal of information about a shape they are very difficult to relate to human perception of shape [85]. Fractal dimension analysis [62, 51] can characterise both large and small scale irregularity in a way which is better in this respect. It was used by Claridge et al. [12] to characterise lesion shape, and is explained and an automatic measurement method presented later in this chapter.

5.3 Bulkiness Measurement

Bulkiness is a dimensionless shape factor indicating the similarity of a shape to an ellipse, first described by Medalia [68] and used by Claridge et al. [12] to characterise melanoma outlines. Here, the method described by Medalia is modified slightly, to sort the boundary points into strips and to allow skipping of duplicated points. The co-ordinate output from lesion boundary detection methods is generally a continuous list in

the order in which the boundary is traced, i.e. co-ordinates which are adjacent in the co-ordinate list are spatially adjacent in the image. The original Medalia method required the co-ordinates to be arranged in strips parallel to the x-axis, hence the co-ordinates were sorted to re-arrange them into this state. After sorting, any duplicate points were skipped to prevent them from affecting the measurements. In addition to measuring bulkiness, the method determines the centroid and area of the shape and the aspect ratio (ansiometry), orientation and axes lengths of the equivalent ellipse.

Bulkiness (B_L) is defined as:

$$B_L = \frac{\text{area of the equivalent ellipse}}{\text{area of the original shape}} \tag{5.1}$$

where the equivalent ellipse has the same moment of gyration as the original shape.

The following description shows how to obtain the bulkiness, and is illustrated in Figure 5-1 with examples in Figure 5-2. To analyse the whole shape, it is processed in strips parallel with the x-axis, which allows the shape moments to be calculated in terms of the moments of these rectangular strips. To obtain the co-ordinates of these rectangles, the boundary co-ordinates are sorted into strips with increasing y and increasing x, i.e. sorted using:

$$(x_1, y_1) > (x_2, y_2) \quad \text{iff } (y_1 > y_2) \text{ or } (y_1 = y_2 \text{ and } x_1 > x_2) \tag{5.2}$$

Each rectangular strip's start and end x co-ordinates (x_{lo} and x_{hi}) are found by sequentially scanning the sorted co-ordinate list for pairs of co-ordinates with the same y value.

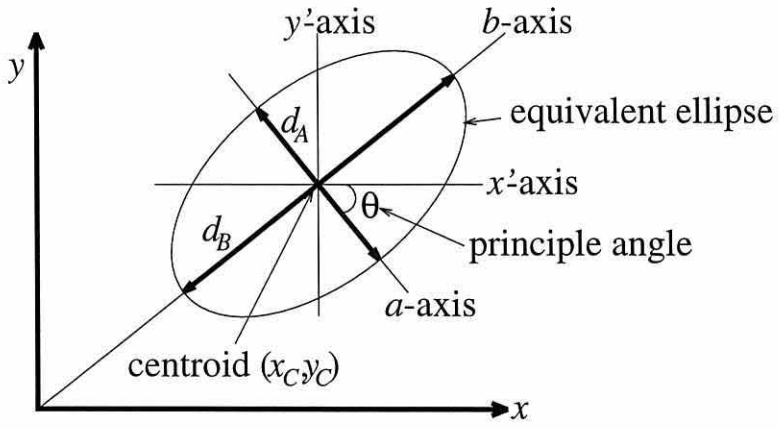


Figure 5-1: Equivalent ellipse.

In this process, duplicate points are skipped and horizontally adjacent points are amalgamated into one strip. After finding each strip's start and end co-ordinates, the calculations described in appendix B are performed to give the following:

Shape area: A
Centroid: x_C, y_C
Moment of inertia about the x'-axis parallel to the x-axis through the centroid: CM_{20}
Product moment about x' and y' axes: CM_{11}
Moment of inertia about the a-axis: MA_{20}
Radius of gyration about the a-axis: rA_{gy}
Radius of gyration about the b-axis: rB_{gy}

These yield the axis lengths of the equivalent ellipse and the principle angle, θ, between the a-axis and the x-axis:

$$d_A = d_S = 4rB_{gy} \tag{5.3}$$

$$d_B = d_L = 4rA_{gy} \tag{5.4}$$

Note: d_B is always larger than d_A.
d_L = Largest diameter of the equivalent ellipse.
d_S = Smallest diameter of the equivalent ellipse.

$$\theta = \arctan\left(\frac{CM_{20} - MA_{20}}{CM_{11}}\right) \tag{5.5}$$

Note: To prevent domain errors when CM_{11} or (CM_{20} - CM_{11}) is zero then θ is set to zero without computing the arctan.

Finally, the aspect ratio and bulkiness can be obtained:

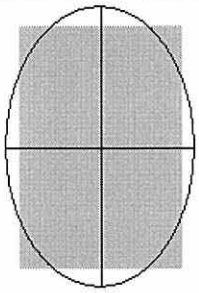
$$\text{Aspect ratio: } A_R = \frac{d_B}{d_A} \tag{5.6}$$

$$\text{Bulkiness: } B_L = \frac{4\pi\ rA_{gy}\ rB_{gy}}{A} \tag{5.7}$$

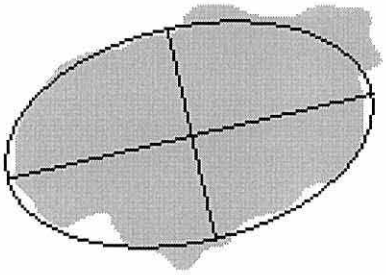
To find the effect of small perturbations in the boundary position on the measurements described in this chapter, the methods were tested with boundaries having added radial Gaussian distributed noise. To add the noise each point on the original boundary was displaced radially by a Gaussian distributed random distance and after removing any duplicate points any gaps in the boundary were bridged by cubic spline interpolation (as described in chapter 8). The standard deviation of the Gaussian distributed numbers was set as a percentage of the average radius of the equivalent ellipse (which is found during

the bulkiness measurement), which allowed the noise to scale with the size of the shape. This method models the effects of small displacements in the boundary, but does not attempt to model larger inaccuracies in the boundary. This allows testing of the effects of slightly misplaced boundaries, but does not consider the effects of the boundary detection method failing to find a reasonable lesion boundary, i.e. one near the true boundary. The bulkiness measurement is inherently scale invariant and hence the effects of changing the image scale were not tested.

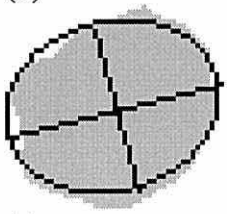
The effect of adding 0 to 5% radial Gaussian distributed noise to the shapes in Figure 5-2 is shown in Figure 5-4 and an example of each shape with 5% noise is shown in Figure 5-3. These graphs show that the average bulkiness increases as the standard deviation of the added noise increases, which is the expected result of the noise increasing the roughness of the boundary and so decreasing the similarity of the shape to an ellipse. As the added noise increases the differences between shapes with the same level of noise added increases and hence the standard deviation of the bulkiness measurement also increases, which can be seen in Figure 5-4.



(a)

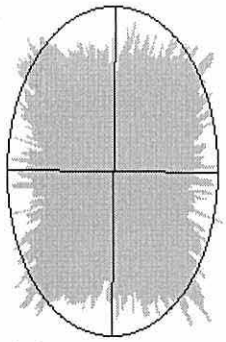


(b)

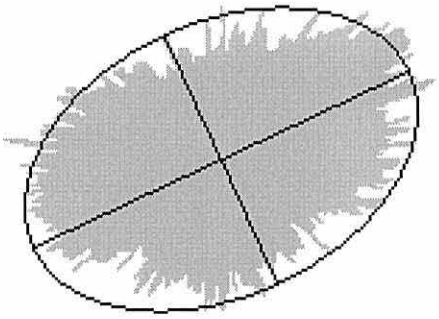


(c)

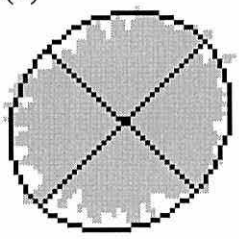
Figure 5-2: Bulkiness of examples shapes with their equivalent ellipses and principal axes. (a) A rectangle with a bulkiness of 1.047 and an aspect ratio of 1.5. (b) Melanoma outline with an area of 7098 and 338 boundary points, a bulkiness of 1.122 and an aspect ratio of 1.736. (c) Benign lesion outline with an area of 1090 and 116 boundary points, a bulkiness of 1.105 and an aspect ratio of 1.282. Note: the images are not shown at the same scale.



(a)



(b)



(c)

Figure 5-3: Example shapes from Figure 5-2 with 5% radially distributed Gaussian noise. (a) Noisy rectangle with a bulkiness of 1.744. (b) Noisy melanoma outline with a bulkiness of 2.406. (c) Noisy benign lesion outline with a bulkiness of 1.631. Note: the images are not shown at the same scale.

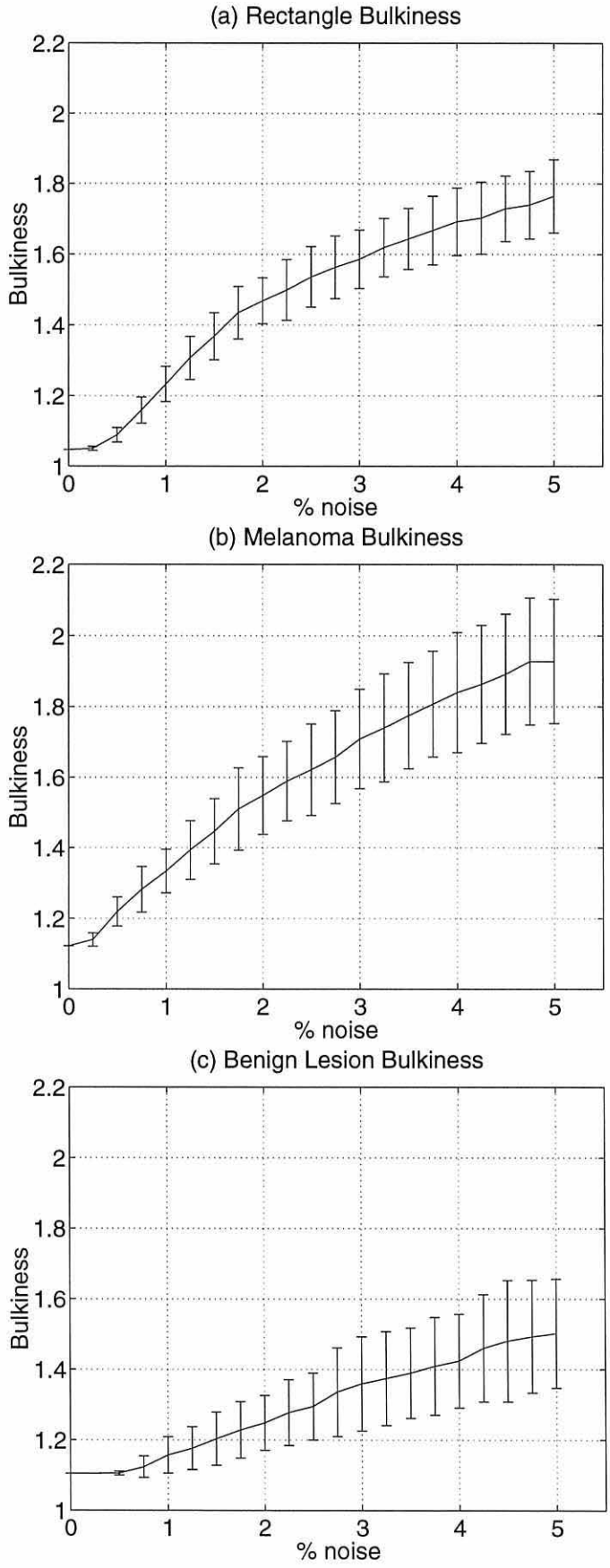


Figure 5-4: Graphs of bulkiness against % noise for the example shapes shown in Figure 5-2. (a) Rectangle. (b) Melanoma outline. (c) Benign lesion outline. The error bars are for ± the standard deviation of a set of 100 shapes.

5.4 Boundary Irregularity

A lesion's outline irregularity is a known diagnostic factor, a smooth outline being likely to have a better prognosis than a rough outline. The measurement of irregularity by determining the structural and textural fractal dimensions, which represent the large and small scale irregularity respectively, is described in this section.

The term fractal dimension was coined by Mandelbrot [62] to denote the space filling ability of a curve. In the two-dimensional boundaries considered here, it describes the plane filling ability of the boundary. The fractal dimension of a two-dimensional object varies from one for a circle to two for a disc (filled circle), which indicates it completely fills a two-dimensional plane. The boundary length of a fractal shape depends on the measurement scale (resolution) and this change in boundary length can be used to determine the fractal dimension. The fractal dimension is given by one minus the gradient of the Richardson plot [51], which is a log-log plot of the estimated boundary length against the measurement scale. At larger measurement scales, the gradient represents the structural fractal dimension and at smaller measurement scales, the textural fractal dimension. The boundary length estimate and measurement units are normalised with respect to the maximum Feret's diameter (f_D) [51], which makes the fractal dimension measurement dimensionless. Feret's diameter is the maximum width of shape in a given direction, hence the maximum Feret's diameter is the largest distance between any two points on the boundary.

To prepare the Richardson plot, it is necessary to calculate estimates of the boundary length at different measurement scales. A range of methods exist for performing this, including structured walk techniques and Minkowski sausage logic [51].

Structured walk techniques create an irregular polygon by stepping around the boundary of the shape [51]. The final vertex of the polygon is joined to the start vertex to form a closed contour. The length of this polygon's boundary gives the estimated boundary length at a particular scale. By changing the length of the steps used to place the polygon's vertices, the measurement scale is changed. The vertices can be manually placed, using a compass, by the three methods:

- **Inswing structured walk**: always pivot the compasses from the outside of the boundary inwards towards the first contact with the boundary.
- **Outward swing structure walk**: always pivot the compasses from the inside of the boundary outwards towards the first contact with the boundary.
- **Alternate swing structure walk**: alternate between swinging inwards and outwards.

These methods can produce different boundary estimates, with the alternate swing method leading to an intermediate length between the inswing and outswing techniques. These techniques are suitable for manually measuring the boundary length and hence fractal dimension, but are not easy to automate. These three methods create polygons with equal length sides (excluding the last side), however there is no fundamental advantage in using equi-sided polygons, provided that the side length varies in an unbiased manner.

A simple unbiased method for placing the polygon's vertices is to move, around the boundary, a fixed number of points between each vertex. By changing the starting point, an average value for the boundary length at a particular scale can be obtained. The number of points stepped over, between each vertex, controls the measurement scale. This method is referred to, in this thesis, as "Structured walk with fixed steps". A slightly more complex method overlays the boundary with a series of parallel lines and uses the intersection of these lines as the vertices of the polygon, which is illustrated in Figure 5-5. It is implemented by searching the boundary co-ordinates for points with y co-ordinates equal to a starting value plus a multiple of the step size. This method is referred to as "Structured walk with parallel lines". Both structured walk methods create a

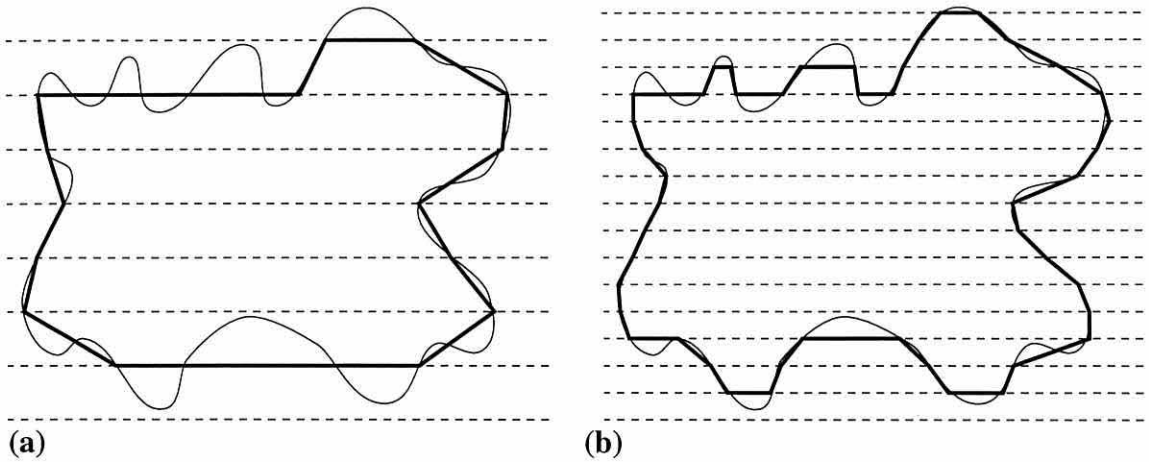


Figure 5-5: A structured walk. (a) At a large scale. (b) At a small scale.

number of boundary length estimates for a particular step size, hence the average and standard deviation can be obtained.

In Minkowski's sausage logic [51], each boundary point is surrounded by a disc (filled circle) and these circles merge to form a ribbon covering the boundary. The ribbon's area can be measured, without knowing how many circles there are, and this area divided by the diameter of the circles gives a boundary length estimate. The circle's diameter is the scale at which the boundary has been measured. Fractal dimension measurement using Minkowski's sausage logic can be implemented using morphological dilation with a disc [31]. Between each boundary length estimation the current boundary area is dilated by a small disc, which creates a series of boundaries with increasing area. The boundary length estimate is the area covered divided by the step size, which can be obtained from the number of dilations performed and the disc size. The dilation used in the original method is time consuming, hence was replaced by the placing of a discrete disc, with a diameter equal to the current step size, centred on each boundary co-ordinate. To ensure that the discrete disc's centre is in the middle of a pixel[1], i.e. it has a centre pixel, and it is horizontally and vertically symmetric, only odd step sizes can be used. This allows the disc's centre to be precisely placed on the discrete boundary. The discrete disc is a discrete approximation of a filled circle, hence its area is not πr^2 which makes the boundary estimate, particularly at small step sizes, inaccurate. To compensate for this the disc area was measured and used to estimate the diameter of the discrete disc (disc diameter = $2\sqrt{(\text{meas. area}) / \pi}$). This improved the boundary estimates at small step sizes. This method is referred to as "Flook's method" [31].

In fractal dimension measurement, the boundary length is estimated over a range of step sizes. For the first structured walk method the largest usable step size is one that reduces the shape to a triangle, which corresponds to the number of points in the boundary divided by three. For the structured walk with parallel lines, the largest usable step size is equal to the shape's height. Kaye [51] recommended using a maximum step of $0.3f_D$ to avoid the inaccurate results which can come from examining boundaries at large scales, caused by the polygon being a poor approximation of the shape in the structured walk methods. For a reasonable size shape, the number of potential step sizes is considerable.

[1] For example, the central pixel of a disk of 5 pixels diameter is at (3, 3). A disk of 4 pixels diameter does not have a central pixel.

At each of these a number of boundary estimates may be made, each requiring significant computation and hence to reduce the computation required the following three methods of selecting the integer step sizes were tested:

1. Linear steps: All values between the minimum and maximum step sizes inclusive.
2. Power of two steps: All powers of two between the minimum and maximum step sizes inclusive.
3. Logarithmic steps at 15 points per decade: Logarithmically (base 10) spaced points at 15 points per decade between the minimum and maximum step sizes inclusive.

After creating the Richardson plot, its gradient is measured to find the fractal dimensions. Claridge et al. [12], who measured the fractal dimension of lesion shapes, observed that the breakpoint between the textural and structural dimensions always occurred, for their data, within a small range of step sizes. Hence, they used values below the minimum of this range to obtain the textural dimension, those above the maximum to obtain the structural dimension and ignored the data inside the range. To find the fractal dimensions automatically, a multi-segment line can be non-linearly fitted to the Richardson plot using all of the data. This allows the breakpoint between the textural and structural dimensions to be determined automatically. A non-linear fit is required as an N_S segment line, of the form shown in Figure 5-6 and the following equations, is not linearly dependent on its $2N_S$ parameters.

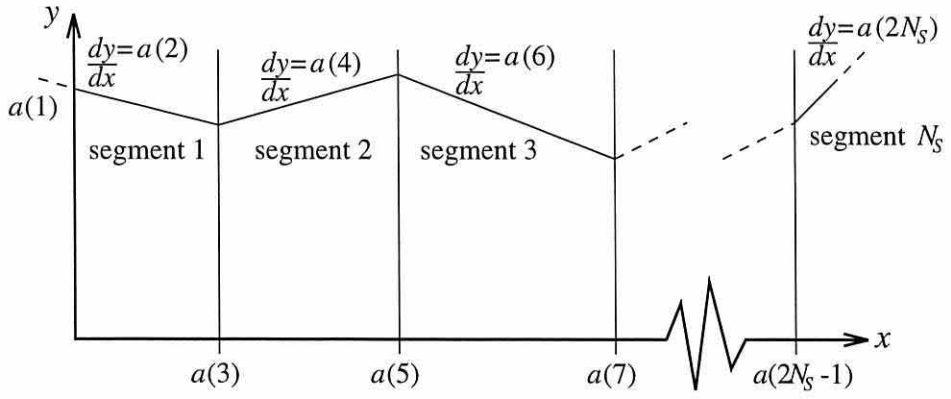


Figure 5-6: Multi-segment line.

For the first segment:

$$y = a(1) + a(2)x \tag{5.8}$$

$$\frac{dy}{da(1)} = 1, \quad \frac{dy}{da(2)} = x, \quad \frac{dy}{da(3)} = \ldots = \frac{dy}{da(2N_s)} = 0 \tag{5.9), (5.10), (5.11}$$

For the n^{th} segment:

$$y = a(1) + \sum_{i=1}^{n-1} a(2i)\big(a(2i+1) - a(2i-1)\big) + a(2n)\big(x - a(2n-1)\big) \tag{5.12}$$

$$\frac{dy}{da(1)} = 1, \quad \frac{dy}{da(2)} = a(3), \quad \frac{dy}{da(2n)} = x - a(2n-1) \tag{5.13), (5.14), (5.15}$$

$$\frac{dy}{da(2n+1)} = \ldots = \frac{dy}{da(2n)} = 0 \tag{5.16}$$

For i = 3 to $2n$-1 odd only: $\dfrac{dy}{da(i)} = a(i-1) - a(i+1)$ (5.17)

For i = 3 to $2n$-1 even only: $\dfrac{dy}{da(i)} = a(i+1) - a(i-1)$ (5.18)

where $a(1)$ = y intercept of segment 1,
$a(2n)$ = gradient of segment n,
$a(2n\text{-}1)$ = x co-ordinate of start of segment n,
$\dfrac{dy}{da(i)}$ = differential of y with respect to the i^{th} parameter.

The Levenberg-Marquardt method [81] of non-linear least-squares fitting was used to fit the multi-segment line. This is an iterative method which requires an initial estimate of the function parameters ($a()$), an estimate of the standard deviation of each point and the ability to calculate the differential of y with respect to each of the parameters at a particular x. The following equations set these initial parameters to give each line segment approximately the same number of points and set the breakpoints, between line segments, to coincide with measured data points and the line segment gradients using existing points.

$$a(2)=\frac{y\left(\frac{n_p}{N_S}\right)-y(1)}{x\left(\frac{n_p}{N_S}\right)-x(1)},\quad a(1)=y(1)-x(1)a(2) \qquad (5.19), (5.20)$$

and for $i = 2$ to N_S

$$a(2i-1)=x\left(\frac{n_p i}{N_S}\right),\quad a(2i)=\frac{y\left(\frac{n_p(i+1)}{N_S}\right)-y\left(\frac{n_p i}{N_S}\right)}{x\left(\frac{n_p(i+1)}{N_S}\right)-x\left(\frac{n_p i}{N_S}\right)} \qquad (5.21), (5.22)$$

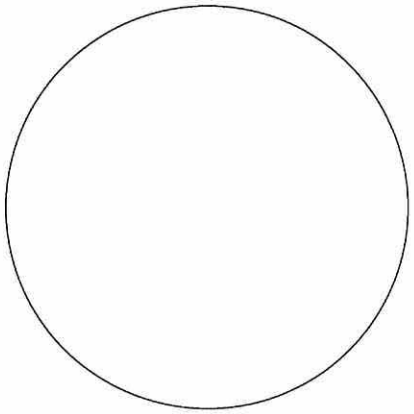
where n_p = the data points, i.e. the number of fractal dimension measurements.

To test the accuracy and measurement time of the fractal dimension (FD) measurements, shapes with known fractal dimensions are required, hence lesion shapes cannot be used as their FDs are not known. Claridge et al. [12] used a circle (FD = 1) and a Triadic Koch curve [62] (FD = 1.262, shown in Figure 5-7(b)). With these shapes, they found their method had an average error of 2.3%. However, this error measurement was obtained using only two shapes, with the Triadic Koch curve having a fractal dimension much higher than that of lesion shapes. To discriminate between benign and malignant lesions they used a structural fractal dimension of 1.070 and a textural fractal dimension of 1.075. Consequently, it is important to assess the accuracy using shapes with FD's nearer to these values than the Triadic Koch curve to obtain a better idea of the accuracy. In addition, to obtain a more reliable assessment of accuracy it is necessary to use more test shapes.

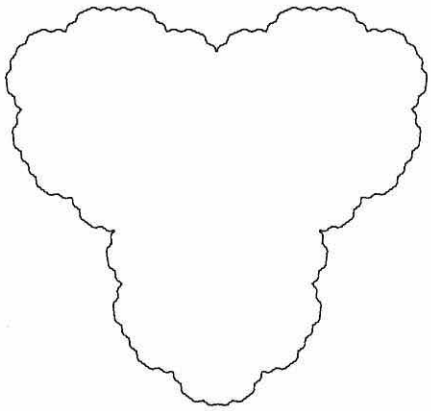
Both of these problems can be addressed by using the five test shapes, shown in Figure 5-7, with FD's between 1 and 1.262. These shapes have the same fractal dimension at all measurement scales, so only one fractal dimension can be measured. When measuring a single fractal dimension, a straight line linear least square fit [81] was used, instead of non-linear fitting of the multi-segment line previously described.

The last four shapes shown in Figure 5-7 are Koch curves [62] and were constructed by the method described here, which is illustrated in Figure 5-8 by a Triadic Koch curve. The shape construction starts with the initiator (Figure 5-8(a)), which controls the overall shape. To create a first approximation of the Koch curve, each side of the initiator is replaced by a scaled, rotated and translated version of the generator (Figure 5-8(b))

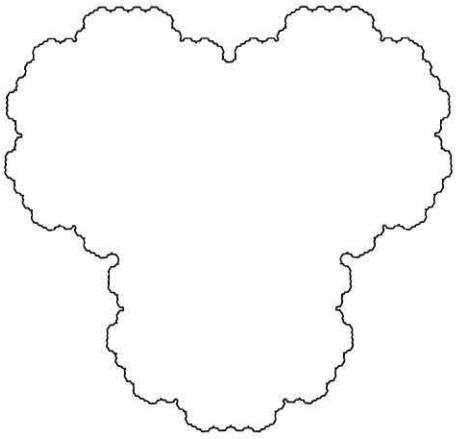
to give the shape shown in Figure 5-8(c). This process of replacing each side of the current shape with the generator is repeated an infinite number of times to create a Koch curve. Figure 5-8 shows the first four iterations of this process. To create an approximate curve, which is suitable for assessing the FD measurement, this process is stopped once the side length is less than the available resolution.



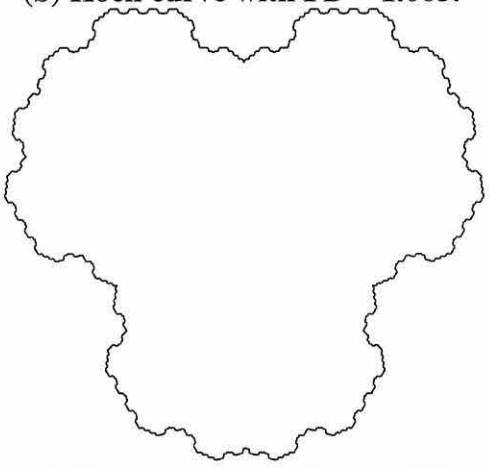
(a) Circle with FD = 1.000.



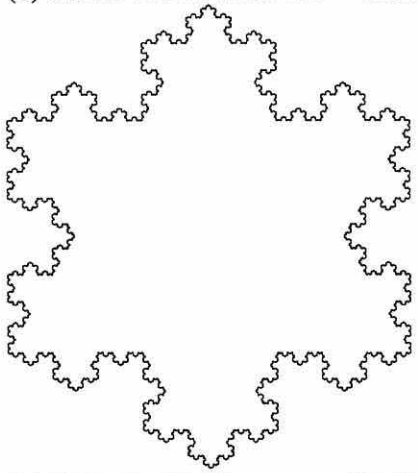
(b) Koch curve with FD = 1.065.



(c) Koch curve with FD = 1.131.

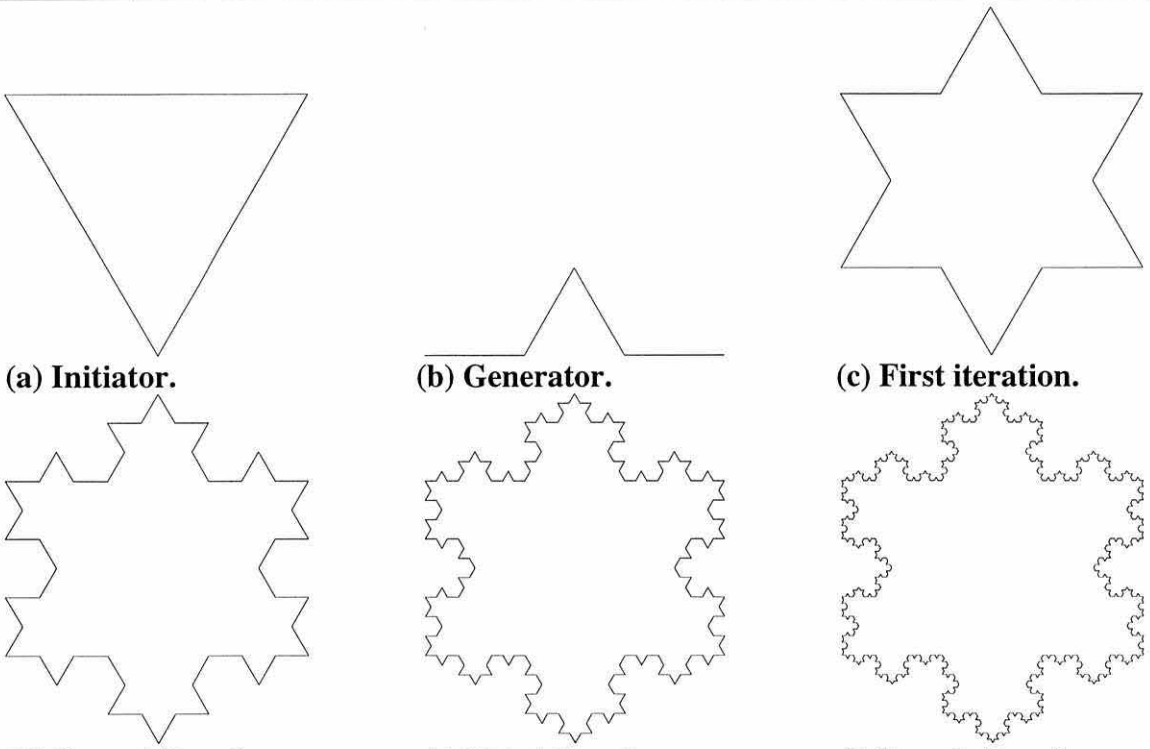


(d) Koch curve with FD = 1.196.



(e) Triadic Koch curve with FD = 1.262.

Figure 5-7: Fractal dimension analysis test shapes. Note: The shape boundaries are drawn 3 pixels wide.



(a) Initiator. **(b) Generator.** **(c) First iteration.**

(d) Second iteration. **(e) Third iteration.** **(f) Fourth iteration.**

Figure 5-8: Koch curve generation.

The fractal dimension of a Koch curve is controlled by the generator, as this controls the increase in boundary length with increasing resolution, and is given by [62]:

$$f_D = \frac{\log N}{\log 1/r} \tag{5.23}$$

where N = the number of segments in the generator. For the Triadic Koch curve, $N = 4$.
r = the ratio of the length of the generator segments to the length of the whole generator. For the Triadic Koch curve, $r = 1/3$.

Equation (5.23) gives the fractal dimension for a shape with a known generator, however by specifying either N or r the method presented here can create a generator which gives a specified fractal dimension. To allow a generator, of the shape shown in Figure 5-9, to be easily constructed by a regular method, the number of segments (N) is set to an even number. Once the number of segments is fixed, the fractal dimension is controlled by the ratio (r) of the generator length (g_L) to the segment length (s_L). For convenience, the generator length is fixed, at for example one, hence the segment length can be found by:

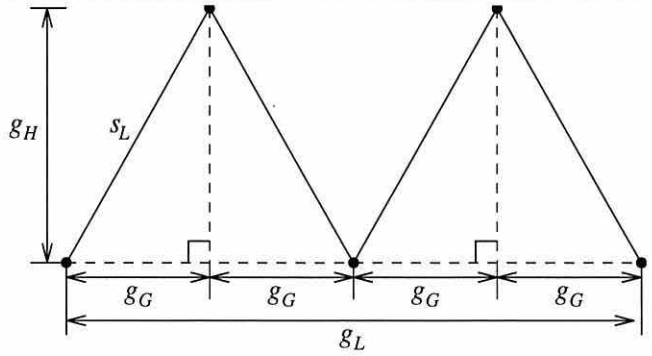


Figure 5-9: Generator construction, for a 4 segment generator.

$$s_L = \frac{g_L}{\exp\left(\frac{\log N}{f_D}\right)} \tag{5.24}$$

where g_L = generator length,
N = number of segments in the generator,
f_D = desired fractal dimension.

From the segment length and the number of segments, the remaining dimensions can be calculated and the generator constructed in a shape similar to Figure 5-9:

$$g_G = \frac{g_L}{N} \tag{5.25}$$

$$g_H = \sqrt{s_L^2 - g_G^2} \tag{5.26}$$

where g_L = length of the whole generator,
g_H = generator height.

Before analysing the shapes just described, their boundaries must be converted to the same form as real lesion boundaries. This conversion was performed by rounding the boundary co-ordinates to the nearest integer and removing any points which were not on the shape's exterior by contour tracing an image of the shape. This process made the boundary similar to those from real images, which allowed testing on more realistic data.

Before comparing the three methods, it is necessary to decide on the size of the test shapes and the starting and stopping measurement scales. As the test shapes are discrete approximations, the fractal dimensions will not exactly match the theoretical ones. To minimise this difference larger scale test shapes can be used, but they bring the penalty of larger computation times. Hence, one shape (the Triadic Koch curve) was tested at a

number of sizes (250 to 1500 in steps of 250), with all the three methods, to assess the effects of changing the shape size and the effects of changing the starting scale. The size of the Triadic Koch curve was set by the side length of the initiator. As the shape's size increased the accuracy of the fractal dimension measurement increased, but the computational time also increased. Hence, it was necessary to find a compromise between shape size and computation time. By examining the Richardson plots of the shapes, it was possible to estimate the minimum usable step size. Below the minimum step size, the downward gradient of the Richardson plot decreased (i.e. the fractal dimension decreased) due to the boundary becoming more Euclidean. On the basis of these initial tests, it was decided to use an initiator side length of 1000 and a minimum step size of 3% of the maximum Feret's diameter.

Table 5-I presents the results of testing the three fractal dimension analysis methods with the three step size selection methods on the five test shapes, shown in Figure 5-7. Flook's method (method 3) gave the best overall performance with all step selection methods, except with the "power of two steps". The slightly greater accuracy of method 2 ("structured walk with parallel lines") using the "power of two steps" over method 3 is not significant as the measurement is only based on, in these examples, three or four points. The very small number of points used in the "power of two steps" method makes it unusable when the textural and structural dimensions are required. This method was included in the tests to show the effect of using a very small number of points which requires very little computation.

These test results compare favourably with the errors obtained by Claridge et al. [12]. They obtained errors of 2.0% and 2.6% for a circle and Triadic Koch curve respectively, compared with errors of 0.5% and 0.4% for the method selected here. They also used Flook's method [31], but their implementation was different in using repeated dilations by a small disc. It is not possible to compare the results for the three shapes with FDs between 1.065 and 1.196 with Claridge et al. [12] as they only used two test shapes. When using only two test shapes, it is difficult to assess the measurement accuracy, especially when neither of the shapes has FDs similar to those they measured for melanomas and benign lesions.

On the basis of these tests "Flook's method" with logarithmic step sizes, at 15 points per decade, was selected as the most appropriate method for assessing skin lesion shape.

Figure 5-10 shows Richardson plots, for a malignant and benign lesion, created using "Flook's method". The test shapes were higher resolution versions of those shown in Figure 5-2(b) & (c). Structural and textural fractal dimensions were found using the non-linear fitting method presented earlier.

The effects of adding radial Gaussian distributed noise to these shapes is shown in the graphs in Figure 5-11. (The addition of radial Gaussian distributed noise was described in section 5.3.) These show the textural fractal dimension increasing as the size of the added noise increases, which is the expected effect of the noise increasing the small scale variability of the shape. The structural fractal dimension also increases, although not as much as the textural dimension. As the added noise increases the differences between shapes with the same level of noise added increases and hence the standard deviation of the fractal dimension also generally increases. Theoretically, the structural fractal dimension should be unaffected by the addition of small scale Gaussian noise as there is no change in the large scale shape. However, these tests indicate that, in this method, the structural dimension is affected by small scale Gaussian noise. Another problem is the difficulty of assessing its accuracy with test shapes, with known fractal dimensions, similar to lesions. In this thesis, this problem has been addressed with a method which creates shapes with a specified fractal dimension. Although, these shapes have fractal dimensions similar to lesions, they do not have overall shapes similar to those of lesions. To obtain lesion boundaries with sufficient resolution for measuring boundary length over a reasonable scale range, it was necessary to use high resolution images. In practical system, it may be difficult to use images of such a high resolution because of the capture, storage and processing requirements. These problems indicate that fractal dimension measurements may not be suitable for assessing lesion shape. The fractal dimension measurement is inherently scale invariant, as it scales with the maximum Feret's diameter and hence the effects of changing image scale were not tested.

Shape	Fractal dimension						
	Theoretical	Method 1: Structured walk with fixed steps					
		linear steps		power of 2 steps		log base 10 steps	
		value	% error	value	% error	value	% error
Circle	1.000	1.005	0.5%	1.005	0.5%	1.004	0.4%
Koch curve 1	1.065	1.047	1.7%	1.044	2.0%	1.045	1.9%
Koch curve 2	1.131	1.069	5.5%	1.077	4.8%	1.070	5.4%
Koch curve 3	1.196	1.070	10.5%	1.071	10.5%	1.071	10.5%
Triadic Koch curve	1.262	1.208	4.3%	1.207	4.4%	1.209	4.2%
Average % error			4.5%		4.4%		4.5%
Shape	Fractal dimension						
	Theoretical	Method 2: Structured walk with parallel lines					
		linear steps		power of 2 steps		log base 10 steps	
		value	% error	value	% error	value	% error
Circle	1.000	1.066	6.6%	1.059	5.9%	1.057	5.7%
Koch curve 1	1.065	1.108	4%	1.097	3%	1.101	3.4%
Koch curve 2	1.131	1.118	1.1%	1.121	0.9%	1.113	1.6%
Koch curve 3	1.196	1.119	6.4%	1.115	6.7%	1.109	7.3%
Triadic Koch curve	1.262	1.266	0.3%	1.261	0.1%	1.265	0.2%
Average % error			3.7%		3.3%		3.6%
Shape	Fractal dimension						
	Theoretical	Method 3: Flook's method					
		linear steps		power of 2 steps		log base 10 steps	
		value	% error	value	% error	value	% error
Circle	1.000	1.005	0.5%	1.006	0.6%	1.005	0.5%
Koch curve 1	1.065	1.045	1.9%	1.044	2.0%	1.045	1.9%
Koch curve 2	1.131	1.076	4.9%	1.077	4.8%	1.077	4.8%
Koch curve 3	1.196	1.078	9.9%	1.078	9.8%	1.078	9.8%
Triadic Koch curve	1.262	1.257	0.4%	1.257	0.4%	1.256	0.4%
Average % error			3.5%		3.5%		3.5%

Table 5-I: Comparison of theoretical and measured fractal dimensions, for the test shapes in Figure 5-7. Note: The measured fractal dimensions will not exactly match the theoretical ones, as discrete approximations of the shapes were used.

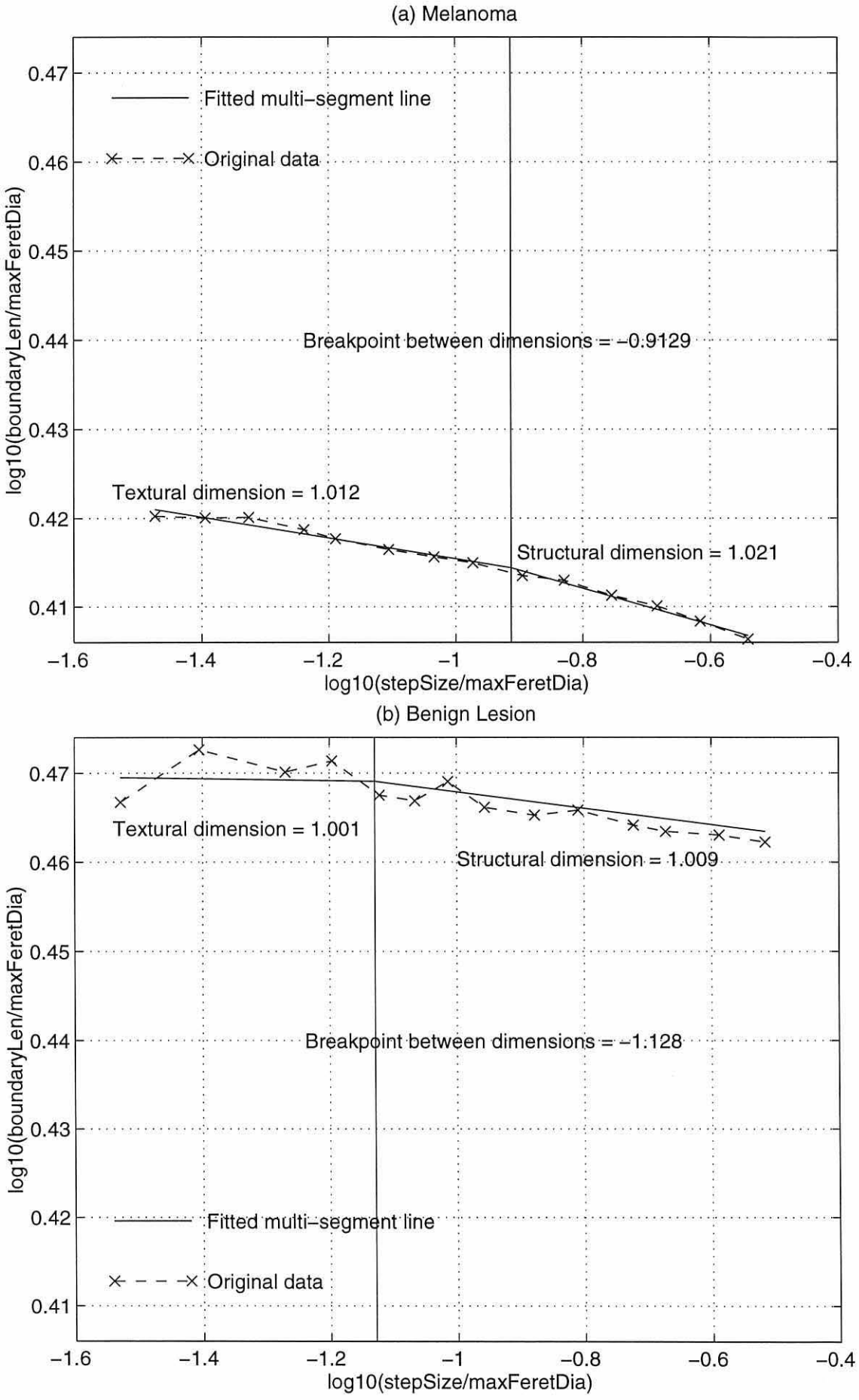


Figure 5-10: Example Richardson plots for a malignant and benign lesion.

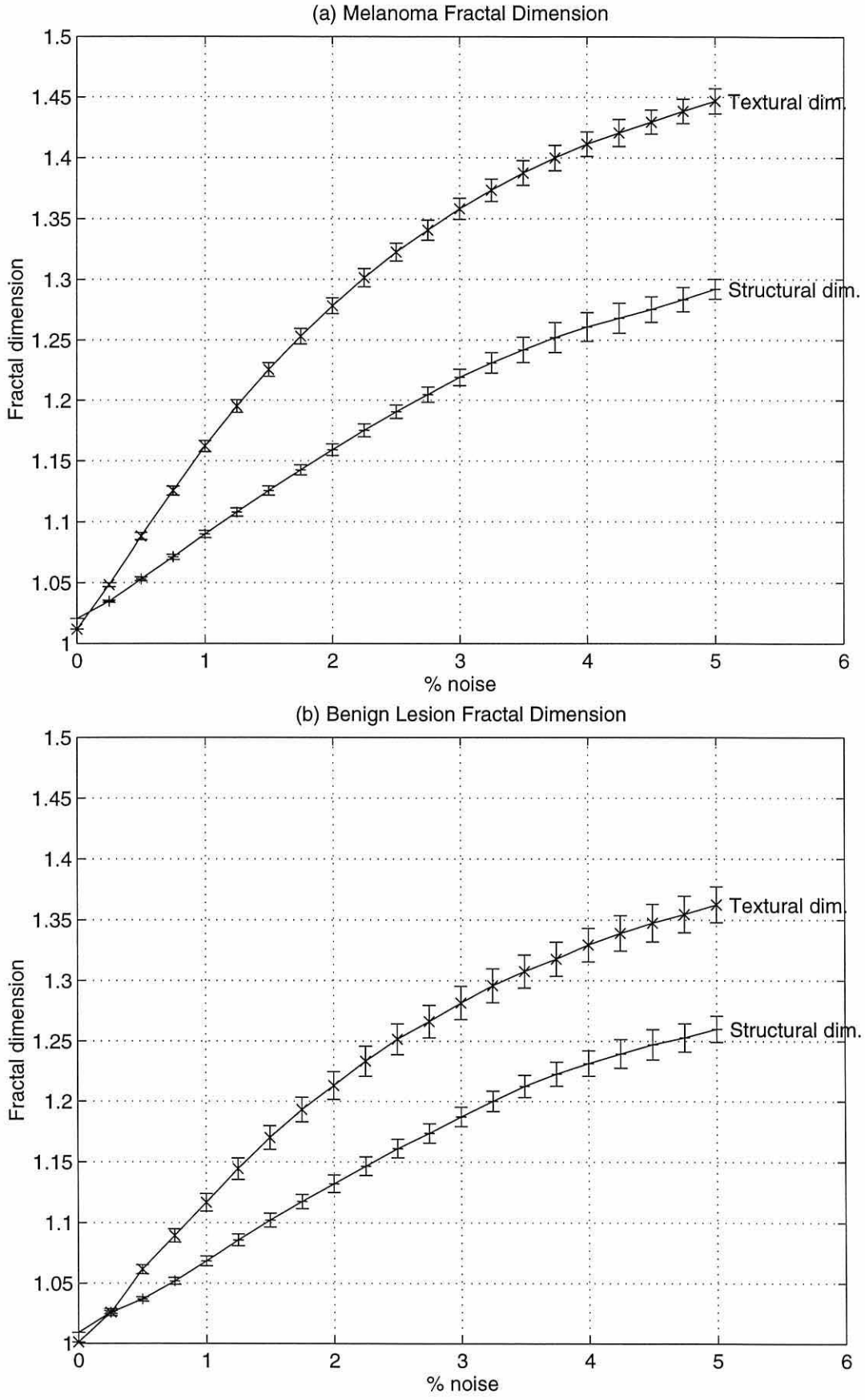


Figure 5-11: Effects of adding radial Gaussian noise on the fractal dimensions of a malignant and benign lesion. (a) Melanoma. (b) Benign lesion. The error bars are for ± the standard deviation of a set of 100 shapes.

5.5 Conclusions

This chapter described the measurement of two shape parameters which are known to be useful in diagnosing skin cancer. A method of obtaining bulkiness, which is an enhancement of the method proposed by Medalia [68], was presented. The measurement of fractal dimension was discussed and an implementation of "Flook's method" [31] used to find the structural and textural dimensions of lesion shapes. To select this method, three methods of fractal dimension measurement were tested on shapes with known fractal dimensions. To create these test shapes, a new method of creating shapes with specified fractal dimensions was presented. Using the test shapes "Flook's method" was shown to give the best overall performance. The application of these shape measurements in a diagnostic system is discussed in chapter 10.

The creation of shapes with known fractal dimensions allowed the assessment of the fractal dimension measurement methods on shapes with fractal dimensions comparable with those of real lesions. This allowed more reliable assessment of the fractal dimension measurement methods which was not possible in the previous application of fractal dimension measurement to lesion analysis [12].

The effects of adding radially distributed Gaussian noise to the shapes under assessment was investigated. The bulkiness and fractal dimension were shown to increase with increasing Gaussian noise, which demonstrates the importance of accurate boundary detection. The effects of noise on the fractal dimension measurement and the necessity of using high resolution shapes indicated that it may not be suitable for assessing lesion shapes. However, bulkiness does not require the use of such high resolution shapes and hence is more suited to assessing lesion shapes.

The lesion boundaries used initially to test these methods were obtained by binary thresholding of monochrome skin images. However, this method was observed to be inaccurate and unreliable in finding boundaries on many images. Hence, a more reliable and accurate method of determining lesion boundaries was required. Methods for achieving this are described in the following chapters.

Chapter 6

Skin Lesion Location and Isolation

6.1 Introduction

This chapter examines the problem of approximately locating and isolating lesions in an image. This process is important as it allows the lesion analysis to be tolerant of image capture conditions and image contents. An image may contain many background objects (e.g. clothing and rulers) which require identification as extraneous image areas so that the area containing a lesion can be identified and separated from the rest of the image. This chapter presents a process which can locate and isolate a single lesion and provide the following information, which is required or useful in the analysis of that lesion:

- **cropped image** containing a single lesion with sufficient surrounding skin image to allow accurate processing, but excluding background objects and other skin features.
- **approximate location of the lesion** within the cropped image and the location of the cropped image within the original image.
- **lesion size and shape** measurements.

The process, illustrated in Figure 6-1, was developed for images containing a single lesion and its extension to images containing multiple lesions is discussed in the chapter 10. It is essentially the locating of a number of dark regions (lesions), which can be characterised by their size, shape, intensity and texture, on a light background (skin).

The process is described in the order in which it is performed. At each stage alternative methods are described, evaluated, compared and the most appropriate method and its associated parameters presented. To enable this evaluation, all of the methods read and produce 256 grey-level images. This allowed each stage to be developed independently and the intermediate outputs to be examined.

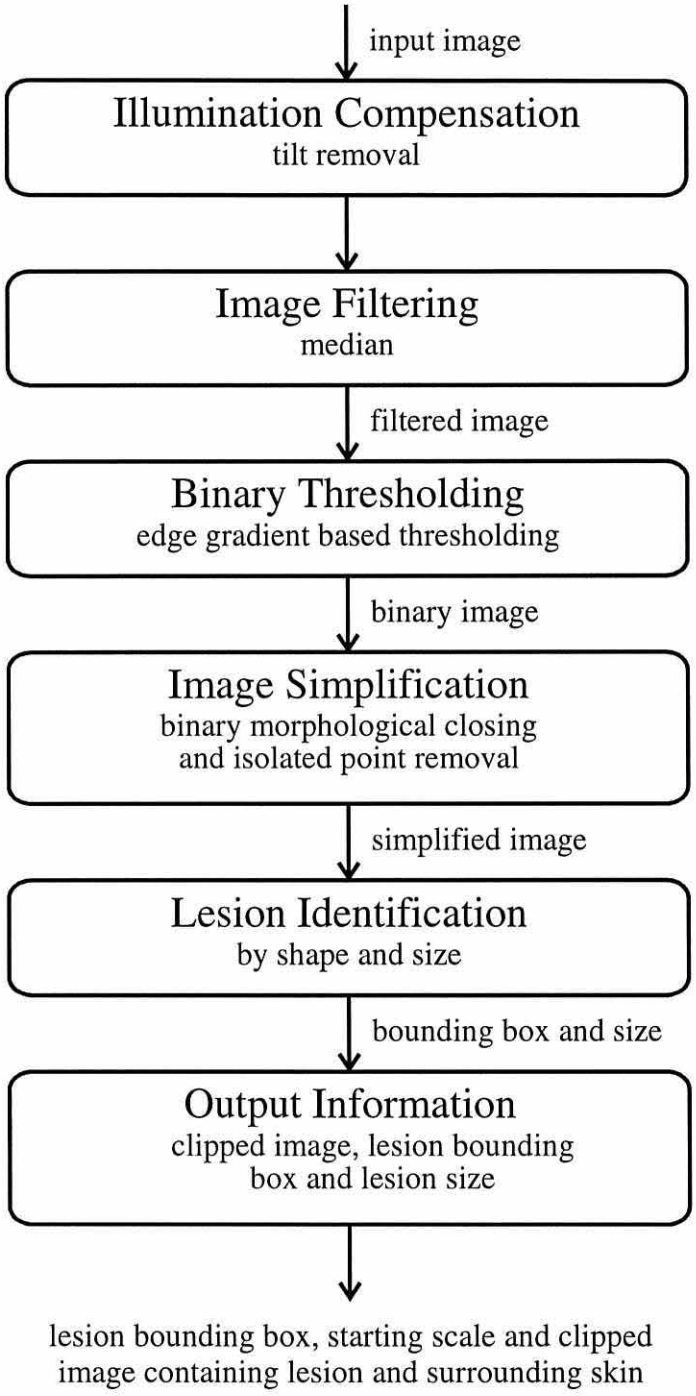


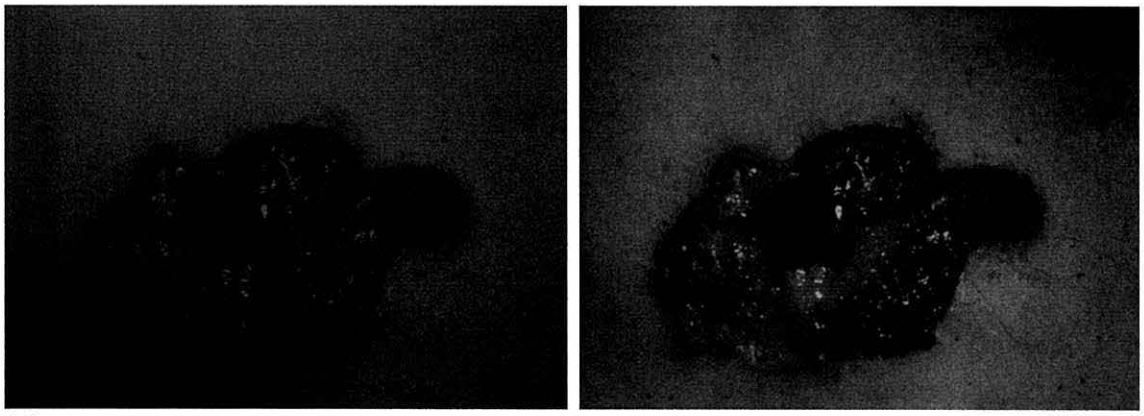
Figure 6-1: Lesion selection process.

6.2 *Illumination Compensation*

Skin images may have non-uniform illumination, which causes difficulties in identifying the lesion by affecting the binary thresholding of the images. One technique for reducing the impact of non-uniform illumination is described in this section and its usefulness assessed in sections 6.4.3 and 6.6.

6.2.1 Tilt Removal

Tilt removal, which was performed by a Khoros [83] program, can compensate for a constant illumination gradient. It is performed by computing a least square best fit plane, of the form $f(x,y) = Ax + By + C$, for the original image and subtracting this plane from the image. This sets the average intensity of the image to zero, but this can be restored by adding the average intensity of the original image which ensures that the output occupies a similar range to the input. An example tilt removal is shown in Figure 6-2. Tilt removal is suited to images with unknown uniform illumination gradients, such as that caused by non-uniform lighting.



(a) (b)

Figure 6-2: Image tilt removal using Khoros on a 668 by 480 image. (a) Before. (b) After.

6.3 *Image Filtering*

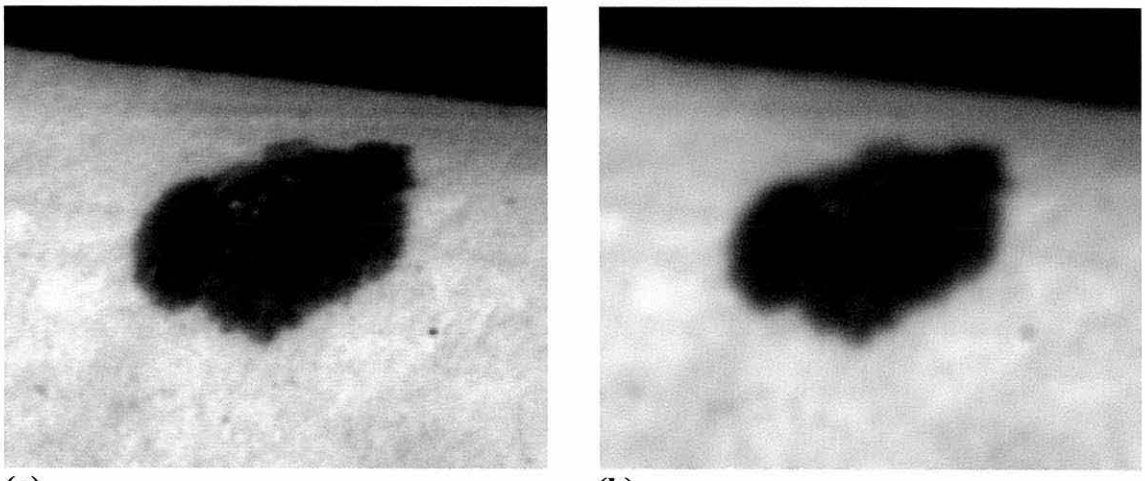
The thresholding, described in section 6.4, is affected by the edges and high frequency components of an image. By filtering the image these can be modified which changes the thresholding and hence the ability to identify the lesion. Three image filtering methods (low-pass filtering, median filtering and unsharp masking) are described in this section and tested in conjunction with thresholding methods in section 6.4.3.

6.3.1 Linear Low-pass Filtering

Low-pass filtering attenuates high frequencies whilst leaving low frequencies unaffected, thereby blurring the image. A simple form of low-pass filtering, average filtering, which replaces each pixel by the average of itself and its neighbours, is described in this section.

An average filter finds the average of the pixels in a square neighbourhood centred on the current output pixel. Hence, for an n by n filter n^2 values must be added for every pixel in the image, which may be a time consuming task. However, by using a *running average filter* the average computation can reduced to n additions and n subtractions per pixel. This method updates the average by removing the effect of the left-hand column and adding the effect of the right-hand column as the image is processed in a left to right, top to bottom scan. At the beginning of each line, the average is calculated for the whole neighbourhood. The neighbourhood is progressively truncated at the edges of the image to ensure that it is contained within the image. This truncation changes the effect of the filter near the image borders, but allows all of the image to be filtered and has less effect on the output than surrounding the image with zeros.

Low-pass filtering blurs the image, reducing the sharpness of edges and may affect the performance of thresholding. The effects of average filtering are shown in Figure 6-3.



(a) **(b)**

Figure 6-3: Low-pass filtering of an example image (246 by 210) using a 7 by 7 running average filter. (a) Original image. (b) Low-pass filtered image.

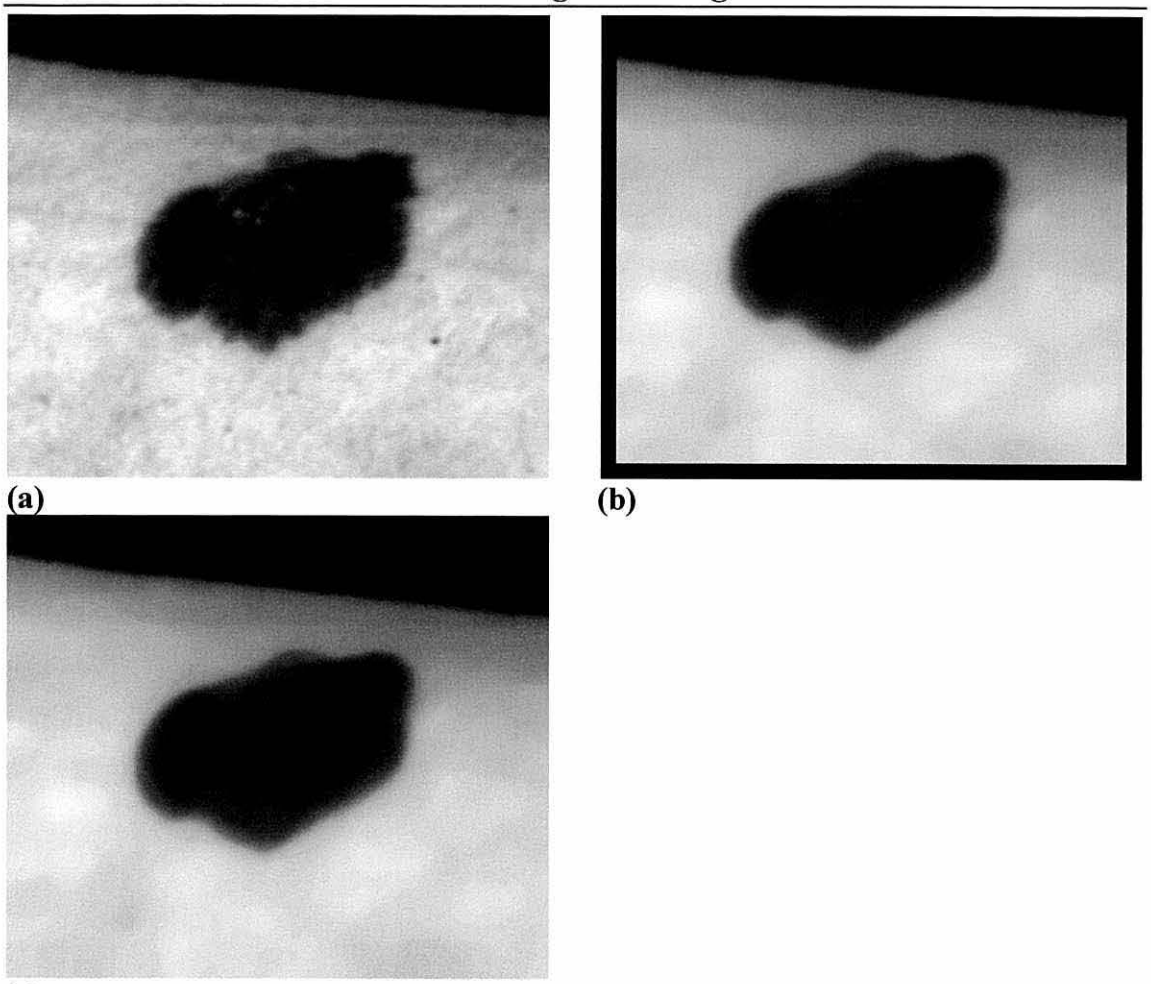
6.3.2 Median Filtering

A median filter replaces each point in the image with the median of all the points in a rectangular window centred on that point. This non-linear filtering, described in section 3.11, removes isolated spike noise but preserves distinct edges, which overcomes the problems of low-pass filtering.

To find the median the values of all the pixels in the filter window must be sorted, which is a time consuming process even for a small filter. For example, for a 5 by 5 pixel filter 25 values are sorted for every point in the image, which with a Shell sort [81] would in the worst case take 125 operations (Shell sort requires $O\left(N^{3/2}\right)$ operations). For a reasonable sized image and filter this is impractical and consequently the histogram based method, described below, was used.

Efficient image median filtering can be achieved using the running median algorithm published by Huang et al. [44], which uses the histogram of the pixels in an m by n filter window to find the median. As the image is scanned from left to right this histogram is updated to remove the effect of n left-hand column pixels and add the effect of the new n right-hand pixels. The remaining central $mn-2n$ pixels in the window are unchanged. At the beginning of each line the histogram is initialised for the first window. The published algorithm does not filter the image borders of $m/2$ and $n/2$ pixels at the left/right and top/bottom respectively. Although it is not possible to filter this area exactly, the algorithm can be adapted to allow partial filtering of this area, by truncating the filter window near the image borders. This truncation assumes that the data outside the image would have no effect on the median.

The results of filtering an example image, with and without filter truncation, are shown in Figure 6-4. While median filtering preserves edges, unsharp masking can enhance the visual appearance of edges.



(a) **(b)**

(c)

Figure 6-4: 2-D median filtering using a 15 by 15 window on a 246 by 210 image. (a) Original image. (b) Median filtered without filter truncation. (c) Median filtered with filter truncation.

6.3.3 Unsharp Masking

The performance of skin image segmentation by thresholding may be affected by the nature of the edges in an image. Unsharp masking, which is a form of edge crisping (high boost filtering) used in astronomy, can enhance the visual appearance of edges [80]. An unsharp masked image is created from the weighted difference of the unfiltered and low-pass filtered images:

$$i_{UM}(x,y) = \frac{c}{2c-1} i(x,y) - \frac{1-c}{2c-1} i_{LP}(x,y) \tag{6.1}$$

where $i_{UM}(x, y)$ = unsharp masked image at x, y,
$i_{LP}(x, y)$ = low-pass filtered image at x, y,
c = weighting constant: $0.5 \le c \le 1$.

The unsharp masked image has wider edges with overshoot and undershoot as illustrated in Figure 6-5. The unsharp masking process is shown on an example image in

using a simple running average filter (section 6.3.1) to perform the low-pass filtering. The sharpening effect is increased by either decreasing the weighting constant, c, or by increasing the degree of low-pass filtering.

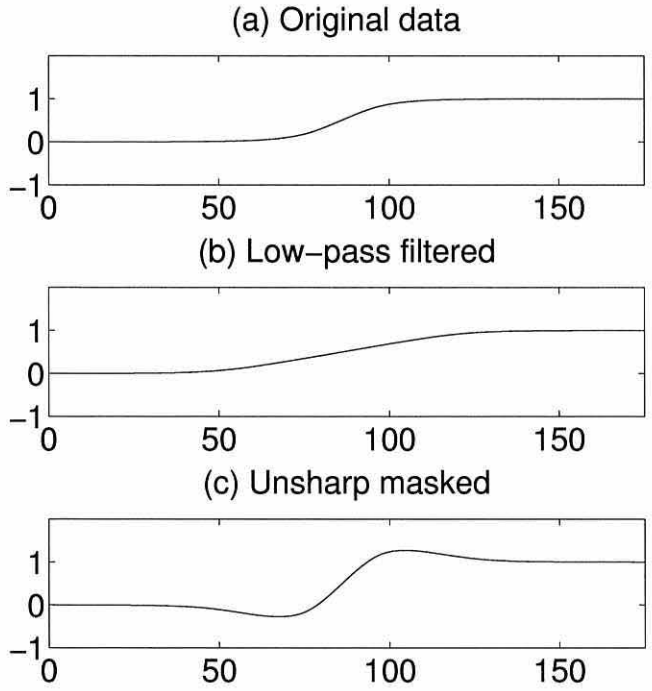
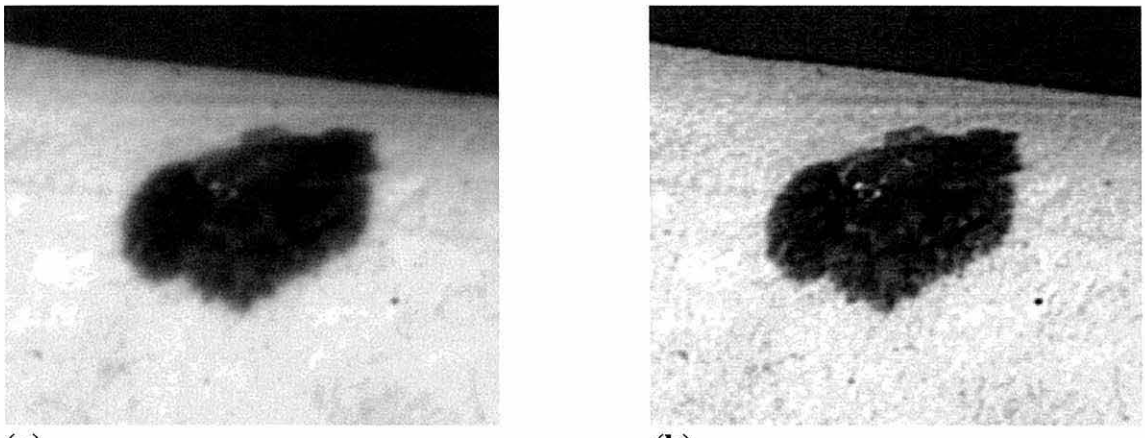


Figure 6-5: 1-D unsharp masking using an 1-D running average filter with length 71 points and c = 0.55. The 1-D running average filter is 1-D equivalent of the filter described in section 6.3.1



(a) **(b)**

Figure 6-6: Unsharp masking using a 7 by 7 average filter and c = 0.6 on a 246 by 210 image. (a) Original image. (b) Unsharp masked image.

6.4 Binary Image Thresholding

Binary thresholding can separate objects from a different intensity background by converting grey-scale images into binary images. Two global thresholding methods are described in this section and their performance assessed in combination with the image filtering methods described in section 6.4.3. Only global thresholding methods were tested as there is insufficient *a priori* knowledge to allow parameters to be deduced for local thresholding methods.

6.4.1 Histogram based Thresholding

A simple nonparametric and unsupervised method of selecting a threshold from an image's normalised histogram was published by Otsu [72]. The description given here uses the histogram rather than the normalised histogram and is based on the description given by Low [58]. The first stage of the method is the formation of the cumulative histogram:

$$h_C(k) = \sum_{j=0}^{k} h(j) \tag{6.2}$$

where k = grey-level,
$h(j)$ = number of pixels in the image with grey-level j.

From this the means of the pixels with grey-levels between 0 and i can be found:

$$m_G(i) = \frac{\sum_{j=0}^{i} jh(j)}{h_C(i)} \tag{6.3}$$

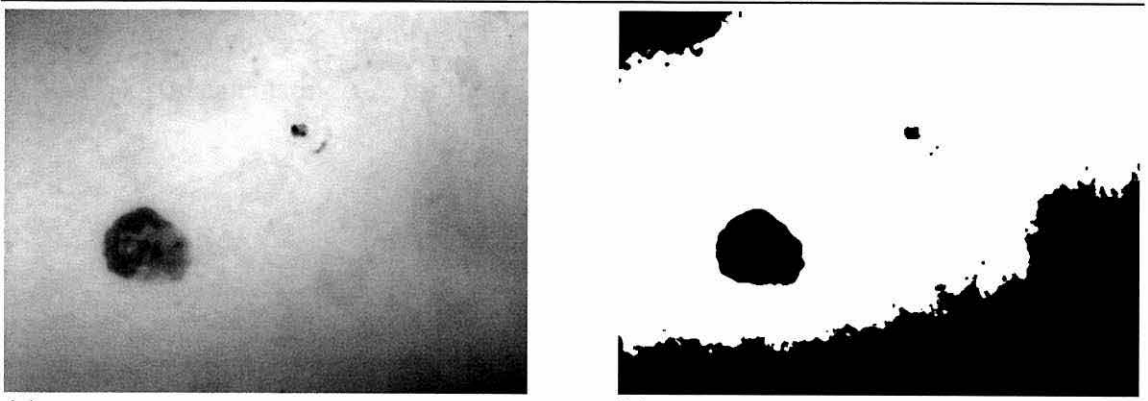
$$\bar{I} = m_G(L) \tag{6.4}$$

where $m_G(i)$ = mean of pixels with grey-levels between 0 and i,
$\bar{I}$ = mean grey-level of the whole image,
L = maximum possible grey-level.

The threshold (T_0) is set at the grey-level which maximises the following expression:

$$\frac{h_C(T_0)\left(m_G(T_0) - \bar{I}\right)^2}{MN - h_C(T_0)} \tag{6.5}$$

An example of an image thresholded by this technique is shown in Figure 6-7.



(a) (b)

Figure 6-7: Histogram based thresholding on a 652 by 478 image. (a) Original image. (b) Image thresholded at 172.

6.4.2 Edge Gradient based Thresholding

The threshold selection technique published by Kittler et al. [52] uses the edge gradients instead of histogram analysis. Two variations of the technique were tested:

1. The whole image (excluding a single pixel border) is used in calculating the threshold. (Kittler W)
2. Only pixels with high edge gradients are used to find the threshold. (Kittler HG)

In the first case, the values of s_{MG} and s_{IG} (from equations (6.6) and (6.7)) are used to obtain the threshold using equation (6.8). The edge gradients are calculated from the difference of the adjacent pixels. Thus:

sum of the maximum edge gradients:

$$s_{MG} = \sum_{\text{all pixels } (x,y)} i_M(x,y) \tag{6.6}$$

sum of the product of the intensity and maximum edge gradient:

$$s_{IG} = \sum_{\text{all pixels } (x,y)} i(x,y) i_M(x,y) \tag{6.7}$$

and the threshold:

$$T_0 = \frac{s_{IG}}{s_{MG}} \tag{6.8}$$

where $i(x, y)$ = image intensity at x, y,

$i_H(x, y)$ = horizontal gradient at $x, y = \left|i(x-1,y) - i(x+1,y)\right|$,

$i_V(x, y)$ = vertical gradient at $x, y = \left|i(x,y-1) - i(x,y+1)\right|$,

$i_M(x, y)$ = maximum gradient at $x, y = \max\left\{\left|i_H(x,y)\right|, \left|i_V(x,y)\right|\right\}$.

In the second case, an estimate ($\hat{\sigma}$) of the standard deviation of the edge gradients is obtained by calculating s_{MG} for the whole image:

$$\hat{\sigma} = \frac{\sqrt{2\pi}}{4} \frac{s_{MG}}{MN} \tag{6.9}$$

Then s_{MG} and s_{IG} are calculated for those pixels with edge gradients greater than $6\hat{\sigma}$ and these values used to obtain the threshold using equation (6.8). Assuming that the non-edge pixels have a Gaussian distribution then over 99.99% of them are removed by the $6\hat{\sigma}$ threshold [52]. Figure 6-8 shows this technique used on an example image and the performance of this technique is evaluated in the next section.

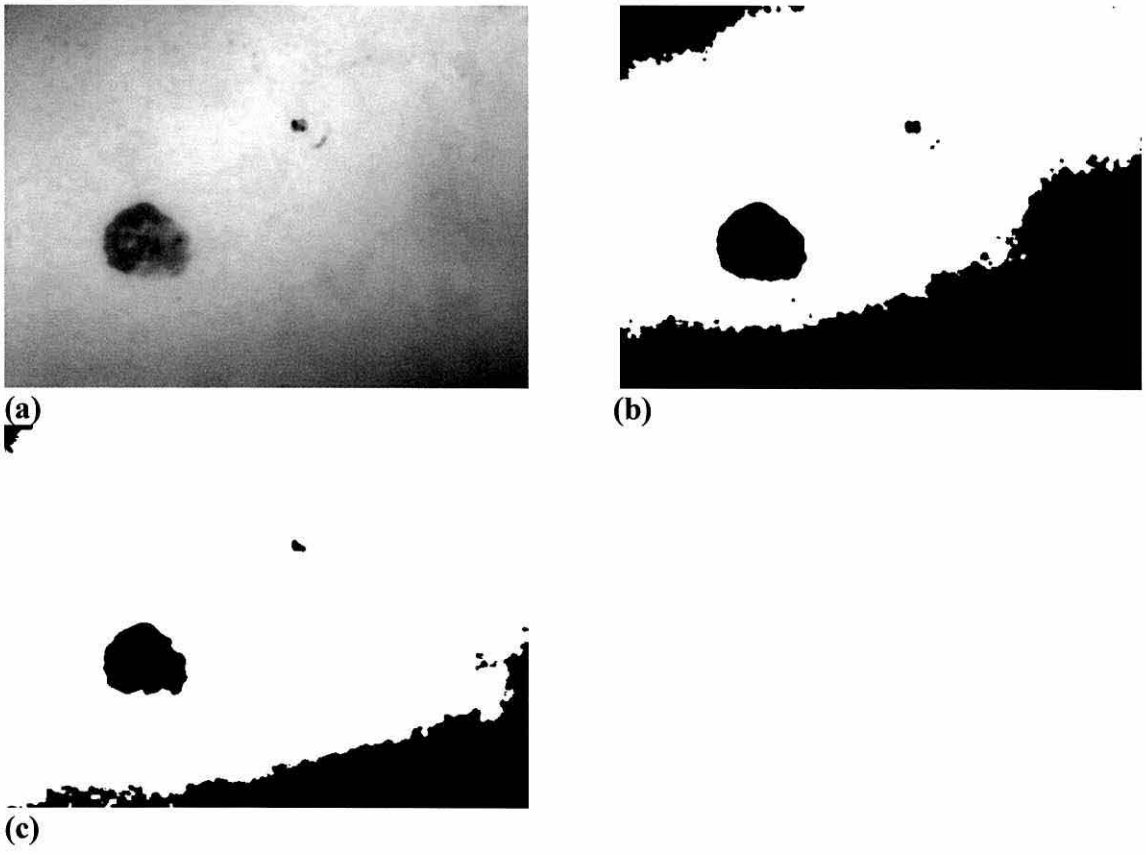


Figure 6-8: Edge gradient thresholding on a 652 by 478 image. (a) Original image. (b) Thresholded image (177) using the whole image. (c) Thresholded image (143) using pixels with high edge gradients. Threshold values are shown in brackets.

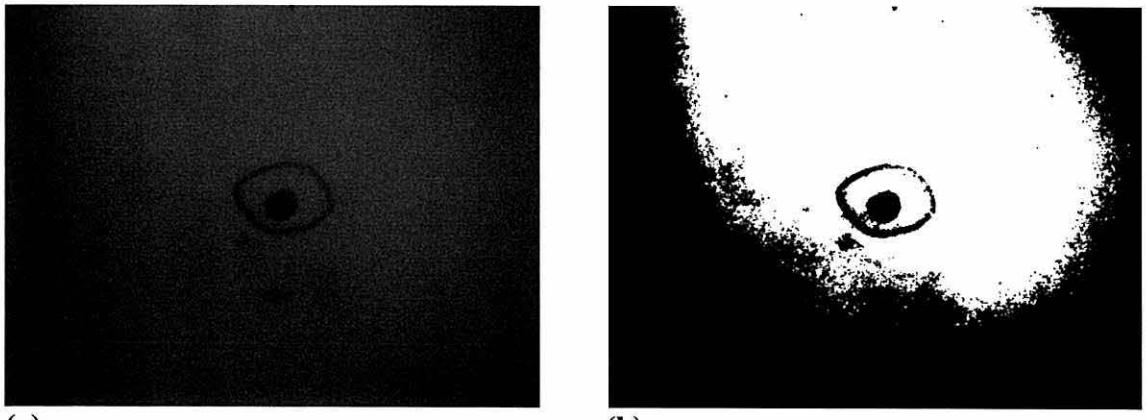
6.4.3 Selection of Filtering and Thresholding Method

The thresholding method used in this application must meet the following criteria:

1. It must find a threshold for every image.
2. The selected threshold must separate the lesion from the background.

The precise segmentation of the image and resulting exact shape of the lesion are not important as the exact boundary is found by the process presented in chapter 8. However, the lesion's shape must be sufficiently accurate to allow it to be identified. The separation of the lesion from the background was visually assessed by examining the original image and thresholded image side by side. Figure 6-8 shows a successful segmentation and Figure 6-9 shows an image in which the lesion cannot be separated from the background. In Figure 6-9(a), a pen mark can be seen around the lesion. This mark is also present in the thresholded image (Figure 6-9(b)) and is, unfortunately, connected to the area representing the lesion. Therefore, the lesion cannot be separated from the background, which may prevent its identification and give an incorrect indication of its extent. The slightly darker area around some of the image border is also present in the thresholded image. However, this can be separated from the lesion as it is not connected to the lesion.

To find a suitable combination of thresholding and filtering methods, the separation of the lesion from the background was visually assessed in 12 test images. The test set, which is shown in Appendix C, was chosen to represent a wide range of skin images and to include some multiple images of the same lesion to insure that the process was



(a) (b)

Figure 6-9: Example of an unsuccessful segmentation of the lesion from the background. (a) Original image (666 by 502). (b) Image thresholded with Kittler W thresholding (196) after filtering with a 3 by 3 average filter (section 6.3.1). Threshold value is shown in brackets.

tolerant of different image conditions. This visual assessment was performed by displaying each original image along side the images produced by a single thresholding method used in conjunction with 13 filters. The three thresholding and 13 filtering methods are listed in Table 6-I and Table 6-II. The visual comparison was performed on the original images and those with illumination tilt removal applied. The thresholding methods were selected to have simple implementation and rapid execution, to allow their use in practical systems, and not to require *a priori* knowledge of the image. The three main filtering methods (low-pass, median and unsharp masking) were chosen for their effect on the thresholding and for their simple implementation and rapid execution. Filter parameters were chosen to maintain or improve lesion visual appearance and to give reasonable execution times.

Table 6-III and Table 6-V show, for the original and tilt removed images respectively, the success rate of separating the lesion from the background for each thresholding and filtering method. Although many of the methods separated the lesion from the background consistently the resulting lesion shape was sometimes not similar to the visually perceived shape. Hence, for each image and thresholding method the filtering method which provided the region that was most similar to the visually perceived lesion shape was selected. In some images two or more different filtering methods produced indistinguishable results, so all of the best filtering methods were selected. In addition to the visual examination of the thresholded images, the number of regions in each thresholded image was counted. When more than two methods produced visually similar lesion regions the one with fewer non-lesion regions was selected as best. By selecting the method with fewer non-lesion regions, the expense of analysing the additional regions was reduced and the chances of a non-lesion region being incorrectly identified as the lesion was reduced. Table 6-IV and Table 6-VI show the percentage of the images for which each filtering method was selected as best. (Note: as multiple filtering methods could have been selected for each image thresholding method the rows of this table may not total to 100%.)

Summaries of the effects of the different thresholding and filtering methods are presented in Table 6-I and Table 6-II respectively. On the basis of these tests, the Kittler HG thresholding method with filtering by 7 by 7 median filter was selected. (Note: With these filtering and thresholding methods, the image shown in Figure 6-9(a) can be

successfully segmented, i.e. the lesion can be separated from the pen mark.) Although the visual similarity of the lesion regions in the thresholded image and the original images from the illumination tilt removed images was greater than that of the images without tilt removal the ability to separate the lesion from the background was only slightly better. Consequently, it is difficult, with these test images, to decide whether the illumination tilt removal is worth the extra computational expense. Hence, this decision is left until the lesion identification performance is assessed in section 6.6.

Thresholding method	Section	Comments
Kittler W	6.4.2	The number of regions in the images thresholded by this method was generally greater than the number produced by the other two methods.
Kittler HG	6.4.2	This method almost always found thresholds which were lower than Kittler W and Otsu.
Otsu	6.4.1	This method generally found thresholds which were lower than Kittler W.

Table 6-I: Thresholding methods.

Filtering method	Section	Comments
Unfiltered	N/A	These images often contained many small regions.
Average 3x3	6.3.1	The number of regions generally decreased with increasing filter size.
Average 5x5	6.3.1	
Average 7x7	6.3.1	
Median 3x3	6.3.2	The number of regions generally decreased with increasing filter size.
Median 5x5	6.3.2	
Median 7x7	6.3.2	
Unsharp masking avg. filter 3x3 $c = 0.6$	6.3.3	The unsharp masked images with $c = 0.8$ were generally better than those with $c = 0.6$. Also, those using the smaller average filters were better. Both of these factors produce images with less unsharp masking, i.e. with less edge emphasis. Generally, the unsharp masked images produced poorer segmentations than any of the other methods.
Unsharp masking avg. filter 3x3 $c = 0.8$	6.3.3	
Unsharp masking avg. filter 5x5 $c = 0.6$	6.3.3	
Unsharp masking avg. filter 5x5 $c = 0.8$	6.3.3	
Unsharp masking avg. filter 7x7 $c = 0.6$	6.3.3	
Unsharp masking avg. filter 7x7 $c = 0.8$	6.3.3	

Table 6-II: Filtering methods.

Thresholding method	Unfiltered	Average filtered		
		3 by 3	5 by 5	7 by 7
Kittler W	67%	67%	75%	75%
Kittler HG	92%	92%	92%	92%
Otsu	75%	67%	75%	75%
	Median filtered			
	3 by 3	5 by 5	7 by 7	
Kittler W	58%	67%	75%	
Kittler HG	92%	92%	92%	
Otsu	75%	75%	75%	
	Unsharp masked using an average filter					
	3 by 3 $c = 0.6$	3 by 3 $c = 0.8$	5 by 5 $c = 0.6$	5 by 5 $c = 0.8$	7 by 7 $c = 0.6$	7 by 7 $c = 0.8$
Kittler W	50%	58%	50%	50%	58%	58%
Kittler HG	75%	75%	67%	75%	67%	75%
Otsu	75%	75%	58%	67%	50%	75%

Table 6-III: Percentage success in visually separating the lesion from the background.

Thresholding method	Unfiltered	Average filtered		
		3 by 3	5 by 5	7 by 7
Kittler W	0%	0%	50%	50%
Kittler HG	0%	0%	17%	50%
Otsu	0%	0%	33%	67%
	Median filtered			
	3 by 3	5 by 5	7 by 7	
Kittler W	0%	0%	83%	
Kittler HG	0%	0%	75%	
Otsu	8%	17%	58%	
	Unsharp masked using an average filter					
	3 by 3 $c = 0.6$	3 by 3 $c = 0.8$	5 by 5 $c = 0.6$	5 by 5 $c = 0.8$	7 by 7 $c = 0.6$	7 by 7 $c = 0.8$
Kittler W	0%	0%	0%	0%	0%	0%
Kittler HG	0%	0%	0%	0%	0%	0%
Otsu	0%	0%	0%	0%	0%	0%

Table 6-IV: "Best" thresholding and filtering methods.

Thresholding method	Unfiltered	Average filtered		
		3 by 3	5 by 5	7 by 7
Kittler W	75%	83%	100%	100%
Kittler HG	92%	92%	100%	100%
Otsu	92%	92%	100%	100%
	Median filtered			
	3 by 3	5 by 5	7 by 7	
Kittler W	75%	100%	100%	
Kittler HG	92%	100%	100%	
Otsu	92%	92%	92%	

	Unsharp masked using an average filter					
	3 by 3 $c = 0.6$	3 by 3 $c = 0.8$	5 by 5 $c = 0.6$	5 by 5 $c = 0.8$	7 by 7 $c = 0.6$	7 by 7 $c = 0.8$
Kittler W	58%	67%	58%	67%	58%	67%
Kittler HG	83%	92%	83%	92%	67%	92%
Otsu	83%	83%	83%	83%	83%	83%

Table 6-V: Percentage success in visually separating the lesion from the background for illumination tilt removed images.

Thresholding method	Unfiltered	Average filtered		
		3 by 3	5 by 5	7 by 7
Kittler W	0%	0%	58%	83%
Kittler HG	0%	0%	58%	75%
Otsu	0%	0%	17%	75%
	Median filtered			
	3 by 3	5 by 5	7 by 7	
Kittler W	0%	0%	100%	
Kittler HG	0%	0%	92%	
Otsu	0%	0%	92%	

	Unsharp masked using an average filter					
	3 by 3 $c = 0.6$	3 by 3 $c = 0.8$	5 by 5 $c = 0.6$	5 by 5 $c = 0.8$	7 by 7 $c = 0.6$	7 by 7 $c = 0.8$
Kittler W	0%	0%	0%	0%	0%	0%
Kittler HG	0%	0%	0%	0%	0%	0%
Otsu	0%	0%	0%	0%	0%	0%

Table 6-VI: "Best" thresholding and filtering methods for illumination tilt removed images.

6.5 Simplification of the Thresholded Image

The image thresholding generates a complicated segmented image containing a number of regions. The lesion itself may be broken into separate regions because of variations in intensity, possibly caused by variegated colouring, may have internal holes and have a very irregular boundary which is unrepresentative of its true shape. These factors make identifying the lesion difficult or even impossible if it has been broken into separate regions. To overcome this problem, a "simplification" process that meets the following requirements is required:

1. Preserve approximate region shape and size - to allow each region to be identified.
2. Merge adjacent regions - lesions can vary in intensity, causing the global thresholding to break these lesions in two or more adjacent regions or generate inlets. Amalgamating these regions and removing the inlets enables lesion identification.
3. Smooth region outlines - to remove small scale irregularities which can appear after the global thresholding, but retain the overall region shape to allow identification.
4. Remove isolated very small regions - isolated regions that are too small to be lesions may be removed. This can be performed either at this stage or during the lesion identification.

A simple and quick method that meets the first three of these requirements is binary morphological closing [38, 41, 78]. Binary closing, which was introduced in section 3.14, smoothes region outlines and at the same time preserves approximate region shape and size (fulfilling requirements three and one respectively.) It also fuses small gaps between regions, which allows adjacent regions to be merged into larger regions (requirement two). It does not fulfil the fourth requirement of removing isolated small regions, but this can be performed during the lesion identification. Before "simplifying" the image, the image intensity is inverted (i.e. black and white are swapped) because in the original thresholded images the black lesion is classified as background. This allows binary closing to be applied as the closing affects the white foreground regions.

Binary closing uses a structuring element whose shape and size control its effects. In this application, the orientation of area of interest, i.e. the lesion, is not known hence the structuring element's effect should be direction independent. This criteria leads to the use of a circular structuring element which has equal effect in all directions. Hence,

Figure 6-10: Binary closing, on a 652 by 478 image, with a 9 pixel diameter disc. (a) Original thresholded image. (b) Closed image. Note: black and white have been swapped in these images to allow clearer printing.

discrete discs (filled circles) with odd sized diameters were tested as structuring elements. The odd disc diameter ensured that the discs always had a discrete centre.

By comparing Figure 6-10(a) and Figure 6-10(b) the effects of binary morphological closing with a disc can be seen. The outlines of the lesion region and the other regions have been smoothed. In addition, it can be seen that small gaps between regions have been bridged, e.g. the three very small regions on the right-hand side of the image are fused into one region.

The structuring element's size is dependent on the distance between the regions that require merging, on the degree of outline smoothing required and the size of the smallest lesion that is to be reliably analysed. Smaller sizes lead to less smoothing and the image

possibly containing more regions. This lack of smoothing may result in the lesion appearing to have a very irregular boundary that is not representative of its actual shape. Larger disc sizes increase the degree of smoothing and possibly reduce the number of regions. Reducing the number of regions removes regions which could be mistakenly identified as the lesion and also reduces the processing time required to identify the lesion. However, using a disc size which is too large would result in the lesion being merged with an adjacent region and also increases the processing time for the morphological closing.

To find a suitable disc diameter the thresholded images, from the test set used earlier, were visually examined without binary closing and with closing with odd disc diameters from 3 to 21 pixels. This range of diameters was used as dilating by a single pixel produced results very similar to the original images and diameters larger than 21 pixels caused many of the lesions to be amalgamated with adjacent regions. The thresholded images were produced by Kittler thresholding after filtering by a 7 by 7 median filter, which was found in section 6.4.3 to give the best separation of the lesions from their backgrounds. In addition to a visual examination of the thresholded images and corresponding original images, which showed whether the lesion was identifiable, the number of regions in each thresholded image was counted. This generally showed that as the disc size increased the number of regions present in the thresholded image decreased as adjacent regions became amalgamated. Table 6-VII and Table 6-VIII show the results of assessing the structuring element diameter without and with illumination tilt removal. The "best" structuring element diameter was that which gave regions most similar to the visually perceived lesion boundary. In many images two or more structuring element diameters produced indistinguishable results, so all of the "best" diameters were selected. (Note: As a range of disc diameters were sometimes selected the second row of both tables does not sum to 100%.)

For the test images without illumination tilt removal structuring elements with diameters 3 to 9 pixels (and without binary closing) were the most reliable at allowing the lesion to be identified. In particular, a structuring element disc of 9 pixels consistently gave the "best" segmentation of the image and is consequently the most suitable structuring element diameter for these images. When the test images had illumination tilt removal applied before filtering and thresholding, structuring elements with diameters 3 to 9

pixels (and without binary closing) were the most reliable. Consequently with these test images, a discrete disc with a diameter of 9 pixels was the most suitable structuring element for binary closing. The larger disc diameter was chosen as it reduces the number of regions in the thresholded images. An example of closing with a 9 pixel diameter disc was given in Figure 6-10.

The structuring element's diameter is dependent on the test images used, hence the diameter found in these tests may not always be the most suitable. However, these tests show the advantages of applying binary closing, with a disc, to the thresholded images before lesion identification is attempted. At this stage, the performance difference between the illumination tilt removed images and original images was small and hence the choice of method was left until lesion identification performance was assessed (section 6.6).

	Structuring element diameter					
	none	3	5	7	9	11
Lesion identifiable	92%	92%	92%	92%	92%	75%
"Best" disc size	67%	67%	67%	67%	92%	58%
	Structuring element diameter					
	13	15	17	19	21	
Lesion identifiable	75%	75%	75%	75%	75%	
"Best" disc size	58%	58%	58%	58%	58%	

Table 6-VII: Disc size for binary closing.

	Structuring element diameter					
	none	3	5	7	9	11
Lesion identifiable	100%	100%	100%	100%	100%	83%
"Best" disc size	58%	67%	67%	75%	75%	67%
	Structuring element diameter					
	13	15	17	19	21	
Lesion identifiable	83%	83%	83%	83%	83%	
"Best" disc size	75%	75%	75%	75%	83%	

Table 6-VIII: Disc size for binary closing with illumination tilt removal.

6.5.1 Isolated Pixel Removal

Single isolated pixels, which are too small to be of interest, are left intact by the binary morphological closing. This section describes how these isolated pixels may be removed. The only benefit of removing isolated pixels is that it reduces the number of regions present and consequently reduces the time required to identify the lesion. The isolated pixels are removed by setting those white pixels with no white neighbours to black.

6.6 Lesion Identification

One of the regions in the simplified thresholded image should be a lesion, which can be identified using the following general characteristics which lesions possess:

- low aspect ratio - artificial objects, such as rulers, tend to have higher aspect ratios.
- smooth outline - artificial objects tend to have sharp corners.
- elliptical - lesions are more elliptical in shape than most artificial objects. Bulkiness (section 5.3) is used to assess the smoothness and similarity to an ellipse.
- larger than a minimum size - if the lesion is too small it cannot be reliably analysed and is not significant.
- completely contained in the image - when the lesion is not completely contained in the image it cannot be reliably analysed.
- darker than the surrounding skin - this is used during the binary thresholding.

These factors and testing on real images led to the heuristic algorithm which is presented here. The algorithm examines all of the regions (found by a left to right, top to bottom scan) to decide which is a lesion based on the characteristics listed above.

Each region's external boundary is traced to create a list of boundary co-ordinates [76] (section 3.5) and this boundary used to delete the region. The region is deleted by either drawing the boundary, in black, if it has less than 4 points or by repeatedly flood filling (section 3.17) using each non-black boundary point as a seed point.

After finding each region's boundary it is characterised by a bounding rectangle which surrounds the region and just intersects with its boundary, its bulkiness (section 5.3), aspect ratio (the ratio of the equivalent ellipse's two diameters) and its size (the average

of the equivalent ellipse's two diameters expressed as a percentage of the average width and height of the whole image). The bulkiness and aspect ratio are both scale and rotation invariant and the size scales with the size of the image. To find suitable parameters for identifying lesions, all of the regions in a test set of thresholded images (excepting very small regions with widths or heights less than or equal to two pixels) were analysed to find the above parameters. (The 12 test thresholded images were produced by the filtering and thresholding methods selected at the end of section 6.4.3 from the test images introduced at the beginning of that section and were binary morphological closed by a 9 pixel diameter disc.) The results of this analysis are shown in Table 6-IX and Table 6-X for lesion and non-lesion regions respectively. These show that the range of bulkiness, aspect ratio and size is similar for the lesions in the original and illumination tilt removed images. However, for the non-lesion regions these measurements have considerably greater range and the bulkiness and aspect ratio are generally higher. From these measurements, the maximum allowed bulkiness and aspect ratio for a lesion were set to 1.3 and 2 respectively and the minimum size set to 4%. Using the following method on each region these settings produced a 100% success rate, on the 12 test images introduced in 6.4.3, at identifying the lesion in the original and illumination tilt removed images:

1. Ignore regions with widths/heights less than or equal to 2 pixels. These very small regions are too small to be lesions.
2. Find the bulkiness, aspect ratio and size of the region.
3. Ignore regions with high bulkiness or aspect ratios.
4. Ignore regions which are smaller than the minimum size.
5. Of the remaining regions, that with the smallest bulkiness was selected as the lesion.

The ability of the process, described in this chapter, to identify the region containing the lesion was the same for both the original and illumination tilt removed images. Hence, it was decided that the illumination tilt removal was not worth the extra computation and was not used. However, it is possible that tilt removal will make lesion identification more robust in the presence of uniform illumination gradients, but this is dependent on the lesions having uniform intensity. The highly variable nature of lesion images makes this unlikely, so it is much better to uniformly light the original scene. The extension of this lesion identification method to images containing multiple lesions is discussed in chapter 10.

It should be noted that this 100% success rate was obtained on a small test set and the performance on a larger test set is unlikely to be as good. However, the method, possibly with different parameters, would still be applicable on other images.

Original images		
	Minimum	Maximum
Bulkiness	1.02	1.17
Aspect ratio	1.14	1.87
Size	4.38%	73.7%
Illumination tilt removed images		
	Minimum	Maximum
Bulkiness	1.02	1.27
Aspect ratio	1.11	1.88
Size	5.05%	62.39%

Table 6-IX: Lesion shape identification parameters.

Original images		
	Minimum	Maximum
Bulkiness	0.96	6.47
Aspect ratio	1.03	6.47
Size	0.54%	74.98%
Illumination tilt removed images		
	Minimum	Maximum
Bulkiness	0.99	7.74
Aspect ratio	1.05	18.89
Size	0.54%	74.11%

Table 6-X: Non-lesion region shape parameters.

6.7 Output Information

Once the lesion has been identified in the cleaned, thresholded image the following information is required to enable its further analysis:

- A sub-image (from the original input image) containing the lesion and sufficient surrounding image. The image border enables comparison of the lesion with the skin and provides data for image processing techniques, such as edge detection, to function correctly.
- Approximate lesion size.
- Approximate lesion location, which is indicated by an area that completely contains the lesion.

The following description of how to obtain this information is based on the Laplacian of Gaussian (LoG) edge focusing algorithm (described in detail in chapter 8), but could be adapted for other edge detectors or alternative analysis methods.

An estimate of the lesion's size is required to set the edge detector's size to just detect the lesion, but not overlook any part of the lesion. If the edge detector's size is set incorrectly the following can happen:

- much too large - no boundary or an incomplete boundary is found and the lesion is overlooked.
- slightly too large - initial boundary is not correctly placed and the initial edge detection takes longer, but the boundary can be found.
- slightly too small - initial boundary is not correctly placed and extra edges may be detected, but the lesion boundary can be distinguished.
- much too small - many edges are detected and the lesion boundary cannot be found.

The estimate of the lesion's size should indicate the size of the smallest dimension of the lesion. During the lesion identification an equivalent ellipse, which is aligned with the lesion's outline, is obtained and this ellipse's smallest diameter can be used as an estimate of the lesion's minimum width. (The equivalent ellipse is shown in Figure 6-12.) The size of the LoG edge detector (which is specified by its space constant (σ_{SC})) is set from the ellipse's smallest diameter by the following equation (see chapter 7):

$$\sigma_{SC} = \frac{d_S}{4\sqrt{2}} \tag{6.10}$$

where d_S = smallest diameter of the equivalent ellipse.

The above equation assumes that the lesion is a ridge edge and sets σ_{SC} to place the edges (zero-crossings) detected by an LoG convolution on the sides of the ridge, which is illustrated in Figure 6-11(a). Although a real lesion image would not have this intensity profile, this assumption will give a suitable space constant on a range of lesion intensity profiles. For example, using the same σ_{SC} on a pulse edge (Figure 6-11(b)) gives edges which are within σ_{SC} of the sides of the pulse. It is possible to set the space constant by assuming other shapes, but this method was successful when used with the LoG edge focusing algorithm.

In the edge focusing algorithm σ_{SC} is successively decremented by 0.5 to generate a sequence of boundaries. (The reason for decrementing by 0.5 is explained in chapter 8.)

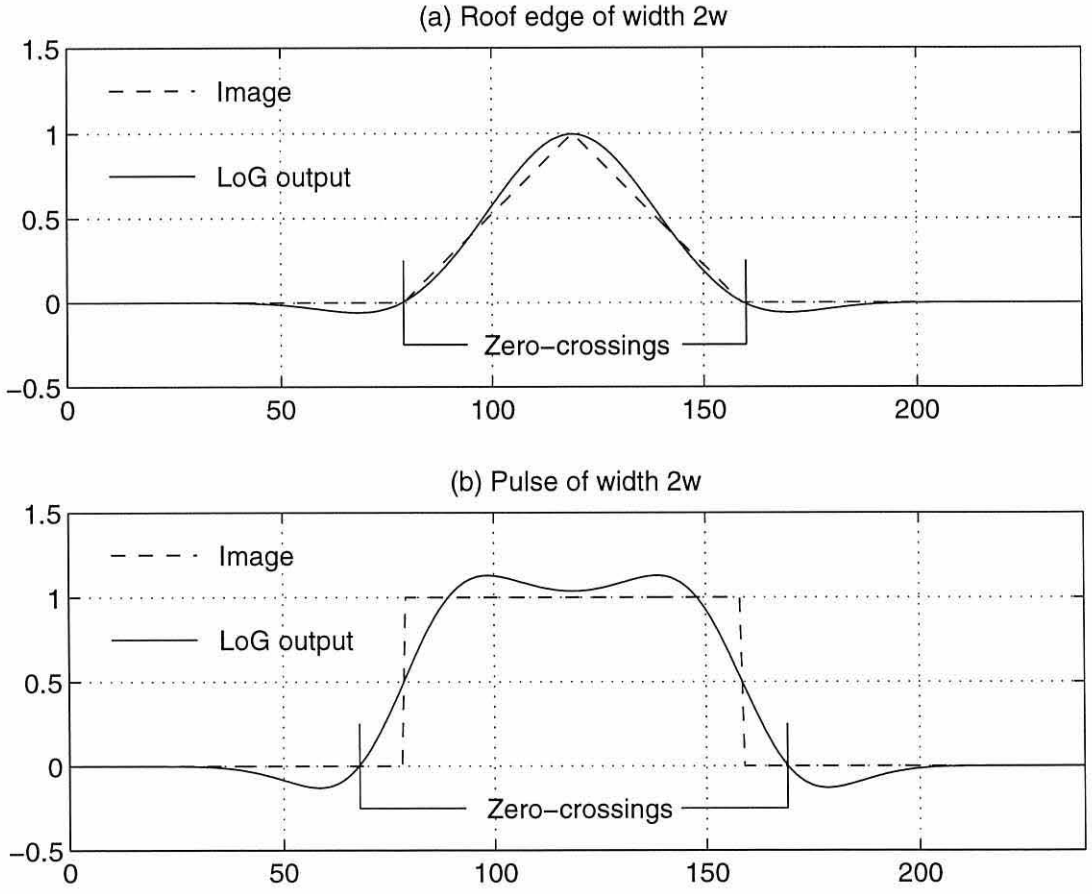


Figure 6-11: LoG output. (a) Ramp edge of width 2*w*. From the width of this ramp edge using equation (6.10) σ_{SC} = 14.142. Using equation (6.11), the σ_{SC} which would be used for this edge is 13.828. (b) Pulse edge of width 2*w*, with LoG output with σ_{SC} = 13.828.

When using equation (6.10), the sequence is different for every different sized lesion. To allow comparisons between different images it is convenient to have a fixed sequence of σ_{SC} values. Consequently, σ_{SC} is rounded to the nearest multiple of 0.5 plus the minimum σ_{SC} less than the value obtained from equation (6.10):

$$\sigma_{SC} = \tfrac{1}{2}\left\lfloor \left(\frac{d_S}{4\sqrt{2}} - \frac{s_{min}}{2\sqrt{2}} \right) \Big/ 0.5 \right\rfloor + \frac{s_{min}}{2\sqrt{2}} \tag{6.11}$$

where s_{min} = minimum permitted LoG sensitive region in pixels, which is defined in chapter 8 as 8,
$\lfloor x \rfloor$ = floor - the largest integer less than x.

The lesion is approximated by an image oriented bounding box L (shown in Figure 6-12), which completely contains the lesion and just intersects its boundary. This simplifies the image processing as it is generally easier to process a rectangular area than an irregularly shaped area; however extra image area, which may not be of interest, is included. Box L must contain the boundary detected by the LoG operator at the starting scale, as edges outside this box are ignored. However, additional edges inside the lesion bounding box can be dealt with by the edge focusing algorithm. Even in the ideal case where the lesion boundary in the cleaned thresholded image is the boundary finally found by the edge focusing algorithm, the initially detected boundary will not coincide with the boundary in the cleaned thresholded image. This edge displacement is discussed in chapter 8 and can be allowed for by considering the idealised case when the box L is a black filled area. The edges detected at the initial scale will not coincide everywhere with the box's boundary and will be displaced by up to σ_{SC} [92] outwards from the box near the corners. To allow for this potential displacement, a border of $\lceil \sigma_{SC} \rceil$[1] pixels is added to box L to give box A. This displacement of $\lceil \sigma_{SC} \rceil$ assumes that the boundary has a large curvature at all points, which is very unlikely to happen and consequently this border will be larger than necessary. However, the edge focusing algorithm (chapter 8) is tolerant of this extra border area being included.

To find the size of the required cropped image C, a border is added around the bounding box A. The size of this border is determined by the area required for the correct functioning of the edge detector and is half the detector's width (M_{LoG}), which for the LoG

[1] $\lceil x \rceil$ = ceiling - the smallest integer greater than x.

detector is $6\sqrt{2}\sigma_{SC}$ (this is rounded to the nearest odd number, as the filter width must be odd to allow its actual use on an image).

When the lesion is near the edge of the original image, the required cropped image may not be completely contained within the original image. In this case, analysis may be inaccurate or not possible. When the bounding box A is not within the original image, the lesion cannot be guaranteed to be contained within the image and hence analysis is not possible. When the cropped image C is not completely within the original image, analysis may be less accurate or not possible depending on the techniques used. The maximum percentage of the image which can be filled by the lesion before the required cropped image overlaps the original image's edge can be obtained by making assumptions about the lesion's shape and location. When the lesion is centrally placed and the image is square, the situation shown in Figure 6-13 occurs. The minimum width (M) of the original image is:

$$M = 3w + 2w + 2\sigma_{SC} = (5 + \frac{\sqrt{2}}{2})w \tag{6.12}$$

where $w = 2\sqrt{2}\sigma_{SC}$ = size of the LoG sensitive region.

This gives a maximum lesion diameter of 35% of the original image width:

$$\text{Max. lesion diameter} = 2w = \frac{2M}{5 + \frac{\sqrt{2}}{2}} = 35\% \text{ of } M \tag{6.13}$$

The edge focusing algorithm can work when the required cropped image exceeds the original image, but initial edge detection may not be as accurate as some of the image data required is not available.

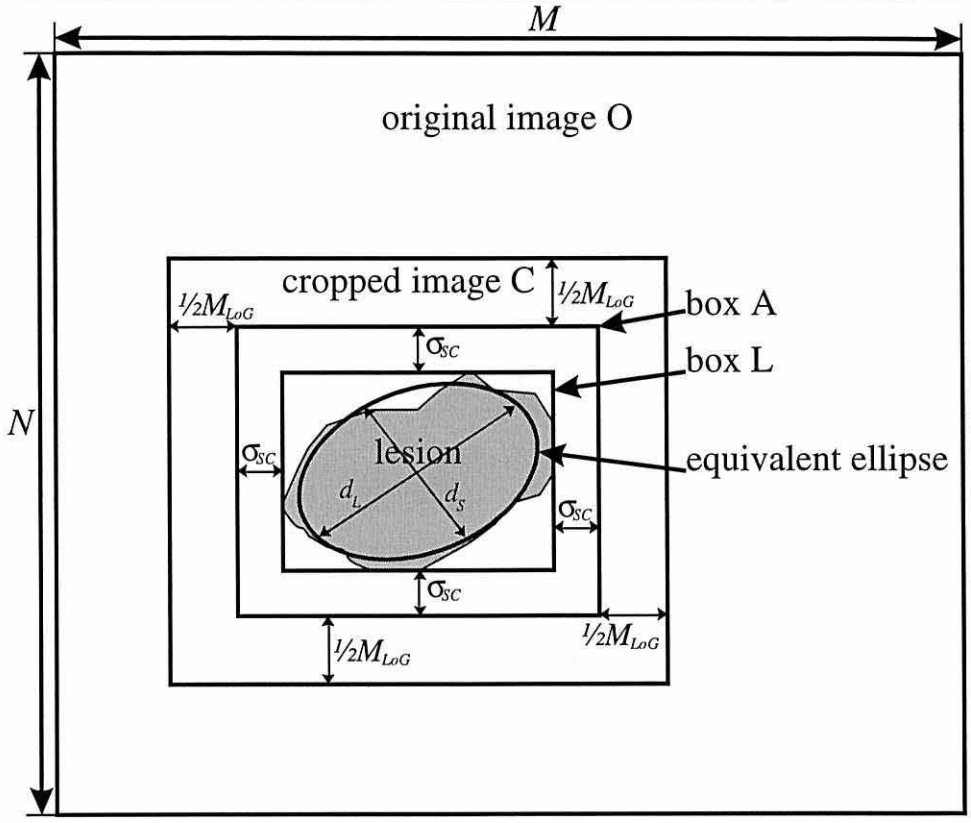


Figure 6-12: Lesion output information (not to scale).

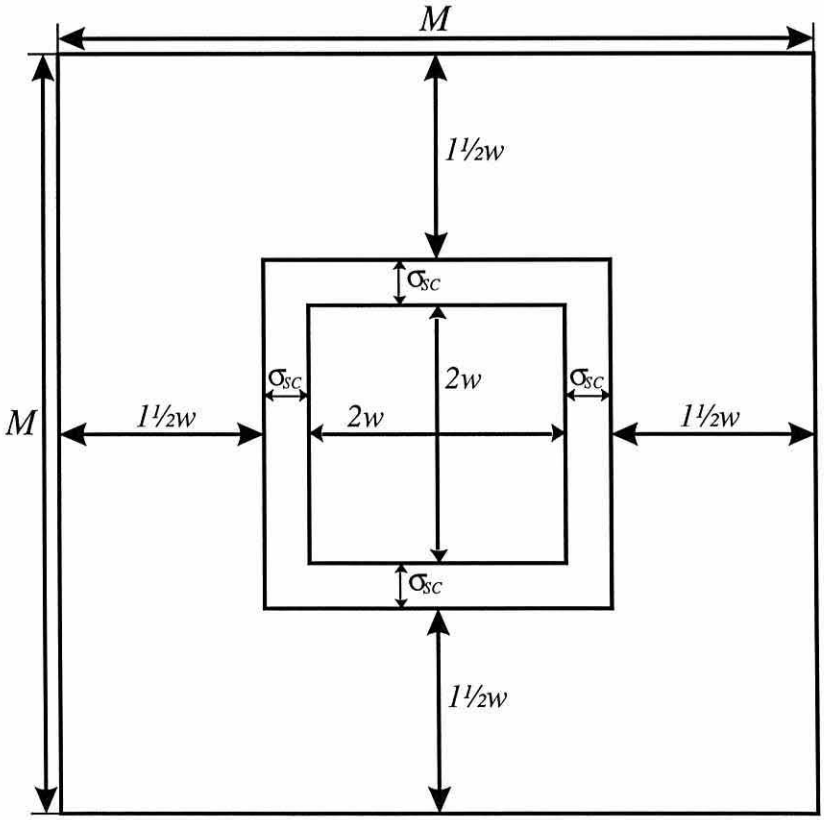


Figure 6-13: Maximum lesion size (not to scale).

6.8 Conclusions

This chapter described a method of analysing an arbitrary image containing a single lesion to generate a cropped image containing the lesion with sufficient surrounding image and an indication of the lesion size, location and bounding area. This information is the starting point for further analysis of the lesion, such as boundary detection which can be performed by the edge focusing algorithm presented in chapter 8. The further development of the method presented in this chapter is discussed in chapter 10.

The process presented in this chapter used the following sequence of operations: filtering, thresholding, simplification, lesion identification and information output. At each stage alternative methods were presented, tested and the most appropriate selected. Three types of image filtering methods, with a range of parameters, were tested with three thresholding methods to find an appropriate combination of filtering and thresholding which separated the lesion from its background. After separating the lesion from its background, a method, using binary morphological closing, for simplifying the thresholded image was evaluated. This allowed the lesion to be identified using heuristic rules based on shape. Finally, the size and position of a cropped image, containing the lesion and sufficient surrounding image, was calculated together with the starting edge detection scale for the LoG edge focusing algorithm.

The work in this chapter, described in the previous paragraph, showed that Kittler thresholding in conjunction with a 7 by 7 median filter is a suitable method of segmenting lesion images to allow identification of the lesion. After morphological closing with 9 pixel diameter disc, heuristic rules, based on shape and a test set of 12 lesion images, provided a 100% success rate at identifying the lesion and providing a cropped lesion containing image.

Chapter 7

Edge Detection Methods and their Implementation

7.1 Introduction

This chapter examines the detection of luminance edges in images and explains the techniques used in the edge focusing algorithm described in the next chapter. It does not consider the detection of edges in colour images and those caused by texture changes. Edge detection was introduced in section 3.8 and is expanded upon in this chapter by studying the first three stages shown in Figure 7-1:

1. **Edge detection.**

   The first section considers a selection of simple edge detectors, by explaining their implementation and limitations. In later sections, a second derivative (Laplacian of Gaussian) and a first derivative (Canny) edge detector are examined in detail by explaining their implementation.

2. **Edge localisation.**

   Edge localisation generates an edge map from the edge detector's output, which gives the location of edges and may indicate their direction. For first derivative edge detectors non-maxima suppression is used (section 7.5.2) and for second derivative edge detectors zero-crossing (ZC) detection is used (section 7.4).

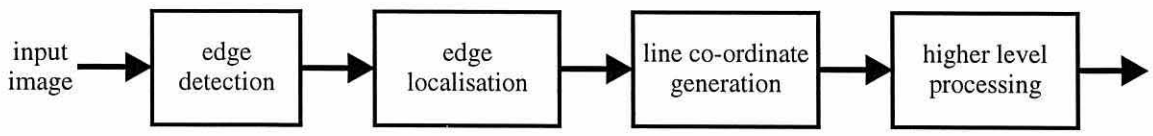


Figure 7-1: Edge analysis.

3. **Edge co-ordinate generation.**

   Two methods for converting an edge map into the co-ordinates of the connected edges are presented in section 7.6. The first of these methods uses the edge direction information, provided by the ZC detection, to assist in following edges and the second method uses the relative position of the previous point on the line to guide the edge tracing and is used for the non-maxima suppression output.

The problems of image border erosion and processing irregularly shaped areas are discussed in sections 7.7 and 7.8.

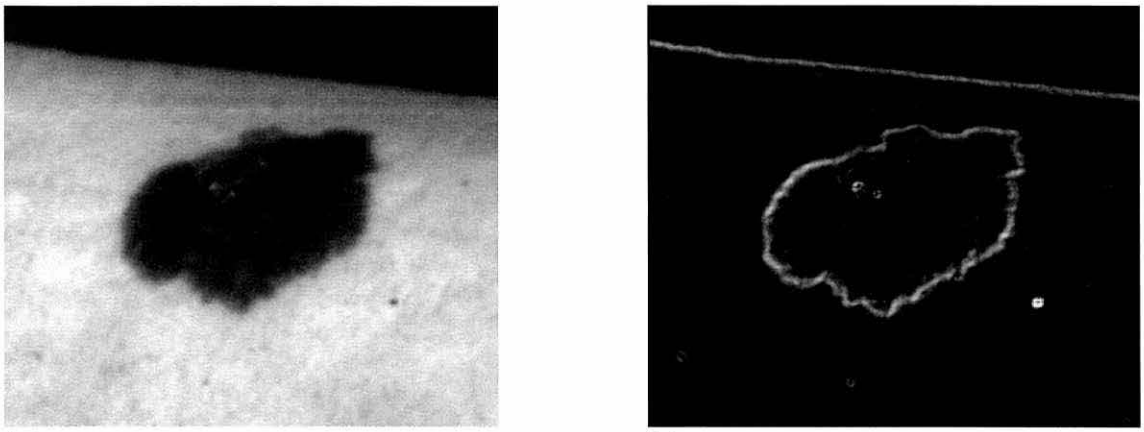
7.2 Simple Edge Detectors

Simple edge detectors can be based on small filters, such as those shown in Table 7-I [80], which are convolved with an image to give images indicating the local horizontal and vertical gradients. The horizontal and vertical gradients from these heuristically obtained filters can be combined to yield edge intensity and direction. An example of a common simple edge detector, the Sobel operator, applied to a lesion image is shown in Figure 7-2.

There are two problems with simple edge detectors: variable directional sensitivity and inability to accurately detect edges in noise. Their sensitivity to edges varies with direction, for example the Sobel operator is 1.06 times more sensitive to diagonal edges than horizontal and vertical edges [80]. Their sensitivity to noise is caused by the small area (3 by 3 pixels) used and can be alleviated by properly increasing their size; however this is difficult as the filters are heuristically obtained [80]. The example image in Figure 7-2 shows that although the lesion border can be seen, many other edges have been detected. None of the simple edge detectors can localise edges to a single pixel and differ in their directional sensitivity, localisation ability and their probability of correct edge detection. These problems led to the investigation of more complex edge detectors such as those described in the next section.

Operator	Horizontal gradient	Vertical gradient
Roberts	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
Prewitt	$\frac{1}{3}\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{3}\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Sobel	$\frac{1}{4}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{4}\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
Frei-Chen	$\frac{1}{2+\sqrt{2}}\begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{2+\sqrt{2}}\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$

Table 7-I: Simple edge detectors.



(a) **(b)**

Figure 7-2: Sobel edge detection. (a) Original image (246 by 210). (b) Edges detected by a Sobel operator.

7.3 Laplacian of Gaussian Edge Detection

Marr and Hildreth [65, 64] suggested using the zero-crossings in the output of a Laplacian of Gaussian (LoG) operator for edge detection based on the human visual system. They argued that to build a description of an image required the location of edges at a range of different scales, which the combination of the two parts, Laplacian and Gaussian, of the LoG operator can provide. The Laplacian function is a second order differential operator which can detect intensity changes and the Gaussian part blurs the image, effectively wiping out all structure at scales much smaller than the space constant σ_{SC}. The two-dimensional form of the LoG operator, shown in Figure 7-3(a), is:

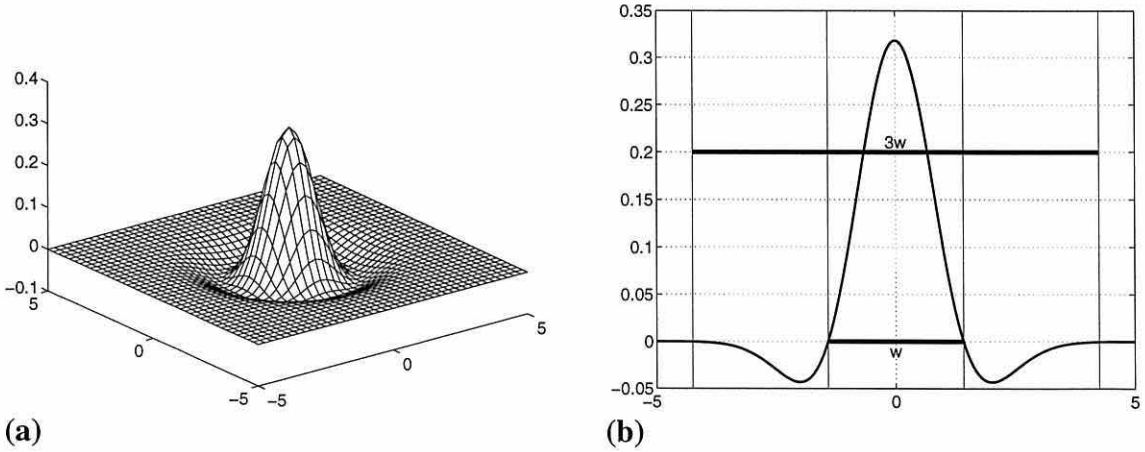
$$\nabla^2 G(x,y) = \frac{1}{2\pi\sigma_{SC}^{\ 4}}\left(2-\left(\frac{x^2+y^2}{\sigma_{SC}^{\ 2}}\right)\right)\exp\left(-\frac{x^2+y^2}{2\sigma_{SC}^{\ 2}}\right) \tag{7.1}$$

where σ_{SC} = space constant (standard deviation) of the Gaussian.

Edges are indicated by zero-crossings in the output of the convolution of this function with the image and can be located by the process described in section 7.4. This operator can be scaled, using σ_{SC}, to vary its sensitivity to small edges and is rotationally invariant, i.e. it is equally sensitive to edges in all directions.

The sensitive region of the filter (w), which indicates the size of features detected, has a width of $2\sqrt{2}\sigma_{SC}$ and is the central positive region of the filter, which is shown in Figure 7-3(b). Features with a width smaller than w have their edges displaced and may become fused with the edges from adjacent features. This fusing is useful in obtaining a global description of an image as small details are not present in the output. The overall width (M_{LoG}) of the filter is set at $3w$ [45] and for implementation, M_{LoG} is rounded to the nearest odd number larger than $3w$, which ensures the filter has a central point.

The filter can be applied by directly convolving a 2-D mask with the image, which for large σ_{SC} is impractical, as it requires $M_{LoG}^{\ 2}$ multiplications and additions per pixel and is too time consuming. For example, the mask size for σ_{SC} = 10 is 85 by 85 which requires 7225 multiplications and additions per pixel. To overcome this problem, the 2-D LoG can be decomposed into the sum of two separable 1-D filters [45]:



(a) (b)

Figure 7-3: Laplacian of Gaussian, with σ_{SC} = 1. (a) 2-D shape. (b) 1-D shape showing sensitive region (w) and overall width ($3w$).

$$\nabla^2 G(x,y) = h_1(x)h_2(y) + h_2(x)h_1(y) \tag{7.2}$$

where

$$h_1(i) = \frac{1}{\sqrt{2\pi}\sigma_{SC}^{\ 2}}\left(1 - \frac{i^2}{\sigma_{SC}^{\ 2}}\right)\exp\left(\frac{i^2}{2\sigma_{SC}^{\ 2}}\right), \tag{7.3}$$

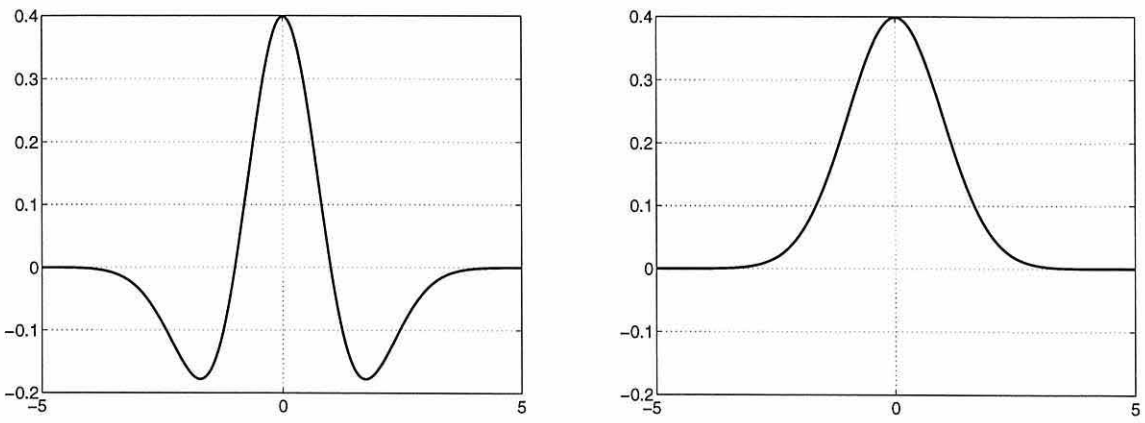
$$h_2(i) = \frac{1}{\sqrt{2\pi}\sigma_{SC}^{\ 2}}\exp\left(\frac{i^2}{2\sigma_{SC}^{\ 2}}\right). \tag{7.4}$$

$4M_{LoG}$ multiplications and additions per pixel are required for this version, but the number of multiplications can be reduced by a factor of two by noting that the filters h_1 and h_2, shown in Figure 7-4, are symmetric. The separated LoG filter is applied using the following steps:

1. **Calculate maximum and minimum filter outputs.**

   The input to the LoG filter is an unsigned byte image and for convenience and to minimise the storage requirements the output is scaled to fit into a signed byte. The scaling is found by calculating the worst case maximum and minimum outputs by assuming an input of 255 for +ve and 0 for -ve filter coefficients for the maximum output and visa versa for the minimum output. Single precision floats were used for the filter and intermediate storage. An integer convolution kernel was not used, as it is not possible to ensure that the sum of the filter coefficients is always zero whilst retaining the filter shape. The zero-crossings will be offset if the filter coefficient sum is non-zero.

2. **Generate filters h_1 and h_2,** using equations (7.3) and (7.4).



(a) h_1 **(b) h_2**

Figure 7-4: Separated LoG, with $\sigma_{SC} = 1$. (a) h_1. (b) h_2.

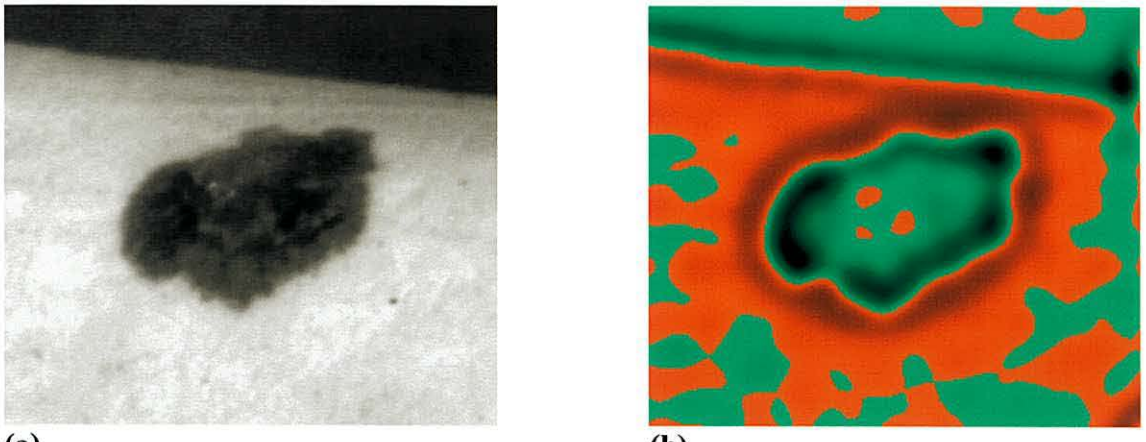
3. **Partition the convolution to use the available memory.**

   For an image of M by N two sets of intermediate storage of M by $\left(N + \lfloor 3w/2 \rfloor\right)$ are required, which may occupy considerable memory. To reduce the memory usage, the calculation can be partitioned into vertical slices with widths that are calculated from the available memory and image height.

4. **Perform convolution on each slice.**

   The image is row convolved with the two filters h_1 and h_2 and the output of these convolutions column convolved with the other filter, i.e. convolution h_1 output convolved with h_2. The output of the last two convolutions are summed and scaled to provide the final output.

Figure 7-5 shows the results of applying the above steps to generate an image in which edges are indicated by zero-crossings. In this example, the image was convolved with an LoG filter with $\sigma_{SC} = 8$ which gave an overall filter width of 69 pixels. (Note: DC padding, which is described in section 7.7, was used during this convolution to improve the quality of edges near the image borders.) In Figure 7-5(b), small positive and negative numbers are shown by bright red and bright green respectively; larger numbers are represented by darker colours. At this edge detection scale, the zero-crossings indicating the approximate lesion boundary and other edges inside and around the lesion can be seen. LoG convolution, in conjunction with zero-crossing detection, is used as part of an algorithm for finding lesion boundaries which is presented in chapter 8. Consequently, the features of fixed scale LoG edge detection are discussed at the beginning of that chapter.



(a) **(b)**

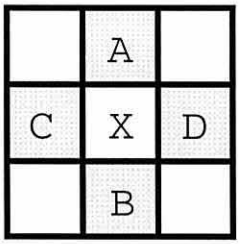
Figure 7-5: Example LoG output, with $\sigma_{SC} = 8$ and $w = 22.6$. (a) Original image (246 by 210). (b) Output of LoG convolution with DC padding.

7.4 Zero-Crossing Detection

In second derivative edge detection, the locations of the edges are indicated by zero-crossings in the output and, ideally, edges would be marked where the output is zero and a zero-crossing occurs. However, in real images the zero-crossing points are unlikely to occur at the discrete points (pixels) at which the output is sampled. In the one-dimensional case, zero-crossings are detected by searching for two adjacent points of opposite sign or three adjacent points of which the middle one is zero and the outer two have opposite signs [45]. In the two-dimensional case, care must be taken to ensure that continuous, accurately placed edges are generated and extraneous points are not produced, i.e. there is a single response to a zero-crossing. In the discrete 2-D domain there are a number of methods for marking the zero-crossing points, which include the following:

1. **Cross-shaped zero-crossing rule.**

   A simple zero-crossing rule [46] can be created by considering a cross-shaped region:



   The central pixel X is marked as a zero-crossing if AXB or CXD contains a zero-crossing. By mapping 0, -ve and +ve to 0, 1 and 2 respectively, this can be implemented as a 27 element look-up table, shown in Table 7-II, which shows two sets of rules. Figure 7-6(a) shows the result of applying the first set of rules, which produce double thickness edges and some extraneous edges, as a result of +- and -+ transitions being marked when both the +ve and -ve points are in the centre. The second set of rules, which are illustrated in Figure 7-6(b), are a modification removing the possibility of double thickness edges by marking the edge on the positive pixel when a +- or -+ transition occurs. Both of these methods have the disadvantage that edges may not be accurately placed, as the magnitude of the LoG output and diagonal zero-crossings are not considered. The next method uses four points to find the zero-crossings and indicates the edge direction.

Pixel pattern	Edge presence	
	Rule Set 1	Rule Set 2
0 0 0	no edge	no edge
0 0 -	no edge	no edge
0 0 +	no edge	no edge
0 - 0	no edge	no edge
0 - -	no edge	no edge
0 - +	edge	no edge
0 + 0	no edge	no edge
0 + -	edge	edge
0 + +	no edge	no edge
- 0 0	no edge	no edge
- 0 -	no edge	no edge
- 0 +	edge	edge
- - 0	no edge	no edge
- - -	no edge	no edge
- - +	edge	no edge
- + 0	edge	edge
- + -	edge	edge
- + +	edge	edge
+ 0 0	no edge	no edge
+ 0 -	edge	edge
+ 0 +	no edge	no edge
+ - 0	edge	no edge
+ - -	edge	no edge
+ - +	edge	no edge
+ + 0	no edge	no edge
+ + -	edge	edge
+ + +	no edge	no edge

Table 7-II: Cross-shaped zero-crossing rule look-up table.

(a) (b)

Figure 7-6: Cross-shaped zero-crossing rule output, for the image shown in Figure 7-5. (a) Rule set 1. (b) Rule set 2.

2. **Pratt Zero-crossing Patterns.**

These rules, suggested by Pratt [80], are shown in their original form in Figure 7-7(a). For implementation the bottom two lines of rules can be combined and edge direction added as shown in Figure 7-7(b). The example output shown in Figure 7-8 shows that double thickness edges are generated in horizontal and vertical directions, but the rules always result in continuous contours. The double thickness edges are due to duplication between rules A/CD, E/GH, I/KL and M/OP. These rules indicate edge direction, but do not create single thickness edges or use the magnitude, so produce multiple responses to a single edge and inaccurately place edges. The next method addresses both of these problems.

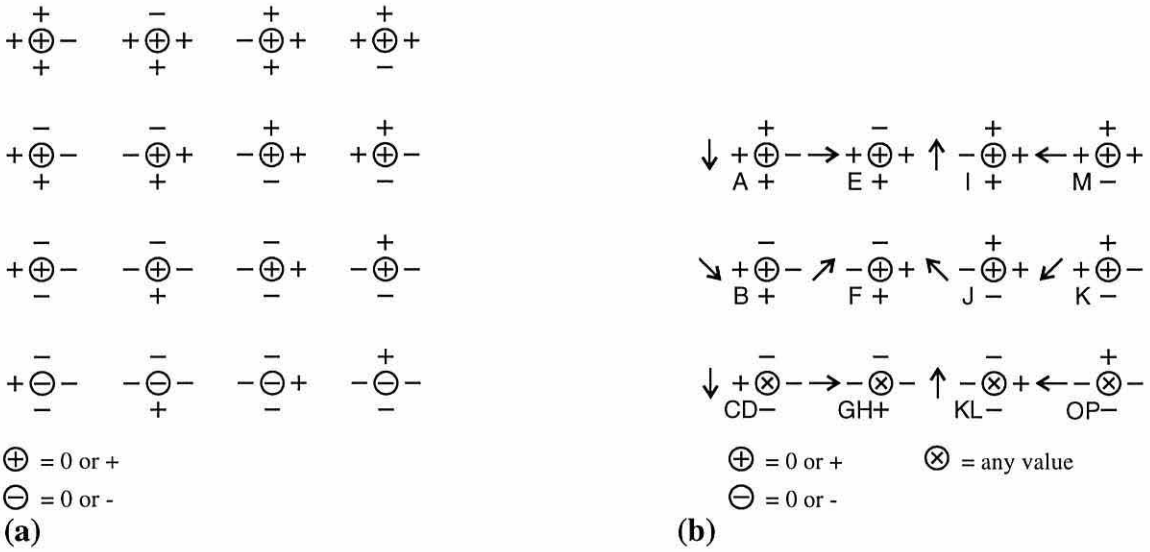


Figure 7-7: Pratt zero-crossing patterns. (a) Original patterns. (b) Simplified patterns with edge direction.

Figure 7-8: Zero-crossings detected using Pratt zero-crossing patterns, for the image shown in Figure 7-5.

3. **Huertas and Medioni Zero-crossing Detection.**

This sophisticated method [45] produces i-connected edges with 8-way directional information and is based on 11 predicates, which decide the direction and location of an edge point based on the signs and magnitudes of the pixels in a 3 by 3 area. For every point in the image (except for a one pixel border) the following two steps are applied:

A. Predicate selection

The appropriate predicate is selected using the binary decision tree shown in Figure 7-10 based on the sign of the surrounding points.

B. Predicate application

The original published predicates, with a few minor corrections[1], are shown in their original form in Figure 7-11, which combines selection and application. They are difficult to implement from these diagrams and their output contains gaps and incorrect directional information. The predicates were modified to remove the gaps and yield correct directional information by studying faults in the output of the original predicates and making appropriate modifications. These modified predicates are shown in Figure 7-12 in the form of binary decision trees, which are easier to understand and implement. An example output of the modified predicates is shown in Figure 7-13.

While testing these rules, it was found that there were sometimes gaps in the edges. Examination of the LoG output for the affected areas showed that the presence of zeros in the LoG output sometimes affected the performance of the modified HM zero-crossing predicates. To overcome this problem, the scaling in the final stage of the LoG convolution was modified. This modification changed the scaling of any value which originally scaled to zero and remapped it to ±1 when it was greater than 0.0001*(the value which exactly mapped to one). After applying this empirical modification, the presence of edge gaps was reduced considerably. These modified HM predicates produce accurately placed edges, as they consider both sign and magnitude, and are used in the edge focusing algorithm presented in the next chapter.

[1] Predicate B: The edge direction for the last case was changed from South to North.
Predicate D: Plus in pixel b changed to minus.
Predicate J: In the third line the output pixel f was highlighted.

a	b	c
d	e	f
g	h	i

Figure 7-9: Pixel names.

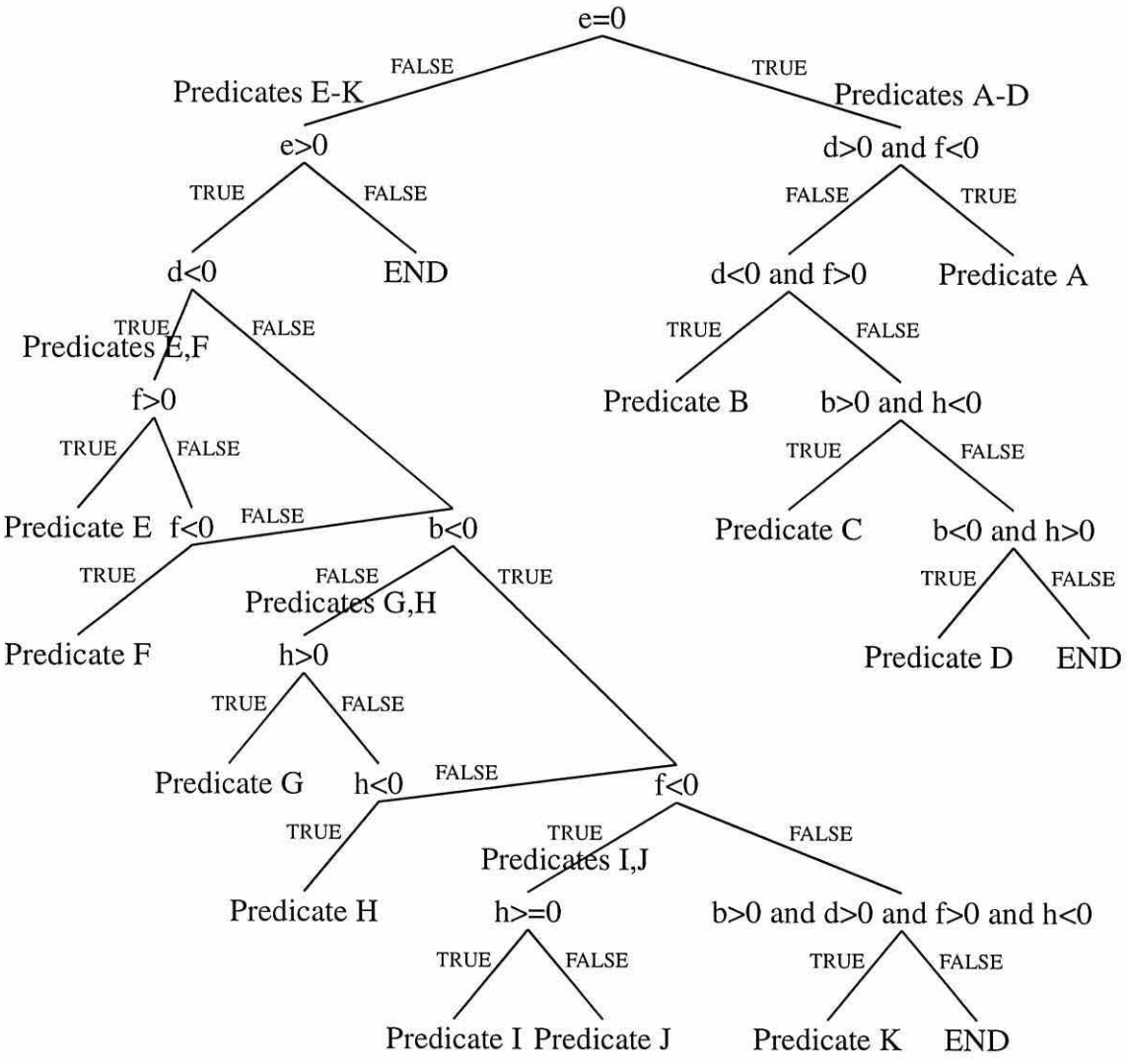


Figure 7-10: Zero-crossing decision tree. Pixel names are shown in Figure 7-9.

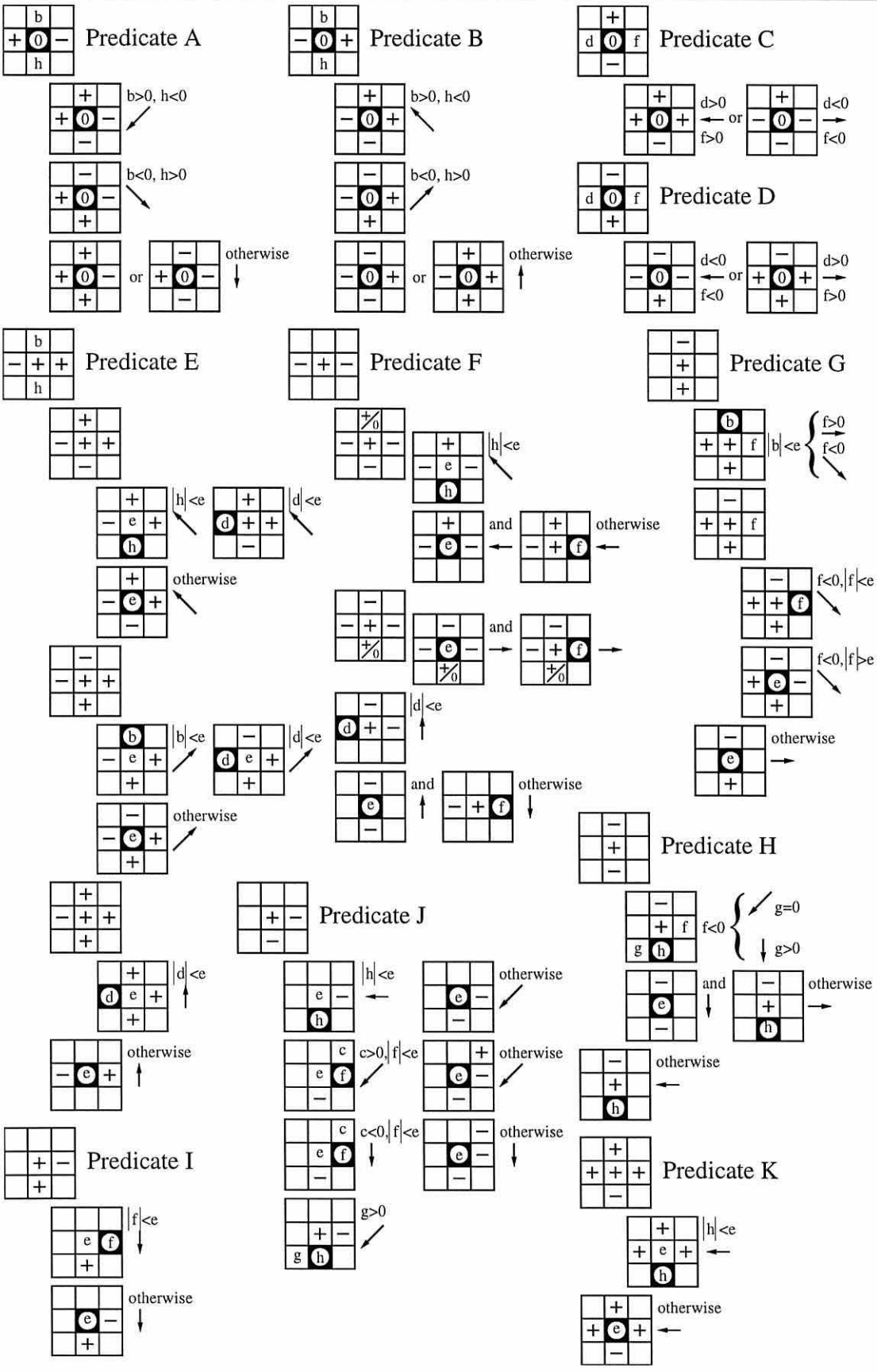


Figure 7-11: Original HM zero-crossing predicates. The highlighted pixels indicate where the edge is placed and the edge direction is given by the arrows. Pixel names are shown in Figure 7-9.

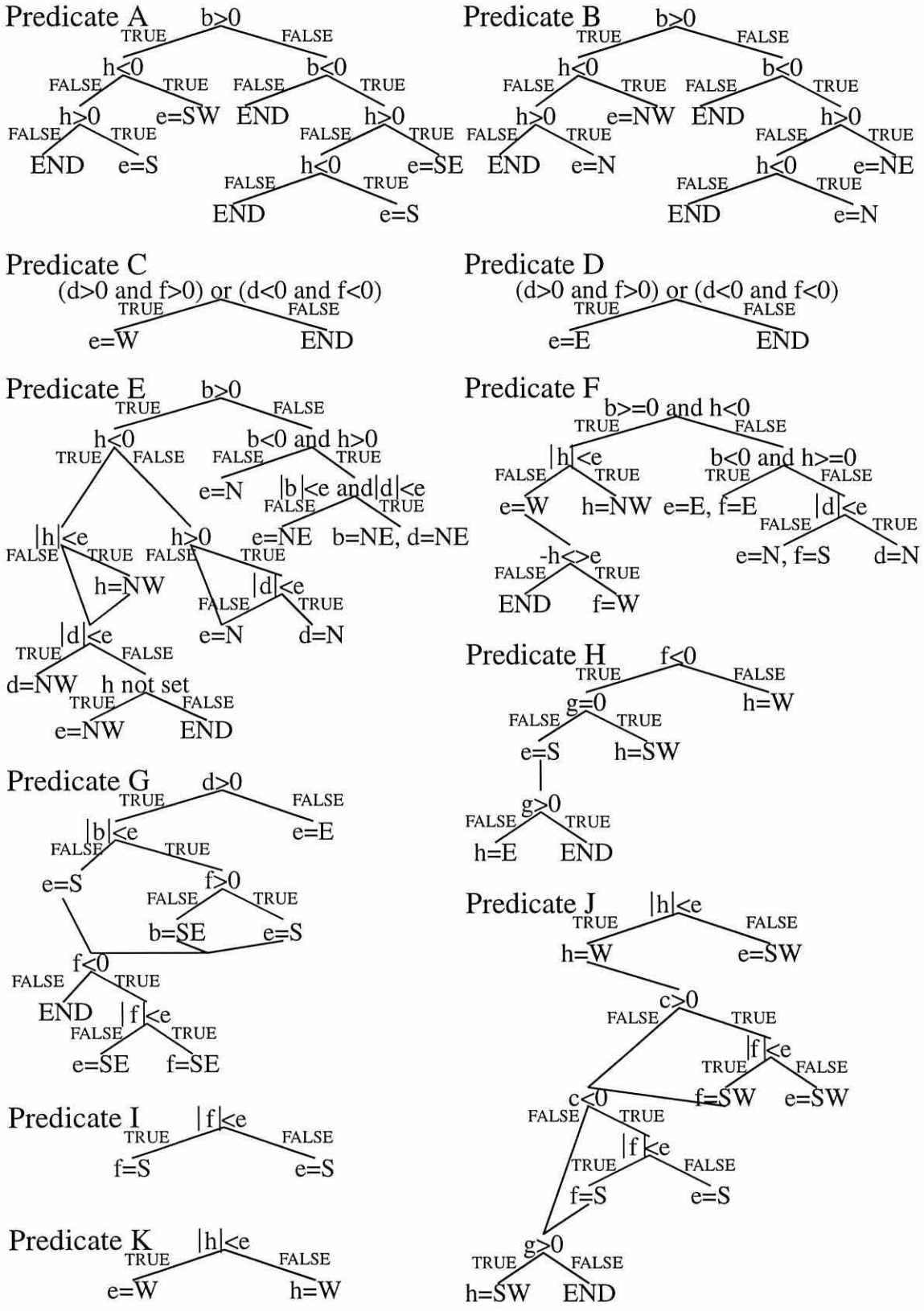


Figure 7-12: Modified HM zero-crossing predicates, shown as binary decision trees. Pixel names are shown in Figure 7-9.

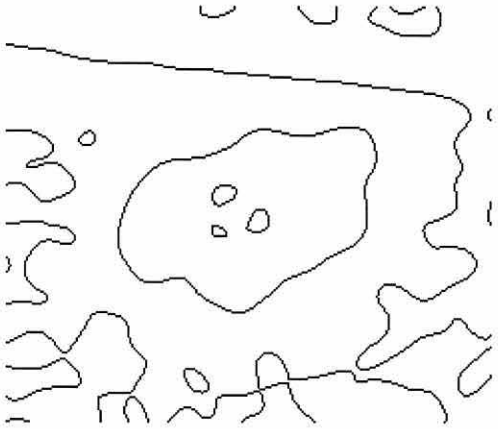


Figure 7-13: Zero-crossings detected by the modified HM zero-crossing predicates, for the image shown in Figure 7-5.

7.5 Canny Edge Detection

The Canny edge detector [10] is a first derivative edge detector with different properties from the second derivative LoG edge detector. The Canny and LoG detectors differ in their detection of ramp edges (see section 3.8) and their detection of curved edges (discussed later in section 8.6). These differences mean that one may be more appropriate than the other for a particular application, hence both were investigated.

Canny [10] took an analytic approach to designing a first derivative edge detector, based on a numerical optimisation of finite anti-symmetric function for detecting a one-dimensional continuous step edge of a given height in white Gaussian noise. Three criteria were used in optimising the function:

1. **Good detection**. The function should have a low probability of failing to mark real edge points and a low probability of marking non-edge points.
2. **Good localisation**. The points marked as edges should be as close as possible to the centre of the real edge.
3. **Single response**. There should only be one response to an edge.

The numerically obtained function is similar to the first derivative of a Gaussian ($g'(x)$), which is shown in Figure 7-14(a), and for implementation this anti-symmetric function is used:

$$g'(x) = \frac{-x}{\sqrt{2\pi}\sigma_C^{\,3}} \exp\left(\frac{-x^2}{2\sigma_C^{\,2}}\right) \tag{7.5}$$

where σ_C = scale of the edge detector.

The performance of $g'(x)$ is slightly worse than that of the optimal operator, obtained by numerical optimisation, however this difference would probably be difficult to detect in real images and $g'(x)$ can be computed with much less effort in two dimensions [10].

7.5.1 Application of Canny Edge Detection to an Image

The edge detection function $g'(x)$ was designed for 1-D edges, and is anti-symmetric so cannot be directly extended to two-dimensions as the corresponding 2-D operator would not be rotationally symmetric [10, 92]. However, by applying a Gaussian projection function (Figure 7-14(b)) perpendicular to the edge detection function the detector can be applied to 2-D images:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_C} \exp\left(\frac{-x^2}{2\sigma_C{}^2}\right) \tag{7.6}$$

where σ_C = standard deviation of the Gaussian function.

The combination of detection ($g'(x)$) and projection ($g(x)$) functions allows the location and orientation of all of the edges in a 2-D image to be found by two perpendicular edge detectors [92].

In the following operations used to apply Canny edge detection the filter coefficients, horizontal, vertical and overall gradients were stored as double precision floats and the final edge strength and direction maps, produced by the non-maxima suppression, were scaled to fit into an unsigned byte.

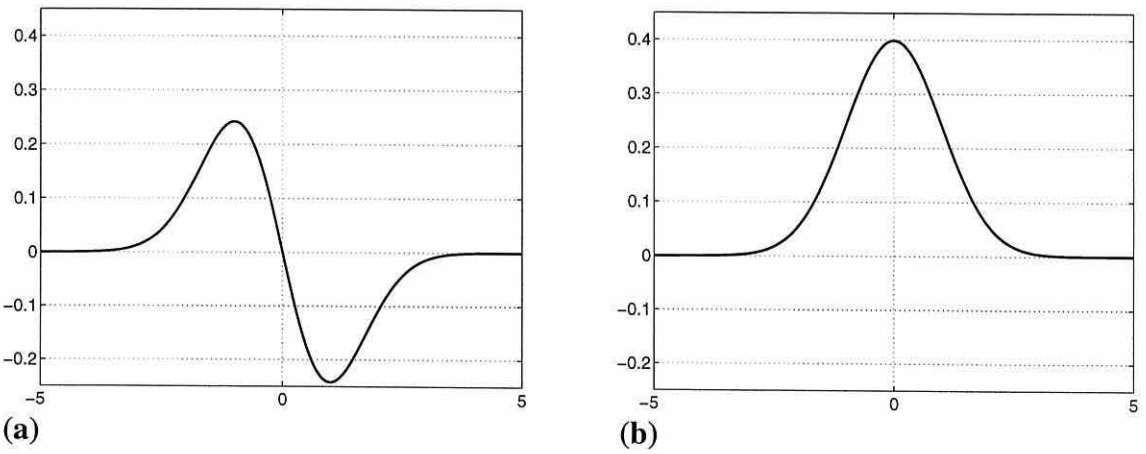


Figure 7-14: Gaussian functions. (a) First derivative of Gaussian. (b) Gaussian.

1. **Filter width calculation.**

Both $g(x)$ and $g'(x)$ have infinite extent, so must be truncated before application to an image. The filter widths can be obtained by considering the area under each function. The area under $g(x)$ from minus infinity to plus infinity is one and the area under $g'(x)$ from zero to plus infinity is one. From the integrals the following expressions can be obtained for the number of coefficients required for half the filter width, including the central point:

$$g_W = \lceil \sqrt{2}\sigma_C \operatorname{erfinv}(c_P) + 0.5 \rceil + 1 \tag{7.7}$$

$$dg_W = \left\lceil \sigma_C \sqrt{-2\log\left(\sqrt{2\pi}\,\sigma_C (1 - c_P)/2\right)} + 0.5 \right\rceil + 1 \tag{7.8}$$

where g_W = number of coefficients in half the Gaussian filter,
dg_W = number of coefficients in half the first derivative of Gaussian filter,
c_P = portion of the area under the curve which is included, which is set to 0.99,
erfinv(x) = inverse error function [91].

2. **Calculate filter coefficients.**

The filter coefficients are computed using the symmetry of $g(x)$ and the anti-symmetry of $g'(x)$. The following expressions use the integral to obtain an average value for each coefficient:

$$\hat{g}(x) = \frac{\operatorname{erf}\left((x + 0.5)/\sqrt{2}\sigma_C\right) - \operatorname{erf}\left((x - 0.5)/\sqrt{2}\sigma_C\right)}{2} \tag{7.9}$$

$$\hat{g}(g_W - 1) = \frac{\operatorname{erfc}\left((g_W - 1.5)/\sqrt{2}\sigma_C\right)}{2} \tag{7.10}$$

$$\hat{g}'(x) = \frac{\exp(-(x + 0.5)^2/2\sigma_C{}^2) - \exp(-(x - 0.5)^2/2\sigma_C{}^2)}{\sigma_C\sqrt{2\pi}} \tag{7.11}$$

$$\hat{g}'(dg_W - 1) = \frac{-\exp(-(dg_W - 1.5)^2/2\sigma_C{}^2)}{\sigma_C\sqrt{2\pi}} \tag{7.12}$$

where $\hat{g}(x)$ = estimate of $g(x)$,
$\hat{g}'(x)$ = estimate of $g'(x)$,
erf(x) = error function [91],
erfc(x) = 1 - erf(x) = complementary error function [91].

The final coefficients, obtained by integrating from x to infinity, contain the area under the rest of the curve which ensures that the total area is correct. The sum of these coefficients is used to obtain the maximum edge strength, which is used in the non-maxima suppression.

3. **Gaussian smoothing in the vertical direction by convolution.**
   The Gaussian function's symmetry enables the number of multiplications to be reduced.

4. **Calculate horizontal gradient i_X.**
   The horizontal edge gradient is obtained by convolving the first derivative of Gaussian with the vertically Gaussian smoothed image. The smoothing (projection) and detection functions are associative and so can be applied in either order. The detection function is anti-symmetric and is zero valued at its centre, which reduces the number of multiplications and additions required.

5. **Gaussian smoothing in the horizontal direction.**

6. **Calculate vertical gradient i_Y.**

7. **Calculate edge strength,** $e_S(x,y) = \sqrt{i_X(x,y)^2 + i_Y(x,y)^2}$.

8. **Non-maxima suppress.** This is described in section 7.5.2.

After non-maxima suppression, hysteresis thresholding [10] based on the edge strength can be used to remove isolated false edge points and generate continuous edges. This is not described as the edge focusing algorithm uses other assumptions which are specific to the problem to remove unnecessary points.

7.5.2 Non-maxima Suppression in the Gradient Direction

When using a first derivative edge detector, such as Canny, the edges are indicated by maxima in edge gradients. To produce an edge map it is necessary to locate these maxima by comparing each point with its two neighbours (a_1 and a_2) which are along the edge gradient direction and at unit distance from the point. To calculate these two

points, the non-maximum suppression performs the following process, using a 3 by 3 neighbourhood, at every point:

1. **Check edge strength.**

   The edge strength is examined and those points with very small edge strengths are ignored, as they correspond to very weak (faint) edges. When used in the edge focusing algorithm, presented in the next chapter, the minimum edge strength was set to half the minimum representable output, as any value smaller than this would be rounded to zero. In other applications, the minimum edge strength may be set to larger values to prevent the detection of weak edges.

2. **Determine amplitudes of the two neighbouring points a_1 and a_2.**

   The horizontal and vertical edge gradients of the central point give the edge direction which determines the location of the two neighbouring points a_1 and a_2, which are placed on the lower and higher intensity sides of the edge respectively. From neighbouring points' location, the four points in the 3 by 3 neighbourhood needed to calculate the edge strengths of the neighbouring points (a_1 and a_2) can be found. Figure 7-15 shows the points used for an edge pointing in an approximately south-east direction. The edge strengths of a_1 and a_2 were obtained by bilinear interpolation [81] of the four appropriate edge strengths. Bilinear interpolation is simple to implement and provides reasonable interpolation which varies smoothly from pixel to pixel. However, the gradient of the interpolated function has discontinuities at the boundaries of each pixel, e.g. when point a_1 moves from pixel b to c.

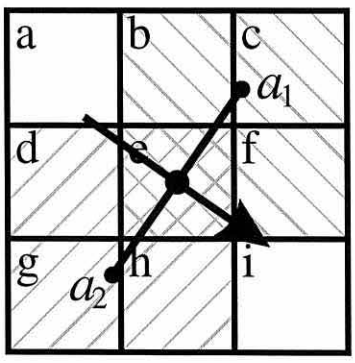


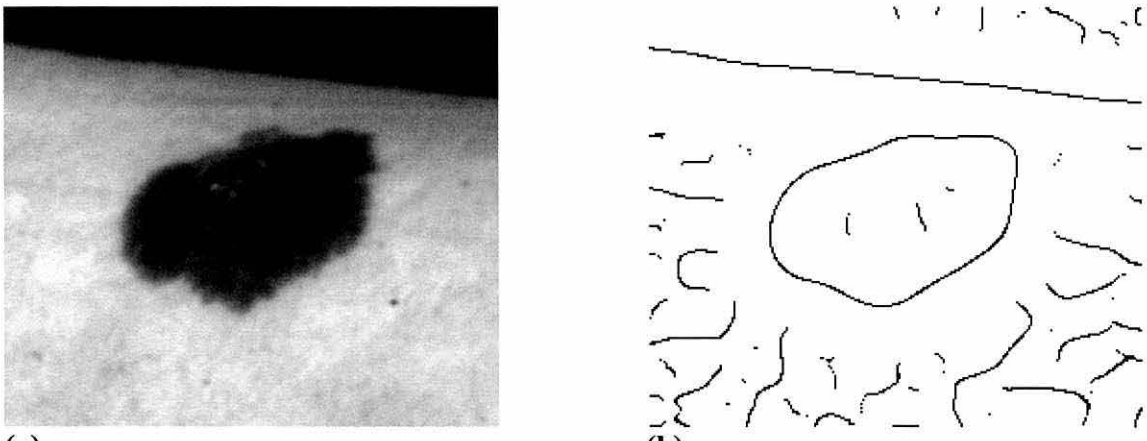
Figure 7-15: Non-maxima suppression. Hatching shows the points used for calculating a_1 and a_2.

3. **Check for maxima in gradient direction.**

   If the central point's edge strength is greater than or equal to a_1 and is greater than a_2, it is marked as an edge and its edge direction calculated from the vertical and horizontal edge gradients. The edge strength and direction were scaled, using the maximum edge strength, to fit into an unsigned byte. The placing of a_1 and a_2 on lower and higher intensity sides of the edge respectively combined with the greater than or equal test for a_1 ensures that edges are marked on the higher intensity side.

As the non-maxima suppression uses a 3 by 3 neighbourhood, it is necessary to increase the size of the area over which the edge gradients are calculated by adding a single pixel border. When a mask (see section 7.8) is used to specify where edges are required, the mask must be dilated by one pixel to allow for this.

Figure 7-16 shows the results of Canny edge detection using the methods described in this section. Comparison of Figure 7-16 and Figure 7-13 illustrates the differences between similarly scaled Canny and LoG edge detections. The two detectors place the lesion boundary differently due to the differing effects of curved edges [92]. (This is discussed later in section 8.6.)



(a) (b)

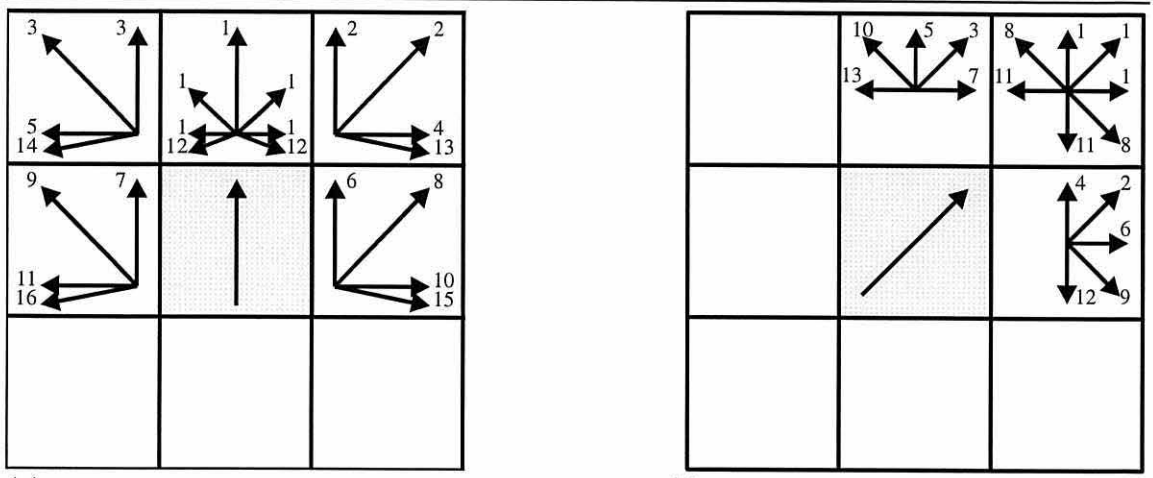
Figure 7-16: Canny edge detection, with $\sigma_C = 8$. (a) Original image (246 by 210). (b) Canny edge detector output.

7.6 Edge Tracing

Edge tracing (following) converts an edge map, containing isolated edges, into co-ordinate lists which are an early stage in analysing an image and can be used for calculating parameters such as a region's area. Edge tracing is distinct from contour tracing, which traces region boundaries and was used in section 6.6. The output of edge tracing should be either continuous i-paths or d-paths with closed contours and isolated edges identified. Closed contours and crossing line segments must not affect its operation. The edge direction or line direction, which are the direction of each point and the direction obtained from the relative position of the previous point on the line respectively, may assist in following an edge. The edge map should contain a single response for each edge; multiple responses can be removed during the zero-crossing detection or non-maxima suppression. The methods described here do not bridge gaps in edges, which is considered in the next chapter, and can be used to trace multiple edges. When tracing multiple edges, each edge is deleted after it has been found and the process repeated.

7.6.1 Edge Tracing using Directional Information

This method uses 8-way directional information and i-connected (8-way) edges, which are provided by the modified Huertas and Medioni zero-crossing method (described in section 7.4), but could be used with other edge detectors that provide edge direction output. Its first stage is to find a starting pixel anywhere on an edge by using a left to right, top to bottom scan of the image. Once this starting pixel has been found, its neighbours are determined. The edge tracing is performed on an 8-way connected image and thus two types of neighbours exist: i-neighbours and d-neighbours, which simplifies finding the next point on an edge, as the search can be reduced to examining two directions: north and northeast. The current point's edge direction is used to rotate it and its neighbours by a multiple of 90°, so it points either in a north or northeast direction, which aligns it with Figure 7-17(a) or Figure 7-17(b) respectively. A list of directions to available next points is created, with those involving the smallest turn first, by searching in the order shown in Figure 7-17. The directions to the next points are rotated back to compensate for the rotation performed to align the current point in a north or northeast direction. This pre- and post- rotation reduces the complexity by removing the requirement for two sets of four similar rules. From these directions, the co-ordinates of available next edge points are obtained and these points checked against the points already



(a) (b)

Figure 7-17: Edge trace, using edge direction, search order. (a) For north. (b) For northeast.

visited to ensure that closed contours are detected and each part of the edge is traced once. After removing points already visited the first remaining next point, which is the one requiring the smallest change of direction, is used as the next point on the edge. This process is repeated for the rest of the edge and once the end of the edge is reached the search continues in the opposite direction from the start of the edge. To allow the same set of rules to be used again, the direction of the current point and its neighbours are rotated by 180°. Once both sections of the edge have been traced, the co-ordinates are re-arranged to follow the direction of the edge and to be continuous, i.e. each co-ordinate in the list is next to its spatial neighbours.

7.6.2 Edge Tracing for 4-way Connected Edges without using Directional Information

This method is designed for d-connected (4-way) edges, which are provided by the non-maxima suppression method used in Canny edge detection, and does not use edge direction. When the previous method was tested on the output of the non-maxima suppression method some edge points were missed because the edges tended to be 4-way connected. Hence, this new method was developed and the differences from the previous method are explained here. It is similar to the previous method, except that each pixel's edge direction is not used and instead the relative position of the previous pixel on the line is used to guide the search for next point, on the basis that the line will continue in the direction it is already going or will make the smallest possible turn, with 4-way connected points preferred. The relative position of the previous point on the line determines the direction in which the line is travelling, except at the start where the line

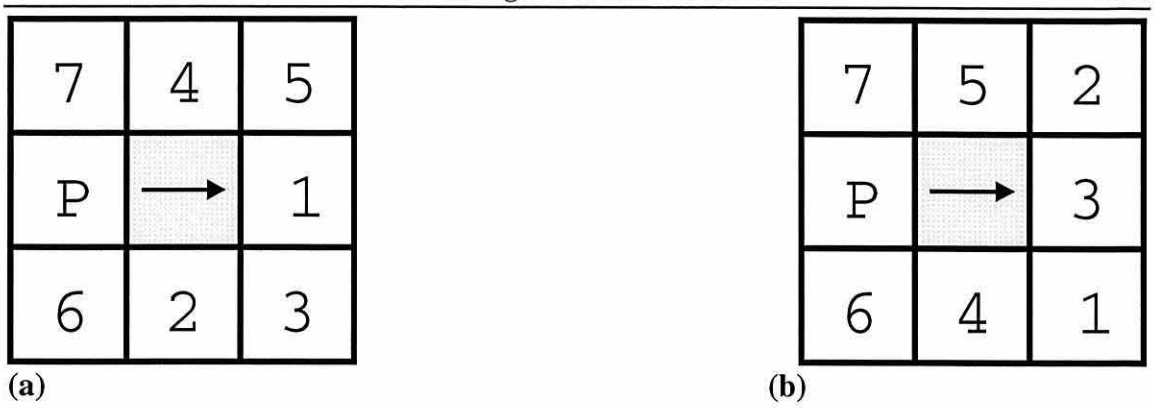


Figure 7-18: Edge tracing, without edge direction, search order. P = previous point. (a) For horizontal and vertical directions. (b) For diagonal directions.

is assumed to be going horizontally in the positive x direction. This relative position is used to rotate the neighbourhood by a multiple of 45° to place the previous point in position P shown in Figure 7-18. The next point is searched for in the order shown in Figure 7-18(a) or (b) depending on whether the previous point is horizontally/vertically or diagonally adjacent to the current point. A list of possible next points is created in order of smallest turn first. The first point, which has not been visited already, is used as the next point on the line. The same pre- and post- rotation is used to simplify the search for the next point and the edge can be traced in two sections in the same manner as used in the previous method.

7.7 Image Border Erosion

When a mask of M by M pixels is convolved with an image, $\lfloor M/2 \rfloor$ pixels are lost from the border, as output from this area would require data from outside the image. For small filters this may not be significant; however, for large filters, for example edge detectors which use 20-30% of the image, this loss is significant. For example, an LoG filter with $\sigma_{SC} = 10$ requires a mask of 85 by 85 which when convolved with a 512 by 512 image results in an output image of 428 by 428 which is a 30% loss in image area. The significance of this loss depends on the importance of the information in the image borders and the size of the mask. The loss complicates the building of an image description based on information from several scales as some information will be missing at larger scales. The examples in this section are based on LoG edge detection, but similar problems occur for other filter based edge detectors.

To obtain information from the image borders, assumptions based on the available image data must be made about the area outside the image. The data outside the image can be assumed to be similar to that inside the image, so may be approximated by a simple function. Any approximation made must have minimal impact on edges inside the image, i.e. it must not create new edges and have minimal impact on the location, direction and strength of existing edges. As a minimum requirement, a uniform input image must produce a uniform output. The following methods, which are illustrated in Figure 7-19, can be used to handle image borders [89]:

1. **Zero padding or constant padding.**

   This is the simplest form of padding, where the image is surrounded by zeros or another constant value. An example of zero padding for edge detection is shown in Figure 7-19(c). It can generate extra false edges, especially for bright regions near the image border, as a result of false luminance changes. It may also interfere with other edges near the border, causing their location, direction and strength to change. This method is simple, but inadequate because it can introduce extra edges and cause significant distortion of existing edges.

2. **DC padding by border replication.**

   In this method, the image is extended by replicating the border pixels. This is a zero order 1-D extension and assumes that the border pixels are representative of the local image. Figure 7-19(d) shows the edges detected using an LoG convolution which incorporates DC padding. With this method, a constant input produces a constant output. Hence, it is an improvement on the first method, but remains simple to implement. Edges parallel to the image border are less affected than edges at oblique angles, as luminance changes across and parallel to the border are not generated.

3. **Extension by a higher order approximation.**

   It is possible to extend the border by a higher order extension such as a quadratic or cubic spline, which could increase the accuracy of the edge location and strength at the expense of increased computation. This method was not tested because of the increased computation involved.

As the image padding is approximate it affects the location, strength and direction of edges near the borders. This effect can be reduced by combining the edges obtained on a

number of scales [3, 64]. At smaller scales, fewer edges are affected and those influenced are less affected; at larger scales, more edges are affected with greater distortion.

These methods may be incorporated into the image processing algorithm or be applied before any processing. If the image padding is applied before processing the padding is calculated only once, it increases the size of the image and is not adaptable to have minimal impact on a particular algorithm. When the padding is incorporated into the algorithm, it requires less memory and can be adapted to the algorithm. However, it increases the complexity of the algorithm, hence possibly reducing its speed. DC padding was incorporated into the LoG and Canny convolution algorithms, described earlier in this chapter and used in the next chapter.

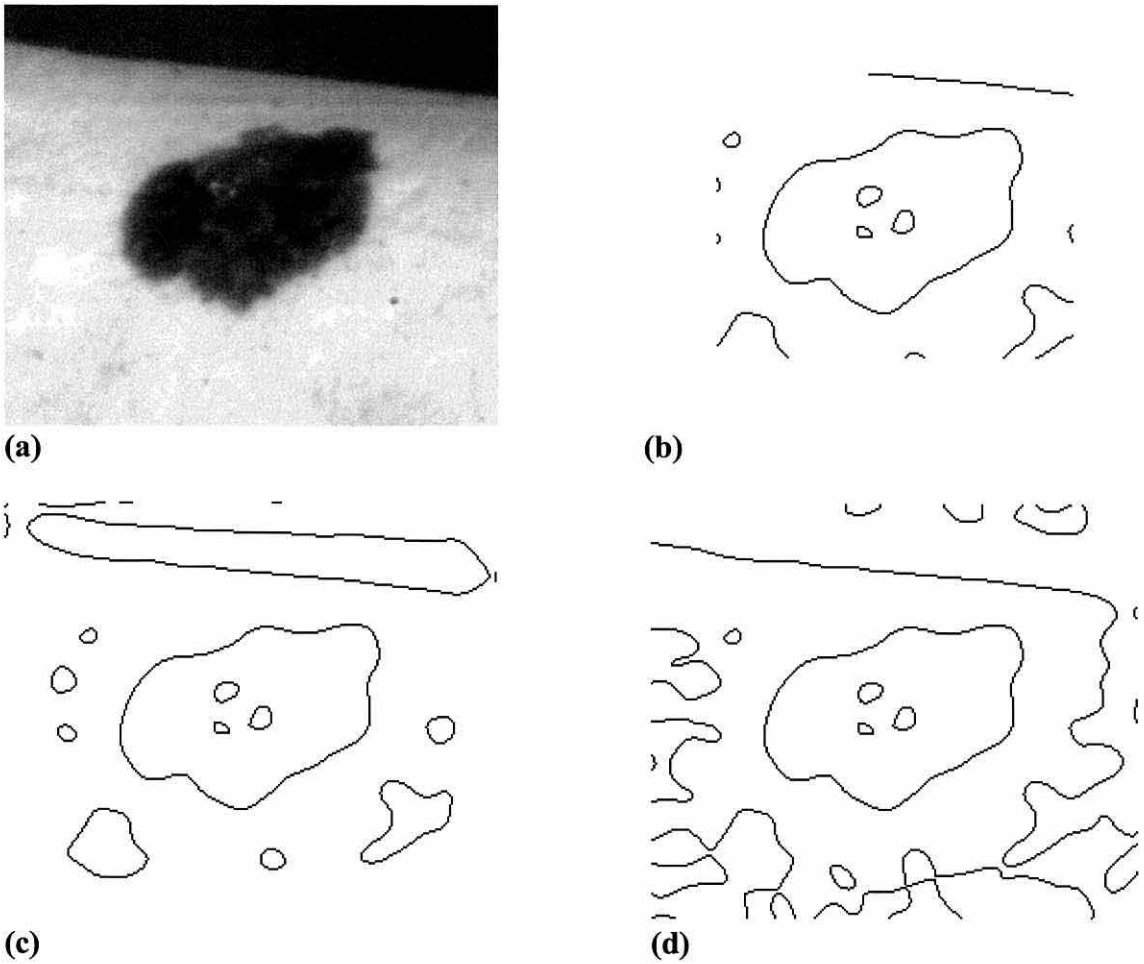


Figure 7-19: Image padding for LoG edge detection. (a) Original image (246 by 210). (b) Edges detected without padding, 34 pixels are lost from each side. (c) Edges detected with zero padding. (d) Edges detected with DC padding. LoG performed with $\sigma_{SC} = 8$ giving a filter width of 69 pixels.

7.8 Image Processing of an Irregularly Shaped Area

Whilst analysing an image, image processing may be required in only part of the image, i.e. the image is segmented into regions of interest and regions that are not currently of interest. For example, during coarse-to-fine or fine-to-coarse edge focusing, existing edges are refined by detecting edges on and adjacent to the existing edges. These regions can be defined in the following ways: a rectangular sub-image, an irregularly shaped area, a list of points and a list of points with their neighbourhoods (e.g. coarse-to-fine edge focusing). The rectangular sub-image case is a simple modification of processing the whole image. It is beneficial to minimise the number of points processed, as the effort in processing each point is considerably greater than that of selecting points. The following methods for processing an irregularly shaped area are intended for edge detection, but could be used in other tasks:

1. **Process the whole image and use only the required output.**

   This is the same as processing the whole image, but with a post processing stage which selects the required output.

2. **Process each point or sub-image individually.**

   Each point is processed in turn, treating it as the whole image. This is slow as there is no sharing of information between adjacent points, but can be slightly improved by grouping vertically and horizontally adjacent points into rectangular sub-images. These sub-images do not share information with adjacent regions, which for a large scale edge detector introduces considerable inefficiency.

3. **Mask.**

   Masks that specify where processing is required are drawn on rectangular sub-images, which bound them. During processing, the mask is checked to determine if a particular point requires processing. By appropriately dilating a mask, the area which is needed, for example, for an edge detection convolution (Figure 7-20(b)) can be found from the area where edge detection (Figure 7-20(a)) is required. This method was used and is described in the next chapter. Masks are generally the fastest approach to processing an irregularly shaped area as only the necessary points are processed.

(a) (b)

Figure 7-20: Example image processing masks. (a) Example edge detection for edge focusing. (b) A mask indicating where LoG convolution is required for the edge detection mask (a).

7.9 Conclusions

This chapter examined some methods of edge detection and forms the background for the work presented in the next chapter. It started with a description of simple edge detectors, such as Sobel, which showed their unsuitability for lesion boundary detection. The implementation of the LoG second derivative edge detector was described. Three methods of zero-crossing detection were examined and improved versions of the first two methods (a cross-shaped ZC detector [46] and Pratt's ZC detector [80]) were presented. The final, more sophisticated, method (Huertas and Medioni zero-crossing detection [45]) was found to give the best results and was refined to give more accurately placed and continuous zero-crossings. A clearer way of describing this method, using binary decision trees which allows easier implementation, was presented. An implementation of the Canny edge detector, which used integrals to find the number of coefficients and their values, was presented. In addition, an implementation of non-maxima suppression using bilinear interpolation was given. Two methods of edge tracing, which are suited to the output of the LoG and Canny edge detectors, were presented. The effects of image border erosion on edge detector output were considered and DC padding selected as the most suitable method of improving the output, near the image borders, for the LoG and Canny edge detectors. The image processing of irregularly shaped areas was examined and the use of masks selected as the most suitable method of processing these areas. In the next chapter, masks are used for a number of purposes and further explanations of their use are given where required.

Chapter 8

An Edge Focusing Algorithm for Skin Lesion Boundary Detection

8.1 Introduction

Accurate and reliable boundary detection is important in the automated diagnosis of skin lesions, in order to segment the image into lesion, skin and other background, thereby ensuring that colour and texture measurements are carried out only on the lesion image. In addition, given an accurate outline, the important diagnostic factor of lesion shape [59] can be analysed to provide quantitative measurements of size, asymmetry [90], border irregularity [37, 12] (chapter 5) and roundness [12] (chapter 5). The characteristics of skin images are very variable (e.g., lighting source, lesion size and nature, skin texture, hair and background objects) which produces problems in obtaining methods which are reliable, repeatable and robust. In many cases, the position of the "best" boundary is very subjective. However, an objective and consistent definition of the boundary location, which can be reasonably implemented, is required for an image processing system.

Researchers developing automated diagnostic systems [55, 29, 53] have found current automatic boundary detection methods to be inadequate. Consequently, they have used manual boundary detection as the starting point for many of their diagnostic measurements. This means that they have not fully automated the diagnostic process as dermatologists are required for some diagnostic measurements. To increase the ease of use and repeatability whilst reducing usage costs, it is desirable to automate boundary detection to reduce the reliance manual input. To fulfil this requirement, this chapter presents a new algorithm for finding lesion boundaries, which starts with the information provided by the process described in chapter 6: an image containing the lesion with sufficient sur-

rounding image with an indication of the approximate lesion size, location and bounding area.

General filter based edge detectors, for example the Sobel [80] and the Laplacian of Gaussian (LoG) [65, 64] operators (chapter 7), with fixed sizes are unreliable because of the nature of the image. Small scale edge detectors provide large numbers of unwanted edges by detecting image features such as skin texture and hair. Even when a complete boundary can be found there are a large number of other edges present which makes it difficult to identify the genuine border. If the image is low-pass filtered or a large scale edge detector used the resulting boundary is less affected by the unwanted image features, but the locational accuracy is reduced as a result of inadequate resolution. These effects are shown in Figure 8-1. A method that accurately finds the border and is tolerant of image noise is required. This can be achieved through the use of "edge focusing", which is the coarse-to-fine tracking of edges [3]. In this method, the final edges consist of refined versions of those detected by a large scale edge detector, but have the locational accuracy of those produced by a small scale edge detector. Bergholm [3] presented the mathematics used for the gradual edge focusing from coarse (large scale) edges to fine (small scale) edges and applied it to arbitrary general images. In the algorithm presented here, edge focusing is applied to medical images, in particular skin lesion images, for the first time. The algorithm uses new methods to control the boundary during edge focusing and to select the output boundary.

Two variations of an edge focusing algorithm, designed to find skin lesion boundaries, using LoG and Canny [10] edge detection were developed, because these two types of edge detector (first and second derivative) have different properties. These two scaleable edge detection methods were described in the previous chapter and their differences in detecting ramp and curved edges are explained in sections 3.8 and 8.6 respectively. The algorithm is presented using the LoG edge detector and then the modifications required to use the Canny edge detector are explained.

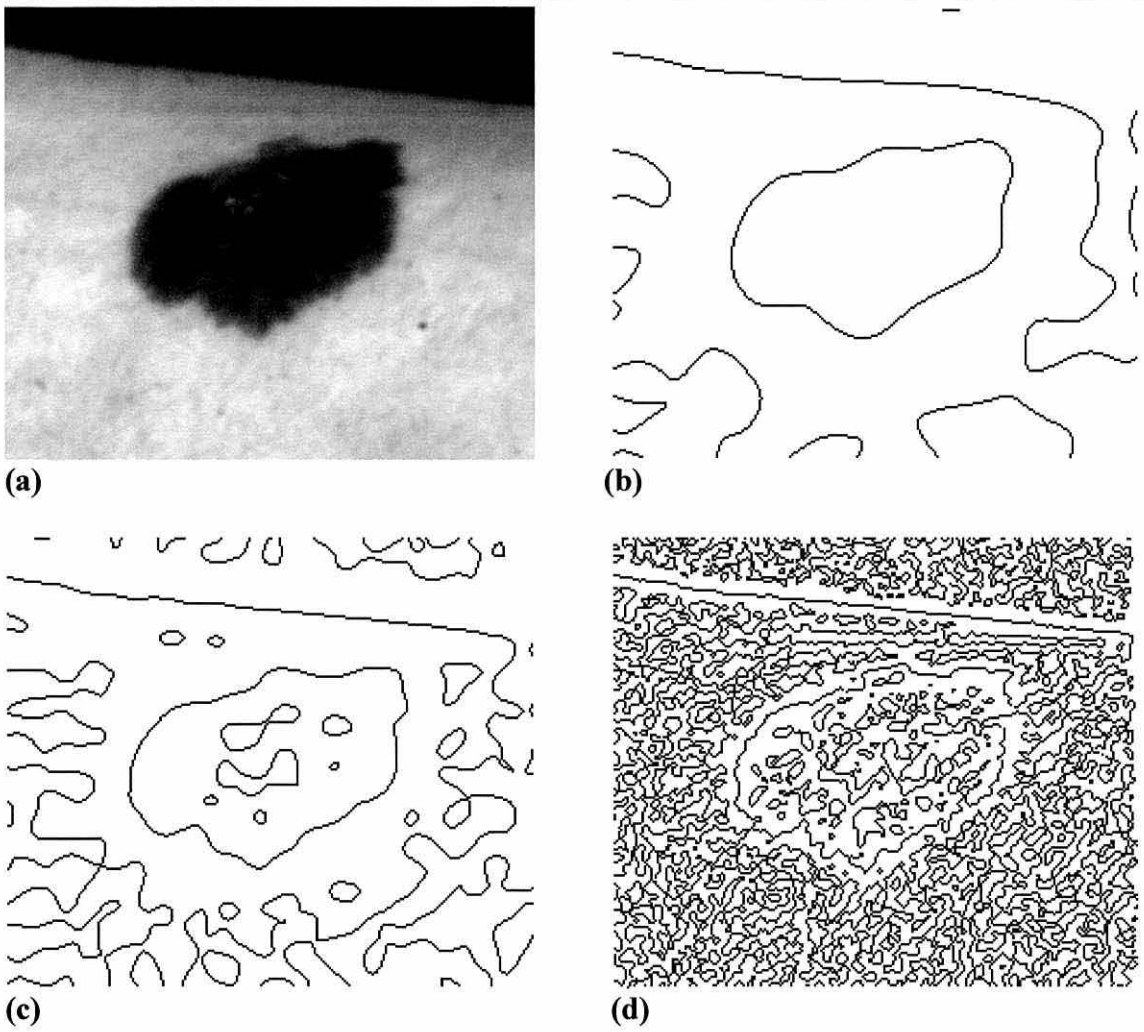


Figure 8-1: Fixed sized edge detection, using LoG operator. (a) Original image (246 by 210). (b) Large scale edge detection $\sigma_{SC} = 8\sqrt{2}$. (c) Medium scale edge detection $\sigma_{SC} = 4\sqrt{2}$. (d) Small scale edge detection $\sigma_{SC} = \sqrt{2}$.

8.2 Algorithm Outline

The edge focusing algorithm starts by applying a large scale LoG edge detector to the lesion bounding rectangle in the cropped image, provided by the process presented in chapter 6. The initial lesion boundary is selected from the detected edges by the process explained in section 8.4. This boundary is gradually refined by decreasing the LoG's scale and at each step detecting new edges only adjacent to or on the existing boundary [3], thereby preventing the detection of new unwanted edges. At each step the boundary is "cleaned", by the same method used for selecting the initial boundary, which ensures a closed contour is always produced. This creates a series of boundaries, with increasing detail, from which the most suitable is selected by the method explained in section 8.5. The edge focusing algorithm is outlined in the following steps:

1. Initial edge detection - by performing LoG edge detection using the initial value of σ_{SC} from the process described in chapter 6.
2. Edge cleaning - to select initial approximate boundary. The initial boundary selection and subsequent "cleaning" are performed by joining adjacent edges (using a cubic spline), selecting the largest closed contour and removing any redundant pixels (by drawing and re-tracing the boundary) to give an 8-way connected boundary.
3. Store boundary. Keep a copy of the current boundary.
4. While space constant $\sigma_{SC} \geq$ minimum space constant do steps 5 to 8. The minimum space constant ($\sigma_{SCMin} = 2\sqrt{2}$) corresponds to a filter sensitive region width ($2\sqrt{2}\sigma_{SC}$) of 8 pixels, which was chosen to prevent the edge focusing detecting small scale skin texture.
5. Decrease space constant by 0.5 - which ensures that the boundary moves, at most, one pixel [3]. This enables refinement of the existing edge, whilst preventing detection of new unwanted edges.
6. LoG edge detection on and adjacent to existing boundary.
7. Boundary cleaning - to join gaps in the boundary and remove unneeded edge points (see step 2).
8. Store boundary. Keep a copy of the current boundary.
9. Selection of the "best" boundary - This relies on examining the image near the boundary. The boundary with the greatest contrast between the average intensity just inside and just outside the border is selected, corresponding to a minimum in the intensity ratio because the lesion is darker than the surrounding skin. The width of the area is based on the size of the initial edge detector and so scales with the lesion size.

8.3 Initial and Subsequent Edge Detections

The initial edge detection is performed in the lesion bounding rectangle, identified by the process described in chapter 6, and produces edges such as those shown in Figure 8-3(a). The edge map (which indicates the position and direction), produced by the LoG convolution and zero-crossing, is traced, by the method described in the previous chapter, to create co-ordinate lists with the isolated lines and closed contours identified. Although the LoG is a rotationally invariant second derivative operator and hence always produces closed contours in an unrestricted image [92], the image is restricted, hence non-closed contours, i.e. isolated lines, can occur because part of a contour may be outside the image. This in conjunction with the ZC detection (section 7.4) occasionally

marking edges incorrectly creates isolated lines and closed contours which are processed, by the method explained in the next section, to create the initial boundary. After processing, the initial boundary will be a closed contour around the lesion providing that the lesion is completely contained within the lesion bounding rectangle and it has been correctly identified from the other closed contours.

Subsequent edge detections are performed on and adjacent to the current boundary, using a three pixel wide line mask created from the current boundary to identify where edge detection is required. This edge detection mask is sized and placed to just surround the current boundary, with sufficient space to include a two pixel margin to allow the zero-crossing detection to work correctly. An example boundary and edge detection mask are shown in Figure 8-2(a) and Figure 8-2(b). The mask was originally created by drawing the current boundary and morphologically dilating it by one pixel. However, binary dilation is a time consuming operation, so instead, the edge was drawn directly 3 pixels wide. (The advantages of this approach are discussed in section 8.5.) From the edge detection mask, a mask indicating where LoG convolution is required can be obtained by dilating with a 5 by 5 square. This adds two pixels to either side of the 3 pixel width line, giving a total width of 7 pixels. The width of the line is determined by the area required to allow the modified HM zero-crossing detection (section 7.4) to function. Each zero-crossing predicate uses a 3 by 3 pixel area and usually places its output in the centre of that area. However, some predicates do not place their output in the centre, hence an extra pixel is required to accommodate this. Hence, when edges are to be detected across a 3 pixel wide line, the convolution is performed across a line 7 pixels wide. The convolution mask, shown in Figure 8-2(c), is directly obtained from the current boundary without the intermediate creation of the edge detection mask to reduce the complexity and save time.

The LoG convolution is performed by 1-D convolutions to reduce the computation (see section 7.3). Although the final output is only required in the area specified by the convolution mask, the second set of 1-D column convolutions uses the output of the set of first 1-D row convolutions from outside this area. To create the mask for the first set of 1-D convolutions, the convolution mask is vertically dilated downwards and upwards by half the filter width (rounded down to the nearest integer). This is illustrated in Figure 8-2(d) for a filter of 21 pixels wide, i.e. it is vertically dilated by 10 pixels.

After performing the convolution, the zero-crossing predicates are applied where any part of the 3 by 3 predicate area overlaps the three pixel wide edge detection mask. This creates a new edge map which is traced (section 7.6.1) and the resulting co-ordinate lists processed to create a new lesion boundary with greater detail. Between each edge detection stage the space constant (σ_{SC}), which controls the detail of the edges produced, is reduced by 0.5. This ensures the boundary moves at most by one pixel [3], which enables it to be re-detected on the next pass while not introducing spurious edges not connected to the lesion boundary.

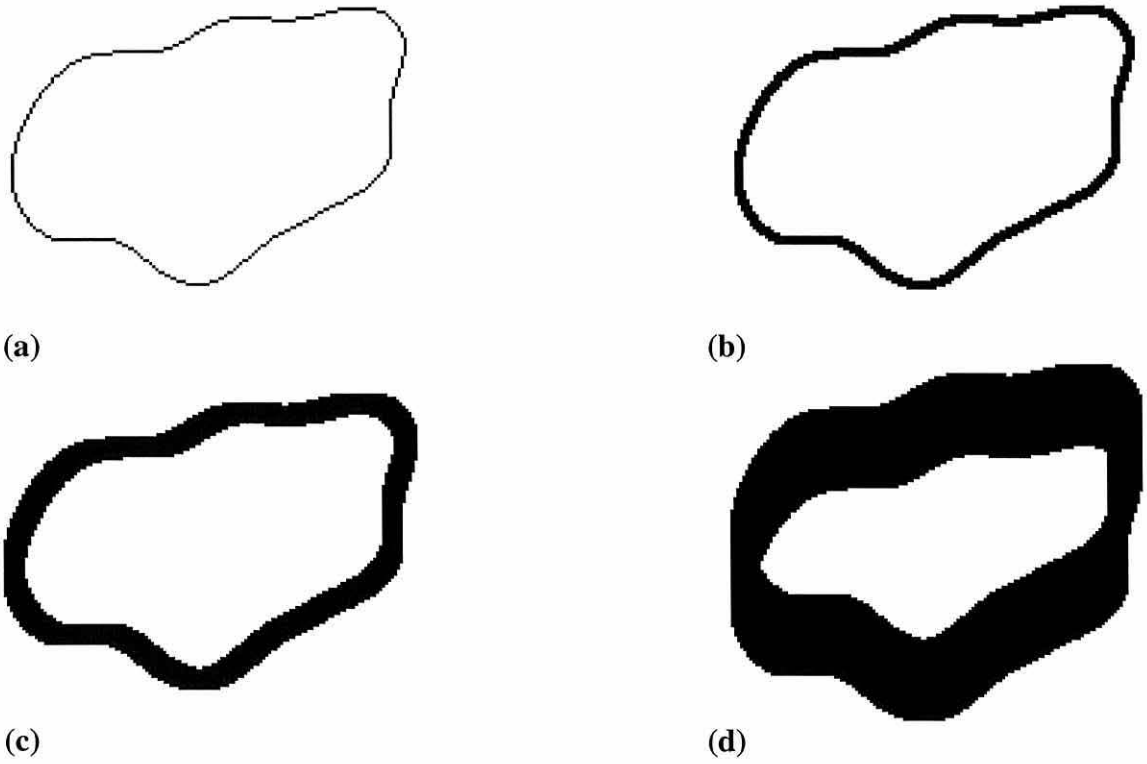
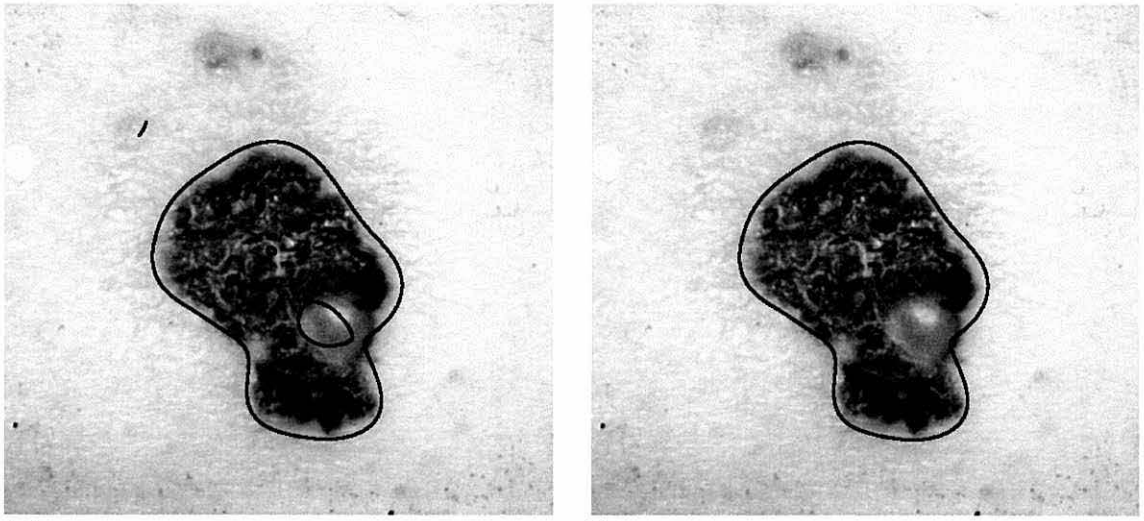


Figure 8-2: Edge detection masks. (a) Original boundary. (b) Edge detection mask, which is (a) drawn 3 pixels wide. (c) Convolution mask, which is (a) drawn 7 pixels wide. (d) Mask for the first set of 1-D convolutions, which is (c) vertically dilated by 10 pixels each way.

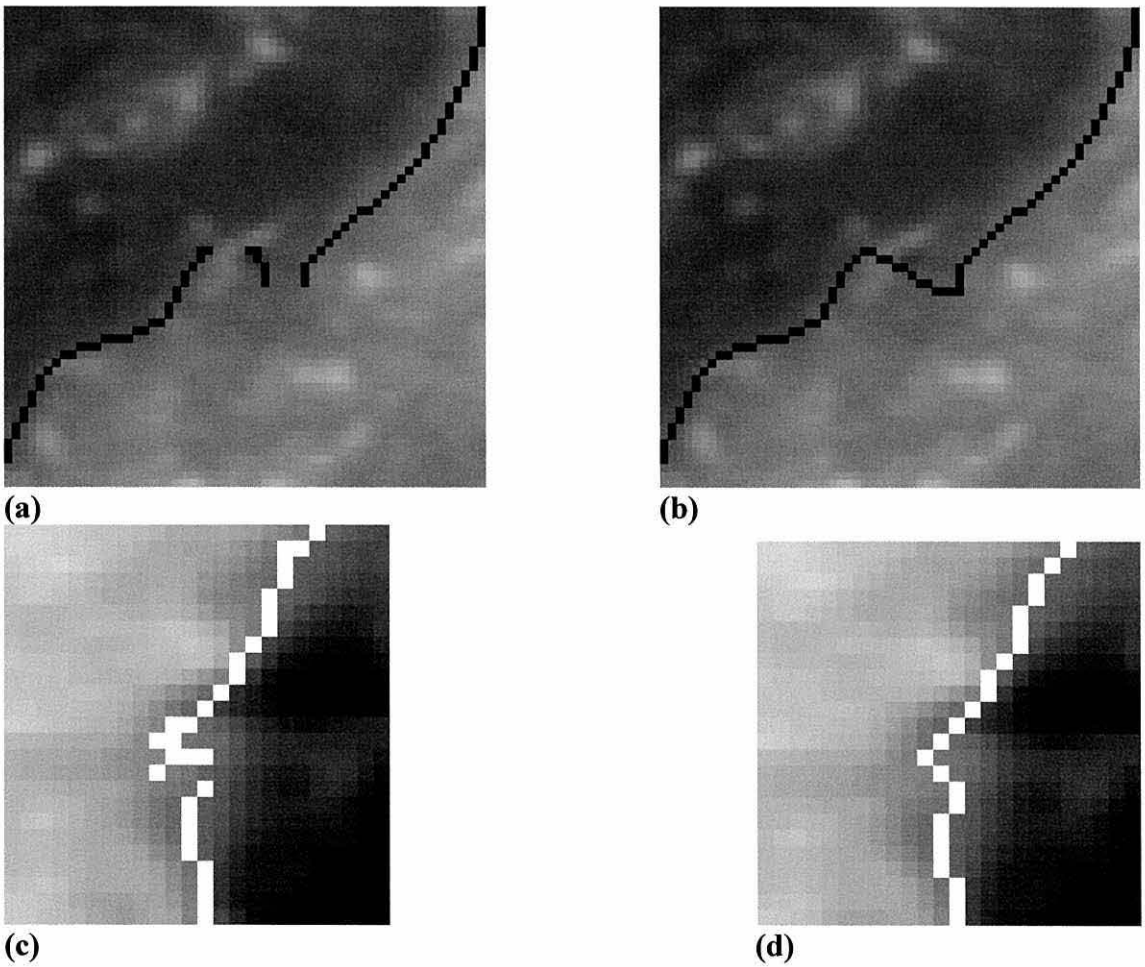
8.4 Boundary Selection and Cleaning

The edges detected initially and at later stages do not always form a closed contour around the lesion, for the reasons explained in the previous section, and hence require processing to ensure that there is only one complete closed contour. The initial edge map should contain the lesion boundary, but may also contain other edges (e.g. Figure 8-3(a)), hence the lesion boundary must be identified and processed to create a closed contour (e.g. Figure 8-3(b)). The initial edges and subsequent edges may contain gaps and extraneous pixels, caused by the omission of faint edges with very small gradients and imperfections in the ZC detection (see Figure 8-4(a) & (c) for an example of edge gaps and extraneous pixels respectively and Figure 8-4(b) & (d) for the "cleaned" version of these edges). Consequently, the processes described in this section are used to bridge gaps (section 8.4.1), select the correct boundary (section 8.4.2) and remove extraneous pixels (section 8.4.3) and are used in the order in which they are explained. The co-ordinate lists created by edge tracing the edge map are used by this boundary selection and cleaning process, which takes a minuscule amount of time compared with the edge detection.



(a) (b)

Figure 8-3: Example of cleaning of an initial boundary on a 548 by 510 image. (a) Raw initial boundary, from LoG edge detection with σ_{SC} = 31.328, overlaid on the image. (b) Initial boundary after cleaning. Note: Edges are drawn 3 pixels wide.



(a) (b)

(c) (d)

Figure 8-4: Example boundary faults. (a) Part of a raw boundary, showing a gap, overlaid on the image. (b) Boundary after cleaning, showing the effects of cubic spline joining. (c) Part of a raw boundary, showing extraneous pixels, overlaid on the image. (d) Boundary after cleaning. Note: Edges are drawn one pixel wide.

8.4.1 Edge Joining

Edge joining bridges gaps, such as those shown in Figure 8-4(a), to create a single edge that retains all of the co-ordinates of the original edges and includes extra interpolated points to bridge the gaps. These interpolated points should be 8-way connected to themselves and the sides of the gap, and also be arranged so that each point is next to its spatial neighbours in the co-ordinate list.

The 8-way direction edge map, produced by ZC detection, can be processed before edge tracing by Robinson edge linking [80] to remove single pixel gaps and delete inconsistent edge points. This method examines the map in 3x3 blocks and adds or removes the central pixel on the basis of the edge direction of its neighbours. This approach was not used as it can only handle single pixel gaps.

As no *a priori* information exists about the shape of each isolated edge or closed contour the edges cannot be fitted to an existing model. However, it is known that the lesion boundary is a closed contour, and hence all of the edges can be joined to create closed contours. In the initial edge detection one of these contours will be the lesion boundary; in subsequent edge detections the edges should join to create a single contour. To ensure that the correct edges are joined the following steps are used:

1. **Build end list:** Create a list of the edge ends (which excludes closed contours), containing the co-ordinates of the end point and a link to the original edge.
2. **While there are isolated ends, which are reasonably close, do steps 3 and 4:** The largest Euclidean distance (equation (3.1)) between isolated ends was set at 50% of the average of the width and height of the image.
3. **Find the nearest ends:** Search the end list to find the pair of ends which are closest, i.e. have the smallest Euclidean distance between them. The ends can be on the same edge, except when the edge has three or fewer points, which prevents very small edge fragments turning into very small closed contours with no points inside them, when they may be better joined to other edges.
4. **Join the two ends and update the end and edge lists:** The edge list is updated by removing one of the edges and adding the new points and the removed edge to the remaining edge. The end list is updated by removing the two ends which have been joined and modifying the end of the remaining edge to reflect the addition of the

joining and other edge's points. When an edge is joined to itself a closed contour is created, which is removed from the end list. The two methods tested for creating the joining points are described in sections 8.4.1.1 and 8.4.1.2.

Before creating the connecting line, its length is required to allow allocation of space to store its co-ordinates. This length does not include the line's end points as they already exist in the original data. The co-ordinates must be generated in order from the beginning point to the end point, to ensure that the co-ordinates of the resulting joined line are continuous.

8.4.1.1 Straight Line Joining

In this method, discrete straight lines are generated by a modified Bresenham's line drawing algorithm [32] (section 3.17). The algorithm was modified to exclude the two end points and to always draw the line from its starting point to end point. Whereas, the original algorithm draws from the end which it is most efficient. The number of generated points is simply the smallest of the horizontal and vertical lengths of the line minus two (to remove the end points). This method is simple, but does not generate smooth lines and creates discontinuities in the first derivative.

8.4.1.2 Cubic Spline Joining

This method uses cubic splines to provide interpolation which is smooth in the first derivative both within the edge gap and at its boundaries [81]. Hence, they can provide smooth and continuous joins between edges.

The lines being connected and the connecting line are *multi-valued* as they may contain points which have the same x or y co-ordinates as another point on the line. Consequently, the y co-ordinates cannot be generated from the x co-ordinates. This leads to the use of a parametric spline [7] where both the x and y co-ordinates are a function of a parameter t. The values of the parameter t are generated from the Euclidean distance between the points, starting with an initial value of $t = 0$ [7]:

$$t_{i+1} = t_i + \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$$
$$t_1 = 0 \text{ for } (x_1, y_1) \tag{8.1}$$

The 'C' routines in Press et al. [81] were used to generate a *natural cubic spline*, which has zero second derivatives at both of its boundaries, and does not require knowledge of the first derivative at the end of the edges being connected.

The co-ordinates of the edges being connected were combined and arranged to place the ends to be connected next to one another. When a line is connected to itself (i.e. a closed contour is being created) the points are re-arranged to place the gap in the middle of the line (to give equal numbers of points on either side of the gap). As the connecting and existing edges are multi-valued two different cubic splines are created for the x and y co-ordinates, which are both functions of the parameter t (eqn. (8.1)). The generated co-ordinates are those of pixels in a discrete image and consequently must be rounded to integer values. To prevent duplicate points being created, each point is compared (after rounding to the nearest integer) with the start, end and previously generated points. When it is equal to the end point the process is stopped and if it is equal any previous points or the start point it is discarded.

The t values required to generate a continuous line, without duplicate points, cannot be directly determined because the relationship between the parameter t and the x and y co-ordinates is not linear. Consequently, the required values of t are estimated from the number of points, excluding the end points, needed to create a straight line (as described in the previous section) and the values of t on either side of the gap. The number of straight line points is multiplied by a sampling rate of four, which was found empirically to be sufficient to give an estimate of the number of points in the gap. The t increment is obtained from the difference in t on either side of the gap divided by this estimate. The length of the connecting line is obtained by generating and discarding the points and this length used to allocate space for the connecting co-ordinates which are then calculated again. This method of counting and re-calculating the co-ordinates is used because it simplifies the management of the co-ordinate storage at the expense of some repeated calculation.

This method is the one which was implemented since it provides smooth joins with the existing edges and was felt to be worth the additional computation and coding complexity (compared with the straight line joining described in section 8.4.1.1). The connecting edges created by this method can sometimes be 4-way instead of 8-way connected, which is not required in this application and hence the edges are converted to 8-way

connected ones by the process explained in section 8.4.3. (This is illustrated in Figure 8-5.)

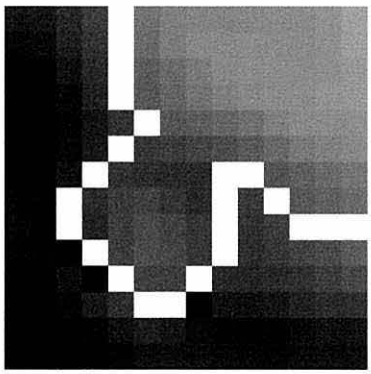


Figure 8-5: Closed contour cleaning. White = fully cleaned boundary with contour cleaning, black = additional pixels left in the partially cleaned boundary. Note: The edge is drawn one pixel wide.

8.4.2 Lesion Boundary Selection

Once any gaps in the edges have been bridged, the edge which is most likely to be the lesion boundary must be selected and all other edges discarded. This stage is only necessary in the initial edge detection when there may be a large number of extraneous edges (e.g. Figure 8-3). The lesion boundary is selected on the basis that it is the largest closed contour (i.e. with the greatest number of boundary points) and all other isolated edges and closed contours are discarded.

8.4.3 Lesion Boundary Cleaning

The closed contour which has been selected as the lesion boundary may contain extraneous points which are not required, such as those shown in Figure 8-5. These extraneous points are generated by imperfections in the ZC detection and edge joining processes. The imperfections in edge joining were discussed at the end of section 8.4.1.2 and illustrated in Figure 8-5. The problems caused by the ZC detection were discussed in section 8.4 and illustrated in Figure 8-4.

At this stage, the only remaining edge is a single closed contour. To remove the unnecessary edge points this contour is drawn and filled, and this filled image boundary traced to generate 8-way connected co-ordinates. A parity check filling algorithm [76] (section 3.17) was used as this does not require a starting point inside the boundary. However, when the region is not full (see section 3.3 for the definition of a full region)

this algorithm fails and hence a seed filling algorithm [76, 35] (section 3.17) was used with the contour's centre of gravity as the starting point. The filled boundary is contour traced using a contour tracing algorithm [76] (section 3.5) to provide the lesion boundary co-ordinates.

The effect of this is illustrated in Figure 8-5 and although proportionally it only affects a small part of the boundary it improves the algorithm's reliability by removing pixels which cause the boundary to become double thickness. These extraneous points are added by imperfections in the ZC detection and edge joining processes and if not removed then in subsequent edge focusing passes extra edges may be generated.

8.5 Selecting a Suitable Boundary

The previously described steps create a series of boundaries (illustrated in Figure 8-6) with decreasing edge detection scale and increasing detail. As the edge detection scale decreases the boundary detail increases as the edge is affected by smaller intensity variations. The edge moves towards areas of higher intensity gradient, which should represent a more accurate lesion boundary. However, as can be seen in Figure 8-6(e), the boundary at very small edge detection scales becomes very irregular and follows minor intensity variations which are considered by humans to be unimportant. It is difficult to identify the "best" boundary, but the chosen boundary must be a reasonable approximation of the lesion outline which can be consistently chosen. Using a human perceived boundary is not possible as this prevents automation of the lesion diagnosis process and their boundaries can vary considerably and may be influenced by other factors, such as the lesion's diagnosis. Identifying the "best" boundary is difficult, but the chosen boundary must be a reasonable approximation of the lesion outline which can be consistently chosen. To select this "best" boundary a new "boundary quality" measure, which gives a consistent and objective measure of the quality of the boundary, was calculated for each prospective boundary.

The unconstrained images being considered may contain features which could affect the choice of the "best" boundary. The area surrounding the lesion may contain other skin features, such as hair or background non-skin objects, some of which are removed by the image cropping performed during the process described in chapter 6. Parts of the lesion may produce "highlights", which are generally caused by non-diffuse lighting

reflected from smooth areas. These highlights can be confused with skin when they are "close" to the boundary since they may have similar brightness to that of the skin. This may result in the edge focusing excluding these areas from the lesion at smaller scales. Consequently, to make the algorithm robust to such problems a "best" boundary needs to be selected, which may not be that produced at the smallest scale.

The inner part of the lesion can vary considerably in intensity, texture and size, so is not used in choosing the boundary. The area around the lesion may contain features which are not "plain" skin, so it is necessary to use only the area near the lesion boundary, preventing unwanted features inside and around the lesion from influencing the choice. Using a large area could de-sensitise the boundary quality measurement as changes caused by a small movement of the boundary would have little impact as they would be swamped by the rest of the large unchanged area.

The first method tested measured the standard deviation of the area outside the boundary and was based on the assumption that the area just outside the "best" boundary would have the smallest standard deviation since it would incorporate a minimum of lesion image. However, this approach selected boundaries with large space constants which resulted in only approximate outlines. Consequently, a second method which assessed the boundary by measuring the characteristics of the image just inside and just outside the lesion was tested. The boundary with the best contrast between the area just inside and just outside the boundary was selected by calculating the ratio of the average intensities of these two areas. Both of these methods are affected by the size of the measurement area and must be adaptable to a range of lesion sizes. To provide this adaptation, the width of the initial edge detector is used to provide an estimate of the approximate lesion size and the measurement area width set at a percentage of this width.

The boundary quality measurement is performed on irregularly shaped areas which can be specified by two masks: just inside and just outside the boundary. A different pair of masks is required for each boundary in the series, which makes the time taken to generate them a consideration. These masks are generated by the following process, which is illustrated in Figure 8-7:

1. Draw a single pixel boundary outline: Figure 8-7(a).
2. Dilate the boundary with a disc: Figure 8-7(b).
3. Create a filled boundary from the image in Figure 8-7(a), using the method described in section 8.4.3: Figure 8-7(c).
4. Logically AND the dilated boundary with the filled boundary to give the area just inside the boundary, including the boundary itself: Figure 8-7(d).
5. Exclusive-OR the image of the area just inside the boundary with the filled boundary to give the area just outside the boundary: Figure 8-7(d).

Initially the dilation, in step 2, was performed using a 'C' morphology library [78]. However, this was found to be taking over 99% of the time required to measure the boundary quality. Hence, to reduce the time this stage was replaced by a method which placed a copy of a disc centred on each point in the boundary. (This idea is also used in the creation of the edge detection masks in section 8.3.) This took approximately $^{1}/_{169}$ of the time required by the original method.

To select the percentage measurement area width, a test set of the images created by processes presented in chapter 6 were visually assessed. For each image, the boundary for 25%, 50% and 100% was superimposed on the original image. These three new images were displayed simultaneously to allow their visual comparison to identify which boundaries were either unacceptable, acceptable or good, i.e. some significant part of the lesion was omitted, the boundary was usable or the boundary closely matched the visually perceived boundary. On the basis of these tests, the results of which are summarised in Table 8-I, a measurement area width of 50% was selected. An example of LoG edge focusing is shown in Figure 8-6.

% of initial edge detector width used as the boundary area	Unacceptable	Acceptable	Good
25%	0%	44%	56%
50%	0%	22%	78%
100%	0%	44%	56%

Table 8-I: Width of the area used for selecting a suitable boundary.

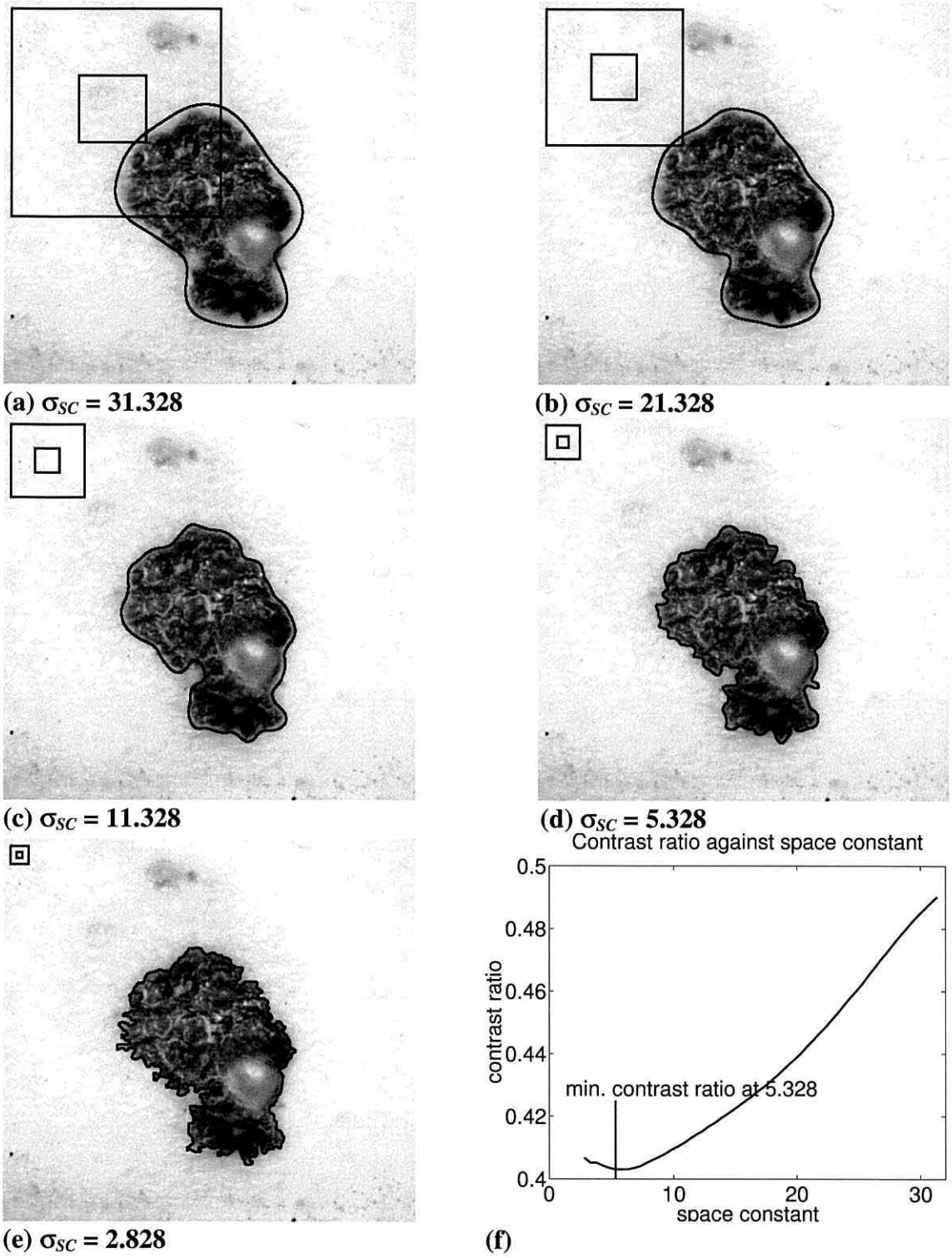
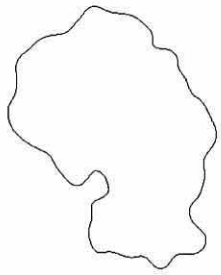
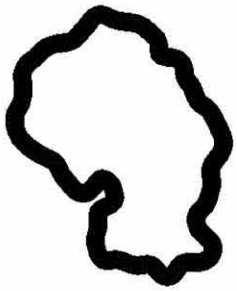


Figure 8-6: Edge focusing on an example image (548 by 510). (a) Initial boundary detected. (b) Boundary at about two thirds of the initial scale. (c) Boundary at about one third of the initial scale. (d) Final selected boundary. (e) Boundary at a space constant smaller than the selected boundary. (f) Graph of contrast ratio against space constant. The best contrast is given by the minimum on this graph as the ratio was calculated by dividing the average image intensity inside the boundary by that outside. The squares show the size of the edge detection filter: outer - area of the image used for each edge point, inner - sensitive region of the filter. The calculation of the filter and sensitive region widths was described in section 7.3. Note: Lesion boundaries and filter widths are drawn 3 pixels wide.

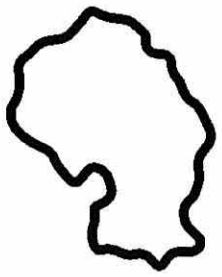


(a)

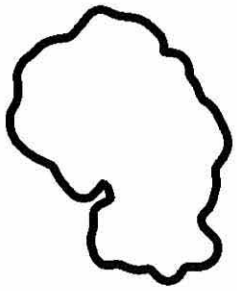


(b)

(c)



(d)



(e)

Figure 8-7: Boundary quality measurement mask generation. (a) Boundary drawn 1 pixel wide. (b) Boundary dilated by a 21 pixel diameter disk. (c) Filled boundary. (d) Image (b) logically ANDed with image (c) to give the area just inside the boundary, including the boundary itself. (e) Image (b) logically XORed with image (d) to give the area just outside the boundary.

8.6 Use of Canny Edge Detection

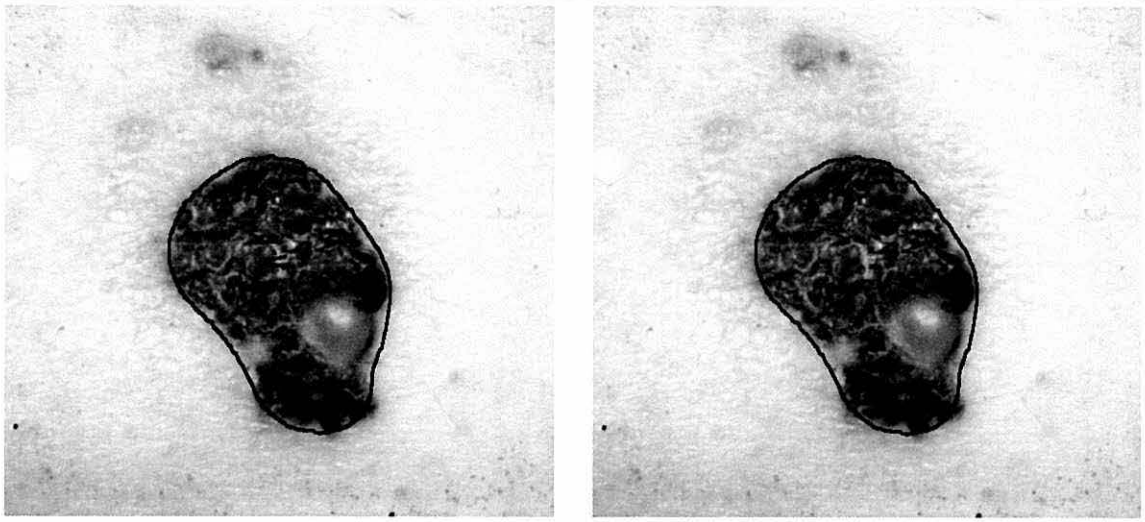
In the first part of this chapter, an edge focusing algorithm for skin lesion boundary detection using LoG edge detection was described. In this section it is demonstrated that other edge detectors can be used, taking the Canny edge detector [10] as an example. Canny edge detection, which was described in the previous chapter, is a first derivative edge detector and hence has different properties from the LoG. The edges detected by Canny have different positions from those detected by LoG edge detection, which can be seen by comparing Figure 8-3 with Figure 8-8. Both LoG and Canny accurately place their detected edges for straight edges, however for curved edges they differ in their edge placement [97, 92]. For a black square on a white background, which is illustrated in Figure 8-9(a), the LoG detector displaces its edge outwards near the corner at a distance related to the true edge curvature and detector space constant. The Canny detector's edge displacement is in the opposite direction, consequently the edges are inside the square, which is shown in Figure 8-9(b). A consequence of these differing edge displacements is that the results of using the edge focusing algorithm with the two edge detectors would be expected to be different.

Three parts of the edge focusing algorithm required modification to adapt it to use the Canny edge detector. The basic edge detection (LoG convolution and zero-crossing detection) was replaced by convolution with Gaussian and first derivative of Gaussian functions followed by non-maxima suppression to locate the edges. The output of the non-maxima suppression is different from the zero-crossing detection in two ways: it can provide both edge direction and strength, and its output tends to be 4-way connected whereas the zero-crossing detection method used generates 8-way connected edges. The edge tracing method using edge direction, which was described in section 7.6.1, was designed to trace 8-way connected edges and on 4-way connected edges may miss some edge pixels. This omission of some pixels creates a number of extraneous edge points, which caused difficulties in the boundary cleaning stage by causing some correct edge points to be discarded which resulted in the cleaned edge being sometimes inaccurately placed. As a consequence a different edge tracing method (explained in section 7.6.2) was used which does not use edge direction and is biased towards 4-way connectivity. This method correctly traces the edge generated by the non-maxima suppression and hence includes all of the edge pixels.

Canny's edge detector produces edge strength, which was used in the initial edge detection to remove "weak" edges. For the initial edge detection and subsequent passes, the minimum edge strength was set at $^1/_{512}$ of the maximum possible edge strength.

The same step size (0.5) as is used for LoG and thus the edge moves at most 1 pixel [3] between edge focusing steps. Consequently, the same three pixel wide line mask is used, but as the non-maxima suppression only requires a one pixel border around this area the mask is only dilated by one pixel to provide the area where the edge detection convolutions are performed. This is different from the LoG detection where the mask was dilated by two pixels because of the functioning of the zero-crossing detection.

An example of Canny edge focusing is shown in Figure 8-10.

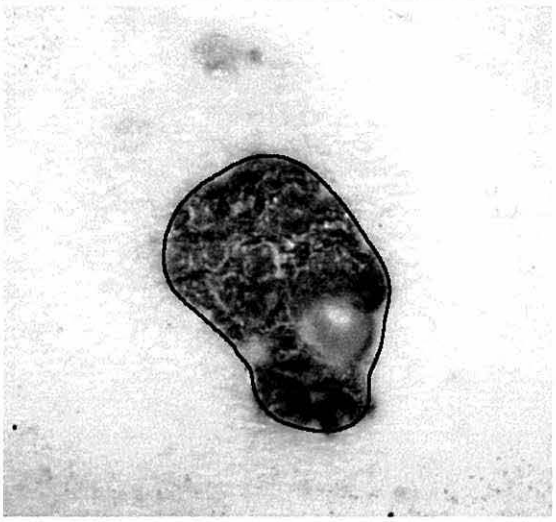


(a) (b)

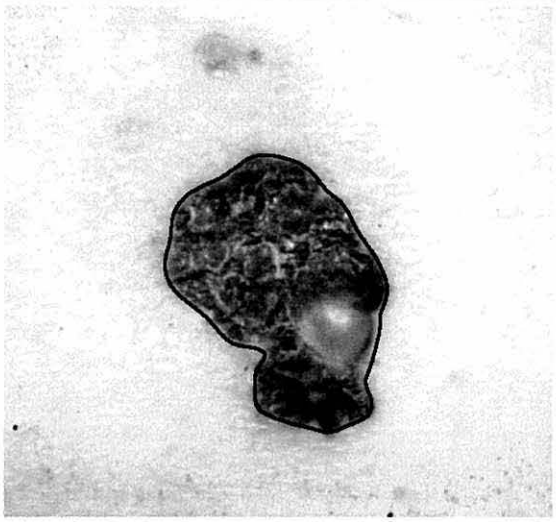
Figure 8-8: Initial boundary for Canny edge focusing. (a) Raw initial boundary, with $\sigma_C = 31.328$, on a 548 by 510 image. (b) Boundary after cleaning. Note: Edges are drawn 3 pixels wide.

(a) $\sigma_{SC} = 25.828$ (b) $\sigma_C = 25.828$

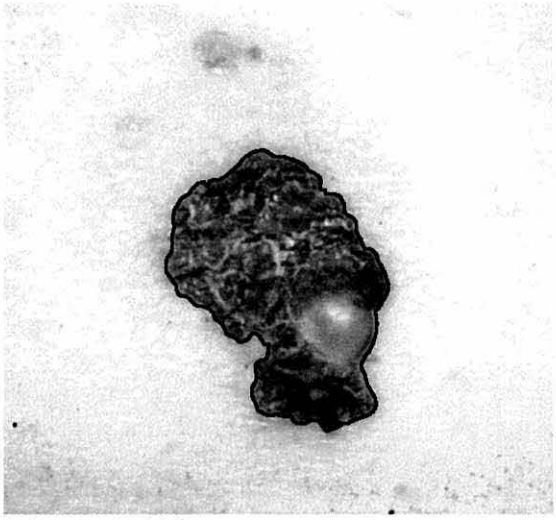
Figure 8-9: Edge displacement for curved edges. (a) LoG edge detection. (b) Canny edge detection. The detected edge is shown in black. The original black 128 by 128 square, on a white background (256 by 256), is shown in grey.



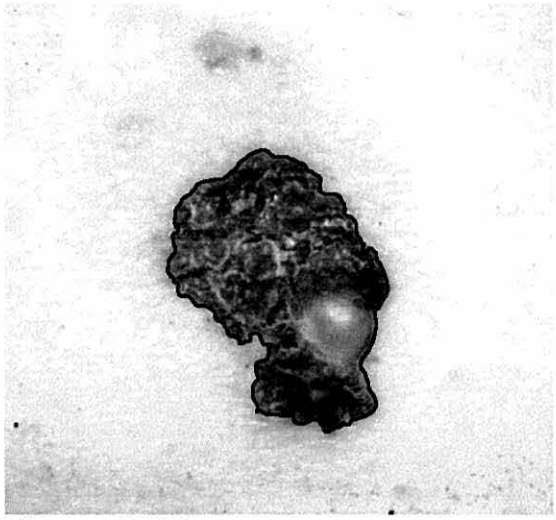
(a) $\sigma_C = 21.328$



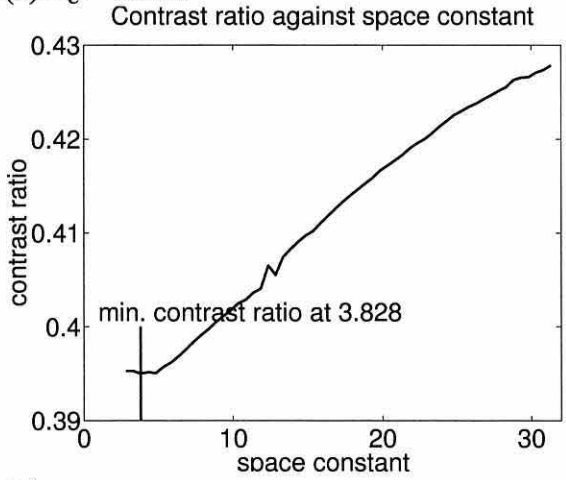
(b) $\sigma_C = 11.328$



(c) $\sigma_C = 3.828$



(d) $\sigma_C = 2.828$



(e)

Figure 8-10: Canny edge focusing on an example image (548 by 510). See Figure 8-8(b) for the initial boundary detected. (a) Boundary at about two thirds of the initial scale. (b) Boundary at about one third of the initial scale. (c) Final selected boundary. (d) Boundary at a space constant smaller than the selected boundary. (e) Graph of contrast ratio against space constant. Note: Lesion boundaries are drawn 3 pixels wide.

8.7 Testing the Edge Focusing Algorithm

The testing of the algorithm was divided into two parts: verification and validation tests, which can be summarised by [88]:

- **Verification: Are we building the system right?**

  Does the algorithm do what it is expected to do? This can be tested by using a set of simple images for which the boundary is known, e.g. the image shown in Figure 8-11. This phase of testing involves ensuring that the boundary is "sensible", which was defined as the boundary being continuous and contained within the original image. This testing was performed during the algorithm development, to ensure that the component parts functioned as expected, and on the complete algorithm to ensure that the components integrated correctly. Verification testing does not attempt to ensure that the produced boundary is useful.

- **Validation: Are we building the right system?**

  Does the algorithm do what we want it do? To answer this much harder question we must decide where the lesion boundary is. One possibility is to use the boundary provided by a human, possibly an expert. However, the boundaries produced by humans are inconsistent in that they can vary considerably from person to person and between different presentations of the same image to the same person. The boundaries can be influenced by external factors, such as what the perceived diagnosis is. The edge focusing algorithm's boundaries were assessed by visual assessment by a non-expert, which checked that the boundaries were in the vicinity of the visually perceived boundary.

  Another method of approaching this problem is to create simulated images which contain "artificial" skin and lesions, which enables specification of the lesion boundary location. This method of assessing the edge focusing algorithm is used in the next chapter where the creation of simulated images is considered.

  The generated boundaries must be consistent: different images of the same lesion should produce similar boundaries. Meeting this criteria will ensure that the system is tolerant of image capture conditions, which ensures that it can be used in a clinical environment where the image capture conditions cannot be accurately controlled. Unfortunately due to the small available test set it was not possible to test the boundaries for consistency.

(a) σ_{SC} = 20.328 (b) σ_{SC} = 2.828

Figure 8-11: Example verification image of a circle (diameter 128 pixels), illustrated with LoG edge focusing. (a) Initial boundary. (b) Final boundary. The original black circle, on a white background, is shown in grey.

8.8 Conclusions

This chapter has presented an edge focusing algorithm for skin lesion boundary detection. Two variations of the algorithm using the LoG edge detector and the Canny edge detector were presented and the testing of the algorithm on artificial images and real images was described. Using real images it was not possible to assess visually the relative performance of these two alternatives, but they are compared in the next chapter. This algorithm can be combined with the work described in chapter 6 to produce a system which is capable of finding lesion boundaries on a wide range of images. From the testing performed the algorithm was shown to be tolerant of image capture conditions and capable of working on a wide range of images.

This algorithm is a new application of edge focusing, which controls the boundary during focusing and selects the output boundary, using image contrast. The lesion boundary is "cleaned" during focusing by bridging gaps in the boundary and removing extraneous pixels, which allows it to function on indistinct boundaries where without this "cleaning" process it is unable to maintain a continuous boundary. The final boundary is selected using image contrast, which enables a suitable boundary to be automatically chosen from the series of boundaries created by edge focusing. Through these developments, edge focusing has been made suitable for lesion boundary detection and may also be suitable for other applications where automatic boundary detection is required for objects with boundaries which are not clearly defined.

Chapter 9

Image Synthesis and Edge Focusing Test

9.1 Introduction

This chapter discusses the synthesis of simulated monochrome skin and lesion images, which are required to investigate the behaviour of the methods presented in the last chapter and chapter 6. Test images are required where the characteristics can be controlled, so that the output of a boundary finding method may be compared with the known input. This comparison allows assessment of algorithm performance under a range of image conditions, which can be created by adjusting the parameters of the image synthesis process. This assessment of algorithm performance can be used to compare different algorithms, such as the LoG and Canny edge focusing algorithms presented in the last chapter, and could also assist in algorithm development by identifying image conditions under which an algorithm succeeds or fails. Previously, algorithm performance has validated by human assessment [30, 36], which is prone to the problems discussed in section 8.7. By the creation of simulated lesion images and automation of boundary quality assessment, these problems can be overcome thereby improving the assessment of the performance of boundary detection and other algorithms. The information obtained from this assessment can be used to help develop new methods and enhance existing methods.

In this chapter, the performance of LoG and Canny edge focusing algorithms and the lesion isolation and location method is assessed using simulated skin lesion images. However, although considered desirable, due to time constraints it was not possible to extend this chapter to cover the assessment of other skin lesion boundary detection methods (such as [12, 30, 77, 36, 26]) and other boundary detection methods (such as

[50, 54, 73]). Such an assessment would have compared the relative performance of edge focusing and other algorithms and could have identified their strengths and weakness. This would have demonstrated whether the edge focusing algorithm provides an improvement over boundary detection methods that have previously been applied to skin lesions. The assessment would have enabled the comparative testing, refinement and improvement of boundary detection and shape analysis algorithms, through gaining a better understanding of the behaviour of the algorithms under known conditions. In addition, again due to time constraints it was not possible to extend the shape generation to synthesise images with "benign" and "malignant" boundaries, which would have allowed the assessment of the effects of different boundary detectors and shape classifiers on the accuracy of computer classification of such boundary types. This would have demonstrated whether the edge focusing algorithm and shape analysis methods presented in this thesis were an improvement over published algorithms previously applied to skin lesions.

The new image synthesis method presented in this chapter is performed in three stages:

- **Shape generation** - the creation of a shape similar to that of a lesion. This simulated lesion shape is used to create a mask which controls the mixing of skin and lesion textures.
- **Boundary transition** - the modelling of the transition between skin and lesion, by controlling the mixing of skin and lesion texture, by modifying the mask created by the shape generation.
- **Texture generation** - the synthesis of textures which are similar to skin and lesion from measurements of real skin and lesion textures. The skin and lesion were assumed to have different, but uniform textures.

The simulated image is created by using the shape image, with the boundary transition applied, to control the mixing of skin and lesion textures.

A method of comparing true and estimated boundaries, based on the area which they enclose, is then presented. This is used to assess the edge focusing and lesion location and isolation methods described in previous chapters. A summary of their performance on a range of skin and lesion textures and boundary transitions is given.

Dhawan and Sicsu [26] synthesised simulated skin and lesion images, which they used to visually assess their segmentation method. They did not specify how these images were created, but their illustrations showed images containing overlapping ellipses of varying colours and textures. These images appear not to have been created from measurements of real lesion images and had clear transitions from skin to lesion. The methods presented here use measurements from real images to create suitable skin and lesion textures and blur the boundary between skin and lesion to make it more representative of real images. In this chapter, methods from other image processing areas are adapted, where appropriate, for use in the synthesis of simulated skin and lesion images.

9.2 Shape Generation

This section describes the creation of a binary image which contains a lesion-like shape. In addition to the binary image, the shape's boundary co-ordinates are stored as the true boundary, to allow their comparison with any estimated boundary. The method creates a simple shape based on an ellipse with random large and small scale irregularities, using parameters which were heuristically chosen to create shapes similar to lesion outlines.

It would be possible to analyse the real lesion shapes and create a model for synthesising lesion boundaries. However, this approach requires accurate boundaries for building a lesion shape model. These accurate boundaries cannot be obtained before the boundary finding method has been shown to be accurate by testing on simulated images. Consequently, the accurate boundaries are not available for developing a model at this stage and hence shape synthesis cannot be based on real lesion shapes. However, it is important, to allow reasonable testing, that the simulated shapes are visually similar to real lesion shapes. To achieve this objective Dhawan and Sicsu [26] used overlapping ellipses with synthetic textures (which were not modelled on real lesion textures) to create the simulated images used to assess their multi-channel intensity, colour and texture segmentation algorithm (section 4.2). This approach created images with an approximately elliptical shape with large scale irregularities, but without any small scale irregularity. The method presented here creates simulated approximately elliptical shapes with both large and small scale irregularities.

The parameters shown in the following three stages used to create lesion-like shapes with large and small scale irregularities were heuristically chosen for an image of 512 by 512 pixels.

1. **Primary shape**

   The primary shape is an ellipse, with a random size, random aspect ratio and random angle placed at a random position in the image, e.g. Figure 9-1(a). The ellipse's minor axis size, aspect ratio and centroid were controlled by Gaussian distributed random numbers with means of 90, one and the image's centre respectively. These random numbers had standard deviations of 25, 0.25 and 51.2 (10% of image width) respectively.

2. **Large scale irregularity**

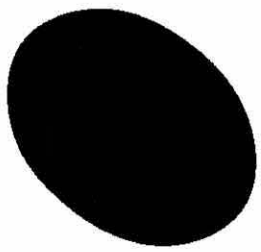
   The ellipse was sampled, a fixed number of times (8), at uniformly spaced angles starting at a random angle. These sampled points were displaced radially by a Gaussian distributed random distance (with zero mean and a standard deviation of 30 pixels) and re-connected to create a shape with large scale irregularities. The parts of the boundary that required interpolation were found by searching the list of boundary co-ordinates for spatially non-adjacent points, i.e. those points adjacent in the co-ordinate list, but with an Euclidean distance of greater than $\sqrt{2}$ between them. The two non-adjacent points and all the other sampled points were connected by parametric cubic spline interpolation (section 8.4.1.2) with the two non-adjacent points placed in the centre of the spline. The sampling, radial displacement and interpolation creates an approximately elliptical shape with large scale irregularities, such as that shown in Figure 9-1(b).

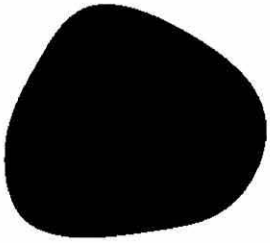
3. **Small scale irregularity**

   The shape with large scale irregularities was re-sampled more frequently (16 times) at uniform points in the co-ordinate list. These points were radially displaced and interpolated by the previously described method, to give the final shape (e.g. Figure 9-1(c)). The points were selected directly from the co-ordinate list rather than by angle (which was used for the large scale irregularities) because of the difficulty of finding points at particular angles in a Cartesian co-ordinate system. Although, the points could have been converted to polar co-ordinates (where it would have been easier to

find points at particular angles), this was not attempted as it would only have produced a different random shape at the expense of extra computation.

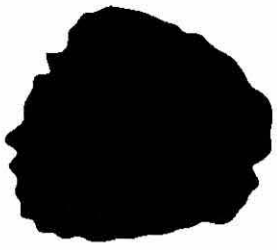
The random numbers were generated from a single seed for the whole shape, which allowed the re-creation of the same shape from the seed value. The Gaussian distributed random numbers [81] used were limited to $\pm 3\sigma$, where σ is the standard deviation of the distribution. This prevented the creation of unreasonable shapes and the placing of the lesion outside the image, but does not affect most shapes as the $\pm 3\sigma$ limit includes over 99% of the Gaussian curve.





(a) Ellipse: aspect ratio = 1.36, major axis = 138.7 pixels, minor axis = 102.0, angle of major axis to vertical (anti-clockwise) = 0.900 radians (51.6°)

(b)



(c)

Figure 9-1: Creation of synthesised lesion shape in a 512 by 512 image. (a) Starting ellipse. (b) Ellipse after adding radial Gaussian distributed noise (with standard deviation of 30) at 8 sampling points and spline interpolating between the points. (c) Second sampling, radial Gaussian noise (with standard deviation of 5) adding and interpolation. Black and white represent lesion and skin respectively.

9.3 Boundary Transition

This section describes the creation of a boundary transition between the simulated lesion and the surrounding simulated skin. The transition is simulated by modifying the shape image which controls the intensity mixing of skin and lesion texture.

The presence of an indistinct transition between skin and lesion, in real images, makes the segmentation of lesion from skin harder. Consequently, to ensure that the simulated images are similarly difficult to segment, the transition between simulated skin and lesion must also be indistinct. In this stage, the problem of requiring an accurate boundary before the accuracy of the boundary detection method has been determined also occurs. Consequently, a method of providing an indistinct skin-lesion transition that does not rely on numerical analysis of real images is required. The simple method presented here creates a boundary transition using a low-pass filter. By visually examining real images and the simulated boundary transitions, a wide range of realistic boundary transitions without the necessity of lesion measurements can be created. Once the performance of the boundary detection method has been assessed it may be possible to create better simulated images using the additional data which could then be obtained.

A first order low-pass butterworth filter was used to blur the synthesised lesion shape image, created by the method presented in the previous section. By changing the filter's cut-off frequency (expressed as fractions of the sampling rate), which represents the point where the filter's output is 50%, the sharpness of the boundary transition can be changed. The butterworth filter was applied, in the frequency domain, to the simulated lesion shape image by Khoros [82] programs. The one-dimensional effect of the filters tested is shown in Figure 9-2 and two example boundary transitions are shown in Figure 9-3. The cut-off frequencies were chosen to create boundary transitions visually similar to those in real images.

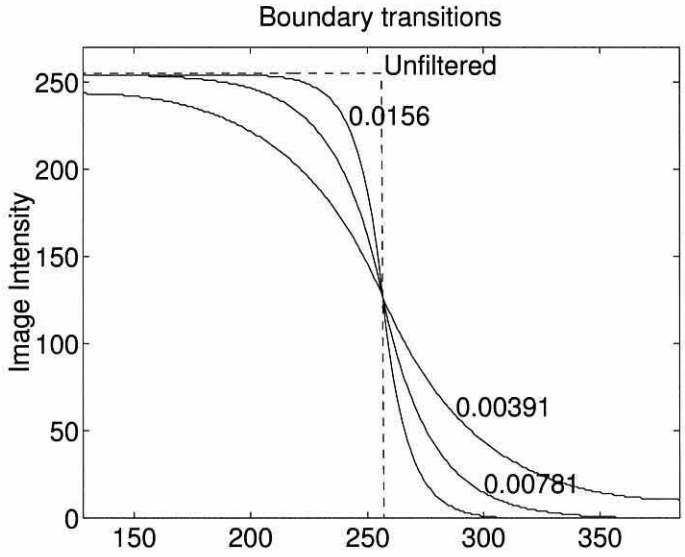


Figure 9-2: 1-D boundary transitions.

(a) **(b)**

Figure 9-3: Example boundary transitions. Filtered with cut-off frequencies of (a) 0.0156, (b) 0.00781. The original synthesised shape is shown in Figure 9-1(c) and original boundary is drawn in black.

9.4 Texture Generation

This section considers the creation of synthesised skin and lesion textures which are visually similar to real skin and lesion textures. To maximise the "realism" of the synthesised images only bi-directional methods, i.e. those which can analyse a texture and synthesise a new similar texture, were considered. The performance of the texture synthesis was assessed visually by comparison with the original texture.

Texture is a property of the local spatial variation of image intensity, which is difficult to define qualitatively and quantitatively [38, 80, 58]. (An introduction to texture was provided in section 3.9.) It is measured over a spatially restricted area and texture measurements must be restricted to areas of uniform texture. Hence, it is necessary to establish the boundary of uniform texture region before making texture measurements. In this work no attempt was made to segment skin images automatically to obtain the relevant textures. Instead, 64 by 64 pixel texture samples were taken manually. In addition to skin and lesion textures, other samples were taken from textured surfaces and one texture (fieldstone) was taken from the Brodatz texture album [6, 5]. The Brodatz fieldstone texture was chosen as it is the closest in this "standard" texture set to lesion texture. The other textures were chosen to be easy to distinguish visually, to facilitate the development and initial assessment of the texture creation methods. Examples of test, skin and lesion textures are shown in Figure 9-4.

In this application, the exact reproduction of the original texture is not required, as this would not allow the creation of a range of simulated textures from one real image. However, it is important to have synthesised textures which are visually similar to real lesion and skin textures. It is possible to measure the similarity between individual pixels in synthesised and original textures, but it is much harder to give an overall assessment of the visual similarity of the textures. Thus, visual comparison of the original and synthesised textures is necessary. Visual texture comparisons were performed by creating composite images containing the original texture in the centre surrounded by synthesised texture (illustrated in Figure 9-5). The image (Figure 9-5(b)) controlling the mixing of the original and synthesised textures was created by low-pass first order butterworth filtering an image of a 64 by 64 pixel black square on a white 128 by 128 pixel background (Figure 9-5(a)). By low-pass filtering the image controlling the mixing of the two textures, the edge between the two textures was de-emphasised. This reduced

the visual impact of the discontinuity between the two textures and made it easier to compare visually the original and synthesised textures. Examples of a texture match and mismatch are shown in Figure 9-5(c) and (d) respectively.

The creation of simulated skin images requires textures which are visually similar to real skin and lesion textures. This means a method that can analyse a real texture and synthesise a similar new texture is required. In image compression similar texture creation methods are used, but in this application (unlike image compression) the size of the data (parameter space) representing the texture and time required for analysis and synthesis are not important. Iversen and Lønnestad [48] surveyed a range of texture analysis/synthesis methods for use in image compression. They assessed the visual performance, the size of the parameter space and analysis/synthesis times of the following methods:

- **Markov models**: the texture is modelled by a stationary first order Markov chain.
- **Autocorrelation and histogram models**: the texture model contains the image histogram and the autocorrelation function.
- **Linear autoregressive models**: each texture pixel is modelled by the sum of a linear combination of the intensities of points in its neighbourhood and Gaussian noise.
- **Fractal models**: the texture's irregularity is modelled by its fractal dimension.
- **Spectral models**: the model is the same as the linear autoregressive model, except the neighbourhood and weights estimation are different.

The Markov models were found to have unsatisfactory visual performance and large parameter spaces, but with small analysis and synthesis times. The performance of the autocorrelation method was better and it uses a small parameter space, but has a time consuming synthesis algorithm. The linear autoregressive model was judged to give the best visual performance with a reasonably sized parameter space. The only problem (which is not relevant in this application) with this method is the very slow analysis algorithm. The performance of the fractal model was very poor, as expected since it had only one parameter and hence Iversen and Lønnestad [48] did not thoroughly investigate it. The spectral model they investigated had a small parameter space, but did not perform visually as well as the linear autoregressive model.

Iversen and Lønnestad's [48] survey showed that the linear autoregressive texture model was a possible texture creation method. In addition to this method, it was decided to investigate a method based on the average power spectrum of the image.

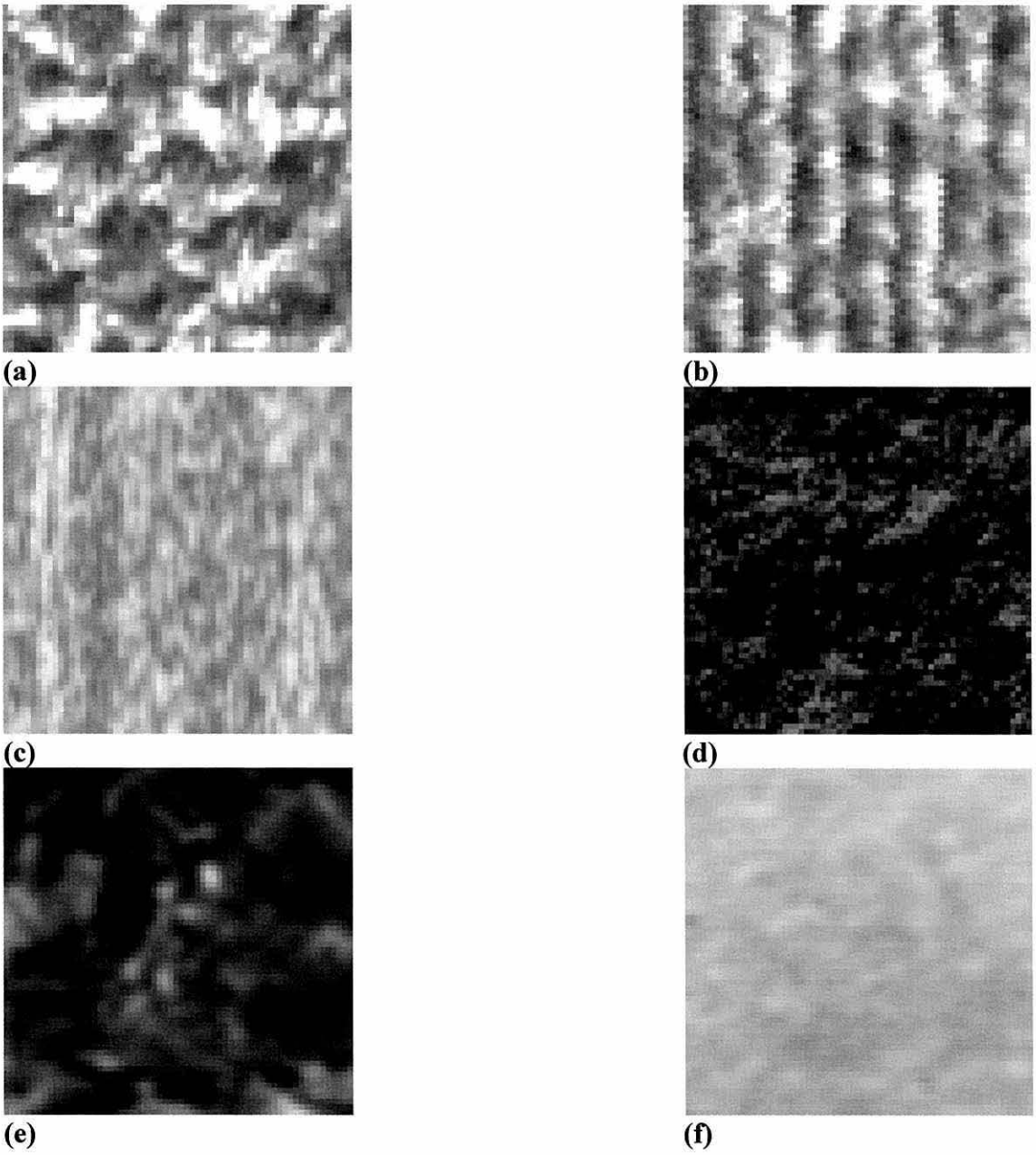


Figure 9-4: Example texture images. (a) Chair covering. (b) Floor. (c) Table top. (d) Fieldstone. (e) Lesion. (f) Skin.

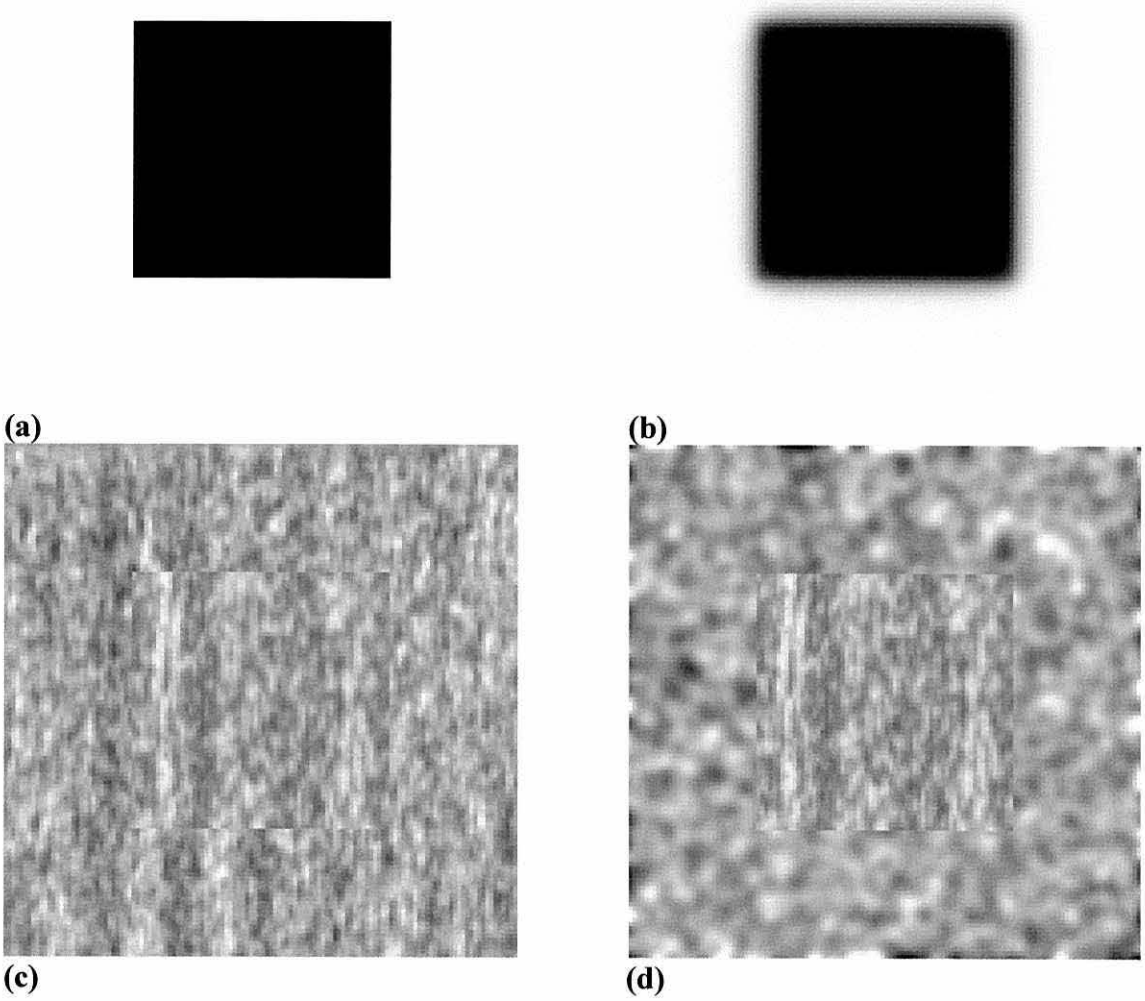


Figure 9-5: Visual texture comparison method. (a) 128 by 128 pixel masking image with central 64 by 64 pixel black square. (b) Masking image after low-pass filtering. (c) Example of matched textures. (d) Example of mismatched textures. The original texture is in the centre of the image with the synthesised texture surrounding it. Both examples use the same original texture.

9.4.1 Power Spectrum Texture Model

The first method investigated used the texture's average spatial frequency power spectrum for analysis and synthesis.

- **Analysis**

  MATLAB's spectrum [66] function was used to measure the power spectrum for each row and column of the image. These were averaged to produce a power spectrum for the whole image. Figure 9-6 shows the power spectra for the example textures in Figure 9-4. The image mean and standard deviation were also measured.

  MATLAB's spectrum function uses Welch's averaged periodogram method [71], which divides the signal into overlapping sections, each of which is detrended (i.e. a straight line fitted to the data and then subtracted to give a mean of zero and average gradient of zero), Hanning windowed [2] and zero padded [81]. Each 32 point section is discrete Fourier transformed [38] and the average magnitude output squared provides an estimate of the power spectrum.

- **Synthesis**

  MATLAB's fir2 [66] function was used to create a 10^{th} order finite impulse response (FIR) filter from the average power spectrum [2]. This was used to filter an image containing pixels with uniformly distributed random intensities in the range [0, 1] inclusive. The output image from this filter was scaled to have the same mean and standard deviation as the original texture. It was also rounded to the nearest integer and limited to the range [0, 255], to convert it into the same format as the original texture. The synthesised textures for the example textures in Figure 9-4 are shown in Figure 9-7.

These textures were not visually similar to the original textures, hence a better method was required. The poor quality of the synthesised textures may be due to the loss of the phase information. In addition, the use of an average of the horizontal and vertical spectrums, which was based on the assumption that the texture had similar properties in horizontally and vertically, may have lost important information. These effects were not investigated further as the next method presented gives significantly better results which are acceptable for this application.

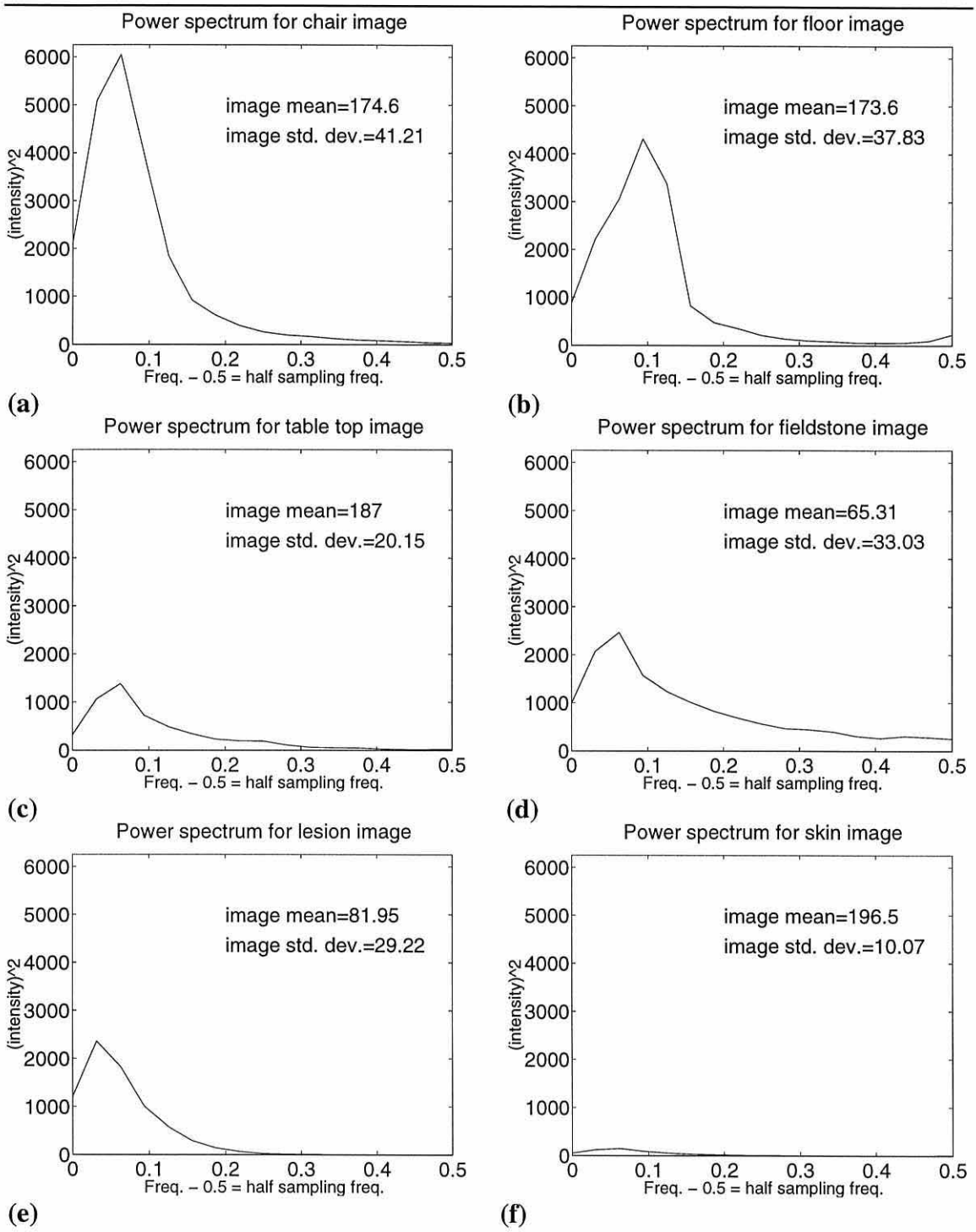


Figure 9-6: Example texture power spectrums. (a) Chair covering. (b) Floor. (c) Table top. (d) Fieldstone. (e) Lesion. (f) Skin.

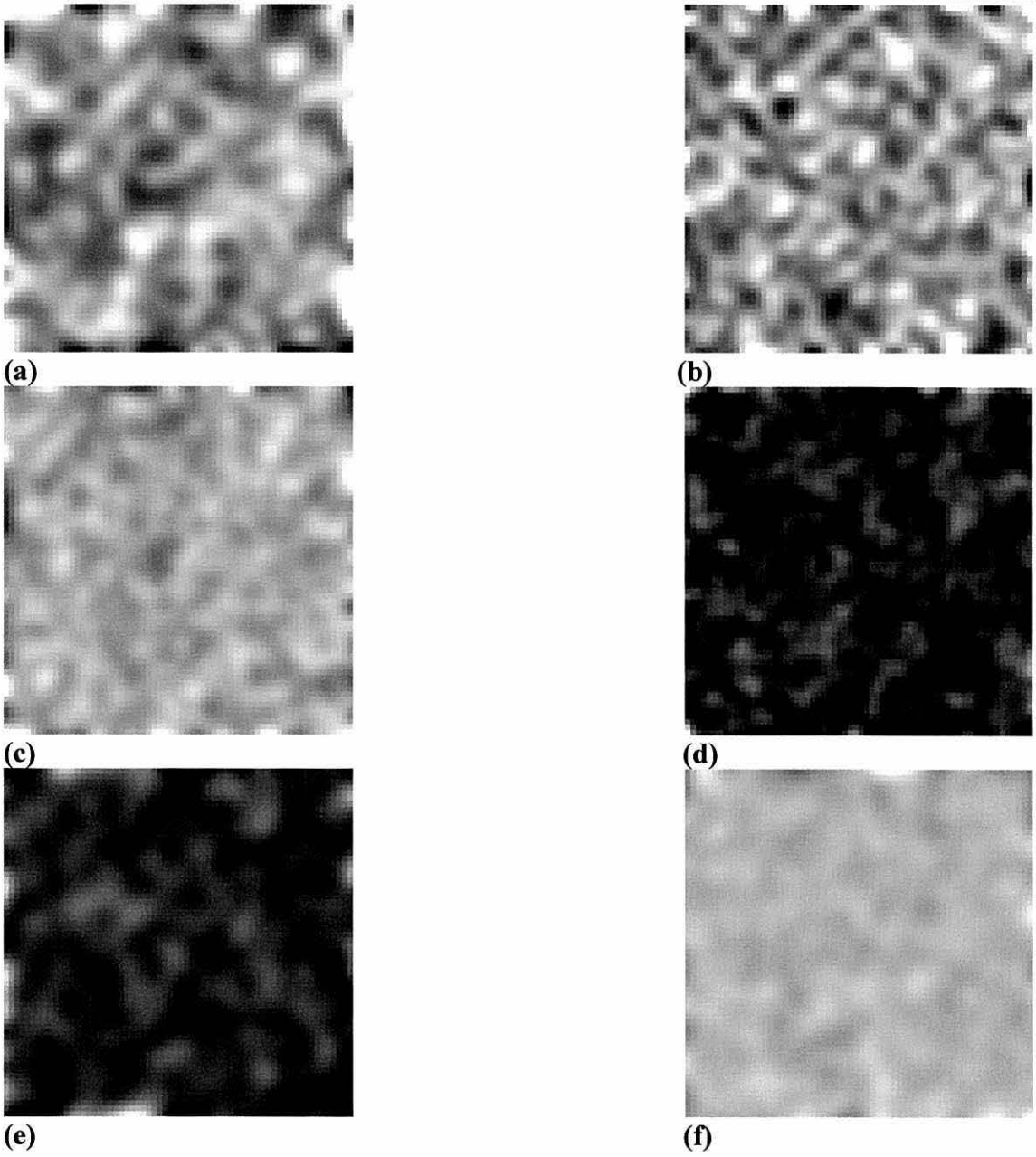


Figure 9-7: Example power spectrum synthesised textures. (a) Chair covering. (b) Floor. (c) Table top. (d) Fieldstone. (e) Lesion. (f) Skin.

9.4.2 Linear Autoregressive Texture Model

In this method the intensity of a texture point is estimated from the sum of a linear combination of the intensities of points in its neighbourhood, represented by the weights (α), and Gaussian noise. The model used here is a modification of the method described by Iversen and Lønnestad [48] (which is an enhancement of the method developed by Delp et al. [16]), with the addition of the measurement of image mean and standard deviation. The scale of texture analysed/synthesised is controlled by the size of a causal neighbourhood (n_d - illustrated in Figure 9-8) which is specified by the neighbourhood distance d_n and contains N_n pixels:

$$N_n = 4\sum_{i=1}^{d_n} i = 2d_n(d_n+1) \tag{9.1}$$

For example, when $d_n = 2$ the neighbourhood contains $N_n = 12$ pixels which are labelled 0 to 11 in Figure 9-8 and are processed in that order. The neighbourhood is causal, so that each pixel in a left to right, top to bottom scan can be created from the already synthesised pixels. To make the neighbourhood causal, the bottom right pixels are not used which can be seen in Figure 9-8, where the bottom right four pixels are not used.

The analysis is performed in two stages and was tested with the analysis limited to pixels where the whole of the neighbourhood, shown in Figure 9-8 fitted within the image and with the neighbourhood allowed to wrap at the image edges by considering the image to be toroidal. Equations (9.2), (9.5) and (9.6) are for the latter wrapped

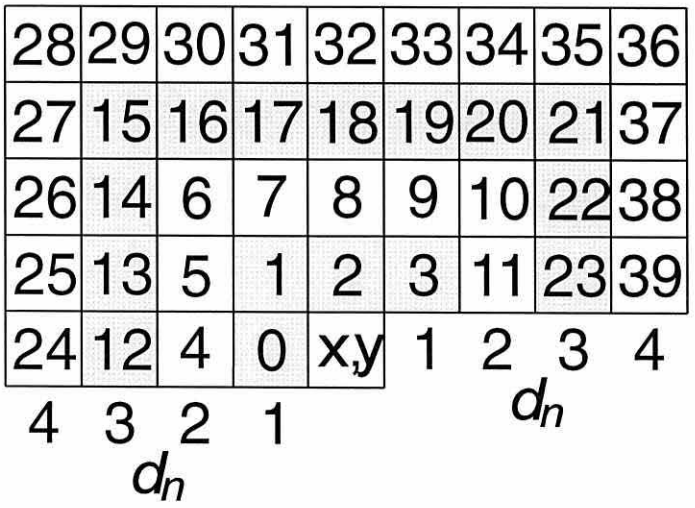


Figure 9-8: Neighbourhood (n_d) for the point (x, y) for the linear autoregressive texture model. d_n = neighbourhood distance.

method, but can be easily changed to the former unwrapped method. The two stages of the analysis are:

1. Measure weights (α, a vector of length N_n), image mean ($\bar{I}$), image standard deviation (σ_I), image minimum (I_{min}) and maximum (I_{max}):

$$\alpha = \left[\sum_{y=0}^{N-1} \sum_{x=0}^{M-1} n_d(x,y) n_d{}^T(x,y) \right]^{-1} \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} i(x,y) n_d(x,y) \tag{9.2}$$

$$\sigma_I = \sqrt{\frac{1}{MN-1} \left(\sum_{y=0}^{N-1} \sum_{x=0}^{M-1} i(x,y)^2 - \bar{I} \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} i(x,y) \right)} \tag{9.3}$$

where $n_d(x, y)$ = Neighbourhood pixels for point (x, y) in a vector in the order shown in Figure 9-8,
x^T = Transpose of vector or matrix x.

The weights (α) are set using the least squares method, hence equation (9.2) is derived from the squared error between the original pixel value and the pixel value created by the model [16]. The matrix inversion in equation (9.2) was performed by LU decomposition with backsubstitution using 'C' code from Press et al. [81]. When using single precision floating point arithmetic and some textures, in particular skin textures, this matrix inversion was unstable because the matrix was nearly singular. This instability caused the weights (α) to be inaccurate which led to large noise means and standard deviations. This was overcome by using double precision in all of the operations.

2. Estimate noise mean ($\bar{n}$) and standard deviation (σ_n). At each available pixel calculate the residual $r(x, y)$, which is the difference between the original image ($i(x, y)$) and that estimated by the weights (α) of the texture model, which have just been measured:

$$r(x,y) = i(x,y) - n_d{}^T(x,y)\alpha \tag{9.4}$$

This provides an estimate of the noise mean and standard deviation:

$$\bar{n} = \frac{1}{MN} \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} r(x,y) \tag{9.5}$$

$$\sigma_n = \sqrt{\frac{1}{MN-1} \left(\sum_{y=0}^{N-1} \sum_{x=0}^{M-1} r^2(x,y) - \bar{n} \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} r(x,y) \right)} \tag{9.6}$$

The noise mean and standard deviation are created from the difference between the original texture and the texture synthesised by the model. Consequently, they indicate the closeness of the model to the original texture with small noise means and standard deviations indicating better fits.

The synthesis algorithm uses the weights and Gaussian random noise to generate a texture, $i_T(x, y)$, given by:

$$i_T(x,y) = \alpha n_d(x,y) + n_g(x,y) \tag{9.7}$$

where $n_g(x, y)$ = Gaussian distributed noise with standard deviation σ_n and mean $\bar{n}$.

The random numbers used to create the Gaussian distributed noise were generated from a single seed for the whole texture. This allowed the re-creation of the same texture from the same seed and the creation of different, but visually similar, textures from the same measurements with different seeds. The synthesised texture is created from left to right, top to bottom to ensure that the neighbourhood is usually within the already created texture. At the top, left and right edges of the image the neighbourhood, shown in Figure 9-8, extends outside the synthesised texture. In this situation, the part of the neighbourhood outside the synthesised texture must be initialised with known values. Iversen and Lønnestad [48] tried three methods of initialising this area: uniformly distributed random numbers in the range $[I_{min}, I_{max}]$, synthesising values using additional autoregressive models for the initial rows and columns, and the mean of the original image, $\bar{I}$. They concluded that using the mean of the original image produced the visually best results. Two of their methods and two new methods were tested for setting the initial values:

1. image mean, $\bar{I}$.
2. uniformly distributed random numbers in the range $[I_{min}, I_{max}]$.
3. Gaussian distributed random numbers with mean $\bar{I}$, standard deviation σ_I and limited to the range $[I_{min}, I_{max}]$.
4. Gaussian distributed random numbers with mean $\bar{I}$, standard deviation σ_I and limited to the range [0, 255].

The input textures are grey-scale images limited to the range 0 to 255 and the output images need to be limited to the range 0 to 255 (to be the same as the real images). Using the linear autoregressive synthesis method the output texture is not limited to this range, hence a method of restricting the range was required. There are two basic approaches to this: limit the output during the texture generation or limit after the texture generation. Limiting during texture generation causes the modified output to be fed back into the synthesis process and hence alters later output. The following methods for limiting the output during generation were tested:

1. No limiting during generation.
2. Noise limited to $\pm 3\sigma_n$.
3. The output including the noise is limited to the range $[I_{min}, I_{max}]$.
4. The output, including the noise, is limited to $\bar{I} \pm 3\sigma_I$.
5. The output, including noise, is limited to the range [0, 255].

The following post-processing methods for limiting the output after generation were tested:

1. Output is limited to the maximum and minimum possible pixel values [0, 255].
2. Output is limited to the original texture minima and maxima $[I_{min}, I_{max}]$.
3. The output range is scaled into the range [0, 255].
4. The output is compressed to have the same mean and standard deviation as original image and then is limited to the range [0, 255].
5. The output is adjusted to have the same mean as the original image and the synthesised minima and maxima mapped into the range $[I_{min}, I_{max}]$.

The large number of combinations of options (200) prevented exhaustive testing to find the visually best generation and measurement method, hence preliminary testing (using a neighbourhood distance of 3), on a small number of textures, was used to eliminate options which were clearly unusable. This showed that the second initialisation method (uniformly distributed random numbers) consistently produced unrealistic textures, in particular creating bars near the image border. The image border is where the impact of the initialisation method can be most clearly seen as the neighbourhood mostly contains the initialisation values. The second output limiting method (limiting to $[I_{min}, I_{max}]$)

created patches of either I_{min} or I_{max} in the synthesised texture which were not present in the original. These two methods were not investigated any further.

The effect of allowing the texture measurement to wrap around the image was tested by using the first initialisation method, the first limiting method and the first post-processing method. These methods were chosen as they involve least modification of the method and hence should have the smallest impact on the results. At neighbourhood distances (d_n) of 3, 4 and 5 the synthesised textures of the wrapped and unwrapped measurements were very difficult to discriminate visually. However, the wrapped measurement produced a visually better texture at a neighbourhood distance of 10. The performance of the unwrapped measurement would be expected to decrease as the neighbourhood distance increases, since it uses less of the image as d_n increases. Subsequent tests used the wrapped measurement method.

After the elimination of the clearly unusable options, the remaining options were tested in the following order:

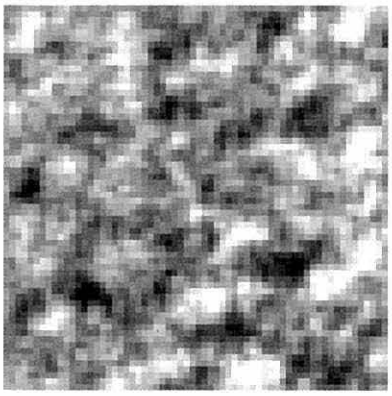
1. Comparison of initialisation methods one, three and four, whilst using the first limiting method and the first post-processing method. 16 lesion and skin textures were compared visually. This comparison showed the first initialisation method (using $\bar{I}$) to be the best. The other two methods sometimes generated bars near the image edges and the overall visual similarity with the original texture was lower. The image mean was used for the rest of the tests.
2. Comparison of the five limiting methods and the four post-processing methods, whilst using the image mean initialisation method. For each of 16 textures, the 20 combinations of options were displayed with the original texture. This comparison clearly showed the third post-processing method (scaling the output to the range [0, 255]) to be unusable. The first post-processing method of limiting the output to the range [0, 255] had poor performance compared with the remaining two methods (methods 4 and 5). The difference between the five limiting methods was not clear, so these tests were repeated with only post-processing methods 4 and 5 and a wider range of textures. These tests showed post-processing method 4 to be slightly better and method 4 to be the visually best limiting method.

The overall method chosen was the following:

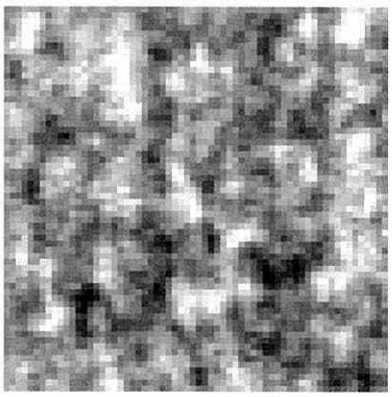
- Wrapped measurement.
- Initialisation method: 1. image mean.
- Limiting during generation method: 4. limiting to $\bar{I} \pm 3\sigma_I$.
- Post-processing method: 4. output compressed to the same mean and standard deviation as original image and then limited to the range [0, 255].

After determining the most suitable method the appropriate neighbourhood size was chosen. Large neighbourhood sizes should create better synthesised textures by capturing more of the large scale texture structure. However, the texture measurement and synthesis times increase more than linearly with the neighbourhood distance, hence it is useful to use the smallest acceptable neighbourhood distance. To assess visually the best neighbourhood distance, 17 skin and lesion texture samples were assessed at neighbourhood distances of 1 to 6, 8, 10, 12, 14, 16, 18 and 20. It was difficult to decide, in nearly all images, which was the best neighbourhood distance. Some textures, even at the larger neighbourhood distances, lacked the highlights and large scale structure present in the original textures.

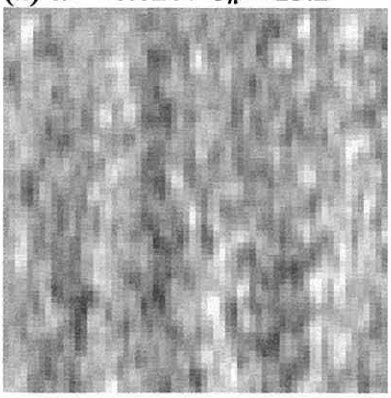
The mean and mode of the visually best neighbourhood distances (d_n) were 13.3 and 14 for skin texture, 16 and 14 for lesion texture and overall 14.3 and 14. To ensure reasonable quality synthesis of most of the textures, a neighbourhood distance of 16 was chosen. Increasing the neighbourhood distance above the minimum acceptable did not degrade any of the synthesised textures. The synthesised textures for the images shown in Figure 9-4 are given in Figure 9-9 with noise means and standard deviations. Comparing corresponding textures in Figure 9-4 and Figure 9-9 shows that the size of the noise parameters (mean and standard deviation) indicates the degree of visual similarity between the synthesised and original textures. For example, Figure 9-4(a) (chair covering) and Figure 9-9(a) with a large noise standard deviation, are not as visually similar as Figure 9-4(c) (floor) is to Figure 9-9(c) which has a smaller noise standard deviation. Comparing Figure 9-7 and Figure 9-9 clearly shows the superiority of the linear autoregressive method, which is used in the rest of this chapter, over the power spectrum method.



(a) $\overline{n} = 0.0207$ $\sigma_n = 13.2$

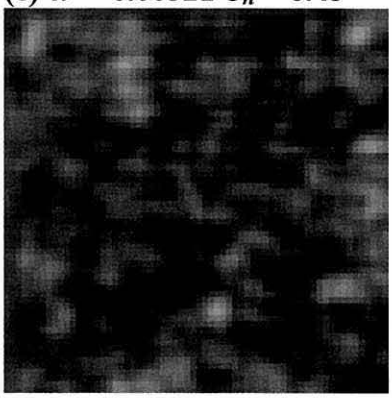


(b) $\overline{n} = 0.0149$ $\sigma_n = 11.6$

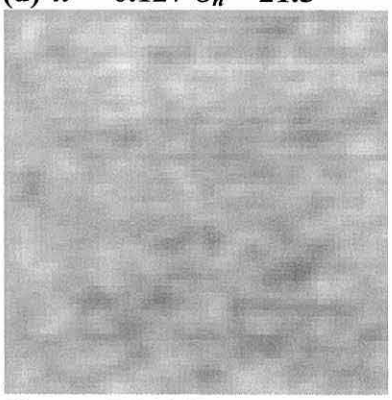


(c) $\overline{n} = 0.00321$ $\sigma_n = 6.45$

(d) $\overline{n} = 0.127$ $\sigma_n = 21.5$



(e) $\overline{n} = 0.0131$ $\sigma_n = 4.09$



(f) $\overline{n} = 0.00148$ $\sigma_n = 1.94$

Figure 9-9: Example linear autoregressive synthesised textures. (a) Chair covering. (b) Floor. (c) Table top. (d) Fieldstone. (e) Lesion. (f) Skin. A neighbourhood distance of 16 was used for all these images. $\overline{n}$ and σ_n are noise mean and standard deviation respectively.

9.5 Creation of the Simulated Image

To create the simulated skin image the lesion shape with the boundary transition, skin and lesion texture were combined. The synthesis process starts by creating a suitable lesion shape (section 9.2, e.g. Figure 9-10(a)) in which black and white represent lesion and skin respectively. A boundary transition is applied (section 9.3, e.g. Figure 9-10(b)) to simulate the indistinct transition between skin and lesion. This simulated lesion image with a boundary transition controls the mixing of the skin and lesion textures. The two textures are intensity mixed with the blurred lesion shape image controlling the proportion of each texture present in the output image. This means where the blurred lesion image is white only skin texture is present and where it is black only lesion texture is present. In intermediate grey areas, the intensity of each texture varies with the intensity of the blurred lesion image. The complete sequence of images is illustrated in Figure 9-10.

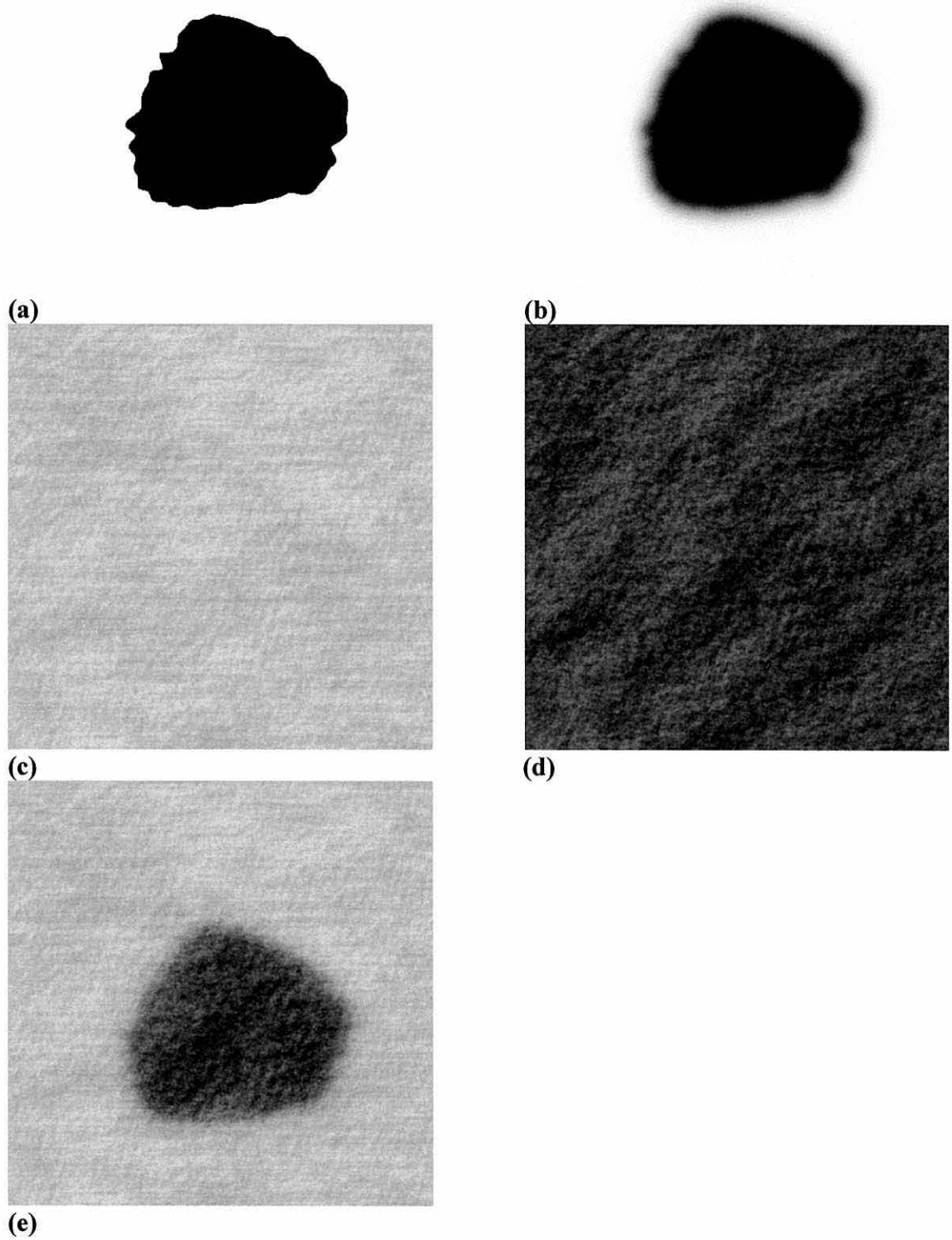


Figure 9-10: Example of the generation of a simulated image. (a) Lesion shape from Figure 9-1(c). (b) Blurred lesion shape from Figure 9-3(a). (c) and (d) synthesised skin and lesion textures created from textures shown in Figure 9-4(e) and (f). (e) Synthesised image. These images are 512 by 512 pixels.

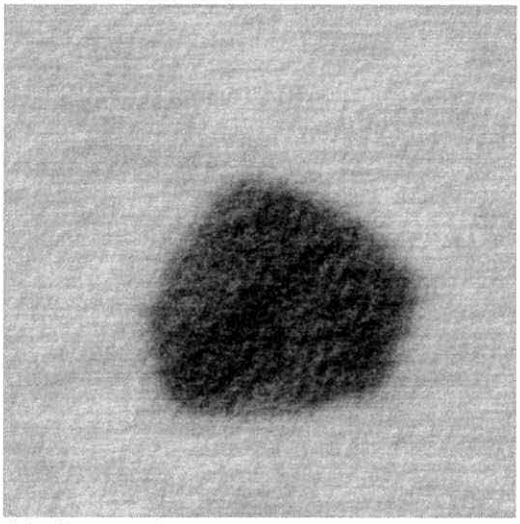
9.6 Comparison of True and Estimated Boundaries

The work described so far in this chapter and in chapters 6 and 8 allows the creation of a simulated skin-lesion image and the determination of a lesion's boundary. This creates true and estimated boundaries which can be compared to measure the performance of the boundary finding method. The true boundary, in the simulated images, is the co-ordinates of the outside of the shape created in section 9.2. A method for comparing these boundaries based on the area which they enclose is presented in this section.

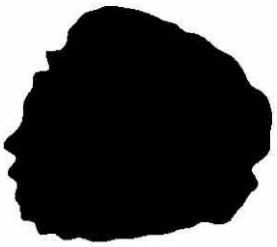
The true and estimated boundaries are unlikely to have the same number of points, hence a direct comparison between the corresponding points, on each boundary, is not possible. Even in cases where the two boundaries have the same number of points, it would be difficult to decide which are the corresponding points. It is possible to re-sample each boundary on a number of radial lines from the boundary's centre, creating two new sets of points, which could be compared by the radial distance between them. This would not use all of the boundary points and hence is an approximation which may be inaccurate, hence was not investigated.

To obtain an accurate comparison between boundaries of different lengths, the area which they enclose can be used. This allows measurement of the area which they have in common (true-and-estimated area or true positive region), the area which is covered by the true boundary and not covered by the estimated boundary (true-not-estimated area or false negative region) and the area which is covered by the estimated boundary and not covered by the true boundary (estimated-not-true area or false positive region). To make these independent of lesion size they are expressed as a percentage of the true lesion area.

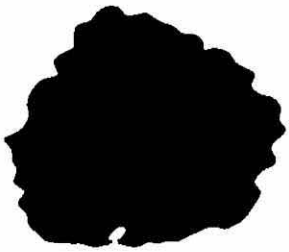
These measurements are performed by creating images with the relevant areas filled in and counting the filled pixels, as illustrated in Figure 9-11. Filled true and estimated boundaries are created by the method described in the previous chapter in section 8.4.3. The true and estimated boundary areas are logically ANDed to create the true-and-estimated area. This true-and-estimated area is logically exclusive-ored with the estimated area to create the estimated-not-true area. The true-not-estimated area is created by logically exclusive-oring the true-and-estimated area with the true area.



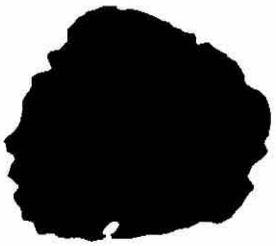
(a) Synthesised image.



(b) True boundary.



(c) Estimated boundary.



(d) True-and-estimated area.

(e) Estimated-not-true area.

(f) True-not-estimated area.

Figure 9-11: True and estimated boundary comparison. (a) True boundary. Area = 46769 pixels. (b) Estimated boundary. Area = 47444 pixels. (c) True-and-estimated area = 46034 pixels or 98.4% of the true area. (d) Estimated-not-true area = 1410 pixels or 3.0% of the true area. (e) True-not-estimated area = 735 pixels or 1.6% of the true area.

9.7 Testing of the Edge Focusing Algorithm on Synthesised Images

This section describes the tests performed on the edge focusing algorithm and lesion location and isolation method presented in chapter 6. In the synthesised images, there are three main factors that can be independently controlled:

- **lesion shape** - each shape can be re-created by using the same parameters and the same seed for the random number generator. As the shape generation is a quick process, the lesion shapes were created as required.
- **boundary transition** - this is performed by low-pass filtering and hence can be time consuming. Consequently filtered lesion shapes were created only once.
- **skin and lesion texture** - for larger neighbourhood sizes the linear autoregressive texture analysis can be time consuming. The synthesis takes less time, but still is time consuming, hence the use of texture analysis and synthesis was kept to a minimum. The skin and lesion textures were analysed and synthesised in pairs taken from the same image, to ensure they matched in scale and lighting.

The edge focusing algorithms (LoG and Canny) were tested on a total of 120 images. To perform tests on a range of lesion shapes, boundary transitions and skin/lesion textures in a reasonable computation time the following sequence was used:

1. shape generation - 10 lesion shapes were created, which provided the true boundary.
2. boundary transition - 4 sets of boundary transitions were created for each lesion shape. The first set had no filtering applied.
3. texture generation - 3 skin/lesion texture pairs were synthesised, using the linear autoregressive method described earlier, for each of the 10 lesion shapes giving a total of 30 skin and lesion texture images. Each of the synthesised skin/lesion textures, created from the same measurements, was different as different random number seeds were used for each. The set-up given at the end of section 9.4.2 with a neighbourhood distance of 16 was used for the skin and lesion textures.
4. for each of the 4 boundary transitions do steps 5 to 11.
    5. for each of 3 skin/lesion textures do steps 6 to 11.
        6. for each of the 10 lesion shapes do steps 7 to 11.
            7. create the synthesised image by mixing the skin and lesion textures.

8. identify the simulated lesion and determine the starting space constant using the process described in chapter 6.
9. edge focusing algorithm - both LoG and Canny edge focusing were tested.
10. selection of the best boundary - the method described in the previous chapter using the average intensity just inside and outside the boundary was used.
11. comparison of the true and estimated boundaries - this was described in the previous section.

The parameters used for the lesion location and isolation process were those presented in chapter 6, which were found by analysing real images. Although, better performance may have been achievable by using analysis of the simulated images to set these parameters, this method is more realistic as the training set is independent from the test set. Similarly, the size of the area used in selecting the most suitable boundary was set by the analysis of real images presented in chapter 8.

The lesion location and isolation process correctly identified all simulated lesions. A summary of the boundary comparison results is presented in Table 9-I and Table 9-II and summarised in Table 9-III. These show the performance, indicated by the common true-and-estimated area, of the edge focusing algorithm decreases as the boundary blurring increases. For the LoG edge focuser the estimated-not-true area, which represents the area outside the true lesion boundary that is included, increases faster than the true-not-estimated area, which indicates the area inside the lesion that is not included. The reverse holds for the Canny edge focuser. This difference between LoG and Canny is the expected effect of the differing edge displacements, which was described in section 8.6 and illustrated in Figure 8-9. For curved edges the LoG edge detector places its edge on the higher intensity side (i.e. skin) and the Canny edge detector places its edge on the lower intensity side (i.e. lesion).

The average performance of the LoG and Canny edge focusers were statistically similar[1] for the true-and-estimated area and true-not-estimated area measurements. However, the Canny edge focuser performed statistically significantly better in the estimated-not-true area. This difference was mainly caused by errors on small lesion shapes. In particular, one shape caused most of the poor performance for the estimated-not-true area, for the LoG edge focuser. This shape always had the highest percentage estimated-not-true area for all textures and boundaries. This shape only occupied about one fifth of the image width and height, causing it to be blended into the white background with the smoother boundary transitions. Hence, this image was not particularly realistic and once it had been removed the estimated-not-true area had an average of 13.2% and standard deviation of 13.3%. This performance is still statistically worse than the Canny edge focuser, but is much closer and has a much smaller standard deviation. This shows the importance of checking that a lesion image is large enough to analyse.

Figure 9-12 shows typical examples of the simulated images with the four boundary transitions together with the true boundary, LoG edge focusing boundary and Canny edge focusing boundary. These examples illustrate the choice of boundary transitions used to create the simulated images. It can been seen from these images that applying no boundary transition (Figure 9-12(a)) results in unrealistic and easily analysed images. This is reflected in the high performance of both edge focusing algorithms. Figure 9-12(b) and Figure 9-12(c) illustrate more realistic boundary transitions which are typical of most real images and result in less accurate boundaries. The final boundary transition, illustrated in Figure 9-12(d), was chosen as an extreme value to show the effects of analysing very difficult images which occur infrequently. In this case, the boundary is very difficult to identify visually and the corresponding edge focusing performance is not as good.

[1] An F-test [75] was used to determine if the samples (LoG and Canny edge focuser area comparisons) had significantly different variances. For samples with significantly different variances, the Student's t-test for significantly different variances [75] was used to determine the similarity between the sample means. For samples with similar variances, the Student's t-test for similar variances [75] was used to determine the similarity between the sample means.

Boundary transition cut-off freq.	% true-and-estimated area		% estimated-not-true area		% true-not-estimated area	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
none	99.6	0.7	1.8	1.1	0.4	0.7
0.0156	94.1	3.1	8.0	5.2	5.9	3.1
0.0078	89.2	4.0	18.0	11.5	10.8	4.0
0.0039	83.0	9.0	52.4	94.2	17.0	9.0

Table 9-I: Performance of the LoG edge focusing algorithm on simulated images.

Boundary transition cut-off freq.	% true-and-estimated area		% estimated-not-true area		% true-not-estimated area	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
none	99.8	0.2	1.5	0.8	0.2	0.2
0.0156	92.8	4.0	3.2	2.0	7.2	4.0
0.0078	88.7	5.0	5.5	3.7	11.3	5.0
0.0039	80.8	10.5	8.0	7.8	19.2	10.5

Table 9-II: Performance of the Canny edge focusing algorithm on simulated images.

Edge focusing method	% true-and-estimated area		% estimated-not-true area		% true-not-estimated area	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
LoG	91.5	8.0	20.0	50.8	8.5	8
Canny	90.5	9.2	4.6	5.1	9.5	9.2

Table 9-III: Average performance of the LoG and Canny edge focusing algorithms on simulated images.

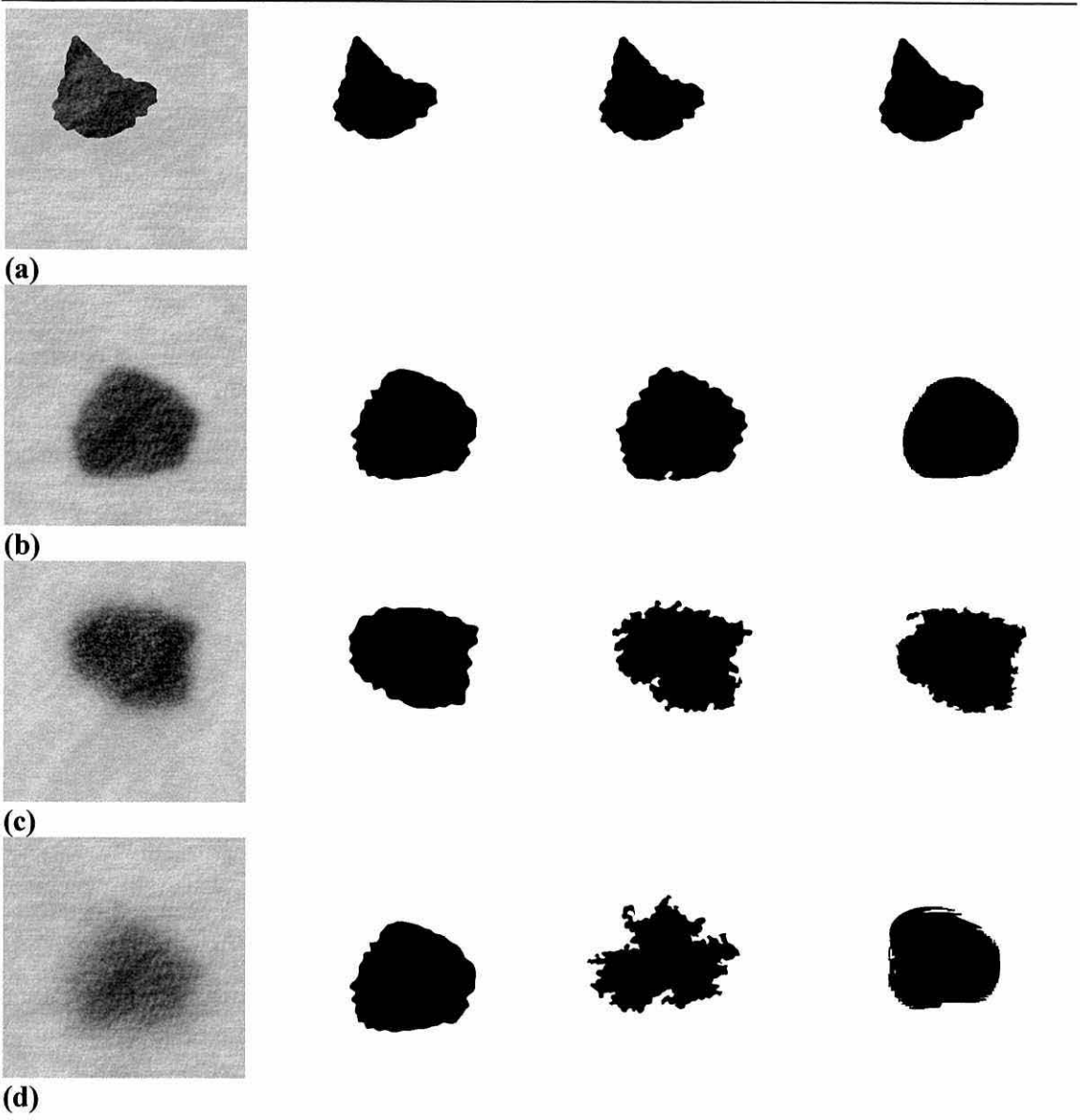


Figure 9-12: Example simulated skin images with true, LoG edge focusing and Canny edge focusing boundaries. Each set of images shows, from left to right, the simulated image, true boundary, LoG edge focusing boundary and Canny edge focusing boundary. (a) No boundary transition. (b) Boundary transition cut-off frequency = 0.0156. (c) Boundary transition cut-off frequency = 0.0078. (d) Boundary transition cut-off frequency = 0.0039.

9.8 Conclusions

This chapter has described the synthesis of simulated skin and lesion images and the comparison of a true boundary with an estimated boundary. This allowed the testing of the edge focusing algorithm presented in the previous chapter and the work presented in chapter 6.

The simulated images were synthesised in three stages: shape generation, boundary transition and texture generation. Lesion-like shapes were created by a new method using ellipses with random large and small scale irregularities. A range of boundary transitions were created by controlling the intensity mixing of the skin and lesion textures with a low-pass filter which "blurred" the lesion shape. Lesion and skin textures were synthesised from measurements of real skin and lesion. Two basic methods of texture synthesis were investigated using visual comparison of the original and synthesised textures. The first method using an image's power spectrum produced textures which were not visually similar to the original textures. A number of variants of the second method (linear autoregressive texture synthesis) were developed and tested to select the most suitable version for this application. Using the selected variant, the effect of the neighbourhood size on the visual quality of the synthesised skin and lesion textures was tested to select the most suitable neighbourhood size. With this method of linear autoregressive texture synthesis and neighbourhood size, the textures were visually similar to the original textures. Extensions of this approach and the use of texture in segmentation are discussed in the next chapter.

A new method for comparing true and estimated boundaries, using the area enclosed, was presented. This method in conjunction with the simulated skin lesion images was used to test the lesion isolation and location method (from chapter 6), and to test and compare the LoG and Canny edge focusing algorithms. The simulated lesion images and boundary comparison method could also be used to compare the relative performance of the edge focusing algorithm and other boundary detection algorithms. However, although considered desirable, due to time constraints it was not possible to undertake this work, which is described at the beginning of this chapter (section 9.1). In addition, again due to time constraints it was not possible to synthesise images with "benign" and "malignant" boundaries, which would have allowed the assessment of the effects of different boundary detectors and shape classifiers on the accuracy of computer classifica-

tion of such boundary types. Both of these areas of work could have demonstrated whether the edge focusing algorithm and shape analysis methods presented in thesis provide an improvement over the results of using published algorithms previously applied to skin lesions.

Using the parameters, derived from real images, given in chapter 6, the lesion location and isolation method correctly identified all the simulated lesions. Assessment of both the LoG and Canny edge focusing algorithm showed that the true-and-estimated area (i.e. the area which the estimated and true boundary have in common) decreased as the boundary became less distinct. For the LoG edge focuser, the estimated-not-true area (i.e. the area outside the true boundary which is included) increased faster than the true-not-estimated area (i.e. the area inside the true boundary that is not included). The reverse was found for the Canny edge focuser, which is the expected effect of the differing edge displacement of LoG and Canny edge detectors, which was described in section 8.6. The average performance of the LoG and Canny edge focusers was statistically similar for the true-and-estimated area and true-not-estimated area measurements. However, the Canny edge focuser performed statistically significantly better in the estimated-not-true area measurements due to better performance on small lesion shapes. Overall, on simulated lesion images with boundaries similar to those observed in real images both algorithms had good performance.

Chapter 10

Development of Computerised Skin Cancer Diagnosis

10.1 Introduction

This chapter discusses the development of a computer based tool to perform or assist in the diagnosis of skin cancer. The discussion shows how the work presented in this thesis and other skin cancer image processing research could be incorporated into such a system. It indicates areas of future research and other research which may be applied to this area.

The chapters starts by discussing the factors that should be considered whilst developing techniques for analysing skin cancer images. With these factors in mind, the development and use of these techniques is discussed. The use of real and simulated skin cancer images for assessing and testing the measurement techniques is examined. The chapter concludes by considering the implicit and explicit acquisition and use of knowledge for the early detection and diagnosis of skin cancer.

10.2 Measurement Techniques

To perform computerised diagnosis of skin cancer, it is necessary to measure lesion features. The measurement techniques, the selection of features to be measured and the points to be considered in developing and using these techniques are discussed in this section. It would be possible for a human to assess a lesion and enter measurements into a diagnostic system. However, it is difficult for humans to make consistent judgements [39, 59, 60] and there may not be a suitable human available, since considerable training is required. Consequently, the techniques discussed here are required to perform this assessment.

A number of requirements that a computerised diagnostic system must fulfil to be useful in a clinical environment were described in section 2.8 and are summarised here:

- **ease of use.**
- **tolerance of image capture conditions.**
- **reasonable equipment and operating costs.**
- **speed.**
- **repeatability.**
- **robustness.**

In the development of an automated diagnostic system, the question of which measurements are required must be considered. Human experts can provide guidance on the types of measurements required and knowledge of the effectiveness of these measurements (when performed by humans). Hence, the initial measurement selection can come from the areas that human experts consider useful, which includes those in the checklists described in chapter 2. These checklists indicate that lesion shape is particularly important and hence accurate, reliable and robust methods of locating lesion boundaries are very important.

The work described in this thesis was based on the application of image processing techniques to digitised optical skin images, which can be acquired easily by non-invasive and non-contact methods, making them quick, easy and cost effective to obtain. Current human diagnostic methods [58, 59, 33, 62] are largely based on the visual assessment of lesions, hence knowledge already exists which can assist in the development of image processing methods. Although the analysis of skin images has these advantages, there are some diagnostic factors that it cannot measure. Consequently, it may be necessary to use image analysis in conjunction with other techniques to obtain sufficient information to perform accurate and reliable diagnosis. One major factor that cannot be measured by the analysis of a single external optical skin image is the Breslow thickness (tumour thickness) [4, 9], which was described in section 2.7. The Breslow thickness can be measured by an instrument called the 'nevoscope' [24, 22], which was described in chapter 4, and is only an experimental technique. High frequency ultrasound can also measure Breslow thickness, but only gives accurate measurements for some lesions [86]. Historical factors such as previous cases of skin lesions in the patient

or the patient's family, life style, age, sex, etc. need to be manually entered into a computer based diagnostic system.

Whilst developing any system it is necessary to ensure that it does what is expected (verification) and it does what we want it to do (validation) [87]. (Verification and validation were discussed in more detail in section 8.7.) This testing should exercise the system on data (images) that are similar to those on which it will operate. It is important to have separate, statistically independent, training (development/verification) and test (validation) data (image) sets [21]. This allows unbiased results to be obtained for the test set, which then can give a more reliable indication of future performance. However, this may be difficult to accomplish because of the difficulties in collecting sufficient images and the time required to process them. Two approaches to creating test data exist: black-box and white-box testing [87]. In black-box testing only an external description of the system, without information on the internal structure, is used to create test data. However, in white-box testing knowledge of the internal structure (i.e. source code) assists in the construction and acquisition of test data. Test images require known characteristics, which can be provided by image synthesis (in which the characteristics can be tightly controlled) and images which have already been diagnosed. However, only the overall diagnosis can be obtained (by histology) for the real images as diagnostic measurements cannot be consistently and objectively found by humans. It is difficult to test a computerised skin cancer diagnostic system in a clinical environment until it is completely developed, hence separately captured images are required to develop and test it. Some testing of tolerance to image capture conditions may be accomplished by altering already captured images. For example, the effects of image rotation, image scaling and lighting changes could be applied by image processing techniques.

During development it would be useful to store all information about a particular lesion together in a "feature file" [92], i.e. a database. Then, separate programs, which communicate via this "feature file", can be used to develop the measurement techniques. This facilitates development by allowing analysis techniques to be independently developed at different times by different people.

To automate the examination of patients with a large number of lesions, a computer controlled "skin scanning" robot could be developed [24], which would be able to

identify individual lesions, determine their characteristics and whether they were new or changed. Development of such a robot would require research into methods for locating the same lesion at different times as well as methods to perform diagnostic analysis. Knowledge of the features of a particular lesion alone may not be sufficient to re-locate it because of the changes which may have occurred in those features and the inaccuracies in assessing the features. Consequently, the relationship of a particular lesion to its neighbours (the "constellation" pattern) could assist in its location [78]. This analysis of the "constellation" pattern could also be used without a robot to identify the lesions within the field of view of a manually aimed camera. Although a "skin scanning" robot would be able to examine a large number of lesions more rapidly than a manually operated camera, its development, equipment and operating costs would be much higher than that of a manual system.

The following sections discuss the work presented in this thesis and further work that could be undertaken to develop a computerised skin cancer diagnosis system, taking into account the points considered above. The sections are presented in roughly the order that they would be used in a system, which is similar to the order in which the topics were presented earlier in this thesis.

10.2.1 Skin Lesion Location and Isolation and Scale Measurement

Approximately locating, isolating and determining the scale of a lesion is the first stage in an automated diagnostic system. Location and isolation are the processes of finding the part of the image containing the lesion and separating that area from the surrounding image. It does not cover finding an accurate boundary for the lesion, which is a more complex task discussed in the next section. This process increases the usability, reliability and robustness of a system by reducing the constraints on the contents of the captured images, so they are not confined to containing skin and lesion alone, and allows images to contain multiple lesions. This tolerance and robustness would make a system easier to use by reducing the requirement for careful preparation before capturing an image.

A process for locating and isolating a single lesion was described in chapter 6 and is summarised by:

1. Image filtering - median filter.
2. Binary thresholding.
3. Image simplification - by binary morphological closing - to produce a cleaned thresholded image.
4. Lesion identification - by shape and size.
5. Output information - cropped image, lesion bounding box and lesion size.

This process could be extended to images containing multiple lesions by enhancing the lesion identification process. This enhanced process could follow the steps listed below and would start from the cleaned thresholded image:

1. Background characterisation - characterise the image background (skin) to allow comparison of suspected lesions areas with the skin for identification and diagnosis. Other objects, such as clothing, must be distinguished from the skin at this stage to prevent their inclusion in the skin characterisation. The binary inverse of the cleaned thresholded image could act as a mask to indicate areas which may be skin, but this may include parts of the image that are not skin. Consequently other methods, such as texture analysis, may be required to identify skin areas.
2. Object characterisation - assign each region, in the segmented image, characteristics based on its size, shape (irregularity, bulkiness, etc.) and grey-level/colour features (texture, average value, etc.).
3. Removal of unmistakably non-lesion objects - remove objects which, by the virtue of characteristics such as shape and colour, are clearly not lesions (e.g. clothing and rulers).
4. Removal and ranking of the remaining areas - use further analysis to remove any remaining non-lesion areas and rank the suspected lesions in order of importance.

This enhanced process would improve the reliability of the identification, allow analysis of multiple lesion images and provide diagnostic information. Each suspected lesion identified by this process would then be examined in detail.

The process presented in chapter 6 uses an image oriented box to represent the lesion. An irregular shaped area, provided by a mask, would be more representative of the lesion shape and may improve the lesion's analysis. The allowance for edge displacement (see section 6.7) would be incorporated by binary dilation with a disc. This should slightly reduce the time required for the first stage of the edge focusing and may improve its reliability.

Lesion size is diagnostically very important, hence the measurement of image scale is necessary. Placing a calibrated ruler near the lesion (to ensure it is at the same distance) would be the simplest and easiest method of determining image scale. By using a ruler of a known colour, compensation for camera set-up and lighting could also be performed. Another method of measuring scale and skin curvature would be by stereoscopic imaging [48], but this has the drawbacks of increased equipment costs, increased computation and reduced ease of use. Other methods of determining the distance of the skin from the camera, and hence the lesion size, include laser range finders, ultrasound and camera focus [48].

To be reliable and robust, a computerised system must be able to determine when measurements from an image are unusable and be able to inform the user of how to improve the image. These checks should be performed early in the processing, so time is not wasted taking unusable measurements. Two main faults in captured images are likely to be inappropriate lesion size and unsatisfactory lighting conditions, which both could be identified during the initial processing discussed in this section. For example, when the lesion either fills too much or too little of the image it is unusable, and the system must ask the user to capture another image at more appropriate scale. When the system controls some parts of the equipment (e.g. camera set-up and lighting) it would be able to make adjustments to minimise or remove the detected problems.

10.2.2 Boundary Detection

Locating a lesion's boundary is important as it directly allows measurement of the diagnostically important factors of lesion size, asymmetry, roundness and border irregularity. (The measurement of roundness and irregularity were described in chapter 5.) It also allows measurements of colour, colour variations and texture to be performed only on the lesion not on a combination of lesion and skin. In addition, many of the points in the

diagnostic checklists, described in chapter 2, can only be assessed once the lesion's boundary has been found. The variable nature of skin images makes it difficult to produce boundary detection methods which are reliable, repeatable and robust. In many cases the border is very subjective, not clearly defined and may even be difficult for the human eye to detect consistently. The lesion boundary found by an automated system may be different from that perceived by a human expert, who can be affected by the underlying diagnosis when determining the boundary. This difference between the automatically and human determined boundary is not important, provided that the automatically determined boundary is consistently and reliably found. Human experts are not generally aware of their methods of boundary detection which makes it difficult to develop algorithms directly based on their methods. However, experts can provide knowledge which may guide the development of algorithms.

The wide range of skin images encountered by a computerised analysis system may produce difficulties in finding a single boundary detection method which is reliable and accurate on all images. To test the performance of algorithms and to discover whether multiple algorithms are necessary will require a test set containing a reasonable number of real test images taken under a range of conditions together with simulated images. This test set should contain multiple images of some lesions to ensure that boundaries are consistently found. The detected boundaries may be compared with one another and with the true boundary, when available, by the enclosed area based comparison method presented in section 9.6. To ensure that the algorithms are image rotation invariant, it will be necessary to present the same image at a number of angles. If a single method which is reliable and accurate on all images cannot be found then a hybrid approach will be required which combines the results from a number of methods [35]. To combine boundaries from different methods or select a single method, confidence levels are required for the whole boundary and for boundary segments. These confidence ratings will allow the selection of either a whole boundary or the combining of the best segments of a number of boundaries.

In chapter 8, an edge focusing algorithm for skin lesion boundary detection was presented. This algorithm, although appearing to be robust, is too slow to be used in a practical system, because of the computation required to perform the large convolutions used in the edge detection. In comparison with the convolution, other parts of the

algorithm take very little time and hence do not require speeding up. One approach to reduce the time required is to parallelise the algorithm, so the time consuming operations are performed on a number of processors. An alternative method would be to change the FIR (finite impulse response) filter based edge detector to one based on an IIR (infinite impulse response or recursive) filter, which should give similar results but require less computation as fewer multiplications and additions are used for each pixel [19, 85, 95].

The edge focusing algorithm produces a series of boundaries from which a boundary for the particular lesion is selected. Enhancement of its performance, i.e. reliability and accuracy, may be achievable in two areas: boundary generation and boundary selection. The boundary generation is affected by the sharpness of lesion boundary which varies between lesions and may vary within an individual lesion, hence a boundary obtained by a small scale edge detector is sometimes most appropriate and at other times the boundary may not be as clearly defined so a less precise boundary produced by a large scale edge detector is the best that can be obtained. Currently, the boundary is assessed as a whole, hence, although the selected boundary may be the best available out of the series of boundaries created, it may not be the best possible boundary that can be created. By measuring the quality of each segment of the boundary as it is created, it may be possible to use the large scale boundary where the boundary is not clearly defined and the small scale boundary where it is better defined. The boundary sharpness information, that would be required to implement this, may also be diagnostically useful and may be obtainable from the edge strength across the boundary (i.e. the intensity gradient). The edge focusing method uses either an LoG or Canny edge detector, which place their output at a distance from the true edge, dependent on the edge curvature and space constant. (For straight edges, both LoG and Canny are correct.) By combining the LoG and second derivative in gradient direction, which displace their outputs in opposite directions, the PLUS operator [96] can be created which provides greater accuracy for curved edges. The boundary selection method, used in chapter 8, was based on the average contrast between an area just inside and just outside the boundary. Alternative boundary selection methods may use the edge strength, texture measurements or a combination of these.

The edge focusing method is based on the image intensity, but there are other methods of image segmentation which use texture. Texture may be used directly to segment an image by classifying each part of the image or indirectly by transforming the image into another domain where the boundary is better defined and another method, such as the edge focusing algorithm or thresholding, may be applied. An example of a texture based transform is the co-occurrence (or spatial grey-level dependence matrices) transform [38] which emphasises the differences between typical and atypical image features. Texture analysis is discussed in more detail in the following section and in chapter 9.

Snakes (or active contour models) [49, 53, 72] may be suitable for finding lesion boundaries. A snake is deformed by "forces" resulting from its geometry (distance between points, curvature, etc.) and from the image (intensity gradient, i.e. edge strength). These "forces" can be used to form an energy function which can be iteratively minimised to find the boundary. The snake is able to lock onto edges and localise them accurately whilst remaining continuous because of the geometry "forces", hence can use small scale detail to find an accurate boundary. Snakes require initialisation by placing them near the desired boundary, which in lesion boundary detection could be taken from the cleaned thresholded image produced in the lesion localisation and isolation process presented in chapter 6.

10.2.3 Texture Analysis

Texture may be used in segmentation (i.e. finding the lesion boundary), for obtaining diagnostic information and to perform measurements for image synthesis. Texture when combined with grey-level intensity and colour may improve the segmentation.

In chapter 9, the linear autoregressive texture model was shown to synthesise reasonably a range of lesion and skin textures. The quality of the synthesised textures is evidence that the model is a reasonable representation of the texture. Hence, it could be used to distinguish between different lesion and skin textures for segmentation and diagnosis. Currently, the texture analysis and synthesis are performed on rectangular areas, which is not a problem for synthesis as the texture can be cropped to the required shape after synthesis. However, it prevents analysis of the whole of a lesion or skin area and thus reduces the accuracy of the texture measurement and resulting synthesised texture. To

increase the accuracy, the texture measurement could be enhanced to analyse irregularly shaped areas, specified by a mask.

Variegated colouring, which can occur in melanomas, could be detected by texture analysis. "Skin creases", which are important as they may indicate the lesion's development stage [39], may also be detected by texture.

10.2.4 Shape Analysis

Lesion shape is a diagnostically important factor, which appears in both MacKie's and the American Cancer Society's diagnostic checklists. In chapter 5, methods of measuring roundness and irregularity through bulkiness and fractal dimension were presented and tested. These tests showed that bulkiness was a robust and suitable method of assessing lesion shape. However, fractal dimension measurement was shown to require high resolution data, which may not be available, and to be affected by the addition of Gaussian noise. Asymmetry is another important shape factor which Stoecker et al. [89] measured using the principal axes, image rotation and image mirroring. The principal axis obtained during the bulkiness calculation and a fast shear based image rotation algorithm [34] could be used to realise this method.

10.2.5 Colour Analysis

Colour could directly provide diagnostic information (e.g. identifying variegated colouring) and could improve the measurement of other factors. Colour may also provide assistance in identifying skin and lesion by separating them from background features, such as clothing, which are likely to have notably different colours.

For colour to be diagnostically useful, calibration may be necessary to ensure consistency between images, which may have been captured in different conditions. When used for a less diagnostically important task, such as approximately identifying skin and lesion, calibration may not be required. Calibration could be performed whilst measuring the image scale, by the method explained in section 10.2.1, and combined with global compensation using a separately digitised white reference image [41]. Using automatic colour calibration would increase the reliability and ease of use by removing the requirement for separate calibration.

To use colour diagnostically, it is necessary to know and examine the important colour ranges. The image capture colour space (probably RGB) may be converted by a fixed transform to an analytically more useful space, such as HSV (hue, saturation, value). Alternatively, the colour space may be changed for each image by an adaptive transform, such as Principal Component Transform [37].

A normal colour camera, which can capture a standard range of colours, has the advantages of low-cost and low-complexity, but may capture colours which are not useful and may not obtain sufficient detail in diagnostically useful colours. Filters with a monochrome camera can capture particular colour ranges, which gives increased information about these colours, but this is more complex than a colour camera. Consequently, a colour camera without filters would be better, provided that its performance was adequate. However, the spatial resolution of colour cameras is less than that of similarly priced monochrome cameras.

10.2.6 Synthesis of Skin and Lesion Images

The synthesis of simulated skin and lesion images, which was described in chapter 9, is required to create test images with tightly controlled characteristics so that measurements can be compared with a known input. The image synthesis was considered in three parts: shape generation, boundary transition and texture generation. Simulated lesion shapes were created from randomly placed ellipses with random diameters with random large and small scale irregularities. A low-pass filter was used to create the transition between skin and lesion. Skin and lesion textures were generated by a linear autoregressive texture model which used the analysis of real skin and lesion images to create the synthesised textures. The simulated images allowed the assessment of the edge focusing algorithm using area based measurements.

Enhancement of the three image synthesis parts and the addition of extra parts to improve the "realism" of the synthesised images could be performed in a number of areas: improving the quality of the synthesised skin/lesion textures, improving the lesion shape synthesis, improving the modelling of the transition between skin and lesion, addition of other skin features, synthesis of images containing multiple lesions, addition of non-skin features and modelling of image capture conditions. To maximise the "realism" of the synthesised images, the nature and probabilities of occurrence of particular features

should be set by studying real images. This principle was used in the selection of a bi-directional texture synthesis method, i.e. a method which could analyse a real texture and synthesise a similar new texture. However, this approach may not be appropriate where it is difficult to obtain accurate measurements from real images, e.g. in the lesion shape synthesis.

In chapter 9, a linear autoregressive texture model was shown to synthesise a range of lesion and skin textures. This texture analysis method uses matrix inversion which was performed by double precision floating point LU decomposition with backsubstitution [80]. It was found that double precision arithmetic was necessary to prevent the matrix inversion becoming unstable on some textures. Different and possibly better results may be obtained using a different matrix inversion method, such as singular value decomposition [80].

The addition of measurement scale (i.e. centimetres per image pixel) to the texture analysis and synthesis would allow the creation of similar textures at a range of scales, which would allow the evaluation of the effects of capturing the same lesion at different scales. Incorporation of the measurement scale may be achievable by an extension of the linear autoregressive texture model. By spatially varying the texture, variations in skin and lesion textures and the transition between skin and lesion could be modelled.

The current lesion shape synthesis method (section 9.2) is not based on measured lesion shapes, but contains the major shape components of most lesions: an approximately elliptical shape with large and small scale irregularities. To use measured lesion shapes for synthesis requires accurate boundary measurements, however the boundary detection method's accuracy needs to be assessed on simulated images before accurate shape measurements can be provided. Consequently, the approximate approach of creating shapes with reasonable shapes and sizes was used in section 9.2. Even with approximate real lesion boundaries the lesion size and large scale structure may be set as these do not depend on accurate lesion boundaries.

The transition between skin and lesion was simulated by using a low-pass filter to smooth the lesion shape image, which controlled the intensity mixing of the lesion and skin textures. This simple approximation provided a wide range of boundary transitions,

but is not based on numerical analysis of real boundary transitions because of the difficulty in analysing real boundary transitions. To enhance this stage, it may be possible to model the lesion/skin transition by finding and modelling the intensity gradient perpendicular to the lesion boundary. This model could control the blurring of the lesion shape image by controlling the intensity gradient between lesion and skin in this image.

The shape, texture and boundary transition synthesis methods could all be extended to model different types of lesions, i.e. benign and malignant ones. For the shape synthesis, this would involve the adjustment of the parameters to enable the generation of different types of boundaries. Similarly, the boundary transition would be changed by adjustment of its parameters. However, it may difficult to select suitable values for the parameters of the shape and boundary transition synthesis methods. Texture synthesis would use sample textures captured from different classes of lesions to generate different synthetic textures. When this is implemented it would be possible to assess the effects of different boundary detectors and shape classifiers on the accuracy of computer classification. This would make it possible to discover whether changes in boundary detection and shape measurement were likely to improve the performance of automatic skin lesion diagnostic systems based on computerised analysis of optical skin images.

Synthesis of images containing multiple lesions would assist in the evaluation of the lesion location and isolation process discussed in section 10.2.1. This enhancement of the current image synthesis process would require an extra image mixing stage for each additional lesion and a new texture for each lesion.

Hair, which commonly appears in real images, may affect the performance of boundary detection methods, texture analysis methods, etc., hence it may require simulation. The effects of hair may be captured in the skin/lesion texture analysis and synthesis. However, to synthesise hair more realistically and to assess the effects of the presence of hair may require a separate process. The first stage in synthesising hair would be its identification and localisation, which because of the small size of individual hairs may not be possible using colour and texture, which require reasonably large areas to assess. However, edge or ridge detection may be able to locate individual hairs and after localisation the shape and placement of individual hairs could be modelled to allow creation of coloured and textured lines to simulate it.

The addition of non-skin features, such as clothing and calibrated scales, would allow testing of system robustness and the ability to find and analyse a calibration scale. The modelling of non-skin features would require analysis of real images to determine the types of features which occur. The effects of image capture conditions, such as non-uniform lighting and camera set-up, could be applied as a post-processing stage to the simulated images.

10.3 Diagnostic Methods

After using a computer to acquire diagnostic measurements, a human expert could perform diagnosis, but they would require additional training to use the additional information [11]. Even with additional training, this approach may not be successful as, although computerised measurements may be more reliable than human assessment, current human diagnosis is relatively inaccurate [39, 59, 60]. In addition to these problems, human experts are expensive due to the extensive training required. Consequently, computer based measurements should be combined with computerised diagnosis to give cost efficient, reliable and accurate early diagnosis.

For a computerised system to perform or assist diagnosis, diagnostic knowledge must be captured. This is an important and difficult task, which can be performed either explicitly or implicitly. To explicitly capture knowledge, human experts are questioned to obtain their knowledge, which is then translated into a formal structure for entry into the system. In contrast, implicit knowledge capture acquires information directly from a training set.

Expert systems [14, 23] can use explicitly captured knowledge in the form of rules (or heuristics) to provide or assist diagnosis. Scores may be associated with the rules to allow calculation of probability of the resulting diagnosis. The chain of reasoning used in reaching a diagnosis can be explained by backtracking through the rules that were triggered (used), which allows an expert to review the diagnosis. The extensive and complex knowledge, obtained from training and experience, used by human experts may be difficult to acquire and condense for use in an expert system because experts are often unaware of the details of their thought processes. This makes explicit knowledge acquisition difficult, hence systems that capture (learn) knowledge from examples are required. However, explicitly obtained expertise can guide the construction of a

computerised skin cancer diagnostic system by assisting in the choice of measurements and providing diagnosed training sets.

Two methods of implicit knowledge capture are rule induction and artificial neural networks. Rule induction generates decision trees, which form an expert system, from the training data [14]. Although, like other expert systems, rule induction systems can produce their chain of reasoning, this explanation may not be fully comprehensible as the individual rules may be difficult to understand. Artificial neural networks [56, 46], which are inspired by the structure of biological nervous systems, contain simple non-linear computational elements (neurons) that are densely interconnected. They can learn using supervised training, in which the training set contains input-output pairs, or unsupervised training in which inputs are grouped into self-similar classes. Some neural networks can adapt to new training examples, without re-processing the existing training set, which is not typical of traditional statistical techniques. In addition, they can be more robust than statistical methods when the input data is generated by a non-linear process and has a strongly non-Gaussian distribution. Neural networks are particularly suited to recognising patterns in noisy data, which is important in this application as digital skin images contain many sources of noise. After training, the network should be able to generalise from the data in the training set to give correct responses to new unseen data. In the diagnosis of skin cancer, the overall system would be trained in a supervised manner as it needs to classify lesions into a small number of predetermined classes. In theory, a neural network could learn directly from a skin image and associated diagnosis. However, in practice this is not possible because the very large quantity of input data would require an impractically large network. Consequently, pre-processing is required in the form of the measurement techniques discussed in the previous section.

10.4 Conclusions

This chapter has placed the work presented in this thesis in the context of the development of a complete computerised skin cancer diagnosis system. It has outlined areas of future research, which could lead to this computerised diagnosis system. The two parts of such a diagnostic system were discussed: measurement techniques and diagnosis methods. The factors that should be considered whilst developing measurement techniques were examined. It concluded that although there are a lot of advantages to using

optical skin images, there are some diagnostic factors, such as historical information and the Breslow thickness [4, 9], which it can not provide. To ensure the quality of a computer based diagnostic system, it is important to have test images with known diagnosis and characteristics to allow extensive testing. It may be necessary to have multiple or hybrid boundary detection algorithms to handle the wide range of skin images. Texture may be used in segmentation, for obtaining diagnostic information and to perform measurements for image synthesis. Lesion shape is a diagnostically important factor, hence, it will be necessary to develop and/or use more shape analysis methods. Colour could directly provide diagnostic information and could improve the measurement of other factors. To use colour, it may be necessary to perform calibration and use an appropriate colour space to extract the relevant information. To allow further investigation of analysis methods, it would be useful to extend the image simulation method to create more realistic skin lesion images by improving the quality of each of its stages and by adding other features (including the synthesis of benign and malignant lesions), which are based on analysis of real skin images.

The inaccuracy and expense of current human diagnosis [39, 59, 60] means that it will be necessary to combine computer based measurements with computerised diagnosis to give cost effective, reliable and accurate early diagnosis. To build a computerised diagnostic system it will be necessary to implicitly and/or explicitly capture diagnostic knowledge. This chapter has presented guidance on producing a computerised diagnostic system for skin cancer, which may be realisable by combining this research with future research and development.

Chapter 11

Conclusions

This thesis has considered the analysis of digital optical skin cancer images and has presented methods for analysing lesion shape, locating and isolating lesions, determining lesion boundaries and synthesising simulated lesion and skin images. These methods would form parts of a computerised system for diagnosing skin cancer. In order for a computerised image analysis system to be practically usable it must be tolerant of its operating conditions. This is particularly important in the analysis of optical skin images as it is difficult to control the image contents and capture conditions. The research in this thesis has aimed to include this tolerance at all stages.

The literature review showed that, although many lesion boundary detection methods have been proposed [11, 98, 12, 30, 77, 36, 26], researchers attempting to construct automated diagnostic methods for skin lesions [29, 53] have found it necessary to use manual boundary detection as they considered the performance of automatic boundary detection to be inadequate. In addition, those methods [30, 36] that have been visually assessed by dermatologists were judged to be inadequate for an automated system, particularly on potentially fatal melanomas. Performance assessment is very difficult as there is no "gold standard" for lesion boundaries. The inadequate performance and the difficulty in assessing performance indicates two areas where research is required: new boundary detection methods and performance assessment methods. Consequently, a new boundary detection method was developed and assessed by a new method using simulated skin images.

Two methods of assessing lesion shape were presented and tested on real and artificial shapes. The "roundness" of a lesion was assessed by the bulkiness, which was obtained by an enhancement of the method proposed by Medalia [67]. A new method of creating shapes with known fractal dimensions allowed the assessment of fractal dimension measurement methods on shapes with fractal dimensions comparable with those of real

lesions. This allowed more reliable assessment of the fractal dimension measurement methods, which was not previously possible. This assessment showed that an implementation of "Flook's method" [30] was the most suitable, of three methods tested, for measuring structural and textural dimensions of lesion shapes. Radially distributed Gaussian noise applied to test shapes was used to assess the effects of errors in the location of lesion boundaries. This assessment showed the bulkiness to increase with the increasing noise, which is the expected effect of increasing the boundary irregularity. This combined with the textural and structural fractal dimensions increasing with increasing Gaussian noise shows the importance of accurate boundary detection. The effects of noise on fractal dimension measurement and the necessity of using high resolution shapes in its measurement indicated that it may not be suitable for assessing lesion shapes.

Before attempting a detailed analysis of a lesion it is necessary to separate it from the rest of the image, which prevents the unnecessary and possibly inaccurate analysis of parts of the image that are not of interest. To do this, a process was developed to isolate a single lesion and produce an image containing the lesion and sufficient surrounding skin with an indication of lesion size, location and bounding area. This process was able, using shape based heuristic rules, to correctly identify and isolate all of the lesions in a test set and also, using the same parameters, isolate all of the lesions in simulated skin lesion images, which were created to test this process and the edge focusing lesion boundary detection algorithm.

Initial examination of lesion boundary detection problem showed that fixed scale edge detectors are not suitable. When a small scale edge detector is used large numbers of unwanted, accurately located, small scale edges are detected and when a large scale edge detector is applied the boundary can be identified, but it is not possible to accurately locate it. Consequently, an edge focusing algorithm for lesion boundary detection was developed, which combines the advantages of small and large scale edge detection. This algorithm is a new application of edge focusing, which automatically controls the boundary during focusing and selects the output boundary, using image contrast. Through these developments, edge focusing was made suitable for lesion boundary detection and may also be suitable for other applications where automatic boundary detection is required for objects with boundaries which are not clearly defined. By combining

this algorithm with the process of isolating lesions, a system that can find lesion boundaries in a range of images was produced. Two variants, using Laplacian of Gaussian (LoG) and Canny edge detection, of this system were tested on real and simulated images. The tests with simulated skin images, which are described in the next paragraph, were used to compare the performance of the two variants. During the development of this algorithm, an improved zero-crossing detection method, with improved edge placement accuracy and edge continuity, was created with a clearer method of presentation which allows easier implementation.

In the development of any system it is important to test it with data of known characteristics, which makes testing with real images difficult. To overcome this problem the creation of simulated test images by a three stage process (shape generation, boundary transition modelling and texture generation) was investigated. For the shape generation, ellipses with random large and small scale irregularities were found to be suitable lesion-like shapes. A linear autoregressive texture model was found to create simulated grey-scale textures that were visually similar to lesion and skin textures. Low-pass filtering was used to control the combining of simulated lesion and skin textures to create an indistinct boundary between lesion and skin. This process created a range of images which were used to test the edge focusing algorithm by comparing the boundary of the simulated lesion with the detected boundary using a new area based measurement. This method could also be used to assess other lesion boundary detection methods and be used in other applications. Using parameters, derived from real images, the lesion location and isolation method correctly identified all the simulated lesions. Assessment of both the LoG and Canny edge focusing algorithms showed that the true-and-estimated area (i.e. the area which the estimated and true boundary have in common) decreased as the boundary became less distinct. For the LoG edge focuser, the estimated-not-true area (i.e. the area outside the true boundary which is included) increased faster than the true-not-estimated area (i.e. the area inside the true boundary that is not included). The reverse was found for the Canny edge focuser, which is the expected effect of the differing edge displacement of LoG and Canny edge detectors. The average performance of the LoG and Canny edge focusers was statistically similar for the true-and-estimated area and true-not-estimated area measurements. However, the Canny edge focuser performed statistically significantly better in the estimated-not-true area measurements due to better

performance on small lesion shapes. Overall, on simulated lesion images with boundaries similar to those observed in real images both algorithms had good performance.

Although it was considered desirable, it was not possible due to time constraints to extend this work to cover the assessment of other skin lesion boundary detection methods (such as [12, 30, 77, 36, 26]) and other boundary detection methods (such as [50, 54, 73]). Such an assessment would have compared the relative performance of edge focusing and other algorithms and could have identified their strengths and weakness. This would have demonstrated whether the edge focusing algorithm provides an improvement over boundary detection methods that have previously been applied to skin lesions. The assessment would have enabled the comparative testing, refinement and improvement of boundary detection and shape analysis algorithms, through gaining a better understanding of the behaviour of the algorithms under known conditions. In addition, although desirable, again due to time constraints it was not possible to synthesise images with "benign" and "malignant" boundaries, which would have allowed the assessment of the effects of different boundary detectors and shape classifiers on the accuracy of computer classification of such boundary types.

The use of the research presented in this thesis and further research in a complete computerised skin cancer diagnosis system was discussed. This discussion examined the additional research that could be undertaken and the factors that should be considered whilst doing this research. It was concluded that although there are a lot of advantages to using optical skin images, there are some diagnostic factors, such as historical information and the Breslow thickness [4, 9], which it can not provide. To ensure the quality of a computer based diagnostic system, it is important to have test images with known characteristics and diagnosis to allow extensive testing. It may be necessary to have multiple or hybrid boundary detection algorithms to handle the wide range of skin images. In addition to intensity, it may be possible to use texture or colour information to provide diagnostic information. Lesion shape is a diagnostically important factor, hence, it will be necessary to develop and/or use more shape analysis methods. To allow further investigation of analysis methods, it would be useful to extend the image simulation method to create more realistic skin lesion images by improving the quality of each of its stages and by adding other features which are based on analysis of real skin images. The inaccuracy and expense of current human diagnosis [39, 59, 60] means that it will

be necessary to combine computer based measurements with computerised diagnosis to give cost effective, reliable and accurate early diagnosis. To build a computerised diagnostic system it will be necessary to implicitly and/or explicitly capture diagnostic knowledge. The discussion showed that, by combining this research with future research and development, a computerised diagnostic system for skin cancer could be produced.

11.1 Suggestions for Future Research

The research presented in this thesis could be continued in a number of areas with the goal of automating the diagnosis of skin cancer. This future research was discussed in the previous chapter on the development of computerised skin cancer diagnosis and is summarised here:

- Extension of the lesion location and isolation process, described in chapter 6, to deal with images containing multiple lesions and to use grey-level and colour features to assist in identifying lesions. This extension should also incorporate the automatic measurement of lesion size and compensation for camera set-up and lighting.
- Further development of boundary detection algorithms through the extension of the edge focusing algorithm, presented in chapter 8, and the creation of new methods is required. For the edge focusing algorithm to be usable in a practical system its execution time must be reduced, which may be performed by changing from an FIR edge detector to an IIR detector [19, 85, 95]. Enhancement of the edge focusing algorithm's performance, i.e. reliability and accuracy, may be achievable in two areas: boundary generation and boundary selection. As the algorithm proceeds the detail increases across the whole boundary and hence although the selected boundary may be the best of those available, it may not be the best possible one which could be created. By measuring the quality of each boundary segment as it is created, it may be possible to use a large scale boundary where it is not clearly defined and a small scale boundary where it is better defined. The boundary sharpness information required to implement this may also be diagnostically useful and may be obtained from the edge strength (i.e. the intensity gradient) across the lesion's border. The algorithm selects a boundary using a quality measurement which is critical and application specific. Further work is required to devise quality measurements for this application based on features, such as colour, intensity and texture, which can be applied to the whole and parts of a boundary to enable selection of the most suitable one and generation of an

overall confidence measure. The wide range of skin images may produce difficulties in finding a single method that is reliable and accurate on all images, hence it may be necessary to combine or select the boundaries from different methods using confidence levels. New methods could be developed using texture and techniques such as active contour models [50, 54, 73].

- An additional shape analysis method could be developed to assess lesion asymmetry [89].
- The use of colour directly and indirectly to provide diagnostic information was not considered in this thesis, but is an area of future research. Colour could assist in image segmentation and the identification of variegated colouring, but calibration may be necessary to ensure consistency.
- Development of the synthesis of simulated skin and lesion images could occur in the following areas: enhancement of the synthesised skin and lesion textures, improvement of the lesion shape synthesis, enhancement of the modelling of the transition between skin and lesion, addition of other skin features, synthesis of images containing multiple lesions, addition of non-skin features and modelling of image capture conditions. The nature and probability of occurrence of particular features should be set by studying real images, which would maximise the "realism" of the simulated images. The shape, texture and boundary transition synthesis methods could all be extended to model different types of lesions, i.e. benign and malignant ones. With this development it would be possible to assess the effects of different boundary detectors and shape classifiers on the accuracy of computer classification. This would make it possible to discover whether changes in boundary detection and shape measurement were likely to improve the performance of automatic skin lesion diagnostic systems based on computerised analysis of optical skin images.
- The texture analysis and synthesis method presented in chapter 9 could be extended to allow analysis of irregularly shaped areas and incorporation of scale and colour information. Texture analysis could also be used for segmentation and acquiring diagnostic information.
- After acquiring diagnostically important measurements, diagnosis and prognosis may be obtainable using explicit (e.g. expert systems [14, 23]) and implicit (e.g. rule induction [14] and artificial neural networks [56, 46]) artificial intelligence methods. This research should be guided by the methods used by human experts.

Appendix A

Notation

Variable	Section	Comments
α	9.4.2	Linear autoregressive texture model weight.
σ_{SC}	7.3	Sigma - space/scale constant for LoG convolution.
σ_{SCMin}	8.2	Minimum space constant for LoG edge detection.
σ		Standard deviation.
σ_C	7.5	Scale of the Canny edge detector.
σ_I	9.4.2	Image standard deviation.
σ_N	9.4.2	Noise standard deviation.
μ		Mean.
A		Area, i.e. number of pixels enclosed.
A_I	4.2	Asymmetry index.
A_R	5.3	Aspect ratio.
A_S	5.3	Strip area.
a_1, a_2	7.5.2	Edge strengths of the two neighbouring points in Canny non-maxima suppression.
$a(i)$	5.4	i^{th} parameter of multi-segment line.
B_I	4.3	Border irregularity.
B_L	4.3, 5.3	Bulkiness.
C	4.3	Compactness ratio.
CM_{ij}	5.3	i^{th} moment about the x'-axis and j^{th} moment about the y'-axis of the whole shape.
c	6.3.3	Weighting constant used in unsharp masking.
c_P	7.5.1	Portion of the area under the curve which is included.
$D_E(x_1, y_1, x_2, y_2)$	3.2	Euclidean distance between two points.
dg_W	7.5.1	Number of coefficients in the first derivative of Gaussian filter used in Canny edge detection.
d_L, d_B	5.3	Largest diameter of the equivalent ellipse.
d_S, d_A	5.3	Smallest diameter of the equivalent ellipse.
d_n	9.4.2	Neighbourhood distance for texture analysis and synthesis.
$e_s(x, y)$	7.5.1	Edge strength at x, $y = \sqrt{i_x(x,y)^2 + i_y(x,y)^2}$.
$\mathcal{F}\{f(x)\} = F(u)$	3.10	Fourier transform of $f(x)$.
$\mathcal{F}^{-1}\{F(u)\} = f(x)$	3.10	Inverse Fourier transform of $F(u)$.
f_D	5.4	Maximum Feret's diameter.
G	3.2	Number of possible grey-levels in the image = 256.

Variable	Section	Comments
$g(x)$	7.5.1	Gaussian function.
g_W	7.5.1	Number of coefficients in the Gaussian filter used in Canny edge detection.
$h(i)$	3.13	Histogram, i.e. number of pixels in the image with grey-level i.
$h_1(i)$, $h_2(i)$	7.3	Separated parts of the LoG filter.
$h_C(i)$	6.4.1	Cumulative histogram.
$h_N(i)$	3.13	Normalised histogram.
H, S, I	3.16	Hue, saturation and intensity.
H, L, S	3.16	Hue, lightness and saturation.
H, S, V	3.16	Hue, saturation and value.
$\bar{I}$	6.4.1	Image mean intensity.
I		Whole of the image.
I_1, I_2, I_3	3.16, 4.2	Colour space.
I_A	3.14	Input binary image used in morphology.
I_{BS}	3.14	Binary structuring element image used in morphology.
I_F		Filtered image.
I_{LP}		Low-pass filtered image.
I_{min}	3.2	Minimum intensity of an image.
I_{max}	3.2	Maximum intensity of an image.
I_O	3.14	Output image.
$i_F(x, y)$	3.11	Filtered image at (x, y).
$i_T(x, y)$	9.4.2	Output texture image.
i_k		The k^{th} grey-level.
$i(x, y)$	3.2	Image at x, y.
$i_H(x, y)$	6.4.2	Horizontal gradient at x, y in Kittler thresholding = $\left\|i(x-1, y)-i(x+1, y)\right\|$.
$i_V(x, y)$	6.4.2	Vertical gradient at x, y in Kittler thresholding = $\left\|i(x, y-1)-i(x, y+1)\right\|$.
$i_X(x, y)$	7.5.1	Horizontal gradient at x, y in Canny edge detection.
$i_Y(x, y)$	7.5.1	Vertical gradient at x, y in Canny edge detection.
$i_M(x, y)$	6.4.2	Maximum gradient at x, y = $\max\left\{\left\|i_x(x, y)\right\|, \left\|i_y(x, y)\right\|\right\}$.
$i_{LP}(x, y)$	6.3.3	Low-pass filtered image at x, y.
$i_{UM}(x, y)$	6.3.3	Unsharp masked image at x, y.
k	3.13	Grey-level.
L	3.2	Maximum possible grey-level, $L = G$-1.
M	3.2	Image width.
M_{LoG}	7.3	Width of LoG filter.
M_{ij}	5.3	i^{th} moment about the x-axis and j^{th} moment about the y-axis of the whole shape.
MA_{i0}	5.3	i^{th} moment about the a-axis of the whole shape.
MB_{0j}	5.3	j^{th} moment about the b-axis of the whole shape.
$m_G(i)$	6.4.1	Mean grey-level between 0 and i.
ms_{ij}	5.3	i^{th} moment about the x-axis and j^{th} moment about the y-axis of the current strip.
N	3.2	Image height.
N_n	9.4.2	Number of neighbours for texture analysis and synthesis.
N_S	5.4	Number of parts in multi-segment line.

Variable	Section	Comments
$n_d(x, y)$	9.4.2	Neighbourhood for point $i(x, y)$ for texture analysis and synthesis.
n_p	5.4	Number of data points.
P	4.3	Perimeter, i.e. boundary length.
$P(u)$	3.10	Power spectrum.
R, G, B	3.16	Red, green and blue values.
r, g, b	3.16	Normalised RGB values.
$r(x, y)$	9.4.2	Residual at (x, y).
$rAgy$	5.3	Radius of gyration about the a-axis.
$rBgy$	5.3	Radius of gyration about the b-axis.
s_k	3.13	New grey-level which is assigned to the original k^{th} grey-level.
s_{MG}	6.4.2	Sum of the maximum edge gradients ($i_M(x, y)$).
s_{IG}	6.4.2	Sum of the product of the image ($i(x, y)$) and maximum gradient ($i_M(x, y)$).
T_0	6.4.1, 6.4.2	Threshold.
t	8.4.1.2	Parameter for parametric splines.
w	7.3	Sensitive region width of LoG filter.
w_i	3.11	Filter coefficients.
X	4.2	X-transform.
x, y	3.2	Image co-ordinates, usually from 0 to N-1 and 0 to M-1

Symbol	Comments
$\lfloor x \rfloor$	Floor - the largest integer less than x.
$\lceil x \rceil$	Ceiling - the smallest integer greater than x.
round(x)	Round to x to the nearest integer.
$\|x\|$	Absolute value of x.
$\hat{I}$	Reflection of I about the origin.
$\hat{x}$	Estimate of x.
$(I)_x$	Translation of I by $x = (x_1, x_2)$.
erf(x)	Error function.
erfc(x)	Complementary error function. erfc(x)=1 - erf(x).
erfinv(x)	Inverse error function. x = erfinv(y) satisfies y = erf(x) for $-1 \le y \le 1$, $-\infty \le x \le \infty$.
x^T	Transpose of vector or matrix x.

Appendix B

Bulkiness Measurement

The calculations used in bulkiness measurement (section 5.3) are described in this appendix. For axes names see Figure 5-1. The following steps are applied to each strip in turn, where x_{lo} and x_{hi} are the strip's start and end x co-ordinates:

Strip area: $A_S = x_{hi} - x_{lo}$ (B.1)

Strip's moment[1]: about the x-axis: $ms_{10} = A_S\, y$ (B.2)

This simplification can be used as the strip has unit thickness.

Strip's moment about the y-axis: $ms_{01} = \dfrac{{x_{hi}}^2 - {x_{lo}}^2 + x_{hi} + x_{lo}}{2}$ (B.3)

The strip is considered to be composed of a series of squares of unit area. The discrete summation from x_{lo} to x_{hi} inclusive of the distance of the each square from the y-axis gives the strip's moment about the y-axis.

Strip's moment of inertia about the x-axis: $ms_{20} = A_S\, y^2$ (B.4)

Strip's moment of inertia about the y-axis:

$$ms_{02} = \frac{x_{hi}(x_{hi}+1)(2x_{hi}+1) - x_{lo}(x_{lo}+1)(2x_{lo}+1)}{6} \quad \text{(B.5)}$$

Strip's product moment: $ms_{11} = ms_{10}\, y$ (B.6)

As each strip is processed the following can be accumulated:

Shape's area: $A = \sum_{\text{for all strips}} A_S$ (B.7)

Shape's moment about the x-axis: $M_{10} = \sum_{\text{for all strips}} ms_{10}$ (B.8)

---

[1] The notation used for moments is:
$ms_{ij} = i^{\text{th}}$ moment about the x-axis and the j^{th} moment about the y-axis of the current strip.
$M_{ij} = i^{\text{th}}$ moment about the x-axis and the j^{th} moment about the y-axis of the whole shape.
$CM_{ij} = i^{\text{th}}$ moment about the x'-axis and the j^{th} moment about the y'-axis of the whole shape.
$MA_{i0} = i^{\text{th}}$ moment about the a-axis of the whole shape.
$MB_{0j} = j^{\text{th}}$ moment about the b-axis of the whole shape.

Shape's moment about the y-axis: $M_{01} = \sum_{\text{for all strtips}} ms_{01}$ (B.9)

Shape's moment of inertia about the x-axis: $M_{20} = \sum_{\text{for all strips}} ms_{20}$ (B.10)

Shape's moment of inertia about the y-axis: $M_{02} = \sum_{\text{for all strips}} ms_{02}$ (B.11)

Shape's product moment: $M_{11} = \sum_{\text{for all strips}} ms_{11}$ (B.12)

Once all the strips have been processed the centroid (x_C, y_C) can be found:

$$x_C = \frac{M_{01}}{A}, \quad y_C = \frac{M_{10}}{A} \qquad \text{(B.13), (B.14)}$$

The moments can be shifted to be about the centroid by the parallel axis theorem [90]:

Moment of inertia about the x'-axis parallel to the x-axis through the centroid:
$CM_{20} = M_{20} - y_C M_{10}$ (B.15)

Moment of inertia about the y'-axis parallel to the y-axis through the centroid:
$CM_{02} = M_{02} - x_C M_{10}$ (B.16)

Product moment about x' and y' axes: $CM_{11} = M_{11} - x_C M_{10}$ (B.17)

From these the moment of inertia about the a and b axes can be obtained:

Moment of inertia about the a-axis: $MA_{20} = c\overline{m}_2 + t$ (B.18)

Moment of inertia about the b-axis: $MB_{02} = c\overline{m}_2 - t$ (B.19)

where:

average central moment of inertia: $c\overline{m}_2 = \frac{CM_{20} + CM_{02}}{2}$

$t = \sqrt{c\overline{m}_2{}^2 - CM_{20} * CM_{02} + CM_{11}}$

These allow the calculation of the shape's radii of gyration, which are the same as those of the equivalent ellipse:

Radius of gyration about the a-axis: $rA_{gy} = \sqrt{\frac{MA_{20}}{A}}$ (B.20)

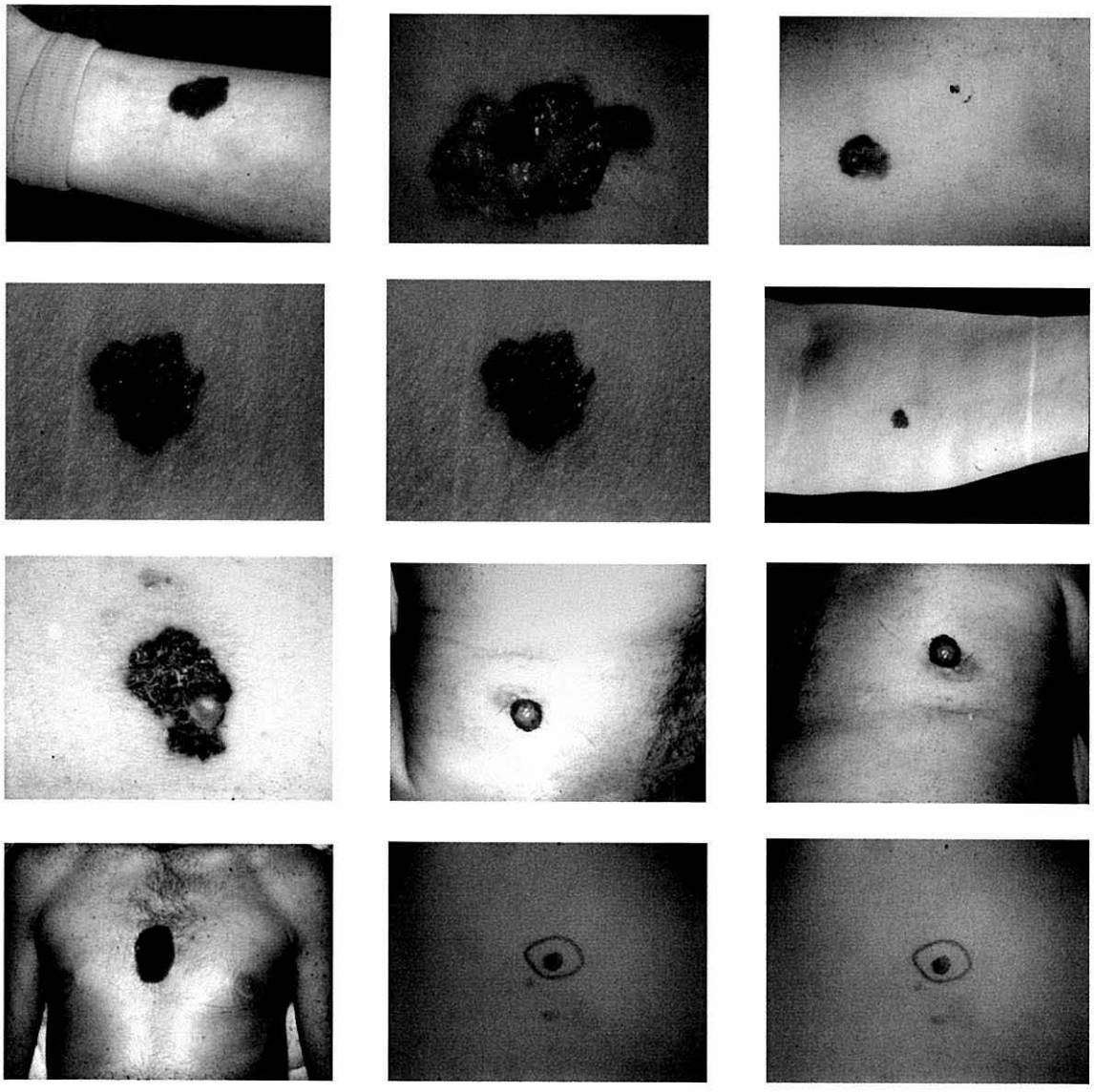
Radius of gyration about the b-axis: $rB_{gy} = \sqrt{\frac{MB_{20}}{A}}$ (B.21)

Appendix C

Skin Lesion Location and Isolation

Test Images

This appendix shows the test images used in developing the lesion locating and isolation process presented in chapter 6.



Appendix D

Publications

The following refereed publications, which are reproduced here, have resulted from this work.

W. E. Denton, A. W. G. Duller, and P. J. Fish. Boundary Detection for Skin Lesions: An Edge Focusing Algorithm. In Proc. 5th Int. Conf. on Image Processing and its Applications, IEE, Edinburgh, 1995.

W. E. Denton, A. W. G. Duller, and P. J. Fish. Robust Boundary Detection for Skin Lesions. In IEEE Eng. in Medicine & Biology 17th Annual Conference, IEEE, Montréal, 20-23 Sept 1995.

W. E. Denton, A. W. G. Duller, and P. J. Fish. Synthesis of skin images for image processing validation. In Proc. of 9th Int. Conf. on Biomedical Engineering, pages 445-447, 1997.

BOUNDARY DETECTION FOR SKIN LESIONS: AN EDGE FOCUSING ALGORITHM

W E Denton, A W G Duller and P J Fish

University of Wales, Bangor, UK

ABSTRACT

The accurate location of the boundary of skin lesions is an important first step in the automatic diagnosis of malignant melanoma. The use of standard edge detectors for skin lesion boundary detection has serious shortcomings since those giving sufficiently detailed borders are sensitive to spurious small scale structure elsewhere in the image while those which ignore small scale structure involve low-pass filtering which loses border detail. Edge focusing uses edge detectors of progressively smaller scale, each focusing only in the region of the boundary estimated by the previous detector. This results in a series of boundary position estimates of increasingly finer scale while eliminating spurious edges resulting from noise and other image detail. The selection of the "best fit" boundary is made by comparison of the contrast between narrow regions just inside and outside the boundary or from the variance of image intensity in the latter region.

INTRODUCTION

Accurate and reliable outline detection is important in the automated diagnosis of skin lesions, in order to segment the image into lesion, skin and other background, thereby ensuring that colour and texture measurements are carried out only on the lesion image. In addition, given an accurate outline, the important diagnostic factor of lesion shape, Mackie (1), can be analysed to provide quantitative measurements of size, asymmetry, Stoecker et al (2), and border irregularity, Golston et al (3) and Claridge et al (4). The characteristics of skin images are very variable (e.g. lighting source, lesion size and nature, skin texture, hair, background objects) and this produces problems in obtaining methods which are reliable, repeatable and robust. In many cases the position of the "best" boundary is very subjective. However, an objective and consistent definition of the boundary location, which can be reasonably implemented, is required for an image processing system.

A number of methods have been proposed to solve the lesion boundary detection problem. However, none of the methods have proved sufficiently reliable on a sufficiently wide range of images. A radial search, Golston et al (6), from the centre of the lesion, for a sustained change in image brightness has been attempted for boundary determination. This method can be inaccurate if the transition between lesion and background is not reasonably sharp. It assumes that no radius intersects the border more than once (radial connectedness) and may not repeatable in that a different border may be found if the image is rotated. Thresholding, preceded by filtering and a colour transform, followed by region growing and contour tracing and smoothing has been used by Ercal et al (5). In this method, the colour transform and threshold level is based on the colour of two windows which have been identified to be inside and outside the lesion. These windows are found by histogramming and approximate colour segmentation. It is applicable to some classes of lesions, but not all. Fixed size Laplacian-of-Gaussian edge detection, with sub-pixel interpolation, has been used by Perednia (7) for boundary detection, but suffers from the problems described below.

General filter based edge detectors such as the Sobel operator or the Laplacian-of-Gaussian operator, Huertas and Medioni (8), with fixed sizes are unreliable because of the nature of the image. Small scale edge detectors provide large numbers of unwanted edges by detecting image features such as skin texture and hair. Even when a complete boundary can be found there are a large number of other edges present which makes it difficult to identify the genuine border. If the image is low-pass filtered or a large scale edge detector used the resulting boundary is less affected by the unwanted image features, but the locational accuracy is reduced as a result of inadequate resolution.

A method that finds the border accurately but is tolerant of image noise is required and this can be achieved through the use of "edge focusing". In this method the final edges consist of refined versions of those detected by a large scale edge detector but have the locational accuracy of those produced by a small scale edge detector.

LESION ISOLATION

Before "edge focusing" can be performed the approximate lesion location and size are required to set the initial Laplacian-of-Gaussian space constant and

allow processing of only the relevant part of the image. This is achieved through bi-level thresholding, Kittler et al (9), (after median filtering using a 7x7 window) followed by binary morphological closing, using a 10 pixel diameter disc, (to smooth region outlines, eliminate small holes in regions, fuse short gaps between regions) and region identification. Each region in the thresholded image is characterised by its shape (bulkiness, Medalia (10), and aspect ratio) and size and these factors are used to determine which is likely to be a lesion. The minimum diameter of the equivalent ellipse (from the bulkiness calculation) is used to set the initial Laplacian-of-Gaussian space constant (σ) to just detect the lesion (minimum diameter = two times the width of the sensitive region of the filter, i.e. $2 * 2\sqrt{2}\sigma$). The location of a rectangle which surrounds the lesion and allows for the uncertainty in the edge location is calculated from the space constant and the location of the thresholded region. This rectangle is used to produce an image which contains the lesion and sufficient surrounding image (half the filter width ($3 * \sqrt{2}\sigma$) around the rectangle) to allow the initial edge detection, to be performed without requiring data from outside the image. Figure 1(a) shows the region selected in an example image.

EDGE FOCUSING

To find the initial boundary a large scale Laplacian-of-Gaussian edge detection is performed on the clipped image within the rectangle which is assumed to contain the lesion. The lesion boundary is then selected from the edges detected by the method described below. The basic edge focusing step is to gradually refine the boundary by decreasing the space constant and at each step detecting new edges only adjacent to or on the existing boundary, Bergholm (11). The reduction in the space constant is 0.5 which ensures that the boundary moves, at most, one pixel. This enables refinement of the existing edge, whilst preventing detection of new unwanted edges. The edges produced by this process do not necessarily form a closed contour and a further "cleaning" stage is required after each edge refinement. The initial boundary selection and subsequent "cleaning" are performed by joining adjacent edges (using a cubic spline), selecting the largest closed contour and removing any redundant pixels (by drawing and re-tracing the boundary) to give an 8-way connected boundary.

Thus a progressively better estimate of the edge is obtained as the edge detection scale is reduced. This process continues to a minimum space constant, which corresponds to a filter sensitive region width of 8 pixels. This size is chosen to prevent the edge focusing detecting small scale skin texture.

With the type of unconstrained images that are being considered, problems are encountered due to parts of the lesion producing "highlights". These are generally caused by lighting which is non-diffuse falling on smooth areas of the lesion. With the algorithm as described these highlights are often confused with skin if they are "close" to the boundary since they generally have an intensity which is near that of the skin. To make the edge focusing robust to such problems it was found necessary to choose the "best" boundary which in some cases was not that produced using the minimum space constant.

Two approaches have been developed for this which both rely on examining the image near the boundary. The first selects the boundary which has the greatest contrast between the average intensity just inside and just outside the border, corresponding to a minimum in the intensity ratio because the lesion is darker than the surrounding skin. The graph of intensity for the lesion in figure 1(a) is shown in 1(c) and the selected boundary is shown in 1(b). The second method examines the variation of the image just outside the border by calculating the standard deviation of the intensity in that area. The "best" boundary is then chosen on the assumption that the skin has a lower standard deviation than mixed skin/lesion. The width of the area is based on the size of the initial edge detector, so it scales with the lesion size.

Figure 2 shows some of the generated boundaries for an example image and the following summarises the method:

1. Median filtering - to reduce image noise, but preserve the lesion border
2. Thresholding
3. Morphological closing - to clean the image
4. Region identification by size and shape - to find the lesion
5. Lesion location indication
6. Initial space constant calculation - from the approximate lesion diameter
7. Image cropping - to include the lesion plus sufficient surrounding image
8. Initial edge detection
9. Edge cleaning - to select initial approximate boundary
10. While space constant ≥ minimum space constant do steps 11 and 12
11. Edge detection on and adjacent to existing boundary
12. Boundary cleaning - to join gaps in the boundary and remove unneeded edge points
13. Selection of "best" boundary

DISCUSSION

The proposed method produces visually accurate continuous boundaries for a range of images and does not make assumptions, such as radial connectedness, about the lesion shape. It is tolerant of lesion size and the presence of other unwanted objects in the initial image. The initial selection of the area which contains the lesion enables it to be used on images which are not centred on the lesion and contain other objects or multiple lesions. Accuracy is obtained by finding the final boundary using a small scale edge detector while robustness is acquired through starting the edge focusing on a large scale. Without the boundary "cleaning" good results can be obtained on lesions with clearly defined boundaries. However, the "cleaning" is required to give the algorithm reliability on a wide range of images.

Currently, a Laplacian-of-Gaussian, which is a second order derivative edge detector, is used but first order derivative edge detectors, such as Canny (12) or those based on IIR filters (for example Sarkar and Boyer (13)), could be used in a similar algorithm. The first derivative edge detectors may be more sensitive to lighting changes, but a considerable time saving would be obtained by using an IIR filter edge detector and the edge strength, which is easily obtained from a first order derivative edge detector, may be useful in evaluating the border quality.

The selection of the most suitable boundary is based on boundary quality measurements, which are critical and application specific. Further work is required to devise quality measurements for this application based on features, such as colour, intensity and texture, which can be applied to parts of the boundary and to the whole boundary to enable selection of the most suitable boundary and generation of an overall confidence measure.

REFERENCES

1. Mackie R M, 1986, "An illustrated guide to the recognition of early malignant melanoma", Department of Dermatology, University of Glasgow

2. Stoecker W V, Li W W and Moss R H, 1992, "Automatic Detection of Asymmetry in Skin Tumors", Computerized Medical Imaging and Graphics, 16(3), 191-197

3. Golston J E and Stoecker W V and Moss R H and Dhillon I P S, 1992, "Automatic Detection of Irregular Borders in Melanoma and Other Skin Tumors", 16(3), Computerized Medical Imaging and Graphics, 199-203

4. Claridge E and Hall P N and Keefe M and Allen J P, 1992, "Shape analysis for classification of malignant melanoma", J. Biomed. Eng., 14, 229-234,

5. Ercal F and Moganti M and Stoecker W V and Moss R H, 1993, "Detection of Skin Tumor Boundaries in Color Images", IEEE Trans. on Medical Imaging, 12(3), 624-627,

6. Golston J E and Moss R H and Stoecker W V, 1990, "Boundary detection in skin tumor images: An overall approach and a radial search algorithm", Pattern Recognition, 23(11), 1235-1247,

7. Perednia D A and White R G and Schowengerdt R A, 1989, "Localization of Cutaneous Lesions in Digital Images", Computers and Biomedical Research, 22, 374-392,

8. Huertas A and Medioni G, 1986, "Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks", PAMI, 8(5), 651-664,

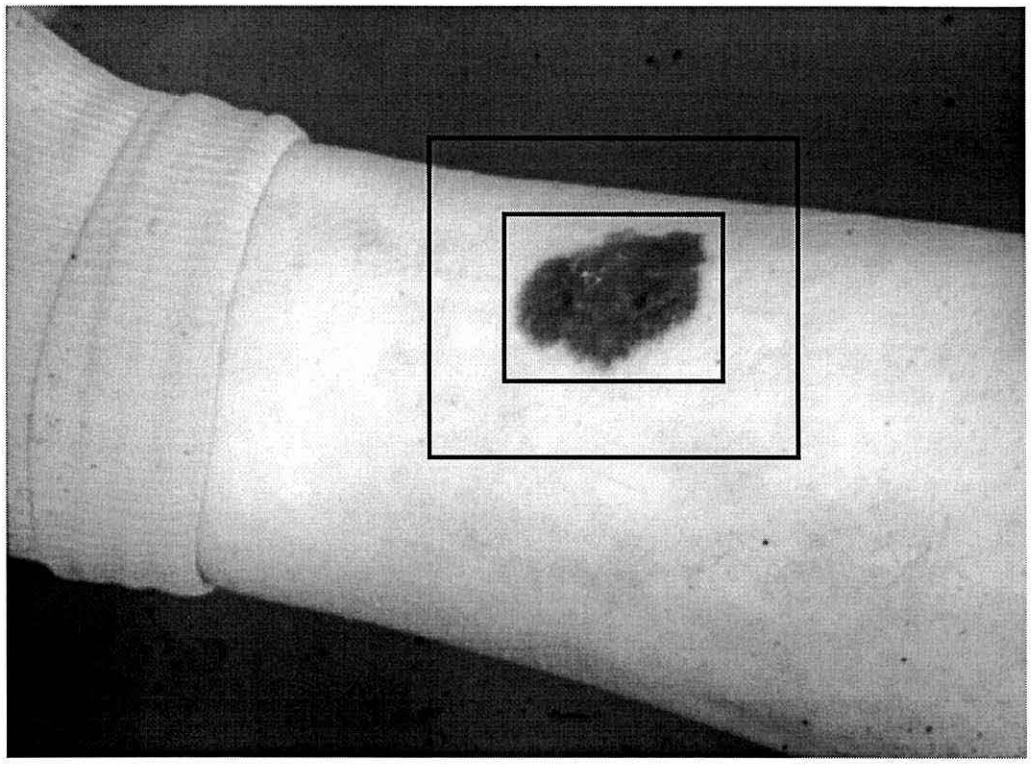
9. Kittler J and Illingworth J and Foglein J and Paler K, 1984, "An Automatic Thresholding Algorithm and its Performance", IEEE Proc. Seventh Int. Conf. Pattern Recognition, 1, 287-289

10. Medalia A I, 1970, "Dynamic Shape Factors of Particles", Powder Technology, 4, 117-138

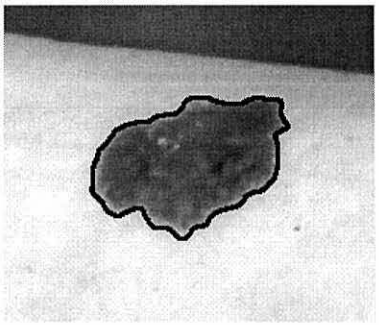
11. Bergholm F, 1987, "Edge Focusing", PAMI, 9(6) 726-741,

12. Canny J, 1986, "A Computational Approach to Edge Detection", PAMI, 8(6), 679-698,

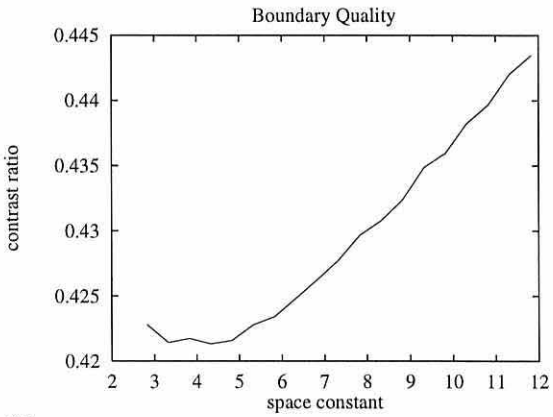
13. Sarkar S and Boyer K L, 1991, "On Optimal Infinite Impulse Response Edge Detection Filters", PAMI, 13(11), 1154-1171,



(a)

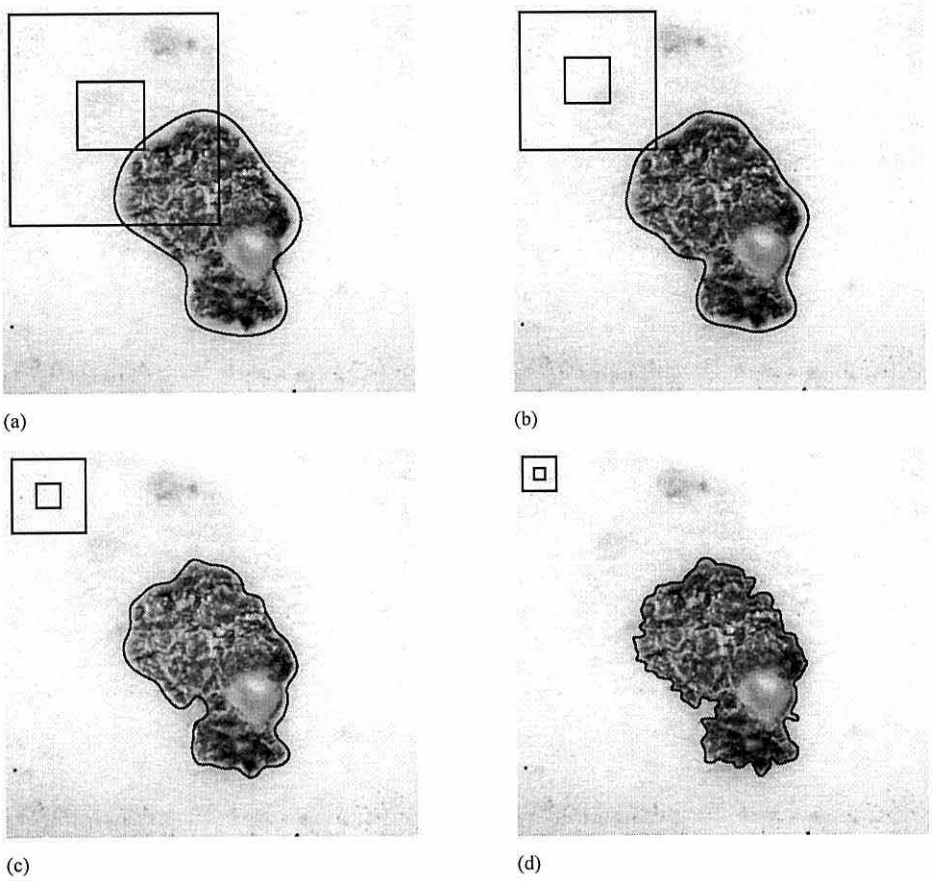


(b)



(c)

Figure 1: (a) Example image showing the area selected for processing. Outer rectangle is complete area being processed. Inner rectangle is the area in which the initial edge detection is performed. (b) Clipped image showing the final selected boundary. (c) Graph showing the contrast ratio between the area just inside and outside the lesion. The area is 50% of the sensitive region of the starting edge detector, which in this case is 16 pixels either side of the border. Note: Lesion boundaries and rectangles are drawn three pixels wide.



(a) (b)

(c) (d)

Figure 2: Edge focusing on an example image.
(a) Initial boundary detected.
(b) Boundary at about one two thirds of the initial scale.
(c) Boundary at about one third of the initial scale.
(d) Final selected boundary.
The squares show the size of the edge detection filter: outer - area of the image used for each edge point, inner - sensitive region of the filter. The filter widths and sensitive region widths are calculated from $6*\sqrt{2}\sigma$ and $2*\sqrt{2}\sigma$ respectively. Note: Lesion boundaries and filter widths are drawn 3 pixels wide.

ROBUST BOUNDARY DETECTION FOR SKIN LESIONS

W E Denton, A W G Duller and P J Fish

School of Electronic Engineering and Computer Systems, University of Wales,
Dean Street, Bangor, GWYNEDD LL57 1UT, UK

***Abstract* - Standard edge detectors for skin lesion boundary detection have serious shortcomings. Those giving sufficiently detailed borders are sensitive to image noise including spurious small scale structure and those tolerant of noise involve low-pass filtering which loses border detail. Edge focusing allows accurate lesion boundary detection whilst being insensitive to image noise.**

***Keywords* - skin lesions, boundary detection, edge focusing**

I. INTRODUCTION

Accurate and reliable outline detection is important in the automated diagnosis of skin lesions, in order to segment the image into lesion, skin and other background, thereby ensuring that colour and texture measurements are carried out only on the lesion image. In addition, given an accurate outline, the important diagnostic factor of lesion shape [1] can be analysed to provide quantitative measurements of size, asymmetry [2], and border irregularity, [3] and [4]. The characteristics of skin images are very variable (e.g. lighting source, lesion size and nature, skin texture, hair, background objects) and this produces problems in obtaining methods which are reliable, repeatable and robust.

General filter based edge detectors, for example the Laplacian-of-Gaussian (LoG) operator [5], with fixed sizes are unreliable because of the nature of the image. Small scale edge detectors provide large numbers of unwanted edges by detecting image features such as skin texture and hair. Even when a complete boundary can be found there are a large number of other edges present which makes it difficult to identify the genuine border. If the image is low-pass filtered or a large scale edge detector used the resulting boundary is less affected by the unwanted image features, but the locational accuracy is reduced as a result of inadequate resolution. A method that finds the border accurately but is tolerant of image noise is required and this can be achieved through the use of "edge focusing". In this method the final edges consist of refined versions of those detected by a large scale edge detector but have the locational accuracy of those produced by a small scale edge detector.

II. METHOD

The following summarises the method:

1. Median filtering (7x7 pixel window) - to reduce image noise, but preserve the lesion border.
2. Bi-level thresholding [6].
3. Morphological closing, using a 10 pixel diameter disc - to clean the image by smoothing region outlines, eliminating small holes in regions and fusing short gaps between regions.
4. Region identification by size and shape - each region in the thresholded image is characterised by its shape (bulkiness [7] and aspect ratio) and size and these factors are used to determine which is likely to be a lesion.
5. Initial LoG space constant (σ) calculation - set to just detect the lesion - $\sqrt{2}/4$ times the minimum diameter of the equivalent ellipse (from the bulkiness calculation).
6. Lesion location indication - the location of a rectangle which surrounds the lesion and allows for uncertainty in the edge location is calculated from the space constant and the location of the thresholded region. This rectangle is used to produce an image which contains the lesion and sufficient surrounding image to allow the initial edge detection to be performed without requiring data from outside the image.
7. Image cropping - to include the lesion plus sufficient surrounding image.
8. Initial boundary detection - by performing LoG edge detection using the initial value of σ from step 5.
9. Edge cleaning - to select initial approximate boundary. The initial boundary selection and subsequent "cleaning" are performed by joining adjacent edges (using a cubic spline), selecting the largest closed contour and removing any redundant pixels (by drawing and re-tracing the boundary) to give an 8-way connected boundary.
10. While space constant $\geq$ minimum space constant do steps 11 to 13. The minimum space constant corresponds to a filter sensitive region width ($2\sqrt{2}\sigma$) of 8 pixels. This size is chosen to prevent the edge focusing detecting small scale skin texture.
11. Decrease space constant by 0.5 - which ensures that the boundary moves, at most, one pixel [8]. This enables refinement of the existing edge, whilst preventing detection of new unwanted edges.
12. LoG edge detection on and adjacent to existing boundary.
13. Boundary cleaning - to join gaps in the boundary and remove unneeded edge points (see step 9).
14. Selection of "best" boundary - This relies on examining the image near the boundary. The boundary which has the

greatest contrast between the average intensity just inside and just outside the border is selected, corresponding to a minimum in the intensity ratio because the lesion is darker than the surrounding skin. The width of the area is based on the size of the initial edge detector and scales with lesion size.

Figure 1(a) shows a "raw" image with lesion. The final image boundary and graph of inside/outside boundary border intensity ratio are shown in figures 1(b) and 1(c) respectively. Figure 2 shows the estimated boundary at the start, during and at the end of the edge focusing process on a second lesion.

III. Results and Discussion

The proposed method produces visually accurate continuous boundaries for a range of images. It is tolerant of lesion size and the presence of other unwanted objects in the initial image. The initial selection of the area which contains the lesion enables it to be used on images which are not centred on the lesion and contain other objects or multiple lesions.

References

[1] R M Mackie, "An illustrated guide to the recognition of early malignant melanoma," Department of Dermatology, University of Glasgow, 1986

[2] W V Stoecker, W W Li and R H Moss, "Automatic Detection of Asymmetry in Skin Tumors," *Computerized Medical Imaging and Graphics*, vol. 16, no. 3, pp. 191-197, 1992

[3] J E Golston, W V Stoecker, R H Moss and I P S Dhillon, "Automatic Detection of Irregular Borders in Melanoma and Other Skin Tumors," vol. 16, no. 3, *Computerized Medical Imaging and Graphics*, pp. 199-203, 1992

[4] E Claridge, P N Hall, M Keefe and J P Allen, "Shape analysis for classification of malignant melanoma," *J. Biomed. Eng.*, vol. 14, pp. 229-234, 1992

[5] A Huertas and G Medioni, "Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks," *PAMI*, vol. 8, no. 5, pp. 651-664, 1986

[6] J Kittler, J Illingworth, J Foglein and K Paler, "An Automatic Thresholding Algorithm and its Performance," *IEEE Proc. Seventh Int. Conf. Pattern Recognition*, vol. 1, pp. 287-289, 1984

[7] A I Medalia, "Dynamic Shape Factors of Particles," *Powder Technology*, vol. 4, pp. 117-138, 1970

[8] F Bergholm, "Edge Focusing," *PAMI*, vol. 9, no. 6, pp. 726-741, 1987

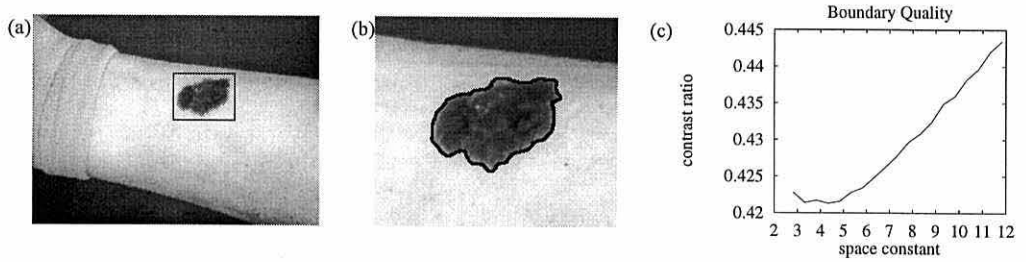


Fig. 1: (a) "Raw" image with lesion containing rectangle. (b) Cropped image with final boundary. (c) Intensity ratio versus filter space constant.

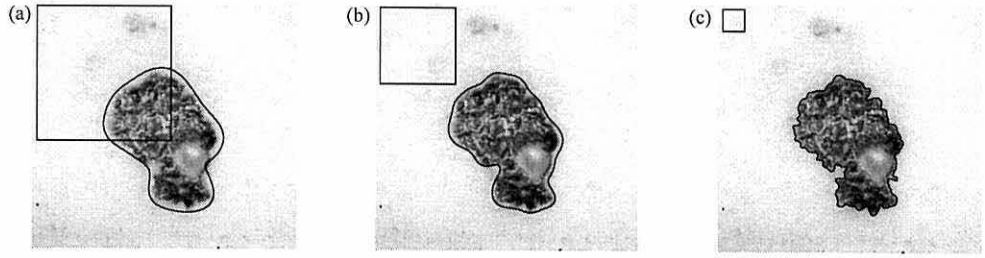


Fig. 2: Image with lesion boundary and filter size ($6 * \sqrt{2}\sigma$ square). (a) Initial boundary (b) Intermediate (c) Final.

SYNTHESIS OF SKIN IMAGES FOR IMAGE PROCESSING VALIDATION

W. E. Denton, A. W. G. Duller and P. J. Fish
School of Electronic Engineering & Computer Systems,
University of Wales, Bangor,
Dean St., Bangor, Gwynedd, LL57 1UT, UK.

Abstract

Our work on the development of a diagnostic aid for skin cancer has involved work in a number of areas including, accurate lesion border detection, lesion shape analysis, skin line texture measurements and colour measurements. A major requirement of these sub-systems is the ability to validate their performance against a set of images which have known characteristics. This is not possible using images of real skin lesions since their true characteristics are insufficiently well defined. This paper discusses the development of methods for generating synthetic images of both skin and lesions which can have tightly controlled characteristics particularly in terms of being a known size and having a known boundary position.

Two methods for modelling skin/lesion texture were investigated, a 2D power spectrum model and a linear autoregressive model. Examples showed that the power spectrum method, while being computationally attractive, did not produce textures which were sufficiently visually similar to real skin - leading to the selection of the autoregressive method.

Synthesised images have been used to compare an edge focusing boundary detector using Canny and Laplacian of Gaussian edge detectors which are both shown to have advantages depending on the type of lesion being analysed; Canny being preferable for small and ill-defined lesions.

INTRODUCTION

This paper discusses the synthesis of simulated monochrome skin and lesion images, which are required to investigate the characteristics of lesion segmentation algorithms which have been developed in previous work [1][2]. Test images are required where the characteristics can be controlled, so that the output of a boundary finding algorithm may be compared with the known input. This comparison allows assessment of algorithm performance under a range of image conditions, which can be created by adjusting the parameters of the image synthesis process. In addition, test images can assist in algorithm development by identifying image conditions under which an algorithm succeeds or fails.

PREVIOUS WORK

Dhawan and Sicsu [3] synthesised simulated skin and lesion images, which they used to visually assess their segmentation method. They did not specify how these images were created, but their illustrations showed images containing overlapping ellipses of varying colours and textures. These images appear not to have been created from measurements of real lesion images and had clear transitions from skin to lesion.

LESION SYNTHESIS

The new image synthesis method presented is performed in three stages:

- Shape generation: the creation of a shape similar to that of a lesion.
- Boundary transition: the modelling of the transition between skin and lesion, which will control the mixing of skin and lesion textures.
- Texture generation: the synthesis of textures which are similar to skin and lesion from measurements of real skin and lesion textures.

Shape generation

Shape generation consists of producing a binary image which contains a lesion-like shape together with the shape's boundary co-ordinates. The method creates a simple shape based on an ellipse with random large and small scale irregularities, using parameters which are heuristically chosen to create shapes similar to lesion outlines. The primary shape is an ellipse, with a random size, aspect ratio and angle placed at a random position in the image. The ellipse's minor axis, aspect ratio and centroid are controlled by Gaussian random numbers giving a range of lesion sizes between 15 and 170 pixels. To create large scale irregularity the ellipse is sampled, a fixed number of times (8), at uniformly spaced angles starting at a random angle. These sampled points are displaced radially by a random distance (with zero mean and standard deviation of 30 pixels) and re-connected to create a shape with large scale irregularities. Finally, small scale irregularity is created by re-sampling more frequently (16 times) at uniform points in the co-ordinate list. These points are radially displaced and interpolated to give the final shape (see figure 1).

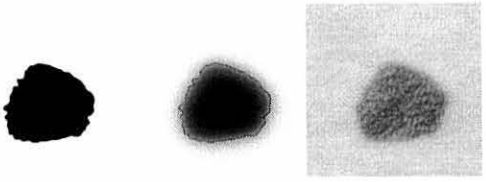


Fig. 1. Typical synthesised lesion shape mask (left), with boundary transition (centre) and final image (right).

Boundary Transition

A boundary transition between the lesion and the surrounding skin is simulated by modifying the shape image which controls the intensity mixing of skin and lesion texture. A first order low-pass butterworth filter is used to blur the synthesised lesion shape image. The cut-off frequencies are chosen to create boundary transitions visually similar to those in real images. An example transition is shown in figure 1.

Texture generation

The most difficult part of creating synthesised lesion images is producing skin and lesion textures which are visually similar to real skin and lesion textures. To maximise the realism of the synthesised images only bi-directional methods, i.e. those which can analyse a texture and synthesise a new similar texture, were considered. The performance of the texture synthesis was assessed visually by comparison with the original texture.

Iversen and Lonnestad [4] surveyed a range of texture analysis/synthesis methods for use in image compression. They assessed the visual performance, the size of the parameter space and analysis/synthesis times of the following methods:

- Markov models: the texture is modelled by a stationary first order Markov chain.
- Autocorrelation and histogram models: the texture model contains the image histogram and the autocorrelation function.
- Linear autoregressive models: each texture pixel is modelled by the sum of a linear combination of the intensities of points in its neighbourhood and Gaussian noise.
- Fractal models: the texture's irregularity is modelled by its fractal dimension.
- Spectral models: the model is the same as the linear autoregressive model, except the neighbourhood and weights estimation are different.

The Markov models were found to have unsatisfactory visual performance and large parameter spaces, but with small analysis and synthesis times. The performance of the autocorrelation method was better and it uses a small parameter space, but has a time consuming synthesis algorithm. The linear autoregressive model was judged to give the best visual performance with a reasonably sized parameter space. The only problem (which is not relevant in this application) with this method is the very slow analysis algorithm. The performance of the fractal model was very poor, as expected since it had only one parameter. The spectral model they investigated had a small parameter space, but did not perform visually as well as the linear autoregressive model.

Linear Autoregressive Texture Model

In this method the intensity of a texture point is estimated from the sum of a linear combination of the intensities of points in its neighbourhood, represented by the weights and Gaussian noise. The model used here is a modification of the method described in [4], with the addition of the measurement of image mean and standard deviation. The scale of texture analysed/synthesised is controlled by the size of a causal neighbourhood. The optimum neighbourhood distance was determined visually and was chosen to be 16 pixels for both skin and lesion textures.

Comparison of True and Estimated Boundaries

Once a simulated lesion has been created the true and estimated boundaries can be compared to measure the performance of the boundary detection method. The method used to compare these boundaries is based on the area which they enclose since the true and estimated boundaries are unlikely to have the same number of points, hence a direct comparison between the corresponding points on each boundary is not possible. Three areas are used, the area which they have in common (true positive region), the area which is covered by the true boundary and not by the estimated boundary (false negative region) and the area which is covered by the estimated boundary and not by the true boundary (false positive region). To make these independent of lesion size they are expressed as a percentage of the true lesion area. Examples are shown in figure 2.

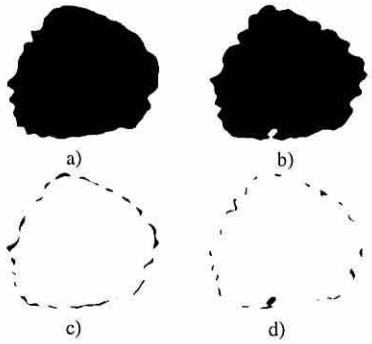


Fig. 2. A comparison of the true and estimated boundaries for a simulated lesion, a) true lesion, b) true positive, c) false negative, d) false positive

Results

The edge focusing algorithm described in [1] is used to demonstrate the use of synthesised lesion images. Two forms of edge detection are compared within the edge focusing framework, namely, Canny and Laplacian of Gaussian (LoG) based edge detectors. Figure 3

shows a number of simulated lesions together with the true boundary and those calculated by the LoG and Canny edge focusers. The three examples show increasingly diffuse boundaries, the blurring filter cut-off frequencies being respectively, 0.0156, 0.0078, 0.0039 (top to bottom). The Canny result (bottom right) is something of an anomaly as the boundary chosen by the edge focusing algorithm was the very first of the sequence of boundaries produced by the algorithm. In contrast the LoG result for the bottom lesion is too highly detailed. Both of these results highlight problems associated with the boundary selection procedure within the edge focusing algorithm.

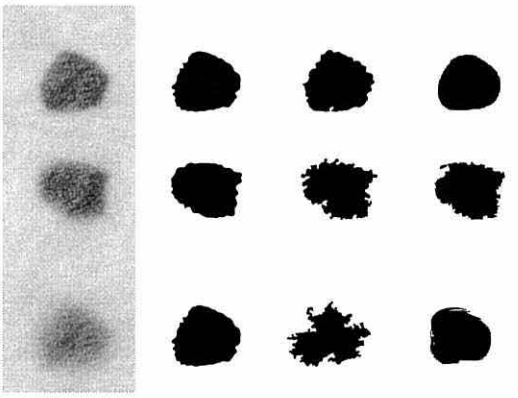


Fig. 3. Examples of simulated lesions: from left to right, lesion, true boundary, LoG boundary, Canny boundary

In the synthesised images, there are three main factors that can be independently controlled, the lesion shape, the boundary transition and the textures of the skin and lesion. The tests reported in this paper were performed on a set of 120 synthesised images made up from 10 lesion shapes, 4 boundary transitions and 3 skin/lesion textures. All of the textures used were synthesised using linear autoregressive model. A summary of the boundary comparison results is presented in table 1. As expected the true positive area reduces as the boundary blurring increases. For the LoG, the false-negative area increases more rapidly than the false-positive, for Canny the converse is true. This is to be expected as for curved edges the LoG edge detector places its edge on the higher intensity side (i.e. skin) and the Canny edge detector places its edge on the lower intensity side (i.e. lesion). The average performance of the LoG and Canny edge focusers are statistically similar for the true-positive and false-positive area measurements. However, Canny performed statistically significantly better in the false-negative area which was mainly caused by errors on small lesion shapes. In particular, one shape caused most of the poor performance for the false-negative area, for the LoG edge focuser. This shape only occupied about one fifth of the image width and height, causing it to be blended into the white background with the smoother boundary transitions. Hence, this image was not particularly realistic and once removed the false-negative area had a mean of 13.2% and standard deviation of 13.3%. This performance is still statistically worse than the Canny edge focuser, but is much closer and has a much smaller standard deviation.

Low pass cut-off freq.	% True pos. area		% False neg. area		% False pos. area	
	mean	SD	mean	SD	mean	SD
	Laplacian of Gaussian edge detection					
none	99.6	0.7	1.8	1.1	0.4	0.7
0.0156	94.1	3.1	8.0	5.2	5.9	3.1
0.0078	89.2	4.0	18.0	11.5	10.8	4.0
0.0039	83.0	9.0	52.4	94.2	17.0	9.0
	Canny edge detection					
none	99.8	0.2	1.5	0.8	0.2	0.2
0.0156	92.8	4.0	3.2	2.0	7.2	4.0
0.0078	88.7	5.0	5.5	3.7	11.3	5.0
0.0039	80.8	10.5	8.0	7.8	19.2	10.5

Table 1 Edge focusing results

CONCLUSIONS

This paper has described the synthesis of simulated skin and lesion images and the comparison of a true boundary with an estimated boundary. This allowed the comparison of two edge focusing algorithms. Two basic methods of texture synthesis were investigated using visual comparison of the original and synthesised textures. The first method using an image's power spectrum produced textures which were not visually similar to the original textures. A number of variants of the linear autoregressive texture synthesis were developed and tested to select the most suitable version for this application. A boundary comparison method based on area was used to compare the Canny and LoG version of the edge focuser. The LoG edge focuser performed slightly better for better defined lesion while performance was considerably better for false-negative areas for small lesions with the Canny based detector.

References

[1] W.E. Denton and A.W.G. Duller and P.J. Fish, "Boundary Detection for Skin Lesions: An Edge Focusing Algorithm," in *IEE Conference on Image Processing*, IEE, July 1995. SEECS DB 455,461.

[2] W. Denton, A. Duller, and P. Fish, "Robust boundary detection for skin lesions," in *Proceedings 17th International Conference, IEEE Engineering in Medicine and Biology Society*, September 1995. SEECS DB 457,464.

[3] A. Dhawan and A. Sicsu, "Segmentation of images of skin lesions using colour and texture information of surfac pigmentation," *Computerized Medical Imaging and Graphics*, vol. 3, no. 2, pp. 54–61, 1984.

[4] H. Iversen and T. Lonnestad, "An evaluation of stochastic models for analysis and synthesis of gray-scale texture," *Pattern Recognition Letters*, vol. 15, no. 6, pp. 575–585, 1994.

References

[1] American Academy of Dermatology. Skin Cancer - An Undeclared Epidemic. http://tray.dermatology.uiowa.edu/PIPS/SkinCa.html. http://www.derm-infonet.com/

[2] H. Baher. Analog and digital signal processing. Wiley, 1990.

[3] F. Bergholm. Edge Focusing. IEEE Trans. PAMI, 9(6):726-741, Nov. 1987.

[4] A. Breslow. Thickness, Cross-sectional Areas and Depth of Invasion in the Prognosis of Cutaneous Melanoma. Annals of Surgery, 172(5):903-908, Nov. 1970.

[5] 256x256 8-bit monochrome digitised brodatz textures. ftp://peipa.essex.ac.uk/disks/pix1/texture/brodatz/.

[6] P. Brodatz. Texture: A Photograph Album for Artists and Designers. Dover, New York, 1957.

[7] K. W. Brodlie. Mathematical Methods in Computer Graphics and Design. Academic Press, 1978.

[8] M. Burgess. C: A Dabhand Guide. Dabs Press, second edition, 1988.

[9] CancerNet: cancer information from the (American) National Cancer Institute. mailto:CancerNet@icicc.nci.nih.go, http://www.arc.com/cancernet/cancernet.html.

[10] J. Canny. A Computational Approach to Edge Detection. IEEE Trans. PAMI, 8(6):679-698, Nov. 1986.

[11] N. Cascinelli, M. Ferrario, T. Tonelli, and E. Leo. A possible new tool for clinical diagnosis of melanoma: The computer. Journal of American Academy of Dermatology, 16(2(part 1)):361-367, Feb. 1987.

[12] E. Claridge, P. N. Hall, M. Keefe, and J. P. Allen. Shape analysis for classification of malignant melanoma. J. Biomed. Eng., 14:229-234, May 1992.

[13] D. Concar. The resistible rise of skin cancer. New Scientist, pages 23-28, 1 May 1992.

[14] T. L. Dean, J. Allen, and J. Aloimonos. Artificial intelligence: theory and practice. Benjamin/Cummings, 1995.

[15] E. X. Dejesus. Face Value. Byte, 20(2):85-88, Feb. 1995.

[16] E. J. Delp, R. L. Kashyap, O. R. Mitchell, and R. B. Abhyankar. Image Modelling with a Seasonal Autoregressive Time Series with Applications to Data Compression. In IEEE Comp. Soc. Conf. On Pattern Recognition and Image Processing, pages 100-104, 1978.

[17] W. E. Denton, A. W. G. Duller, and P. J. Fish. Boundary Detection for Skin Lesions: An Edge Focusing Algorithm. In Proc. 5th Int. Conf. on Image Processing and its Applications, IEE, Edinburgh, 1995.

[18] W. E. Denton, A. W. G. Duller, and P. J. Fish. Robust Boundary Detection for Skin Lesions. In IEEE Eng. in Medicine & Biology 17th Annual Conference, IEEE, Montréal, 20-23 Sept 1995.

[19] W. E. Denton, A. W. G. Duller, and P. J. Fish. Synthesis of skin images for image processing validation. In Proc. of 9th Int. Conf. on Biomedical Engineering, pages 445-447, 1997.

[20] R. Deriche. Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector. International Journal of Computer Vision, pages 167-187, 1987.

[21] D. F. Devereux. Melanoma-Pigmented Lesion Center. New Jersey Medicine, 86(5):401-403, May 1989.

[22] P. A. Devijver and J. Kittler. Pattern recognition: a statistical approach. Prentice-Hall, 1982.

[23] A. P. Dhawan. Early detection of cutaneous malignant melanoma by three-dimensional nevoscopy. Computer Methods and Programs in Biomedicine, 21:59-68, 1985.

[24] A. P. Dhawan. An Expert System for the Early Detection of Melanoma Using Knowledge-Based Image Analysis. Analytical and Quantitative Cytology and Histology, 10(6):405-417, Dec. 1988.

[25] A. P. Dhawan, R. Gordon, and R. M. Rangayyan. Nevoscopy: Three-Dimensional Computed Tomography of Nevi and Melanomas In Situ by Transillumination. IEEE Trans. on Medical Imaging, 3(2):54-61, June 1984.

[26] A. P. Dhawan and A. Sicsu. Segmentation of Images of Skin Lesions Using Color and Texture Information of Surface Pigmentation. Computerized Medical Imaging and Graphics, 16(3):163-177, 1992.

[27] A. Durg, W. V. Stoecker, J. P. Cookson, S. E. Umbaugh, and R. H. Moss. Identification of Variegated Coloring in Skin Tumors. IEEE Eng. in Medicine & Biology Magazine, pages 71-98, Sept. 1993.

[28] G. Elber. GIF library. ftp://garbo.uwasa.fi/pc/gifutil/giflib12.zip, May 1991.

[29] F. Ercal, A. Chawala, W. V. Stoecker, H.-C. Lee, and R. H. Moss. Neural Network Diagnosis of Malignant Melanoma From Colour Images. IEEE Trans. on Biomedical Eng., 41(9):837-845, Sept. 1994.

[30] F. Ercal, M. Moganti, W. V. Stoecker, and R. H. Moss. Detection of Skin Tumor Boundaries in Color Images. IEEE Trans. on Medical Imaging, 12(3):624-627, Sept. 1993.

[31] A. G. Flook. The Use of Dilation Logic on the Quantimet to Achieve Fractal Dimension Characterisation of Textured and Structured Profiles. Powder Technology, 21:295-298, 1978.

[32] J. D. Foley. Fundamentals of interactive computer graphics. Addison-Wesley, 1982.

[33] R. J. Friedman, D. S. Rigel, A. Kopf, et al. Volume of malignant melanoma is a superior to thickness as a prognostic indicator. Dermatologic Clinics, 9(4):643-648, Oct. 1991.

[34] D. J. Gawkrodger. An illustrated colour text, dermatology. Churchill Livingstone, 1992.

[35] A. S. Glassner, editor. Graphics gems, volume 1. Academic Press, 1990.

[36] J. E. Golston, R. H. Moss, and W. V. Stoecker. Boundary detection in skin tumor images: An overall approach and a radial search algorithm. Pattern Recognition, 23(11):1235-1247, 1990.

[37] J. E. Golston, W. V. Stoecker, R. H. Moss, and I. P. S. Dhillon. Automatic Detection of Irregular Borders in Melanoma and Other Skin Tumors. Computerized Medical Imaging and Graphics, 16(3):199-203, 1992.

[38] R. C. Gonzalez and R. E. Woods. Digital Image Processing. Addison-Wesley, 1992.

[39] J. F. Haddon and J. F. Boyce. Co-occurrence matrices for image analysis. Electronics & Communications Eng. Journal, 5(2):71-83, Apr. 1993.

[40] P. N. Hall. Clinical Diagnosis of Melanoma. In N. Kirkham, D. W. K. Cotton, R. C. Lallemand, J. E. White, and R. D. Rosin, editors, Diagnosis and Management of Melanoma in Clinical Practice, chapter 4, pages 35-52. Springer-Verlag, New York, 1992.

[41] R. M. Haralick, S. R. Sternberg, and X. Zhuang. Image Analysis Using Mathematical Morphology. IEEE Trans. PAMI, 9(4):532-550, 1987.

[42] M. Herbin, A. Venot, J. Y. Devaux, and C. Piette. Color Quantitation Through Image Processing in Dermatology. IEEE Trans. on Medical Imaging, 9(3):262-269, Sept. 1990.

[43] T. H. Hong and A. Rosenfeld. Compact Region Extraction Using Weighted Pixel Linking in a Pyramid. IEEE Trans. PAMI, 6(2): 222-229, Mar. 1984.

[44] T. S. Huang, G. J. Yang, and G. Y. Tang. A Fast Two-Dimensional Median Filtering Algorithm. IEEE Trans. on Acoustics, Speech, and Signal Processing, 27(1):13-18, Feb. 1979.

[45] A. Huertas and G. Medioni. Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks. IEEE Trans. PAMI, 8(5):651-664, Sept. 1986.

[46] P. Hughett. Re: Laplacian-of-Gaussian edge detector, 13 July 1993. [e-mail to bill@sees.bangor.ac.uk], Available mailto:bill@sees.bangor.ac.uk.

[47] D. H. Hush and B. G. Horne. Progress in Supervised Neural Networks. IEEE Signal Processing Magazine, pages 8-39, Jan. 1993.

[48] H. Iversen and T. Lønnestad. An Evaluation of Stochastic Models for Analysis and Synthesis of Gray-Scale Texture. Pattern Recognition Letters, 15(6):575-585, June 1994.

[49] R. Jain, R. Kasturi, and B. G. Schunck. Machine vision. McGraw-Hill, New York, 1995.

[50] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. In Proc. First Int. Conf. on Computer Vision, pages 259-268, London, England, June 1987. IEEE.

[51] B. H. Kaye. A Random Walk Through Fractal Dimensions. VCH Publishers, New York, NY (USA), 1989.

[52] J. Kittler, J. Illingworth, J. Foglein, and K. Paler. An Automatic Thresholding Algorithm and its Performance. In IEEE Proc. Seventh Int. Conf. Pattern Recognition, volume 1, pages 287-289, Montréal, P.Q., Canada, 1984.

[53] A. Kjoelen, M. J. Thompson, S. E. Umbaugh, R. H. Moss, and W. V. Stoecker. Performance of AI Methods in Detecting Melanoma. IEEE Eng. in Medicine & Biology Magazine, pages 411-416, July/Aug. 1995.

[54] K. F. Lai and R. T. Chin. Deformable Contours: Modeling and Extraction. In Proc. 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 601-608, Seattle, Washington, June 1994. IEEE.

[55] H. C. Lee. Skin cancer diagnosis using hierarchical neural networks and fuzzy logic. Comp. Sci. Dept, University of Missouri-Rolla. 1994.

[56] C. Lilley, F. Lin, W. T. Hewitt, and T. L. J. Howard. Colour in Computer Graphics. http://info.mcc.ac.uk/CGU/ITTI/Col/colour_announce.html, Jan. 1995.

[57] R. P. Lippmann. An Introduction to Computing with Neural Nets. IEEE ASSP Magazine, pages 4-22, Apr. 1987.

[58] A. Low. Introductory Computer Vision and Image Processing. McGraw-Hill, 1991.

[59] R. M. MacKie. An illustrated guide to the recognition of early malignant melanoma. Department of Dermatology, University of Glasgow, 1986.

[60] R. M. MacKie. Clinical recognition of early invasive malignant melanoma. British Medical Journal, 301(6759):1005-1006, 3 Nov. 1990.

[61] R. M. MacKie. Melanoma Prevention and Early Detection. British Medical Bulletin, 51(3):570-583, 1995.

[62] B. B. Mandelbrot. Fractals. W.H. Freeman, San Francisco, 1977.

[63] R. Marks. Roxburgh's Common Skin Diseases. Chapman & Hall Medical, 16th edition, 1993.

[64] D. Marr. Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. W.H. Freeman, 1982.

[65] D. Marr and E. Hildreth. Theory of edge detection. Proc. Roy. Soc. London, B207:187-217, 1980.

[66] The MathWorks, Inc. MATLAB Signal Processing Toolbox, 1988.

[67] The MathWorks, Inc. MATLAB Reference Guide, 1992.

[68] A. I. Medalia. Dynamic Shape Factors of Particles. Powder Technology, 4:117-138, 1970.

[69] National Skin Centre, Singapore. http://medweb.nus.sg/nsc/nsc.html, 1995.

[70] NetPBM Toolkit. ftp://unix.hensa.ac.uk/mirrors/FreeBSD/distfiles/netpbm-1mar1994.tar.gz, 1994.

[71] A. V. Oppenheim and R. V. Schafer. Digital signal processing. Prentice-Hall, 1975.

[72] N. Otsu. A Threshold Selection Method from Gray-Level Histograms. IEEE Trans. on Systems, Man, & Cybernetics, 9(1):62-66, Jan. 1979.

[73] A. P. Paplinski and J. F. Boyce. An Implementation of the Active Contour Method for Noisy Images Using a Local Minimisation Algorithm. Technical report, Robotics and Digital Technology, Monash University, Clayton VIC 3168, Australia, Jan. 1995.

[74] R. Parson. Ozone depletion faq. http://www.cis.ohio-state.edu/hypertext/faq/usenet/ozone-depletion/top.html, ftp://rtfm.mit.edu/pub/usenet/news.answers/ozone-depletion/, 28 May 1995.

[75] T. Pavlidis. A Review of Algorithms for Shape Analysis. Computer Graphics and Image Processing, 7:243-258, 1978.

[76] T. Pavlidis. Algorithms in Graphics and Image Processing. Computer Science Press, 1982.

[77] D. A. Perednia, R. G. White, and R. A. Schowengerdt. Localization of Cutaneous Lesions in Digital Images. Computers and Biomedical Research, 22:374-392, 1989.

[78] R. A. Peters II. Software for 2D and 3D Mathematical Morphology. Technical report, Vanderbilt University School of Engineering, Nashville, TN 37235, 1991.

[79] I. Poole, A. Carothers, and D. Charleston. Object Relocation in Automated Image Cytometry. Bioimaging, 2(1):62-68, Mar. 1994.

[80] W. K. Pratt. Digital Image Processing. John Wiley & Sons, second edition, 1991.

[81] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. Numerical recipes in C : the art of scientific computing. Cambridge University Press, second edition, 1992.

[82] Rasure and Kubica. The Khoros Application Development Environment. In H. I. Christensen and J. L. Crowley, editors, Experimental Environments for Computer Vision and Image Processing. World Scientific, 1994.

[83] Rasure and Young. An Open Environment for Image Processing Software Development. In 1992 SPIE/IS&T Symposium on Electronic Imaging, SPIE Proceedings, volume 1659, 14 Feb. 1992.

[84] A. J. Reid and S. J. Doyle. IPLIB Transputer Image Processing Library. National Engineering Laboratory.

[85] J. C. Russ. The image processing handbook. CRC Press, second edition, 1994.

[86] S. Sarkar and K. L. Boyer. On Optimal Infinite Impulse Response Edge Detection Filters. IEEE Trans. PAMI, 13(11):1154-1171, Nov. 1991.

[87] J. L. Semple, A. K. Gupta, L. From, K. A. Harasiewicza, D. N. Sauder, F. S. Foster, and D. H. Turnbull. Does high-frequency (40-60 mhz) ultrasound imaging play a role in the clinical management of cutaneous melanoma. Annals of Plastic Surgery, 34(6):599-605, 1995.

[88] I. Sommerville. Software Engineering. Addison-Wesley, third edition, 1989.

[89] G. E. Sotak, J. R. Boyer, and K. L. Boyer. The Laplacian-of-Gaussian Kernal: A Formal Analysis and Design Procedure for Fast, Accurate Convolution and Full-Frame Output. Computer Vision, Graphics, and Image Processing, 48:147-189, 1989.

[90] W. V. Stoecker, W. W. Li, and R. H. Moss. Automatic Detection of Asymmetry in Skin Tumors. Computerized Medical Imaging and Graphics, 16(3):191-197, 1992.

[91] K. A. Stroud. Engineering Mathematics. Macmillan, fourth edition, 1995.

[92] V. Torre and T. A. Poggio. On Edge Detection. IEEE Trans. PAMI, 8(2):147-163, Mar. 1986.

[93] S. E. Umbaugh, R. H. Moss, and W. V. Stoecker. Applying Artificial Intelligence to the Identification of Variegated Coloring in Skin Tumors. IEEE Eng. in Medicine & Biology Magazine, pages 57-62, Dec. 1991.

[94] S. E. Umbaugh, R. H. Moss, W. V. Stoecker, and G. A. Hance. Automatic Color Segmentation Algorithms With Application to Skin Tumour Feature Identification. IEEE Eng. in Medicine & Biology Magazine, pages 75-82, Sept. 1993.

[95] S. E. Umbaugh and W. V. Stoecker. Automatic Color Segmentation of Images with Application to Detection of Variegated Coloring in Skin Tumors. IEEE Eng. in Medicine & Biology Magazine, 8(4):43-52, Dec. 1989.

[96] M. Unser, A. Aldroubi, and M. Eden. Recursive Regularization Filters: Design, Properties, and Applications. IEEE Trans. PAMI, 13(3):272-277, Mar. 1991.

[97] P. W. Verbeek and L. J. V. Vliet. On the Location Error of Curved Edges in Low-Pass Filtered 2-D and 3-D Images. IEEE Trans. PAMI, 16(7):726-733, July 1994.

[98] R. White, D. S. Rigel, and R. J. Friedman. Computer Applications in the Diagnosis and Prognosis of Malignant Melanoma. Dermatologic Clinics, 9(4):695-703, Oct. 1991.