

1993

# Computer-assisted Petrographic Image Analysis And Quantization Of Rock Texture

Abani Kanta Samantaray

Follow this and additional works at: <https://ir.lib.uwo.ca/digitizedtheses>

---

## Recommended Citation

Samantaray, Abani Kanta, "Computer-assisted Petrographic Image Analysis And Quantization Of Rock Texture" (1993). *Digitized Theses*. 2317.

<https://ir.lib.uwo.ca/digitizedtheses/2317>

This Dissertation is brought to you for free and open access by the Digitized Special Collections at Scholarship@Western. It has been accepted for inclusion in Digitized Theses by an authorized administrator of Scholarship@Western. For more information, please contact [tadam@uwo.ca](mailto:tadam@uwo.ca), [wlsadmin@uwo.ca](mailto:wlsadmin@uwo.ca).

**Computer-Assisted Petrographic Image Analysis  
and Quantization of Rock Texture**

**by**

**Abani Kanta Samantaray**

**Department of Geology**

**Submitted in partial fulfilment  
of the requirements for the degree of  
Doctor of Philosophy**

**Faculty of Graduate Studies  
The University of Western Ontario  
London, Ontario  
April 1993**

**© Abani Kanta Samantaray 1993**



National Library  
of Canada

Acquisitions and  
Bibliographic Services Branch

385 Wellington Street  
Ottawa, Ontario  
K1A 0N4

Bibliothèque nationale  
du Canada

Direction des acquisitions et  
des services bibliographiques

385, rue Wellington  
Ottawa (Ontario)  
K1A 0N4

Your file / Votre référence

Our file / Notre référence

**The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.**

**L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.**

**The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.**

**L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.**

ISBN 0-315-84005-6

**Canada**

## ABSTRACT

Under the petrographic microscope, most rocks in thin section appear as an assemblage of tightly interlocked mineral grains, inclusions, pore spaces etc. The geometrical characteristics of these features, including their apparent sizes, shapes, orientations and distributions, define the texture of a rock.

In this study the optical image from a petrographic microscope is processed by a micro-computer. The image is converted into electrical signals by a Sony AVC-D5 monochrome video camera. These signals are digitized by an analog to digital converter in a Imaging Technology PCVISION *plus* frame grabber board which stores and manipulates the resulting digital image in its frame memory. The image is stored in 256-colour PCX format.

The extraction of geological information from the digital image requires that the features in the image be identified and their edges defined. Feature identification is accomplished by manipulation of the digital image which is referred to as image processing. This involves three sequential operations: digital filtering, image segmentation and feature extraction. In the present study fourteen digital filters are evaluated for their abilities to reduce normally distributed additive noise while preserving linear features and image texture. The Sigma Filter is shown to be most suitable for application to petrographic images. The edges of the features of interest are extracted using zero-crossing edge finders with varying window sizes. The sequential capturing of multiple images from one microscope field of view allows a thin section to be analyzed in a manner analogous to the procedure followed in manual petrography. Interactive manipulation of the image containing the detected edges is possible using an image editor.

The feature extraction process identifies and selects features of interest from the detected edges of the petrographic image. This information, the original grey level image and the intermediate segmented edge image, are all used to provide data which are not available from traditional petrography. Salient features of the image processing system developed here are illustrated by application to selected geological problems for which data obtained by conventional techniques are available.

Image processing can provide an initial step in expert systems developed to solve specific petrographic problems. This could allow the processing to be automated using knowledge banks interactively at each stage of image analysis.

***To my parents***

## ACKNOWLEDGEMENTS

I wish to express my deep gratitude to Dr. John Starkey, my supervisor, for patiently offering me advice and guidance, and for spending a lot of time talking to me about new ideas that led to this work.

I wish to thank the management of Central Mine Planning and Design Institute Limited, Ranchi, India, for granting me study leave to carry out this research. This research was supported by operating grant A3555 from the National Science and Engineering Research Council, Canada. Their financial aid is greatly appreciated.

I thank Dr. S. Longiaru for his assistance in developing the computer code for the Frame Grabber Board. I would also like to thank Mr. Simon A.J. Winder, Computer Vision Research, University of Bath Computing Group, U.K., for his assistance in implementing the Canny algorithm.

Thanks are due to Dr. R. Cheel, Chairman, Department of Geology, Brock University, who provided me with thin sections and data sets for the study of sedimentary grain shapes, and to my colleague John McArthur, who provided me with thin sections and data sets for the study of grain shapes in a deformed oolitic limestone. I wish to thank Dr. G.M. Young for his help with sedimentological queries.

# Contents

<b>CERTIFICATE OF EXAMINATION</b>	<b>ii</b>
<b>ABSTRACT</b>	<b>iii</b>
<b>ACKNOWLEDGEMENTS</b>	<b>vi</b>
<b>TABLE OF CONTENTS</b>	<b>vii</b>
<b>LIST OF TABLES</b>	<b>x</b>
<b>LIST OF FIGURES</b>	<b>xi</b>
<b>CHAPTER 1 - INTRODUCTION</b>	<b>1</b>
1.1 Petrographic Image Analysis : an Overview . . . . .	2
1.2 Acquisition, Processing and Measurement of Images . . . . .	5
<b>CHAPTER 2 - IMAGE ACQUISITION</b>	<b>8</b>
2.1 Video Camera . . . . .	12
2.2 Frame Grabber . . . . .	12
2.2.1 Analog to Digital Converter . . . . .	12
2.2.2 Frame Memory . . . . .	13
2.2.3 Display Logic Circuit . . . . .	15
2.3 Nature of the Image . . . . .	17

2.4	Image File Format . . . . .	19
2.5	Interactive Image Acquisition . . . . .	22
<b>CHAPTER 3 - IMAGE PROCESSING</b>		<b>23</b>
3.1	Digital Filtering . . . . .	24
3.1.1	Single-level Filters . . . . .	27
3.1.2	Multilevel Filters . . . . .	35
3.1.3	Evaluation of the Filters . . . . .	40
3.2	Image Segmentation . . . . .	42
3.2.1	Gradient Operators . . . . .	46
3.2.2	Edge Thinning and Following . . . . .	54
3.2.3	Zero-crossing Operators . . . . .	62
3.2.4	Segmentation of Sequential Images . . . . .	69
3.3	Binary Image Editing . . . . .	74
3.3.1	Manual Editing . . . . .	76
3.4	Feature Extraction . . . . .	77
3.4.1	Sorting Boundary Pixel Coordinates . . . . .	86
3.5	Nature of the Information . . . . .	88
<b>CHAPTER 4 - IMAGE MEASUREMENT</b>		<b>90</b>
4.1	Global Measurements . . . . .	91
4.2	Feature Measurements . . . . .	96
4.2.1	Measures of Size . . . . .	96
4.2.2	Descriptions of Feature Location . . . . .	97
4.2.3	Aspects of Shape . . . . .	100
4.2.4	Texture . . . . .	110
<b>CHAPTER 5 - EXAMPLE APPLICATIONS</b>		<b>111</b>

5.1	Sedimentary Grain Shapes . . . . .	112
5.2	Shapes of Deformed Oolites . . . . .	123
5.3	Modal Analysis . . . . .	130
<b>CHAPTER 6 - CONCLUSION AND FUTURE WORK</b>		<b>133</b>
6.1	Conclusion . . . . .	133
6.2	Future Work . . . . .	135
<b>APPENDIX 1. Evaluation of Digital Filters</b>		<b>140</b>
<b>APPENDIX 2. Computational Masks for the Canny algorithm</b>		<b>159</b>
<b>APPENDIX 3. Locating Ends of Edge contours</b>		<b>162</b>
<b>APPENDIX 4. Pascal code for the IMAGING SYSTEM</b>		<b>165</b>
<b>REFERENCES</b>		<b>167</b>
<b>VITA</b>		<b>175</b>

# List of Tables

2.1	Modified header format of 256-colour PCX files . . . . .	19
4.1	Area, Perimeter, and X & Y coordinates of the centroid for grains shown in figure 4.4. . . . .	98
4.2	Aspect ratios and orientation angles for the grains shown in figure 4.4	103
4.3	Formfactor and Roundness parameters for the grains shown in figure 4.4 . . . . .	107
5.1	Robin's strain, standard error, $\phi$ -mean, and angular deviation measured from the ooid particles in an experimentally deformed oolitic limestone by three different methods of image analysis . . . . .	130

# List of Figures

1.1	An overview of the steps in computer-assisted, petrographic image analysis . . . . .	6
2.1	Photomicrograph of a thin section of sandstone between crossed polarizers . . . . .	9
2.2	Schematic diagram of the image digitizer . . . . .	10
2.3	Block diagram of the image digitizing system . . . . .	11
2.4	The coordinate system used to address the memory storage locations in frame memory . . . . .	14
2.5	Organization of the LUTs of the frame grabber . . . . .	16
2.6	Graphical representation of the configuration of the LUTs used in the image digitizing system . . . . .	18
3.1	Microscope image of a quartz grain observed in plane polarized light and frequency distribution of the observed grey levels . . . . .	25
3.2	The nine possible subregions for the Nagao and Matsuyama Filter . .	30
3.3	Subwindows used by unidirectional multilevel filters . . . . .	36
3.4	Microscope image showing an intensity edge between two adjacent quartz grains and a texture edge between a quartz grain and a grain of microcline (crossed polarizers) . . . . .	43
3.5	A convolution operation . . . . .	45
3.6	The masks used for the Sobel and Prewitt operators . . . . .	47
3.7	Microscope image of a thin section showing quartz grains observed in plane polarized light . . . . .	48
3.8	Image showing edges detected using the Sobel operator . . . . .	50

3.9	The four separate templates of a Kirsch operator . . . . .	51
3.10	Image showing edges detected using a Kirsch operator with an un-weighted mask . . . . .	52
3.11	Intensity profiles showing the ramp edges in a microscope image and the multiple indication of the edges detected from the image . . . . .	53
3.12	The edges extracted from an edge image by applying a fixed threshold value of 32 to the edge strength . . . . .	55
3.13	The edges extracted from an edge image by applying an adaptive threshold . . . . .	56
3.14	Examples of undesirable results from edge thinning . . . . .	57
3.15	The manipulation of the v and m counters in the 3 by 1 window used in Lacroix's edge thinning algorithm . . . . .	59
3.16	The 3 by 3 window required to determine the four possible contour following direction in Kunt's algorithm . . . . .	61
3.17	The edges detected by applying Lacroix's edge strategy to the edges located by the Kirsch operator . . . . .	63
3.18	The Gaussian kernel used to smooth the input image before applying zero-crossing operators . . . . .	65
3.19	Image showing edges detected using the Marr-Hildreth operator . . . . .	66
3.20	Image showing edges detected using the Canny operator . . . . .	68
3.21	Two images captured from one microscope field of view of a thin section of quartzite between crossed polarizers in which the relative orientations of the planes of polarization differ by 45 degrees . . . . .	71
3.22	The edges obtained by applying the Canny operator to sequentially captured microscope images . . . . .	72
3.23	Composite grain boundary image obtained by combining two edge images . . . . .	73
3.24	The organization of the eight bit planes of frame memory . . . . .	75
3.25	The possible configurations of pixels in a 3 by 3 window in which the central pixel is the end of an edge contour. . . . .	78
3.26	The edges detected by the Canny operator with the ends of discontinuous edge contours highlighted . . . . .	79
3.27	The output image obtained from an edge image after feature extraction . . . . .	81

3.28	The Freeman chain codes used to identify the orientations of the eight nearest neighbours relative to the central pixel . . . . .	83
3.29	The possible configuration for connecting line segments in three consecutive scans of a raster scanned image . . . . .	85
3.30	Feature boundary information. The boundary coordinates are recorded as a sequence of Freeman chain code directions and as (x,y) coordinates	89
4.1	Measuring the area of inclusions . . . . .	92
4.2	The determination of intercept lengths . . . . .	94
4.3	The association of two mineral grains and the use of a 3 by 3 window to recognise the shared boundary . . . . .	95
4.4	The boundaries of the mineral grains determined automatically by image analysis . . . . .	99
4.5	The true area centroid and the centroid located from the boundary data . . . . .	101
4.6	The long and short axes of grains . . . . .	104
4.7	The 'best-fit' ellipses fitted to grain boundaries . . . . .	105
4.8	The characteristic waveform obtained from a grain outline unrolled with reference to the centroid . . . . .	109
5.1	Rose diagrams showing the orientation of the longest apparent axes of the sand grains in three samples of beach sands. The orientation angles are measured with respect to the internal lamination . . . . .	114
5.2	Rose diagrams showing the orientation of the longest apparent axes of the sand grains in two samples of beach sands. The orientation angles are measured with respect to the flow direction . . . . .	115
5.3	Rose diagrams showing the orientation of the longest axes of the sand grains in three samples of beach sands . . . . .	117
5.4	Rose diagrams showing the orientation of the longest axes of the sand grains in two samples of beach sands . . . . .	118
5.5	Rose diagrams showing the orientation of the longest axes of three subsets of aspect ratio cutoffs of sample A . . . . .	120
5.6	Rose diagrams showing the orientation of the long axes of the best-fit ellipses fitted to the sand grains in three samples of beach sands . . .	121
5.7	Rose diagrams showing the orientation of the long axes of the best-fit ellipses fitted to the sand grains in two samples of beach sands . . . .	122

5.8	Microscope image of a thin section showing oolites observed in plane polarized light . . . . .	125
5.9	$R_f/\phi$ plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using a Zeiss Videoplan digital analyzer system	127
5.10	$R_f/\phi$ plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using the image analysis sytem based on a tracing of the image . . . . .	128
5.11	$R_f/\phi$ plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using the present imaging system . . . . .	129
5.12	Images showing the biotite and quartz content of a Saxony granulite .	132
6.1	Proposed organization of future computer based petrographic analysis	137

The author of this thesis has granted The University of Western Ontario a non-exclusive license to reproduce and distribute copies of this thesis to users of Western Libraries. Copyright remains with the author.

Electronic theses and dissertations available in The University of Western Ontario's institutional repository (Scholarship@Western) are solely for the purpose of private study and research. They may not be copied or reproduced, except as permitted by copyright laws, without written authority of the copyright owner. Any commercial use or publication is strictly prohibited.

The original copyright license attesting to these terms and signed by the author of this thesis may be found in the original print version of the thesis, held by Western Libraries.

The thesis approval page signed by the examining committee may also be found in the original print version of the thesis held in Western Libraries.

Please contact Western Libraries for further information:

E-mail: [libadmin@uwo.ca](mailto:libadmin@uwo.ca)

Telephone: (519) 661-2111 Ext. 84796

Web site: <http://www.lib.uwo.ca/>

# Chapter 1

## Introduction

Image analysis refers to the digital techniques by which numerical information regarding the features of interest can be extracted from an image (Sklansky 1978). In petrography, image analysis can be applied to examine rock thin sections under the microscope to extract information about the constituent mineral grains and their inclusions, and pore spaces. The geometrical characteristics and spatial distribution of these features define the texture of a rock. Characterization of rock textures may require the analysis of hundreds of samples in any particular investigation, hence the desirability of an automated system to obtain the necessary data. Such a system also offers the possibility of obtaining objective and reproducible data, especially if it is under appropriate computer control.

In computer-assisted petrography the microscope image is digitized and stored in a computer as an array of numbers. Each number represents the grey level of the image at the corresponding point and it is the numerical array which is used to determine the different parameters of the rock texture. Petrographic image analysis to date has been applied mainly either to the study of mineral grain morphology, which provides data for interpretations of provenance, lithification and diagenesis,

or to the study of pore spaces, which provides information on porosity and permeability. However, the lack of suitable imaging technology for rapid data acquisition has hindered the efficient implementation of rigorous, quantitative methods in petrography.

Recent improvements in both semiautomatic and automatic analyzing equipment for image analysis and pattern recognition, suggest that we could be on the verge of major advances in petrography. With the help of such equipment, features can be detected, measured, and analyzed in real-time by an appropriate computer-based system. These analyses and measurements can be automated for the acquisition of large amounts of data.

## **1.1 Petrographic Image Analysis : an Overview**

There are few published accounts of image analysis systems in geology and these are concerned with obtaining grain boundaries and pore spaces. Traditionally grain outlines have been digitized from photomicrograph mosaics of rock thin sections by tracing the grain boundaries with a digitizer (e.g. Fabbri 1984, Telford et.al. 1987, Simigian and Starkey 1989). Some dedicated machines with greater analytical potential are also available (e.g. Zeiss Videoplan or Kontron Videoplan digital analyser). These are semiautomatic and require that the grain outlines be provided by an operator using a manually operated contour following device.

More sophisticated approaches have been reported which obtain grain size and shape information from photomicrographs of rock thin sections using image analysis. In these, the grain outlines are traced manually on a transparent overlay on the photomicrograph. The traced boundaries are digitized using either an

Optronics International Photoscan System P-100 (Starkey and Simigian 1987) or a monochrome video camera (Starkey and Simigian 1992). The output grey level image is converted to a binary image by applying a simple thresholding to the grey values, values equal to or above a selected value are set to 0, those below are set to 1. The resulting binary image is processed by a computer program, IMAGE, which recognizes individual grain boundaries from the binary image and computes the area, length of perimeter, aspect ratio, centroid and orientation of the long axis of grains. Image analysis based on photomicrographs has also been reported by Jansenn et.al. (1991). Here, the photomicrographs are digitized using a FEAG drum scanner (Kombinat Carl Zeiss Jena) and the resulting digital data are evaluated using the image processing system BVS A6472 (Kombinat Robotron Dresden). The grain boundaries are extracted by a non-linear digital filtering of the image data which consist of the following steps (Jansenn et.al. 1991): high-pass filtering of the original image, thresholding of the grey values of the filtered image, inversion of the grey values and skeletonization of the resulting stripe-like pattern. In this way the original image is transformed into a line pattern representing the grain boundaries.

Dedicated image processing systems are also available which can be interfaced directly to a microscope, thus reducing the amount of operator's interaction. Examples of these include Quantimet, Magiscan and Intellect (Telford et.al. 1987). However, these imaging systems operate as "black boxes" and in some cases obtaining geologically significant data is impossible.

A more flexible system known as ARTHUR, has been developed for geological use (Fico 1980). This consists of a microprocessor-controlled, microscope-mounted video camera. The associated video digitizer (CAT-100) accepts the analog signal from the video camera and generates a spatial array of picture elements, which

record the grey level at the corresponding point on the image. Simple thresholding of the grey values is used to find the first boundary point and follow the feature boundary until a closure is obtained (Ehrlich and Full 1984). Similar systems have also been reported by Lohmann (1983) and Lenth et.al. (1984). These systems capture, digitize and store the image for subsequent data analysis on a main frame computer. A similar system has been reported by Telford et.al. (1987) which make use of an Apple II plus microcomputer thus making it a stand alone system.

Published computer-assisted petrographic analyses tend to be based on binary digital images. In some cases these are obtained by staining the thin section prior to digitization (Pareschi et.al. 1990, Gerrad et.al. 1992). The requirement, that the grey levels of the feature of interest be clearly differentiated from the background, renders these methods unsuitable for automatic, real-time application to the analysis of petrographic rock thin sections which display a continuous range of grey levels, often with low contrast.

Petrographers examining thin sections routinely change the microscope illumination from plane polarized to cross polarized light in order to identify specific features. Also they rotate the thin section on the microscope stage to create differences in the appearance of grains to observe diagnostic optical properties and grain boundaries. The present study addresses these problems and simulates these techniques for the real time analysis of the petrographic images. The imaging system uses a video camera to digitize the petrographic images and a 80386DX computer to identify and quantify the rock texture.

## **1.2 Acquisition, Processing and Measurement of Images**

Petrographic image analysis is performed by procuring a grey level image, enhancing the feature boundaries to improve their detection, segmenting the grey level image to delineate feature boundaries, selecting the features of interest and measuring their parameters for subsequent analysis. The flow diagram for these steps, as implemented in the present study, is shown in figure 1.1. As indicated in figure 1.1 some parameters are obtained directly from the grey scale image.

Image acquisition involves obtaining an image and storing it in a computer. In computer-assisted petrography the sample is typically a standard petrographic thin section. The section is examined through the petrographic microscope using a magnification selected to obtain a statistically meaningful sample of the feature of interest. Images from the microscope are captured by the video camera and transmitted to a frame grabber board. Each image is composed of picture elements, pixels, and each of these can assume a digital value proportional to the grey level at the corresponding location in the image.

Image processing is required to obtain useful information from the acquired image. For better discrimination of the mineral grains, the grey level image is subjected to digital filtering which reduces variations of the grey levels displayed by the mineral grains due to electronic noise, generated during the image acquisition process, and to intrinsic optical inhomogeneities in the mineral.

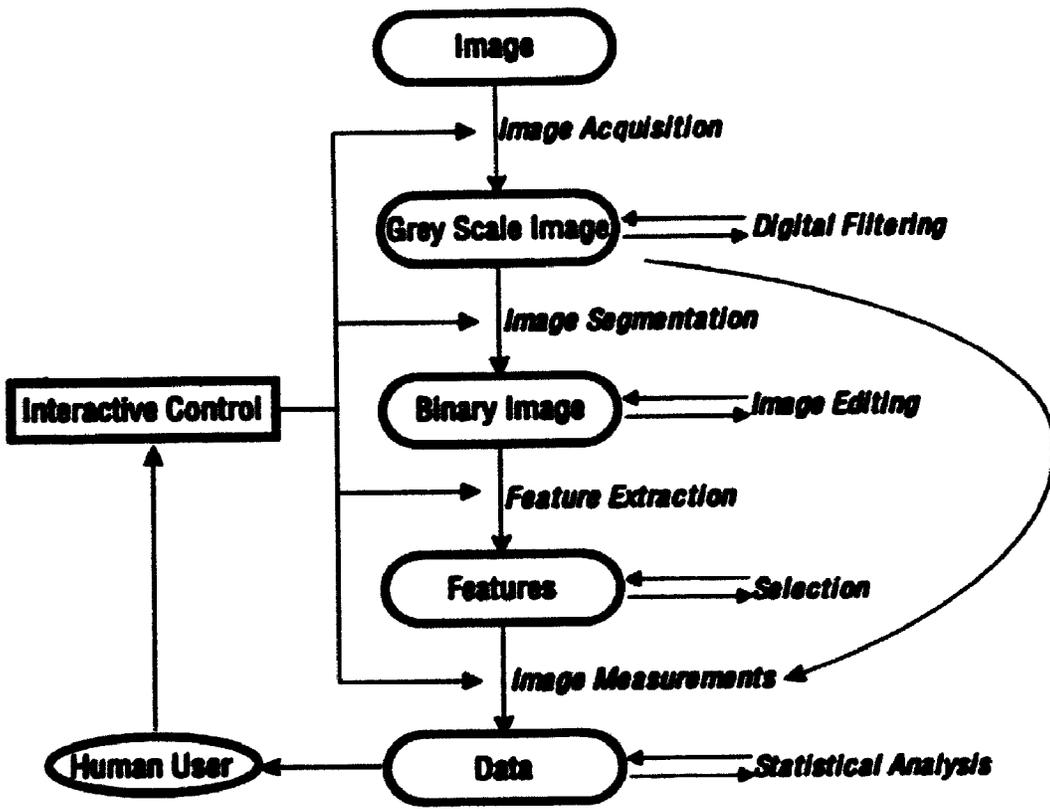


Figure 1.1 : An overview of the steps involved in computer-assisted, petrographic image analysis.

The filtered image displays different shades of grey or image texture for each mineral, but it cannot be analyzed until each mineral grain is separately identified. Boundaries are detected by image segmentation which produces an edge image where the interiors of grains have different intensities to their boundaries. Despite the application of filtering and image segmentation not all the boundaries of the desired features present in the image may be completely delineated. Further editing may be required, this is performed mostly directly on the boundary image itself. Once the boundaries of all the desired features are found, the features are extracted from the boundary image for subsequent data analysis.

In addition to the feature boundary information derived from image processing, as outlined above, the original grey level image and the intermediate segmented image can provide information which permits the quantitative analysis of constituent mineral grains and pore spaces in a rock. These measurements can be either global, where the rock is characterized as a whole, or feature specific, where individual features in the image are measured. Once the individual feature measurements such as size, shape, orientation etc. are obtained they can be subjected to statistical analysis to characterize the aggregate of features.

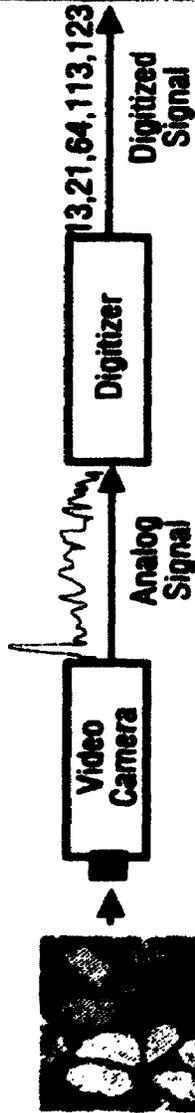
## Chapter 2

# Image Acquisition

Under the optical microscope, most rocks in thin section appear as an assemblage of tightly interlocked mineral grains, inclusions, pore spaces etc. (figure 2.1). The geometrical characteristics of these features, including their apparent shapes, orientations and distributions, define the texture of the rock. For computer processing of the microscope image it must be converted into a form that can be stored and manipulated in computer memory. This conversion is performed by an image digitizer, which produces coded numbers that are a measure of light intensity. This process is called digitization and the stored numerical representation of the original optical image is called a digital image (figure 2.2).

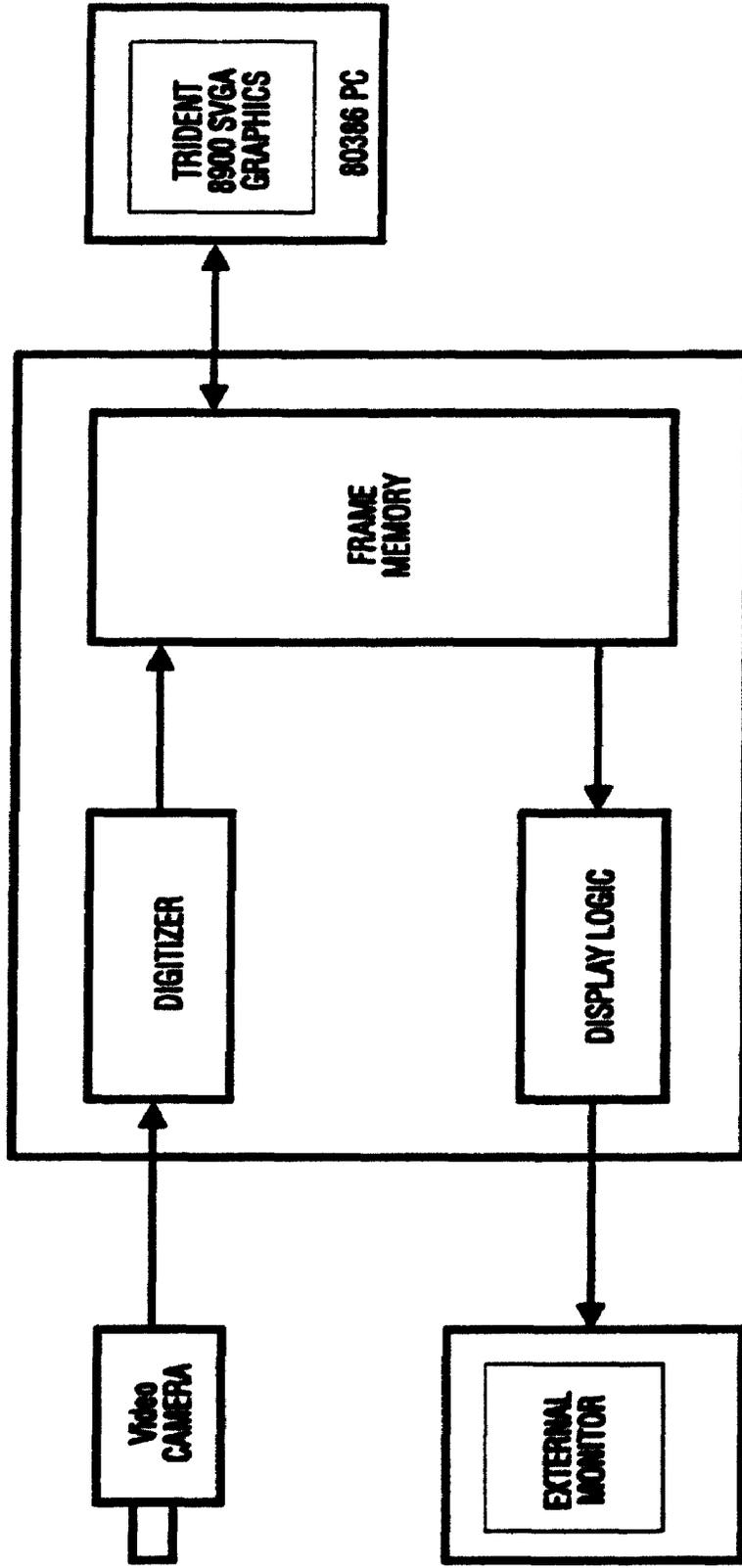
A block diagram of the image digitizing system is shown in figure 2.3. The optical sensor within a video camera converts the image from the microscope into corresponding electrical signals. These signals are digitized by a frame grabber which stores and manipulates the resulting digital image. The frame grabber includes a display logic circuit, which converts the digital image to an analog video output signal for display on an external monitor.

13	21	64	113	123
25	11	93	84	117
75	62	74	50	
103	29	68	14	
120	45	78		
124	49	76		
126				



Digital Image

Figure 2.2 : A schematic diagram of the image digitizing system. The microscope image is captured by the video camera where it is converted to analog signal and passed to the digitizer. The digitized grey values are stored as a digital image in frame memory



**FRAME GRABBER**

Figure 2.3: Block diagram of the image digitizing system. The frame grabber is a circuit board installed inside the 80386 PC.

The frame grabber is installed in a 80386 CPU based PC which executes the software required to control the imaging system hardware and manipulate the digital data. A Trident 8900 Super VGA graphics card is used to display the digital image on the PC video monitor.

## **2.1 Video Camera**

The video camera used in the imaging system is a Sony AVC-D5 monochrome video camera, this has a C mount by which it is attached to the polarizing microscope. The camera uses a CCD (charge-coupled device) as a light sensor. The camera resolution is 512×480 pixels (picture elements) and the individual pixels have an aspect ratio (height to width) of 4:5. The output of the camera conforms to the RS-170 video standard.

## **2.2 Frame Grabber**

The frame grabber used here is a PCVISION *plus* board. The frame grabber consists of an analog to digital converter (digitizer), which converts the electrical signals from the video camera into digital values, a frame memory, which stores the digital data, and a display logic circuit, which converts the digital image to an analog output signal for display on an external TV monitor.

### **2.2.1 Analog to Digital Converter**

The analog image from the video camera is digitized by an analog-to-digital converter which samples the image at discrete time intervals and converts each indi-

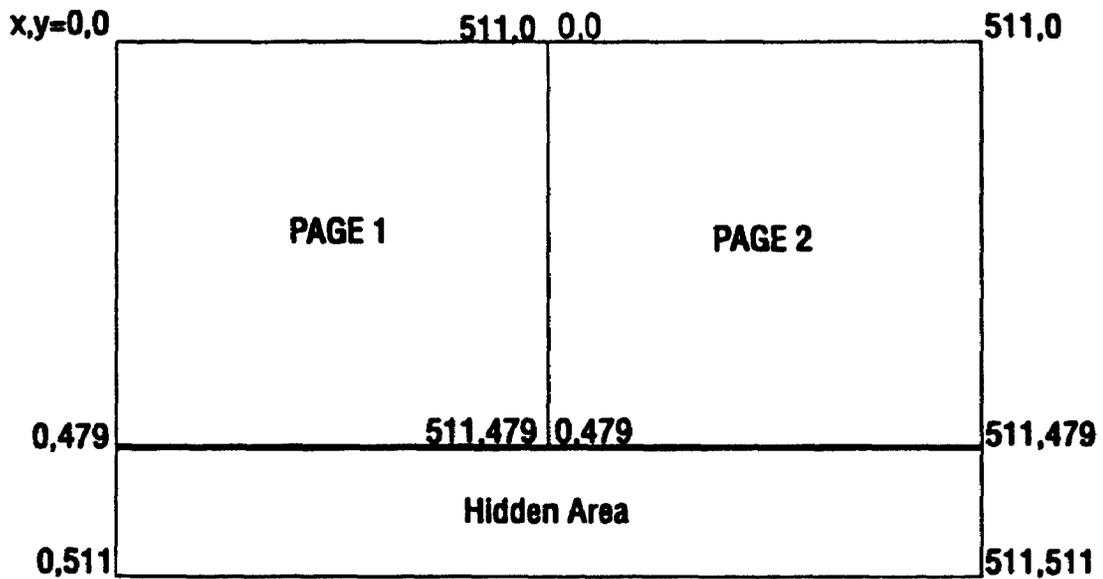
vidual intensity level to a digital value in the range from 0 to 255. One complete sampling of the image is called a 'frame grab'.

Flexibility in digitizing the video source is provided by programmable gain and offset control circuits present in the frame grabber board hardware. The gain boosts or attenuates the incoming signal, it is an adjustable multiplier which can be varied from 0.67 to 1.33 in 100 steps. The offset adds a constant voltage to the signal, this is adjustable from -0.1 Volt to +0.1 Volt in 100 steps. In the present study the defaults for the gain and offset were used, these are midscale in both cases, i.e. gain multiplier 1 and offset 0 volts.

## **2.2.2 Frame Memory**

The frame memory on the frame grabber board is organized as an array of 1024 by 512 eight-bit storage locations. This array is divided into two pages, each with 512 by 512 storage locations, which permits two images to be stored and displayed independently. The coordinate system used to address the storage locations in the frame memory is illustrated in figure 2.4. The external monitor displays a 512 by 480 image, therefore the lower 32 lines of frame memory occur in a hidden area. Only one page of frame memory can be displayed at any one time, this constitutes the active video window.

Each storage location in frame memory can store eight bits which can be addressed separately thus providing eight different bit planes for the image. The frame grabber board provides mask registers which permit write-protection of these bit planes during video acquisition and host computer access. This allows data to be written over the image during subsequent image processing and editing. Host



**Figure 2.4 : The coordinate system used to address the memory storage locations in the two pages of frame memory. The hidden area can not be displayed on the external monitor.**

mask registers are used to protect one or more of the low order bit planes and the video acquisition mask registers are used to protect one or more of the high order bit planes of frame memory.

The lag time between the acquisition and display of an image is  $1/30^{\text{th}}$  second which allows the image of a thin section to be scanned in real-time.

### 2.2.3 Display Logic Circuit

A display logic circuit on the frame grabber board transforms the contents of frame memory into a form compatible with an external monitor. The output of the display logic conforms to the RS-170 video standard which uses an interlaced image display. The external monitor supports an eight-bit image display.

The image can be manipulated by converting individual input data values into predefined output values. This mapping is accomplished using Look-Up Tables (LUTs). The LUTs are located in a block of memory where an input pixel value is used as an address, and the value in the memory block at that address is output as the transformed pixel value. The LUTs can be used to manipulate the contrast range of a displayed image by setting up the tables with nonlinear transformations between input and output grey values. The LUTs can also be used to convert any particular grey value in the image to any arbitrarily chosen display colour (falsecolour).

On the PCVISION *plus* board, there is one group of input LUTs and three groups of output LUTs, one group each for the red, green, and blue channels (figure 2.5). Each group consists of eight blocks of memory (banks) for a total of 32 LUTs, each of which can hold 256 eight-bit values. The host computer can read from or write to only one of the 32 LUTs at one time. To access a LUT, a group (RED,

## LUT Groups

LUT Banks	RED (Channel 1)	GREEN (Channel 2)	BLUE (Channel 3)	INPUT
0	LUT0	LUT0	LUT0	LUT0
1	LUT1	LUT1	LUT1	LUT1
2	LUT2	LUT2	LUT2	LUT2
3	LUT3	LUT3	LUT3	LUT3
4	LUT4	LUT4	LUT4	LUT4
5	LUT5	LUT5	LUT5	LUT5
6	LUT6	LUT6	LUT6	LUT6
7	LUT7	LUT7	LUT7	LUT7

Figure 2.5 : Organization of the eight banks of LUTs of the frame grabber into a group of input LUTs and three groups (channels) of output LUTs.

GREEN, BLUE or INPUT) and its bank (0 to 7) must be specified.

Only one bank of LUTs (bank zero) is used in the present application. The values stored in the active LUTs are illustrated in figure 2.6. As noted above, the grey values from the digitizer are in the range from 0 to 255. The INPUT LUT maps these grey values to a range from 0 to 127, this frees the most significant bit plane (bit-plane 7) for subsequent use in image editing. The output grey values from the INPUT LUT are remapped to a full range (i.e. 0 to 255) in the RED, GREEN, and BLUE LUTs for display.

## 2.3 Nature of the Image

The resolution of the digitized image can be categorized in two ways; grey level resolution, which describes how accurately the digital image represents differences in intensity in the original image, and spatial resolution, which describes how well the digital image represents the position of features in the original image.

The grey level resolution is determined by the number of bits used to store the brightness information of the image. The present imaging system is a monochrome eight-bit system which is capable of storing grey values ranging from 0 to 255 for each pixel, each pixel therefore being represented by one byte. However, only the first 128 (0 to 127) grey values are used to capture the petrographic images in order that the most significant bit (bit-plane 7) remain available for subsequent image editing.

The spatial resolution for the system is  $512 \times 480$ , i.e. the image consists of 480 lines of 512 pixels (for a total of 245,760 pixels). Since the image has a constant

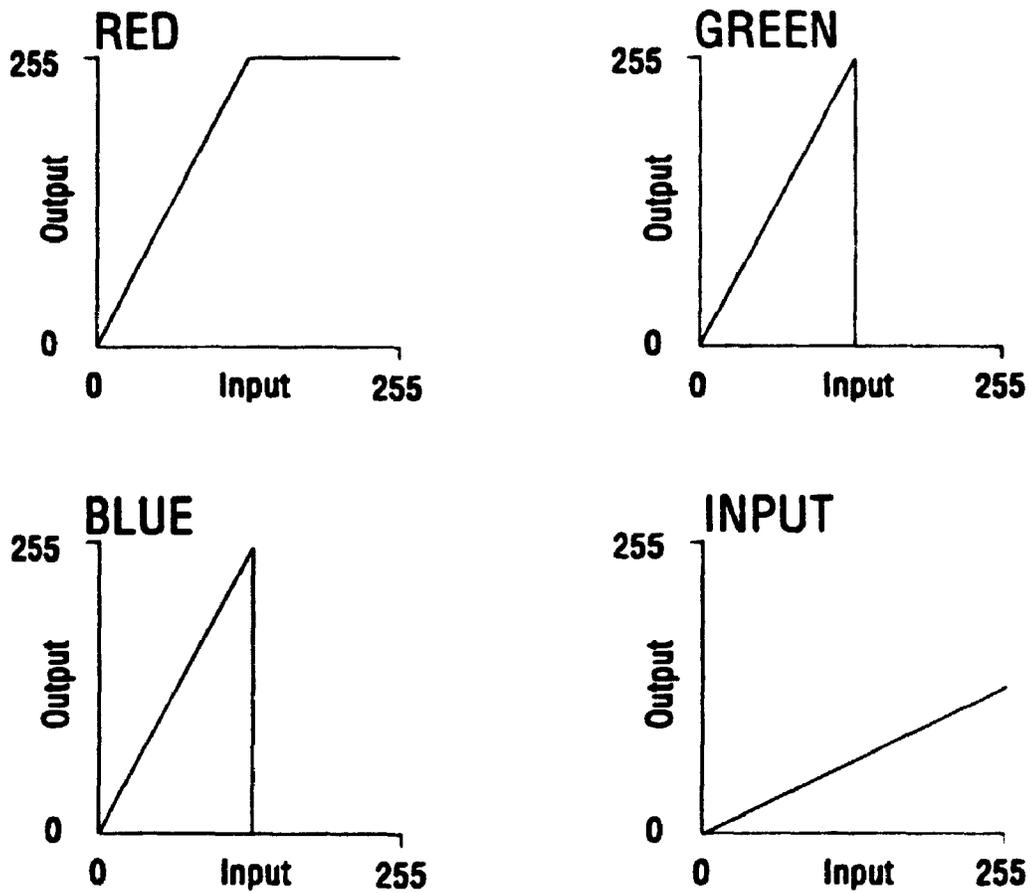


Figure 2.6 : Graphical representation of the configuration of the LUTs used in the image digitizing system.

number of pixels, the area of the thin section represented by each pixel depends on the magnification of the microscope.

## 2.4 Image File Format

The maximum memory requirement for one image is 245,760 bytes, 240K bytes. This is reduced by storing the captured images in 256-colour PCX format (the PC Paintbrush IV file format by ZSoft). The PCX format compresses files by using run length encoding in which successive identical pixel values are grouped and stored in abbreviated form. PCX files carry a 128-byte header, the contents of which are described in Table 2.1. The Scale factor and the Comment are not part of the original ZSoft format, they have been added for the present study to obtain meaningful geological data from the petrographic images. However, the addition of these into the PCX header does not destroy compatibility with standard PCX image files.

Table 2.1: Header format of 256-colour PCX files

Byte	Size (Bytes)	Variable	Description
1	1	Manufacturer	Hexadecimal number, 0x0a designates ZSoft PCX files.
2	1	Version	Versions of PC Paintbrush. For version 3.0 or higher, this byte should contain a value of 5. This byte is important in identifying 256-colour PCX files.

Contd...

3	1	Encoding	The encoding byte should always contain the value of one indicating that the file has been compressed using the PCX run length encoding scheme.
4	1	Bits per pixel	Number of bits used to store data for 1 pixel from 1 plane.
5	8	Window dimensions	4 words (2 bytes each) giving top left and bottom right corners of display in the order <i>xmin</i> , <i>ymin</i> , <i>xmax</i> , <i>ymax</i> .
13	2	Horizontal Resolution	Horizontal resolution of display device that created the image.
15	2	Vertical Resolution	Vertical resolution of display device (lines) that created the image.
17	48	Colour map	Information on colour palette settings if the image has 16 or fewer colours.
65	1	Reserved	
66	1	Colour planes	Number of colour planes in the original image.
67	2	Bytes per line	Number of bytes per scan line in the image.
69	2	Palette type	Palette type, 1 = colour\monochrome, 2 = greyscale
71	6	Scale factor	Scale of the image to real world. This is a floating point value.
77	52	Comment	Description of the image, upto 52 ascii characters.

A Trident 8900 Super VGA display adapter is used in the present imaging system. This manipulates colours in a fashion similar to a standard VGA adapter. The display adapter contains 256 palette registers which are used to display 256 different colours. Each palette register contains three colour registers for red, green,

and blue. The colour registers contain six-bits, thus permitting  $2^6=64$  shades of each primary colour. To create the colour map, the colour registers (RGB values) associated with the palette registers are used. As the RGB values in the palette registers are six-bit values they are multiplied by four in order to store them in a PCX file as an eight-bit byte.

256-colour PCX files use the same format to store palette information as that used for standard 16-colour PCX files, but there is more information since the 256 colour palettes require three bytes per colour, for a total of 768 bytes. These are appended at the end of the PCX file. The presence of this information is indicated by setting the Version byte of the header to 5. The program reading the file, first reads the Version byte and, if it detects a 5, it seeks to the end of the file and counts back 769 bytes. At this location the file contains the value 0x0c (hexadecimal number), indicating the start of the palette information, and the next 768 bytes represent the palettes. The first three bytes of the palette information are the red, green, and blue values respectively for the first colour, the next three are RGB values for the second colour, and so on. After the palette information is read, the program reading the file returns to the beginning of the file to read the image data.

The Trident 8900 Super VGA card uses six-bit values for each of the colour registers to display the image to the screen. Therefore, the eight-bit values used to store the colour registers in the PCX file, noted above, are divided by four.

The image data, which occur after the PCX header, are compressed. For an eight-bit PCX image only one colour plane is used and each pixel value is represented by one byte. The data in a PCX file can be considered as a series of packets containing a key byte and a data byte. The key byte indicates how the data byte

in the packet is to be processed and the data byte indicates what will be decoded to make up the restored image. If the two high order bits of the key byte are set, i.e. non zero, then the six low order bits contain an index indicating the number of times the following data byte in the file is to be repeated. Because the index is stored in six bits, it can indicate a maximum repetition of 63 bytes. Longer runs of identical bytes require more than one packet. If the two high order bits of the key byte are not set, then this byte is considered to be a data byte and is written to the image "as is". If the value in an isolated data is greater than 192, i.e. the two high order bits are set, it must be processed by a key byte indicating a run length of one.

## **2.5 Interactive Image Acquisition**

The use of a video camera and frame grabber board allows the microscope image to be scanned with precision, speed and in real-time. Interactive image acquisition allows instantaneous examination of the rock thin section in plane or cross polarized light or using any variety of illumination techniques to bring out special features of interest in the image. Depending on subsequent feature/data analysis the microscope magnification is selected during image acquisition to generate an image with appropriate resolution. The interactive image acquisition eases the task of subsequent image analysis/measurements required for extraction of geological information from the microscope image.

## Chapter 3

# Image Processing

A digitized petrographic image is a numerical representation of a rock thin section. The extraction of geological information from such a digital image requires that features such as mineral grains, inclusions, pore spaces etc. be identified and that their edges defined. The identification of features requires manipulation of the digital images which is referred to as image processing. The extraction of meaningful information about the features in an image is termed image analysis.

Image processing usually involves three sequential operations: digital filtering, image segmentation and feature extraction (Sklansky 1978, Inoué 1987). Digital filtering removes unwanted frequencies, features etc. to prepare the raw image for digital processing. Image segmentation is the partitioning of the grey level image into a number of feature regions by defining their boundaries, this results in a segmented image. Feature extraction refers to the techniques used to select features of interest from the segmented image for subsequent image analysis.

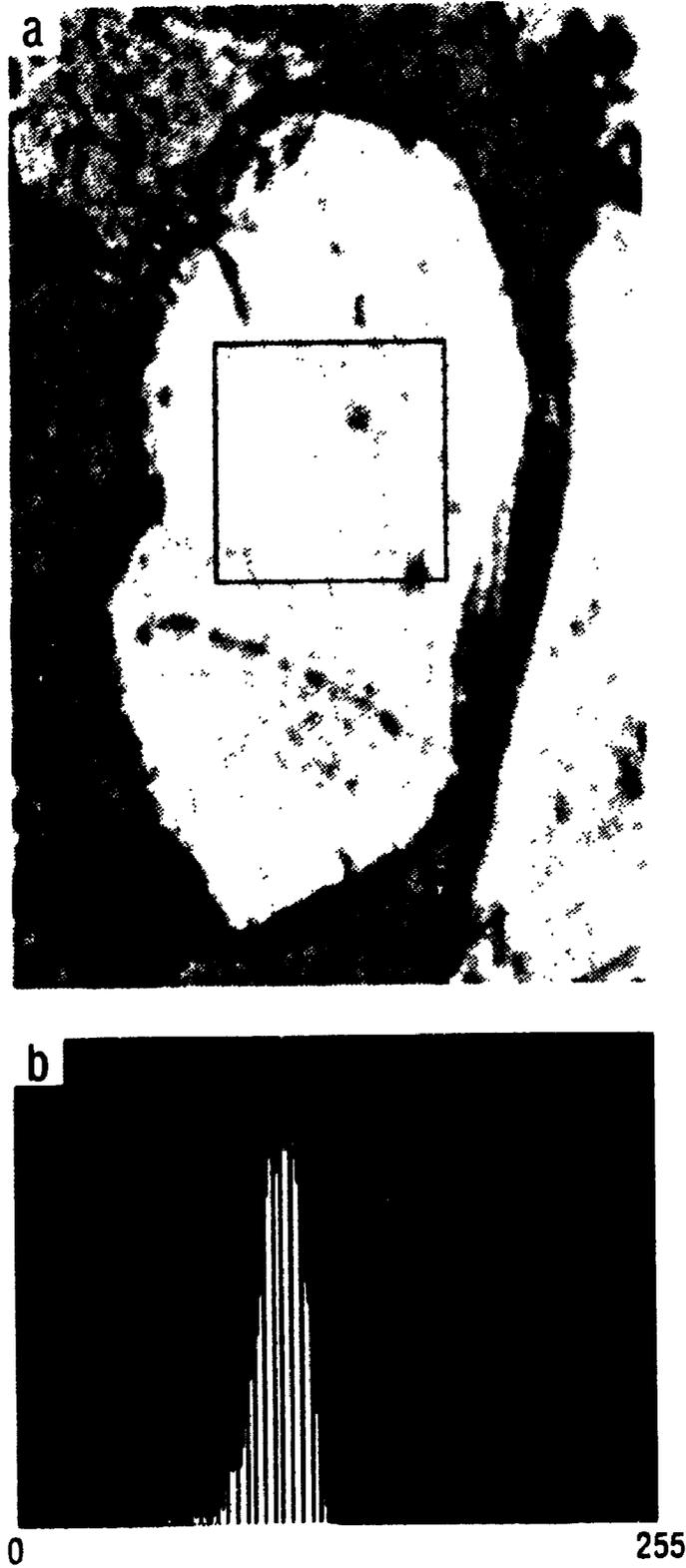
### 3.1 Digital Filtering

The digitized petrographic images from rock thin sections show variations of grey levels within individual mineral grains due to the inherent optical properties of the minerals, such as undulose extinction etc. In addition, the grey levels incorporate noise due to a variety of factors including uneven illumination of the thin section during image acquisition, or due to random electronic noise. The random electronic noise associated with the imaging system hardware used here has been determined using the grey level distribution recorded for a uniform grey area, it has a standard deviation of 1.5.

Digital filtering involves the use of mathematical transformations that reduce inherent grey level variations within mineral grains and the noise present in the image. This also enhance the intensity/texture discontinuities between mineral grains. The selection of an appropriate digital filter depends on the type of grey level variation present in the image. The variations of grey levels within mineral grains can be approximated as additive Gaussian noise (Starkey and Samantaray 1992). Figure 3.1 displays a quartz grain and the variation of grey levels within it.

Digital filtering techniques may be divided into two broad categories: frequency-domain methods, which are based on modifying the Fourier transform of an image, and spatial-domain methods, which are based on direct manipulation of the pixels in an image (Gonzalez and Wintz 1987). Spatial domain methods are used in the present study.

Spatial-domain filters, can be subdivided into global and local techniques. In the former, the raw image is processed globally and the whole, or a large section,



**Figure 3.1 : (a) Microscope image of a quartz grain observed in plane polarized light. The edges of the figure in x-direction represents 0.3 millimeter. (b) Frequency distribution of the grey levels present inside the marked area of the image shown in (a).**

of the image is correlated to a mathematical function to obtain a smoothed image (Rosenfeld and Kak 1976, Andrews and Haunt 1977). The local techniques use local operators in which the filter output at a specific pixel is a function of the input values of the pixels within a selected neighbourhood, for instance, the eight surrounding nearest neighbours. These latter techniques tend to have shorter computation times, since only a small number of input pixel values are considered for each output pixel. Local filters are used in the present study. Most of these filters exist in the literature only as theoretical models and the performances of these have been evaluated only on artificial images (Mastin 1985; Wu, Wang, and Lu 1992). As a part of the present study these local filters are evaluated on petrographic images (Starkey and Samantaray 1992), the results are presented in Appendix 1. The suitability of the filters for petrographic images are assessed using the following criteria:

1. Effectiveness in smoothing - meaning the ability to reduce the grey level variation resulting from noise and due to inherent variation in the optical properties of the mineral grains.
2. Preservation of subtle details of clusters of several pixels and of linear features a few pixels wide.
3. Immunity from shape distortion - in particular the filter must not introduce significant distortion in the shapes of image features.
4. Retention of intensity step and ramp edges between adjacent regions of uniform, but different, intensities.
5. Removal of sparsely distributed, sharp, spot noise consisting of one or two pixels.

## 6. Computational speed.

Both single and multi-level filters were evaluated. The single level filters which appear most commonly in the literature are the Mean, Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma, K-Nearest Neighbour Average (KAVE), Nearest Neighbour Median (NNM) and Weighted Median (WM) Filters. The popularity of these filters may be attributed to several factors, including ease of implementation and conceptual simplicity. Multilevel filters combine the output of basic subfilters and include the Multistage Median, Max/Min Median, Finite Impulse Reduction Median Hybrid (FMH), In Place Growing FMH (IPG-FMH) and Morphological (Two Dimensional Closing Opening) Filters. Multilevel filters have received considerable attention recently because of their ability to smooth images while preserving texture.

### 3.1.1 Single-level Filters

Single-level filters process the image data in a local neighbourhood, or window, which is usually square and the sides of which are an odd number of pixels. The window is moved over the entire image and the output from the filter at the central pixel position is used to create the filtered image. The different filters vary in the manner in which the neighbouring pixels are selected and in the statistical analysis which is applied to the input values.

#### Mean Filter

With the Mean Filter the average of the pixel values within a square window becomes the filter output at the central pixel position. This filter reduces the effect of noise

by reducing the grey level variation in the image. However, the edges between image regions become blurred.

### **Median Filters**

Median filtering was first used by Tukey (1977) in time series analysis and later applied to image smoothing by Pratt (1978). The central pixel value is replaced by the median value within the window. The assumptions behind this filter are that a constant neighbourhood is a region in which the majority of the grey levels represent signal with zero noise and that an edge is a monotonically rising or falling sequence of pixel values which separates two constant neighbourhoods. The Median Filter can be applied repeatedly using the previously filtered image as input. In this case, for a given window size, the filter values converge on to what has been termed the root structure (Gallagher and Wise 1981).

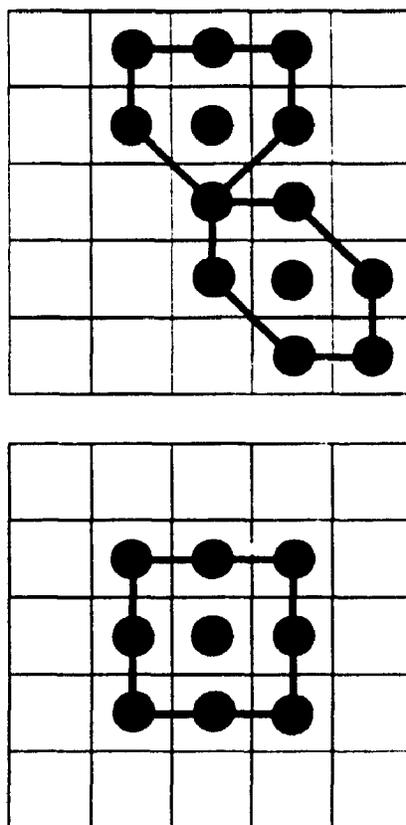
The Recursive Median Filter is a modification of the Median Filter in which the grey levels of the image are replaced by the output of the filter as the process proceeds. Therefore, at any one time, approximately half the values within the window are previously filtered values and the remainder are the original grey levels. The recursive operation determines the root structure of an image in a single pass. However, this root is different from that obtained by repeated passes of the nonrecursive Median Filter although typically they are very similar (Arce et.al. 1986).

Median filters, which are typically based on square windows, are known to cause edge shifts (Hodgson et.al 1985, Arce et.al. 1986). Further, the corners of image regions may be rounded off and artifacts consisting of patches of pattern may be produced in the filtered image. These artifacts persist even if the square win-

dow is replaced by a circular, or near circular, approximation in which the pixel values are weighted according to their distance from the central pixel. These problems are alleviated by the Separable Median Filter (Narendra 1981), which consists of two one-dimensional filters, applied in a two-pass operation. The rows of the two-dimensional image are filtered first by a horizontally oriented one-dimensional median filter to produce an intermediate image. The columns of this image are then filtered by a vertically oriented one-dimensional median filter. The Separable Median Filter can also be applied recursively. The root structure produced by a single pass of the recursive filter is similar to, but different from, that of the nonrecursive filter. The recursive filter has been shown to be better at noise suppression and edge preservation (McLoughlin and Arce 1987) and is therefore evaluated in the present study.

### **Nagao and Matsuyama Filter**

Nagao and Matsuyama (1979) proposed a filter based on a 5 by 5 pixel window. The central pixel value is replaced by the average grey level of the most homogenous subregion from among nine subregions, see figure 3.2. A homogenous neighbourhood is defined as one which does not contain a sharp edge. If an area contains a sharp edge, the variance of the grey levels in that area will be large. Therefore, the variance is used as a measure of the homogeneity of the subregions and the central pixel is replaced by the mean of the subregion which has the minimum variance. This allows noise to be reduced without blurring sharp edges since averaging is not applied to a subregion which contains an edge.



**Figure 3.2 : The nine possible subregions for the Nagao and Matsuyama Filter. (a) One example from each two sets of four subregions. (b) The ninth subregion. The black circles indicate the values used to compute the mean and standard deviation for each of the subregions shown.**

### Gradient-Inverse Filter

The Gradient Inverse Filter proposed by Wang et.al. (1981) assumes that the variation of grey levels within a homogenous region of an image is less than that between different regions, and that the edges between regions are sharply defined. The central pixel value is replaced by the weighted average of all the grey levels inside the window. The weighting coefficients are obtained by computing the inverse gradients between the center and its neighbouring pixel values and normalizing them by dividing by the sum of the inverse gradients in the window. Therefore, pixel values having a greater absolute difference from the central pixel have lower weighting coefficients and contribute less to the output central pixel value. For a square window, where the length of the side is  $2n + 1$ , the output value at the central pixel,  $Y(i, j)$ , is identified by

$$Y(i, j) = \sum_{p=-n}^n \sum_{q=-n}^n W(i + p, j + q) \times X(i + p, j + q) \quad (3.1)$$

Where either

$$W(i + p, j + q) = \frac{1}{2}$$

for  $p = 0$  and  $q = 0$  or

$$W(i + p, j + q) = \frac{1}{2} \left\{ \sum_{V_{(i,j)_n}} (i, j; p, q) \right\}^{-1} \sigma(i, j; p, q)^{-1} \quad (3.2)$$

for  $p$  and  $q = -n, \dots, 0, \dots, +n$ ; and  $p$  and  $q$  not 0 simultaneously.

$V_{(i,j)_n}$  denotes all the pixels within the window.  $\sigma(i, j; p, q)^{-1}$  represents the inverse of the absolute gradient of the neighbouring pixel with the grey value  $X(i, j)$ .

### Lee-Additive Filter

The Lee-Additive Filter (Lee 1980,1981) assumes that the correct value of the central pixel is the mean of the values of all pixels within the window and that the departure of the central pixel value from the mean is within the variance of all pixel values within the window. The corrected image values can be obtained from the difference between the mean and variance of the noise corrupted image and estimates of the mean and variance of the noise alone. The grey value of a pixel  $X(i, j)$ , in a noise corrupted image, is given by the following equation,

$$X(i, j) = Y(i, j) + N(i, j) \quad (3.3)$$

Where  $N(i, j)$  is the white random noise with zero mean and  $\sigma^2$  variance and  $Y(i, j)$  is the pixel value before degradation.

The estimated central pixel value,  $Y(i, j)$ , in the uncorrupted image is given by

$$Y(i, j) = \bar{Y}(i, j) + k(i, j) \{X(i, j) - \bar{Y}(i, j)\} \quad (3.4)$$

Where  $\bar{Y}(i, j)$  is the approximated uncorrupted image mean given by  $\bar{Y}(i, j) = \bar{X}(i, j)$ . The term  $k(i, j)$  is a gain factor which is used to smooth or enhance the image and is defined by

$$k(i, j) = \frac{Q(i, j)}{\{Q(i, j) + \sigma_1^2\}} \quad (3.5)$$

where

$$Q(i, j) = E \left\{ \left\{ X(i, j) - \bar{X}(i, j) \right\}^2 \right\} - \sigma_1^2 \quad (3.6)$$

$\sigma_1^2$  is an estimate of the additive noise variance.

The application of this filter requires estimates of the mean and variance of the pixel values within the window and of the global noise variance. In the present

evaluation the global noise variance is derived from an arbitrarily selected sensibly 'flat' intensity region in the displayed image.

### **Sigma Filter**

The Sigma Filter (Lee 1983) is suggested by the Gaussian distribution, and its standard deviation, which provides an estimate of the probability of a grey value falling within a certain multiple of the standard deviation from the mean, which is the current value of the central pixel within a window. Here, the central pixel value is replaced by the average of those pixel values inside the window which lie within a fixed sigma range of the central pixel grey level. Assuming that the noise has a Gaussian distribution and the a priori mean is the grey level of the central pixel then 95.5% of the pixel values will lie within two standard deviations ( $2\sigma$ ) of the mean. Pixel values outside this range are assumed to belong to a different population. The algorithm consists of computing the two sigma range for a local neighbourhood, and replacing the central pixel by the average of the values which are within that two sigma range of the central pixel value. Alternatively, the median value may be used in place of the average but the resulting output image is almost identical.

The two sigma average does not smooth the sharp spot noise of one or two pixels. In the present evaluation, the minimum number of pixels required to calculate the two sigma average within a window is assumed to equal  $n$  for a square window  $2n + 1$  on a side. If the number of pixels within the intensity range of two sigma is less than or equal to  $n$ , the two sigma average is replaced by the average of the center pixel's eight nearest neighbours.

### **K-Nearest Neighbour Averaging Filter**

The K-Nearest Neighbour Averaging Filter was suggested by Davis and Rosenfeld (1978). This filter relies on the high correlation which must exist between the grey levels of pixels belonging to the same population within a local area. The central pixel value,  $Y$ , of the window is replaced by the average grey level of the  $k$  nearest neighbours with grey levels closest to that of  $Y$ . This filter can be used iteratively, changing the window size and value of  $k$  for successive iterations.

### **Nearest Neighbour Median Filter**

The Nearest Neighbour Median Filter was first applied to image smoothing by Itoh et.al. (1988). For a given window size of  $2n + 1$  the pixel values in the window are ranked in ascending order. A number of nearest neighbours,  $k$ , is selected equal to  $2m + 1$ , where  $m \leq n$ . In the present evaluation, using a 5 by 5 window,  $k$  is chosen as 11. Within the rank ordered array,  $R$ , the position of the central pixel value,  $c$ , is located (Samantaray and Starkey 1993). The output of the NNM filter,  $Y(p)$ , can be defined as follows (Asano et.al. 1990) depending on the location of  $c$ :

$$Y(p) = \left\{ \begin{array}{l} \text{Center Pixel Value;} \\ \text{where } \{m + 1\} < c < \{(2n + 1) - (m + 1) + 1\} \\ \\ R\{m + 1\}; \\ \text{where } 1 \leq c \leq (m + 1) \\ \\ \\ R\{(2n + 1) - (2m + 1) + 1\}; \\ \text{where } \{(2n + 1) - (m + 1) + 1\} \leq c \leq \{2n + 1\} \end{array} \right\} \quad (3.7)$$

### Weighted Median Filter

The Weighted Median Filter was first applied to image smoothing by Brownrigg (1984,1986). Non negative multipliers are applied to the pixel values in the window and the median value is selected as the output. In this evaluation, using a 5 by 5 window, the multipliers are taken as unity except for that of the central pixel which is set at 15.

### 3.1.2 Multilevel Filters

Several multilevel filters based on the Median Filter have been proposed which combine the output of basic subfilters to match the structure spanned by the filter's window (Nieminen et.al. 1987) . They involve the use of the 4 subwindows,  $W_{(1..4)}$ , shown in figure 3.3. Such subwindows are unidirectional since the subwindows span

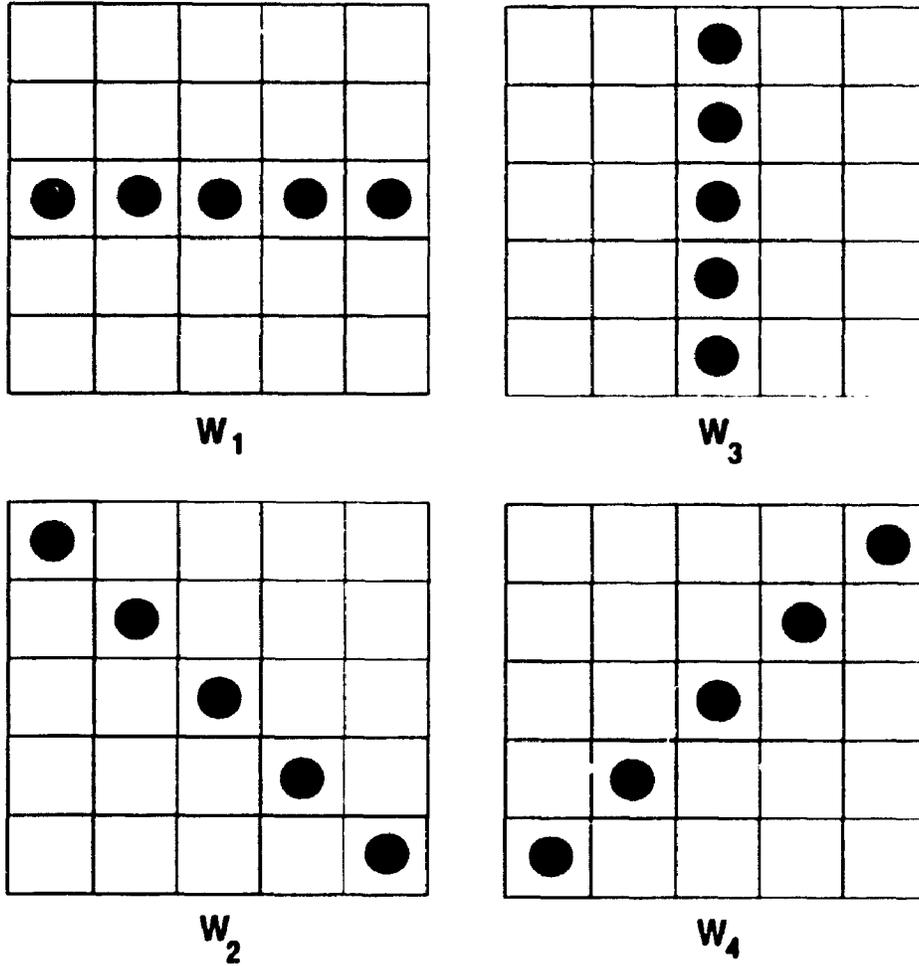


Figure 3.3 : The four subwindows used by unidirectional multilevel filters. The black circles indicate the values considered by each individual unidirectional filters.

one direction only. For the application of unidirectional multilevel filters, the median values,  $Z_{(1..4)}$ , of the subwindows are required (Arce and McLoughlin 1987). For a recursive operation the grey levels of the image are replaced by the output of the filter as the process proceeds.

### Multilevel Median Filter

The output for the Multilevel Median Filter is defined as follows (Nieminen and Neuvo 1988, Arce and Foster 1988)

$$Y(i, j) = \text{median}[Y_{w_1, w_3}(i, j), Y_{w_2, w_4}(i, j), X(i, j)] \quad (3.8)$$

where

$$Y_{w_1, w_3}(i, j) = \text{median}[Z_1, Z_3, X(i, j)]; \quad (3.9)$$

and

$$Y_{w_2, w_4}(i, j) = \text{median}[Z_2, Z_4, X(i, j)]; \quad (3.10)$$

where  $X(i, j)$  is the central pixel value and  $Z_{(1)..(4)}$  are the median values in the four subwindows (figure 3.3 and Arce and McLoughlin 1987). In the present evaluation a recursive operation of the Multilevel Median Filter is used.

### Max/Min Median Filter

A modification of the Multilevel Median Filter was proposed by Wang and Wang (1990) as the Max/Min Median Filter. The output for the filter is

$$Y(i, j) = \left\{ \begin{array}{l} T_1(i, j); \\ \text{if } |T_1(i, j) - T_0(i, j)| \geq |T_2(i, j) - T_0(i, j)| \\ T_2(i, j); \\ \text{if } |T_1(i, j) - T_0(i, j)| < |T_2(i, j) - T_0(i, j)| \end{array} \right\} \quad (3.11)$$

where  $T_0(i, j)$  is the median of all the pixel values within the window.  $T_1(i, j)$  and  $T_2(i, j)$  are respectively the maximum and the minimum median values in all the subwindows (cf. figure 3.3). In the present analysis a recursive Max/Min Median Filter is used.

### FIR-Median Hybrid (FMH) Filter

The output for the FMH Filter is defined as follows (Nieminen et.al.1987, Arce and Foster 1988,1989)

$$Y(i, j) = \text{median}[\bar{Y}_{w_1, w_3}(i, j), \bar{Y}_{w_2, w_4}(i, j), X(i, j)] \quad (3.12)$$

where

$$\bar{Y}_{w_1, w_3}(i, j) = \text{median}[\hat{Z}_1, \hat{Z}_3, X(i, j)]; \quad (3.13)$$

and

$$\bar{Y}_{w_2, w_4}(i, j) = \text{median}[\hat{Z}_2, \hat{Z}_4, X(i, j)]; \quad (3.14)$$

where  $\hat{Z}_p = \text{median}[Z_{pa}(i, j), Z_{pb}(i, j), X(i, j)]$ ,  $1 \leq p \leq 4$ , and where  $Z_{pa}$  and  $Z_{pb}$  are the averages of the values in each of the two parts of the subwindow  $W_p$ , separated by the center pixel (cf. figure 3.3).  $X(i, j)$  is the central pixel value. In the present evaluation the filter is applied recursively by replacing  $Z_{pa}$  by the filter output from the previous window operation.

### In-Place Growing FMH (IPG-FMH) Filter

The IPG-FMH filter was first applied to one dimensional signals by Wichman et.al. (1990). In the present study the filter has been adapted as a IPG-FMH Filter for application to two dimensional images as follows :

$$Y_k(i, j) = \text{median}[\bar{Y}_{k(w1, w3)}(i, j), \bar{Y}_{k(w2, w4)}(i, j), Y_{k-1}(i, j)]; \quad (3.15)$$

where

$$\bar{Y}_{k(w1, w3)}(i, j) = \text{median}[\hat{Z}_1, \hat{Z}_3, Y_{k-1}]; \quad (3.16)$$

and

$$\bar{Y}_{k(w2, w4)}(i, j) = \text{median}[\hat{Z}_2, \hat{Z}_4, Y_{k-1}]; \quad (3.17)$$

where  $\hat{Z}_p = \text{median}[Z_{pa}(i, j), Z_{pb}(i, j), Y_{k-1}(i, j)]$ ,  $1 \leq p \leq 4$ , and where  $Z_{pa}$  and  $Z_{pb}$  are the averages of the values in each of the two parts of the subwindow  $W_p$ , separated by the center pixel (cf. figure 3.3). The number of growth cycles performed at each location is indicated by  $k$ .

The filtered value  $Y(i, j) = Y_{k=m}(i, j)$ , where  $k = 1, \dots, m$ . Where  $k = 1$  then  $Y_{k-1}(i, j) = X(i, j)$ . The size of the subwindow increases with  $k$ . For a recursive operation of the filter  $Z_{pa}$  is replaced by the filter output from the previous window operation.

In the present analysis a nonrecursive IPG-FMH filter is used and  $m$  is set to 2. The window sizes used are 5 by 5 and 7 by 7 with  $k$  equal to 1 and 2 respectively. The output from eq. 3.15, is as follows

$$Y(i, j) = \text{median}[\bar{Y}_{2(w1, w3)}(i, j), \bar{Y}_{2(w2, w4)}(i, j), Y_1(i, j)],$$

where

$$Y_1(i, j) = \text{median}[\bar{Y}_{1(w1, w3)}(i, j), \bar{Y}_{1(w2, w4)}(i, j), X(i, j)]$$

### Two Dimensional Close-Open Filter

The Two Dimensional Close-Open Filter is closely related to ranked-order based filters (Maragos and Schafer 1987). For a window with side length  $2n + 1$ , each subwindow  $W_p$  is partitioned into  $n + 1 = k$  overlapping subsets,  $S_{p,k}$ , of  $n + 1$  consecutive elements. For a 5 by 5 window there are therefore 3 subsets derived from each of the 4 subwindows for a total of 12. Grey scale opening is performed by replacing the central pixel value in the window by the maximum value among the minima in the subsets. This generates an intermediate image on which grey scale closing is performed by replacing the central pixel value in the window by the minimum value among the maxima in the subsets. This filter is known to preserve image geometry and image details.

#### 3.1.3 Evaluation of the Filters

The filters described above are applied to computer simulated images and real images from rock thin sections (Starkey and Samantaray 1992). The filters are applied both as single pass and multiple pass filters using a 5 by 5 pixel window. For single pass operation, all the single level filters except the Nagao and Matsuyama, NNM and

WM Filters are also applied with varying window sizes. Multiple passes consisted of three iterations. For the KAVE Filter the value of ' $k$ ' represents approximately 75%, 50%, and 25% of the neighbourhood elements in successive iterations. The successive passes of the Sigma Filter are applied with intensity intervals  $2\sigma$ ,  $\sigma$ ,  $\sigma/2$ . The results of the evaluation are presented in Appendix 1 and summarized below:

For a single pass operation the RSM Filter most effectively reduces the noise variance in a uniform grey region and retains edges between adjacent image regions. This filter also does not create any significant distortion in the shape of an image region and it effectively smooths sharp spot noise. Thus the RSM Filter meets most of the desirable criteria identified previously. However, it does not sharpen ramp edges, it eliminates small linear features (i.e. those with a breadth less than half the window width), destroys image texture and, since it is a recursive filter, the output depends on the direction in which it is applied. Increasing the window size does not improve the performance of the RSM Filter, the data presented in Appendix 1 suggest that a 5 by 5 pixel window is optimum. The Nagao and Matsuyama Filter is best in sharpening ramp edges with a single pass.

For the preservation of image detail and texture multiple pass applications of the IPG-FMH and Sigma Filter perform best. Not only do they reduce noise in uniform grey areas and retain the edges between adjacent grey areas as well as the RSM Filter, they also preserve linear features and image texture. However, the computation time required for the iterative operation of these filters is significantly greater than that required for the single pass RSM Filter. The Sigma Filter is used here to process the petrographic images because of its superior performance.

## 3.2 Image Segmentation

The filtered image must be segmented to identify regions that have the same properties and to separate dissimilar regions. Image segmentation usually involves the identification of the boundaries in an image that separate one image area from another. The two main approaches to image segmentation of an image are region formation, in which the properties of areas are analyzed to determine whether to merge or split them into regions, and boundary formation, where the boundaries which delimit a feature region are found (Riseman and Arbib 1977). Segmentation algorithms for both approaches are based on one of the two basic properties of the grey values, namely, their similarity and discontinuity (Gonzalez and Wintz 1987). Approaches based on the similarity of grey values include thresholding (Sahoo, Soltani, Wong, and Chen 1988; Tsai and Chen 1992) and region growing (Haralick and Shapiro 1985). Approaches based on the discontinuity of grey values include the detection of boundaries in the image areas across which there are abrupt changes in the image properties. In the present study these latter edge detection techniques are used for segmenting the images.

Edges in petrographic images can be categorized as either intensity edges or texture edges (figure 3.4). Intensity edges arise from abrupt changes in the grey level between adjacent regions. Texture edges are boundaries between regions of the image with different patterns which are represented in the digitized image by different grey level statistics. In addition to the fundamental property that edges separate dissimilar regions, they should also display the following characteristics (Venkatesh and Kitchen 1992):

**Localization :** An edge should lie in the position which partitions the dissimilar

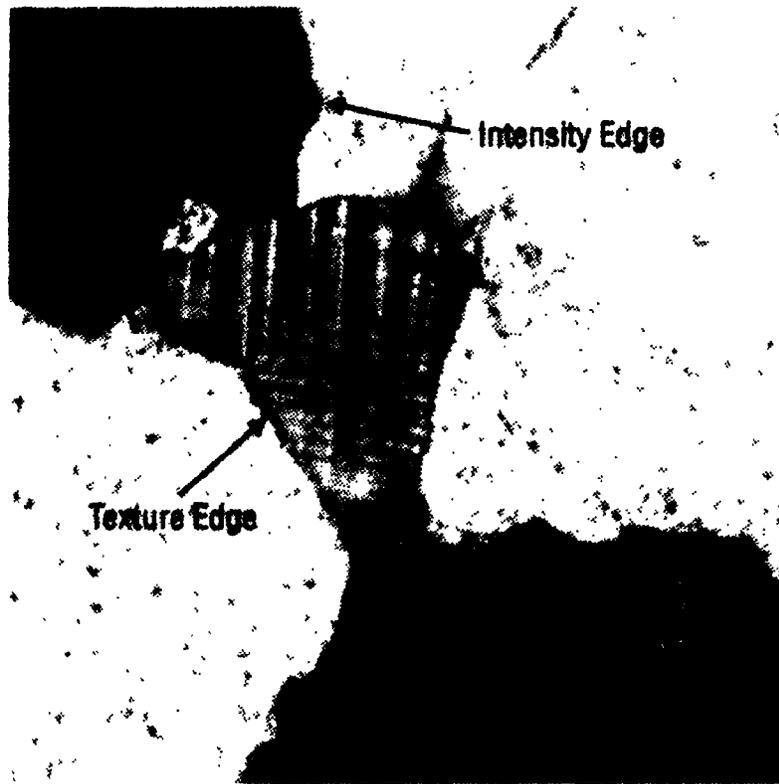


Figure 3.4 : Microscope image showing an intensity edge between two adjacent quartz grains, which show same texture, and a texture edge between a quartz grain and a grain of microcline, which show different textures.

regions in the "best" possible way. An edge is best localized when it coincides with the boundary that results in the maximum degree of dissimilarity.

**Thinness :** Since edges are boundaries, they should form thin lines in the image. Ideally they should be one-pixel wide.

**Continuity :** Since most feature boundaries are continuous in nature, edges should similarly exhibit continuity. It is inevitable that some edge contours will be located in digital images that do not form closed boundaries. Hence, the constraint that edges form closed boundaries should not be imposed on the identification of edges and edges which are discontinuous should be recognized as being fragmented.

Edge detection techniques can be classified on the basis of the edge operators used as either gradient operators, or zero crossing operators (Venkatesh and Kitchen 1992). In either case the filtering operation is performed by a convolution operation using spatial masks to detect edges.

A convolution operation on an image involves a point-for-point multiplication of selected pixel values of the input image with the corresponding values in a second image known as a convolution mask or kernel. Convolution masks usually contain an odd number of rows and columns. The convolution operation involves three sequential operations, an example is illustrated in figure 3.5. In the first step, the convolution mask is overlain on the original image in such a way that the central pixel of the mask is matched with the single pixel location to be convolved in the input image, called the target pixel. Second, each pixel value in the original image is multiplied by the corresponding value in the overlying mask. Finally, the grey value of the target pixel is replaced by the sum of the products determined in the

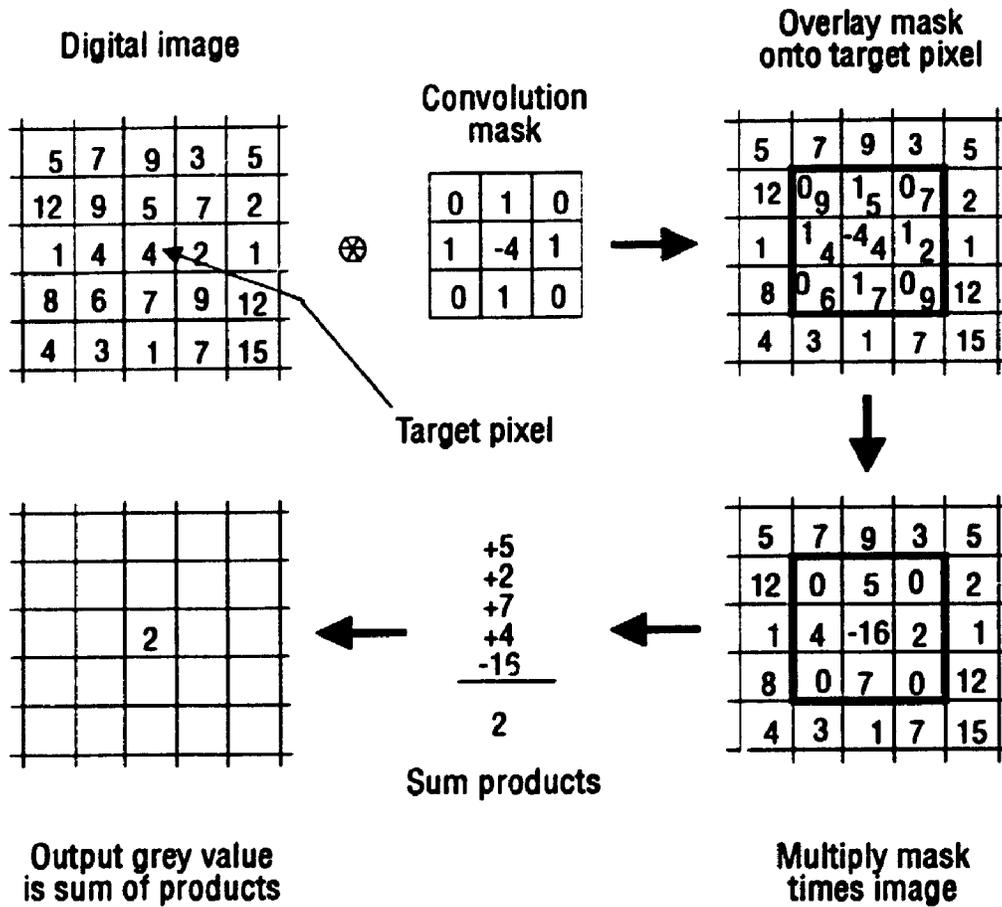


Figure 3.5 : A convolution operation.

second step. To perform a convolution on an entire image, this operation must be repeated for each pixel in the original image. The example shown in figure 3.5 illustrates a Laplacian convolution using a 3 by 3 mask.

### 3.2.1 Gradient Operators

Gradient operators are based on the assumption that edges correspond to intensity changes and therefore they compute the gradient of the intensity changes (Abdou and Pratt 1979). These operators involve computation over small windows, typically  $3 \times 3$  or  $5 \times 5$ . The use of such operators provides an edge strength (gradient) and direction to each pixel in the image. The gradient of an image at pixel location  $(x, y)$  is defined as the two-dimensional vector

$$\mathbf{G}[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (3.18)$$

The magnitude of the vector  $\mathbf{G}$  is referred to as the gradient of the edge and is denoted by  $G[(x, y)]$ , where  $G[(x, y)] = \sqrt{G_x^2 + G_y^2}$ . The direction of the gradient vector is calculated as  $\alpha(x, y) = \tan^{-1}(G_y/G_x)$ , the angle  $\alpha$  is measured with respect to the  $x$  axis.

Figure 3.6.I and 3.6.II shows two pairs of masks used to compute  $G_x$  and  $G_y$  at the center pixel of a 3 by 3 window. These masks are referred to as the Sobel (Sobel 1970) and the Prewitt (Prewitt 1970) operators respectively. These operators have been recognized to perform well at preserving edge orientation (Abdou and Pratt 1979, Kitchen and Malin 1989) and are widely used in image processing applications. Figure 3.7 displays an image of a rock thin section in plane polarized light, the Sobel edges detected in this image, using the masks shown in figure 3.6.I, are shown in

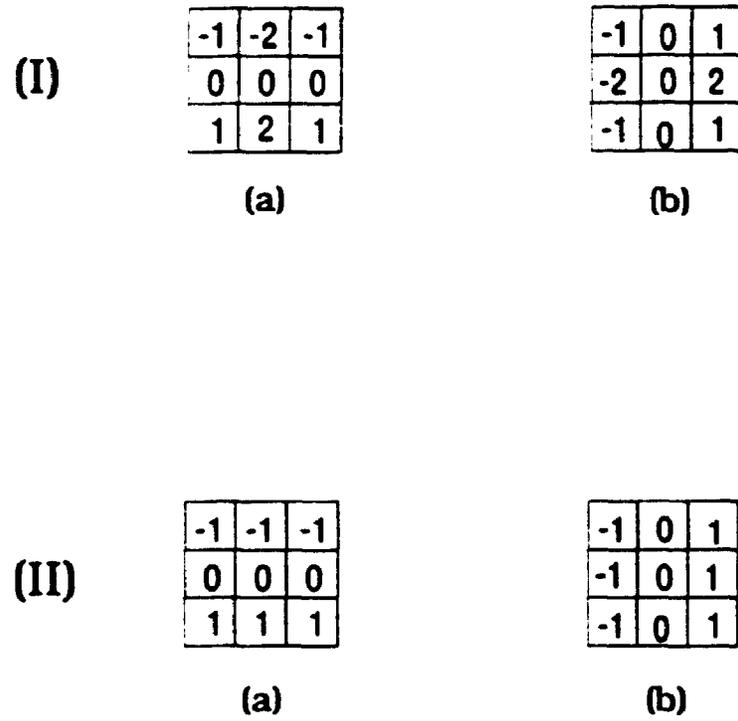


Figure 3.6 : The masks used for Sobel and Prewitt operators.

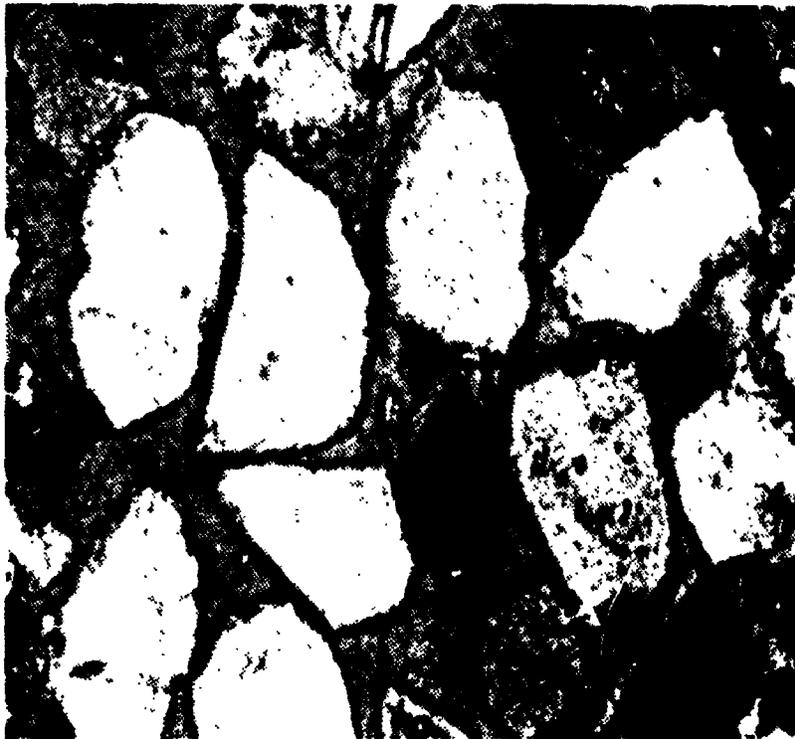


Figure 3.7 : Microscope image of a thin section showing quartz grains observed in plane polarized light. The edge of the figure represents one millimeter

figure 3.8.

The edge strength and direction can also be computed by using multiple edge templates in a local window. The masks for one such operator due to Kirsch (1971) are shown in figure 3.9. The Kirsch operator performs the derivative operation in each of four directions and the output is as follows :

$$G(x, y) = \max |M_i| \text{ and } D(x, y) = \{ i \mid \text{such that } M_i \text{ is max } \}.$$

Where  $G(x, y)$  is the estimate of the gradient of the edge at every pixel location  $(x, y)$  and  $D(x, y)$  is the direction quantized to  $45^\circ$  intervals. If the pixel at  $(x, y)$  is within a uniform area,  $G(x, y) = 0$  and the orientation of an edge is meaningless. Figure 3.10 shows the edges detected by applying the Kirsch operator with the unweighted mask illustrated in figure 3.9 to the image shown in figure 3.7.

All three gradient operators discussed above, the Sobel, Prewitt and Kirsch, have been shown to yield similar results when used to detect edges (Abdou and Pratt 1979). They produce an edge image where each edge pixel is represented by a grey level, indicating the edge strength at that pixel, and an edge direction. Ideally, gradient operators should identify only pixels which lie on the boundaries of the features. However, the detected edge pixels seldom characterize a boundary completely because of noise, breaks in the boundary, and other effects that introduce spurious intensity discontinuities (see figures 3.8 and 3.10).

The original grey level image may have ramp edges in place of step edges, due in part to normally distributed noise (see figure 3.11.a and b). In some cases the application of local area operators produces multiple edges instead of a single one (figure 3.11.c). Such edge detectors do not satisfy the criteria noted previously as suitable for implementation in automated analysis. Figure 3.11.d shows the intensity



Figure 3.8 . Image showing the edges detected using the Sobel operator illustrated in figure 3.6.1 on the microscope image shown in figure 3.7

-1	0	1
-w	0	w
-1	0	1

(a)

1	w	1
0	0	0
-1	-w	-1

(b)

0	1	w
-1	0	1
-w	-1	0

(c)

w	1	0
1	0	-1
0	-1	-w

(d)

Figure 3.9 : The four separate templates of a Kirsch operator. The value of  $w$  varies depending on the type of weighting required. For unweighted  $w=1$ , for weighted  $w=2$ , and for isotropic  $w=\sqrt{2}$ .

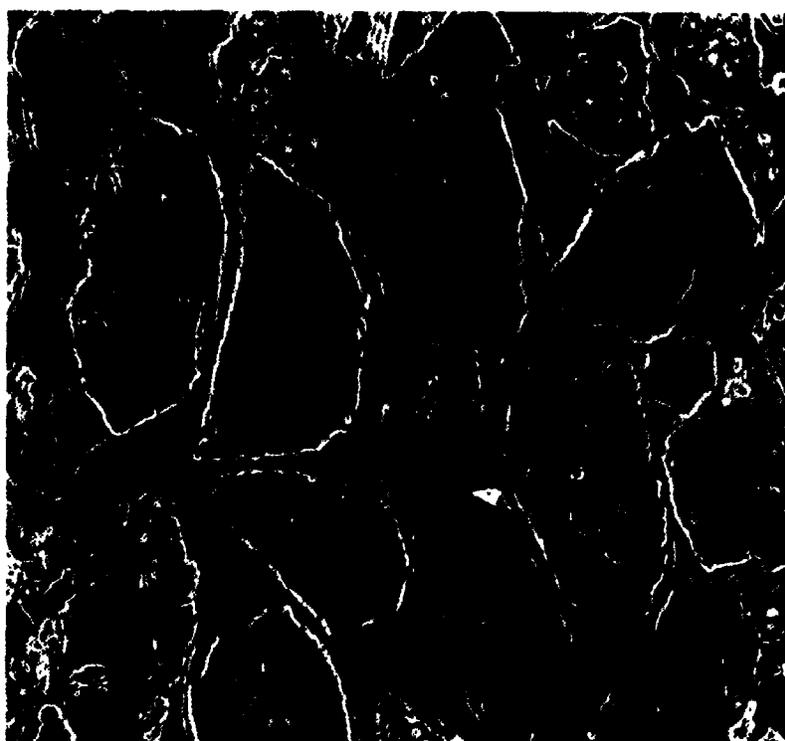


Figure 3.10 : Image showing the edges detected using a Kirsch operator with an unweighted mask illustrated in figure 3.9 on the microscope image shown in figure 3.7

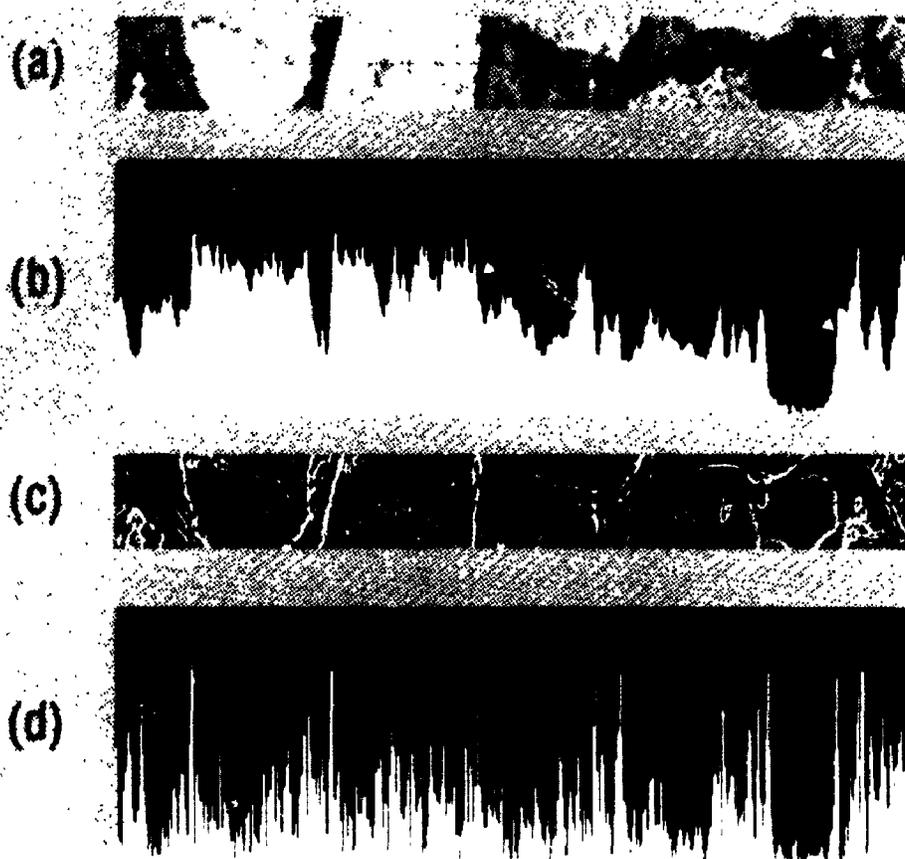


Figure 3.11 (a) Microscope image of a rock thin section observed in plane polarized light (b) Intensity profile along the marked scan line in (a). The arrows indicate the ramp edges. (c) The result of edge detection using the Sobel operator on the image shown in (a) (d) Intensity profile along the marked scan line in (c)

profile along a line through the intensity edges shown in figure 3.11.c. For subsequent analysis, single-pixel wide edges must be selected from the edges detected by the edge operators.

Desirable edges can be selected by applying a fixed or an adaptive threshold to calculated edge strengths. When a fixed threshold is used, all pixels characterized by an edge strength less than the chosen threshold are discarded. Figure 3.12 shows the edge image extracted from the image shown in figure 3.8 after thresholding at a value of 32. In adaptive thresholding the distribution of the grey levels of the edge image is considered, and the threshold is selected automatically based on some chosen criteria (Russ 1990). Figure 3.13 shows the edge image extracted from the grey level image shown in figure 3.8 after thresholding at a value of 16, which is the mode of the frequency distribution of the grey levels present in the edge image.

### **3.2.2 Edge Thinning and Following**

The operation of digital edge detection and thresholding does not necessarily yield one-pixel wide boundaries, therefore a procedure is required to thin the detected edges. A simple technique for thinning the edges is to retain only those edges whose magnitude is a local maximum along the gradient direction of the edge. This sometimes produces undesirable results, such as deletion of the pixel at the junction of two edge contours. An example of such an undesirable result is shown in figure 3.14. The intensity values of the pixels in a section of grey level image are shown in figure 3.14.a, The gradient directions and magnitudes computed for the pixels using the Kirsch operator with an unweighted mask are shown in figure 3.14.b and c and the pixels which contain local maxima are framed in figure 3.14.d. The result

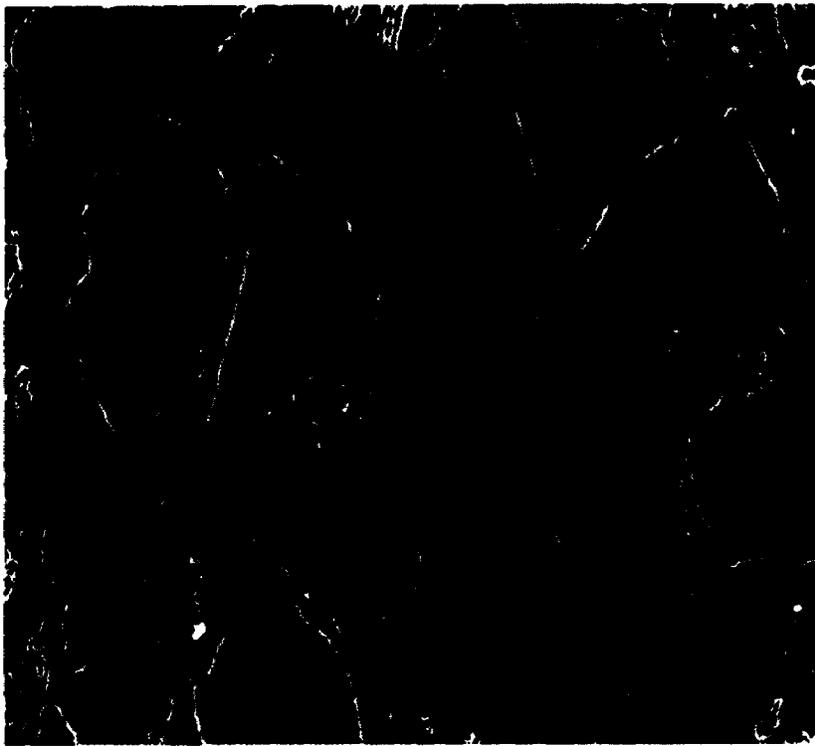


Figure 3 12 . The edges extracted from the image shown in figure 3.8 after applying a fixed threshold value of 32 to the edge strength

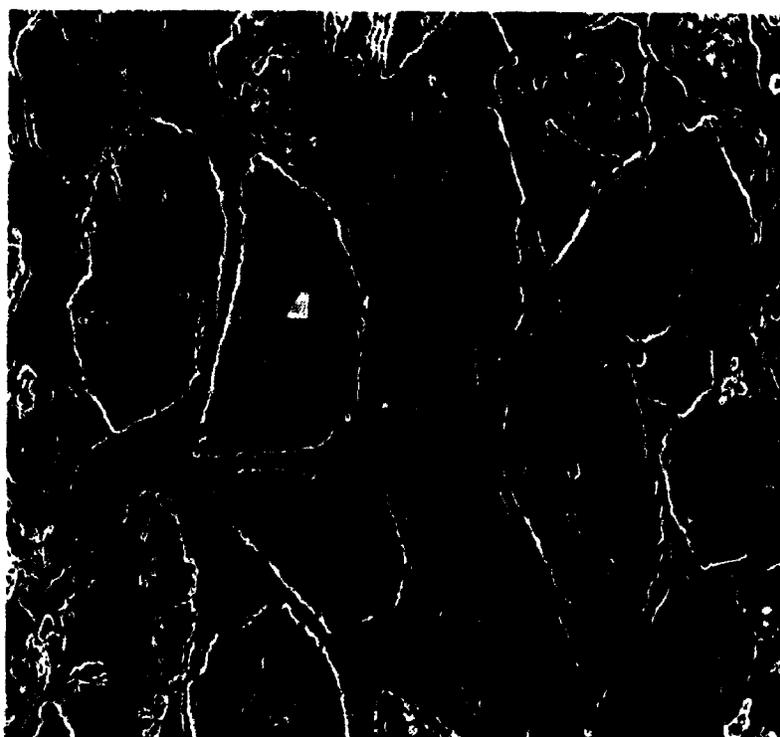


Figure 3.13 : The edges extracted from the image shown in figure 3.8 after applying an adaptive threshold (i.e. the mode of the grey level frequency distribution, 16 in this case) to the edge strength.

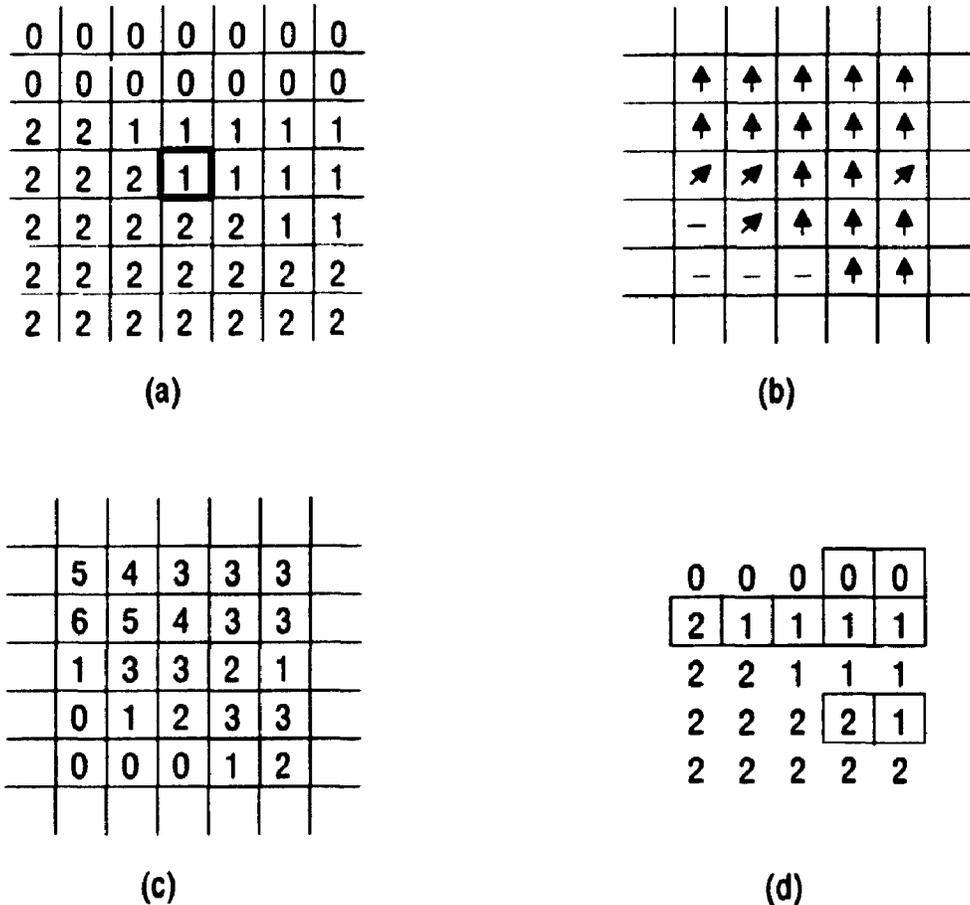


Figure 3.14 : Example of undesirable results from edge thinning on part of an image. (a) The intensity values of the pixels. (b) The gradient directions of the pixels, indicated by arrows. (c) Gradient magnitudes of the pixels computed using a Kirsch operator with an unweighted mask as illustrated in figure 3.9. (d) The framed pixels indicate the local maxima of the gradient magnitudes (see text). The central pixel, highlighted in (a), is eliminated as an edge pixel by edge thinning process.

is that the central pixel, highlighted in figure 3.14.a, is eliminated as an edge pixel by the edge thinning process. Few algorithms have been proposed to thin the detected edges and assemble those remaining into a meaningful set of feature boundaries (Nevatia and Babu 1980, Gonzalez and Wintz 1987, Lacroix 1988, Shu 1989).

In the present study the nonmaximum deletion algorithm proposed by Lacroix (1988) is applied to the edges obtained by a Kirsch operator. This algorithm assigns the likelihood of each edge pixel being an edge (LBE) as follows: Each pixel in the edge image is assigned two counters,  $v$  and  $m$ . A 3 by 1 window is moved over the edge image and centered on each edge pixel so that it is alligned along the gradient direction. The  $v$  counter of the three pixels within the window are incremented and the  $m$  counter of the pixel, or pixels, with the maximum gradient magnitude is incremented. Figure 3.15 shows how the  $v$  and  $m$  counters are incremented when the window is placed over the central pixel of figure 3.14, it also indicates when the  $v$  and  $m$  counters of the central pixel at  $(x, y)$  are incremented during the complete image scan. After the image is processed, the LBE of each edge pixel is evaluated as  $m/v$ . Pixels with  $LBE = 0$  are eliminated as edge pixels. Pixels with  $LBE = 1$  are considered as definite edge elements. Pixels that are some times, but not always a local maximum, will have  $0 < LBE < 1$ , and their status is considered subsequently with contextual information during contour following, see below. The thinned edge contours are converted into a binary edge image for subsequent analysis. At this point it is not uncommon to find edge segments that do not form a continuous outline around a feature.

Discontinuous edge segments which remain are connected into continuous boundaries using Kunt's (1982) contour following technique. The binary edge image is scanned using 3 by 3 window until a pixel is located with an  $LBE = 1$ . The matrix

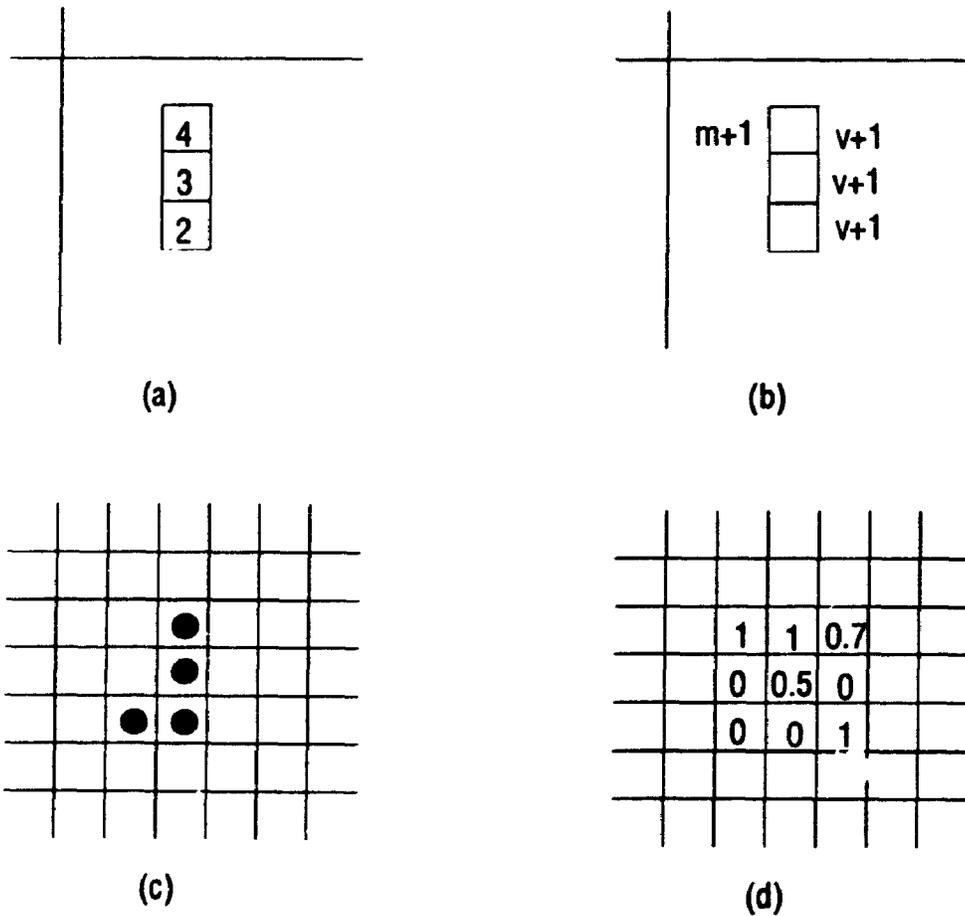


Figure 3.15 : The manipulation of the  $v$  and  $m$  counters in the 3 by 1 window used in Lacroix's edge thinning algorithm, (a) the 3 by 1 window at the central pixel of the image section shown in figure 3.14.c, (b)  $v$  and  $m$  counters of the pixels inside the window are incremented when the window is located over the central pixel, (c) the  $v$  and  $m$  counter of the central pixel are incremented when the window is centered over each of the pixels indicated by black circles, (d) LBE is evaluated as  $m/v$ .

containing the gradient directions is referenced, and left and right contours are started at right angles to the gradient direction of this pixel, This pixel is flagged as a boundary pixel at this time (cf. figure 3.16). The left and right contours are followed in turn according to the following rules. Pixels are identified as  $p$ ,  $u$  and  $d$ , signifying perpendicular, up and down relative to the gradient direction (see figure 3.16). For either contour directions, left or right, let  $max$  be the maximum of the LBEs of  $p$ ,  $u$  and  $d$ . Assuming  $max \neq 0$ , the contour is traced according to the following criteria.

1. Where the LBE of  $p = max$ ,  $p$  is the next pixel of the contour and it is flagged as the next pixel along the contour.
2. Where the LBE of  $p = max$  and the LBE of  $u$  and/or  $d$  is also  $max$ ,  $p$  is flagged as the next pixel along the contour and the LBEs of  $u$  and/or  $d$  are changed to a value less than the LBE of  $p$  but not zero, this prevents the starting of a contour at  $u$  or  $d$  but allows a contour to continue through that pixel.
3. Where either  $u$  or  $d$  has LBE =  $max$ , the one with LBE =  $max$  is the next pixel of the contour and it is flagged.
4. Where both  $u$  and  $d$  has LBE =  $max$ , the current pixel location is identified as a junction pixel and the one with the closest edge strength is chosen as the next pixel on the same physical edge which is flagged. The LBE of the other is changed to one to ensure that it will be selected subsequently as the start of a new contour.

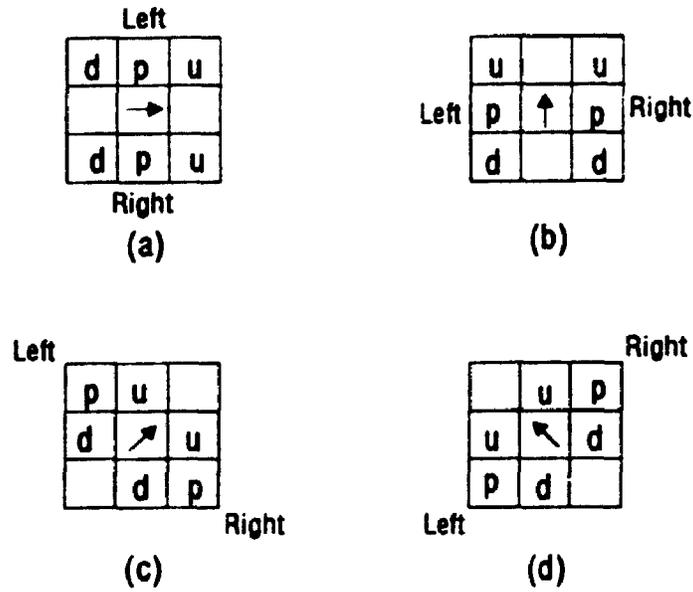


Figure 3.16 : The 3 by 3 window required to determine the four possible contour following directions in Kunt's algorithm. Left and right indicates the contour following directions. p, u and d identify the pixels which are perpendicular, up and down relative to the gradient directions indicated by the arrows.

The contour following process is applied recursively from the last pixel flagged as long as the contour can not be continued further. The success of the technique depends on the choice of a threshold for the edge strength. Figure 3.17 shows the edges extracted from the image shown in figure 3.10 with the threshold set to 8. Here, the edge contours which are at least 10 pixels long are subjected to the contour following process.

### 3.2.3 Zero-crossing Operators

Zero-crossing operators constitute the second group of edge detectors noted above. These operators locate edges at the zero-crossing where the second order derivative of the image value becomes zero (Russ 1990, p. 74, figure 4-4). The zero-crossing operators were specially designed to mimic the way in which the human visual system works, based on physiological studies. Marr and Hildreth (1980) proposed that the detection of intensity changes can be simplified by dealing with the image separately at different scales. Marr and his co-workers (1980, 1983) found that the eye uses smoothing techniques in which the area over which smoothing is applied (equivalent to the kernel size) varies. It is roughly equal to the maximum size of the features to be ignored. The edges are detected based on the finite differences of image intensities (Marr and Hildreth 1980, Hildreth 1983, Canny 1986, Fleck 1992a,b). By forming several different smoothed images, using different windows, edges of differently sized structures can be extracted from the original image.

The zero-crossing operators are applied to grey level images which have been smoothed using a 2D-Gaussian mask of the standard form (Marr and Hildreth 1980):

$$G(r) = \left(\frac{1}{2}\pi\sigma^2\right) \exp\left(-\frac{r^2}{2\sigma^2}\right) \quad (3.19)$$



Figure 3.17 : The edges, shown in red, detected by applying Lacroix's edge strategy to the edges located by the Kirsch operator from the microscope image shown in figure 3.7

where  $r$  is the distance from the center of the operator and  $\sigma$  is the space constant. In the present study, two separable 1D-Gaussian masks are used for increased ease and speed of implementation (Starkey and Samantaray, in press). The rows of the two-dimensional image are smoothed first by a horizontally oriented 1D-Gaussian mask to produce an intermediate image. The columns of this image are then smoothed by a vertically oriented 1D-Gaussian mask. The 1D-Gaussian mask is of the standard form:

$$G(x) = \left[ \frac{1}{\sigma\sqrt{2\pi}} \right] \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (3.20)$$

The value of  $\sigma$  determines the degree of smoothing of the input image and the size of the Gaussian kernel is dependent on the value of  $\sigma$ . The Gaussian kernel trails off towards zero at both ends (see figure 3.18), for the present study it is truncated when  $G(x)$  falls below 1% of its central value. The size of the kernel is chosen  $2n + 1$ , where  $n$  equals three times  $\sigma$ . The value of  $\sigma$  is selected depending on the size of the features to be extracted.

The most common zero-crossing operators are the Marr-Hildreth and Canny operators (Fleck 1992a,b). The Marr-Hildreth algorithm (Marr and Hildreth 1980) computes the second derivative of a Gaussian smoothed image and the zero-crossings of the image indicates the edges present in the image (see figure 3.19). A Laplacian operator is moved over the Gaussian smoothed image. This is known as LOG or a Laplacian of Gaussian operator. The Laplacian is equal to the sum of the directional second differences in all directions. The present implementation computes this using the following mask (Fleck 1992a)

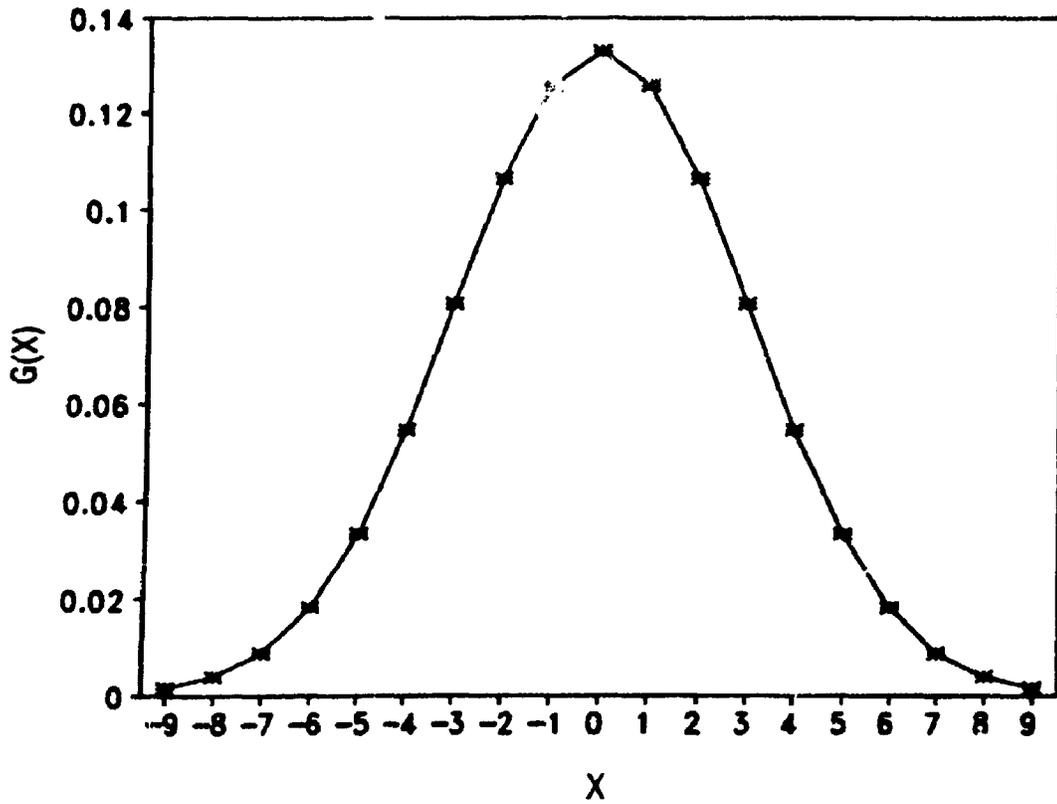


Figure 3.18 : The Gaussian kernel used to smooth the input image before applying the zero-crossing operators. The kernel is derived from eq. 3.20 with  $\sigma = 3$ .

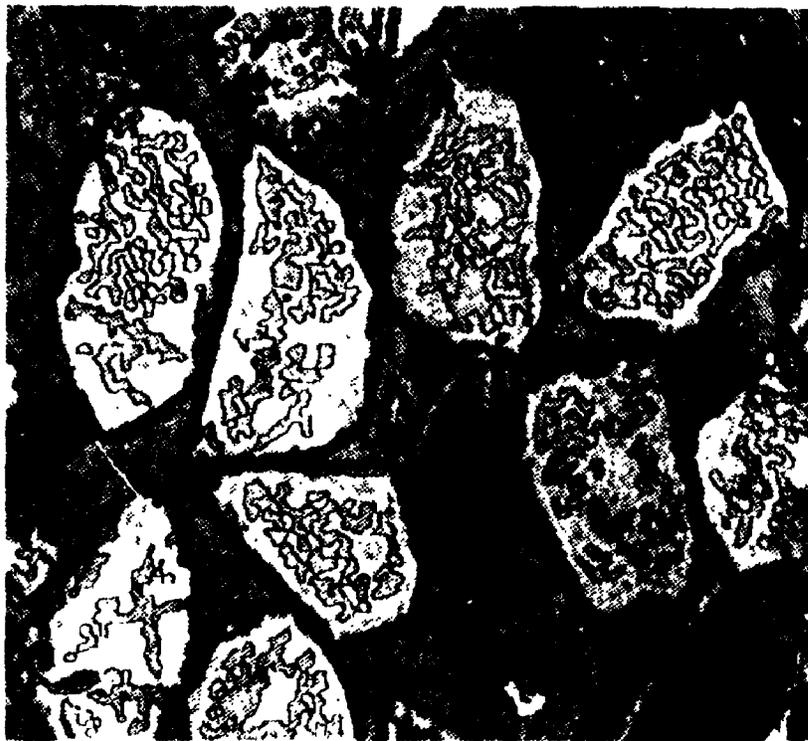


Figure 3.19 : The edges, shown in red, detected by the Marr-Hildreth operator from the microscope image shown in figure 3.7

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} \\ 1 & -4 - 2\sqrt{2} & 1 \\ \frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} \end{bmatrix}$$

The Canny algorithm (Canny 1986) improves on the Marr-Hildreth algorithm and has been recognized to operate well (Fleck 1992b). Canny's algorithm marks edges at maxima in the magnitude of the image gradient in the direction of the gradient of a Gaussian-smoothed image. The magnitude of the gradient,  $G$ , is calculated as  $G = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$ , where  $\frac{\partial f}{\partial x}$  and  $\frac{\partial f}{\partial y}$  are the first order differences computed using the mask  $[-1, 0, 1]$  in the horizontal and vertical directions. A suitable threshold is chosen to select the meaningful edges in the image.

Canny's algorithm determines the location of edges by computing the one-dimensional, second order derivative in the direction of the gradient vector,  $\mathbf{n}$ . For a two-dimensional image this can be approximated as follows (see Appendices 2 and 4):

$$\frac{\partial^2 f}{\partial \mathbf{n}^2} \approx \left[ \frac{\partial^2 f}{\partial x^2} \left(\frac{\partial f}{\partial x}\right)^2 + 2 \frac{\partial^2 f}{\partial x \partial y} \frac{\partial f}{\partial x} \frac{\partial f}{\partial y} + \frac{\partial^2 f}{\partial y^2} \left(\frac{\partial f}{\partial y}\right)^2 \right] = 0 \quad (3.21)$$

where the second order derivatives,  $\frac{\partial^2 f}{\partial x^2}$  and  $\frac{\partial^2 f}{\partial y^2}$ , are computed using the mask  $[1, -2, 1]$  in both the horizontal and vertical directions.  $\frac{\partial^2 f}{\partial x \partial y}$  is approximated by

applying the mask  $\begin{bmatrix} -1 & 1 \\ 0 & \\ 1 & -1 \end{bmatrix}$ . The first order differences are  $\frac{\partial f}{\partial x}$  and  $\frac{\partial f}{\partial y}$  as

before. Figure 3.20 shows the edge pixels obtained from the image shown in figure 3.7, edges are located at the zero crossing.



Figure 3.20 : The edges, shown in red, detected by the Canny operator from the microscope image shown in figure 3.7

The Canny algorithm satisfies most of the criteria recognized as desirable for an edge detector by locating one-pixel wide edges at the zero-crossing. By changing the value of sigma, edges for different size features present in the image can be extracted. Further, by applying a suitable threshold to the gradient magnitude, undesirable edges can be eliminated. Consequently, the Canny edge finding operator is used in the present image analyzing system.

### **3.2.4 Segmentation of Sequential Images**

The detection of boundaries in digital images obtained from a polarizing microscope requires special techniques beyond the usual filtering and edge segmentation of images which have been discussed above. This is because of the nature of petrographic images in which observations in both plane and cross polarized light may be necessary to identify the features of interest. Further, in plane polarized light, many minerals are colourless and only some of those which are coloured show variations of colour due to pleochroism and therefore show boundaries where they are in contact. Similarly, between crossed polarizers, the interference colours displayed by adjacent mineral grains may be similar at some orientations of the thin section relative to the planes of polarization of the microscope. Thus, to aid in the identification of boundaries, thin sections must be examined at different orientations relative to the plane of polarization of the light in the microscope. In petrography this is normally accomplished by examining the thin section alternately in plane polarized and cross polarized light and by rotating the thin section on the microscope stage. For computer based petrography the same effect is achieved by capturing a sequence of images in which the illumination and orientation of the polarizers is changed while the thin section remains stationary (Starkey and Samantaray, in press). Examples

of two such images are shown in figure 3.21.

The two storage pages available in the frame memory are used to process a sequence of images captured from the thin section. For each image the configuration of the polarizers can be changed so that images are obtained in plane or cross polarized light and with different orientations of the planes of polarization. Registration between the individual images is preserved because the thin section is not rotated. The sequential images are captured on the first page of frame memory and segmented by applying Canny's algorithm (see figure 3.22). The resulting edge images are accumulated into a composite image on the second page of frame memory which represents a synoptic image of all grain boundaries detected (figure 3.23, note that all visually apparent edges are not present on this synoptic image, more images must be accumulated for all edges to be detected).

The speed of acquisition of the sequential images by the frame grabber allows examination of thin sections in real-time. The technique is flexible enough to permit the use of the variety of illuminations required to bring out special features of interest in the image. Further, where an individual image shows both intensity and texture edges, different edge detectors may be applied to the same, or different, images to extract the different edge types separately and these can be accumulated on the synoptic image.



Figure 3.21 Two images captured from one microscope field of view of a quartzite between crossed polarizers. The orientations of the planes of polarization differ by 45 degrees in the two images. The arrows indicate boundaries which are not prominent at one or the other of the polarizer orientations.

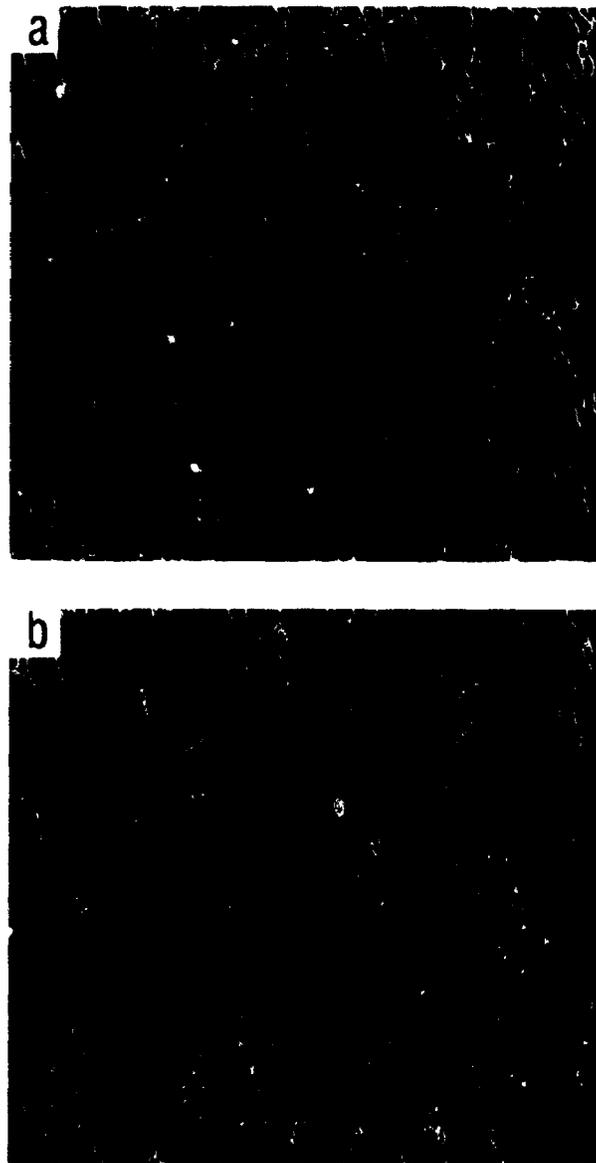


Figure 3.22 : Images of the edges obtained by applying the Canny operator to the two images shown in figure 3.21

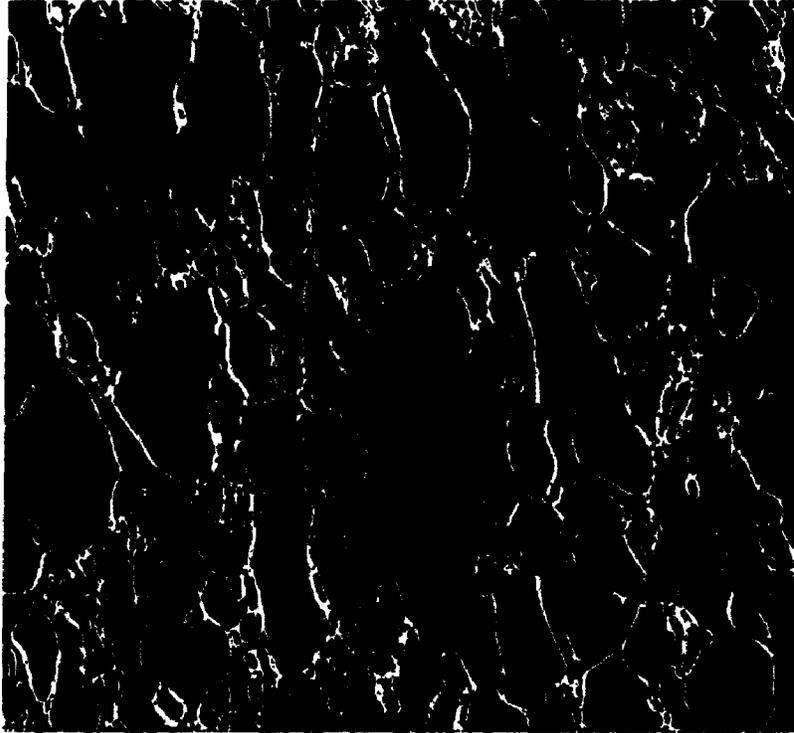


Figure 3.23 . Composite grain boundary image obtained by combining both of the edge images shown in figure 3.22.

### 3.3 Binary Image Editing

Despite the application of filtering and edge segmentation techniques the edges of all the features of interest present in the image may still not be completely delineated. Further editing may be required and most of the editing operations are performed directly on the edge image itself. In the present study, an image editor was developed which allows the superimposition of the binary edge image on the original grey level image. The editor allows the user to write values to or delete values from pixels in the edge image using a mouse, it also includes tools to highlight breaks in edge contours and to fill regions enclosed by edge contours.

Implementation of these editing tools uses the two pages of frame memory, the external monitor of the imaging system and the video display of the micro-computer. As noted earlier, the eight bits of the individual storage locations in the frame memory can be organized in eight different planes which cover the image (see figure 3.24). The original grey level image is displayed using only the lower six bit planes of the first page of frame memory, thus the upper two bit planes are free. The binary edge image, derived from the grey level image, is displayed on bit-plane 7 of the second page of frame memory, bit-plane 6 of the second page is used for filling closed contour regions and highlighting the ends of edge contours. The binary edge image and the editing information stored on page two can be superimposed on the grey level image by copying bit-planes 6 and 7 of the second page onto bit-planes 6 and 7 of the first page. This is accomplished using the host mask register provided by the frame grabber board which protects the low order bit planes containing the grey level image. To re-display the grey level image alone the data on the upper two bit planes of page one are set to zero.

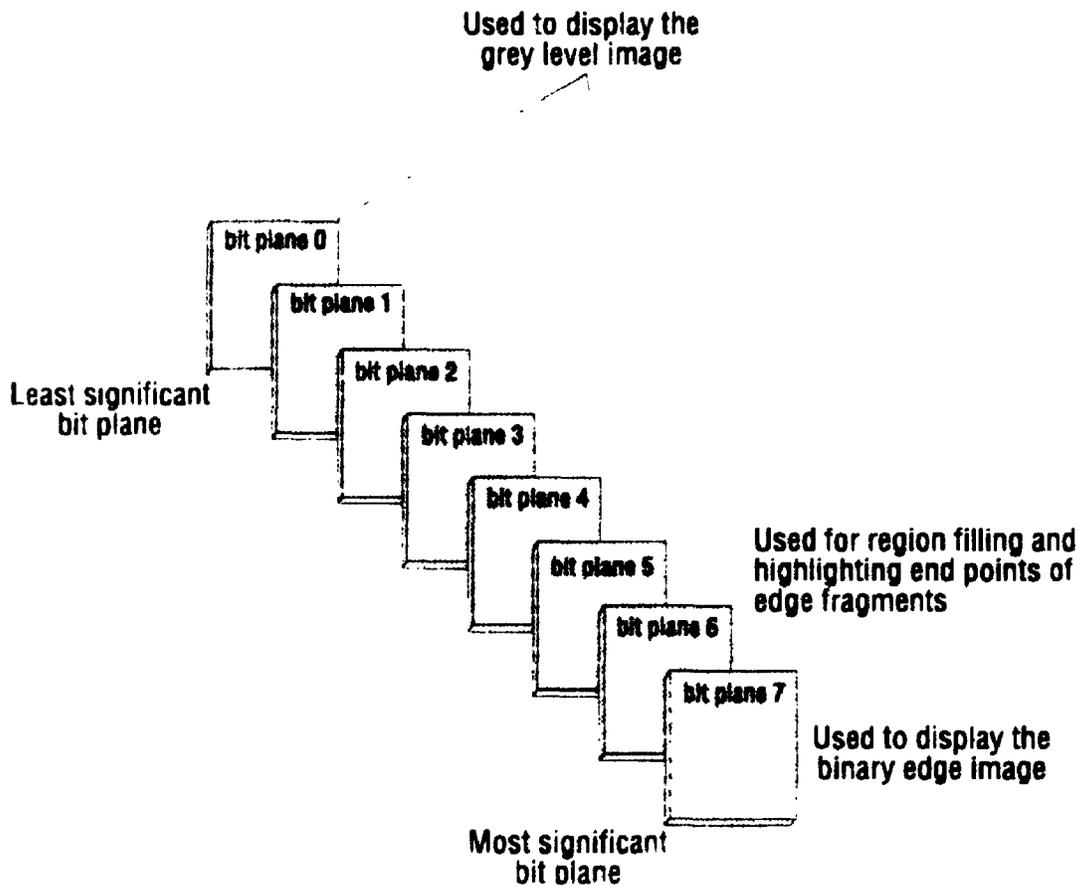


Figure 3 24 : The organization of the eight bit planes of frame memory used for image editing. The original grey level image uses the lower six bit-planes (0..5) for display. The binary edge image uses bit-plane 7 for display. Bit-plane 6 is used for region filling and highlighting the end points of the edge fragments.

### 3.3.1 Manual Editing

A mouse style pointing device is used to edit the image. The pixels in the binary edge image (i.e. bit-plane 7) can be set to ON or OFF as desired, when set to ON the pixel displays a selected colour (red). Therefore, the mouse can be used to connect edge segments or delete unwanted ones. Sometimes manual tracing of the feature boundaries is required. These operations are straight forward, since the mouse generates video coordinates in frame memory and a cursor is displayed on the image to show its current location. The software tracks the motions of the mouse and turns on pixels along its path either by joining those discretely marked by pressing the mouse button or continuously as the mouse is moved.

Another feature incorporated in the editor is a mode of drawing to help select features manually. This is generally called flooding or filling and consists of turning ON the pixels in bit-plane 6 to display a selected color (green) which starts at the current location of the cursor and grows outwards in all directions until a boundary pixel (i.e. one which is ON in bit-plane 6 and therefore displaying red) is met. This fills any closed shape, however intricate. The fill can be removed from any feature by placing the cursor anywhere inside the feature and flooding with OFF (i.e. by turning the bits of bit-plane 6 off). Filling a feature provides confirmation that a feature is completely surrounded by a boundary. Also, filling is used to eliminate features from further consideration because filled features are ignored by the feature extraction algorithm.

When confronted by a complex image containing many discontinuous edges, it is often advantageous to be able to identify the discontinuous edge contours automatically. The ends of the broken contours can be located by a contour following

method, however, this is time consuming. An algorithm developed here uses a 3 by 3 pixel window in a raster fashion to locate the ends of edge contours in the binary edge image. Where the end of an edge contour occurs in the center of a 3 by 3 pixel window the possible configurations of pixels in the window are as shown in figure 3.25. The blackened pixels in the figure correspond to edge pixels and the white pixels correspond to non-edge pixels. The algorithm presented in Appendix 3 recognizes the configurations of pixels displayed in figure 3.25 and identifies the central pixel in the window as an "End" pixel. Figure 3.26 shows the binary edge image of the grey level image shown in figure 3.7, with the end of discontinuous edge contours identified and highlighted in red.

### **3.4 Feature Extraction**

At this point the edges exist only as pixels which are turned on in the video display. Feature extraction is necessary to correlate these pixels with individual features and to define their boundaries for subsequent analysis (Pavlidis 1978, Russ 1990). The third operation of image analysis, feature extraction, selects and groups the detected edges from the edge image into the feature boundaries which are required to describe the image for further analysis.

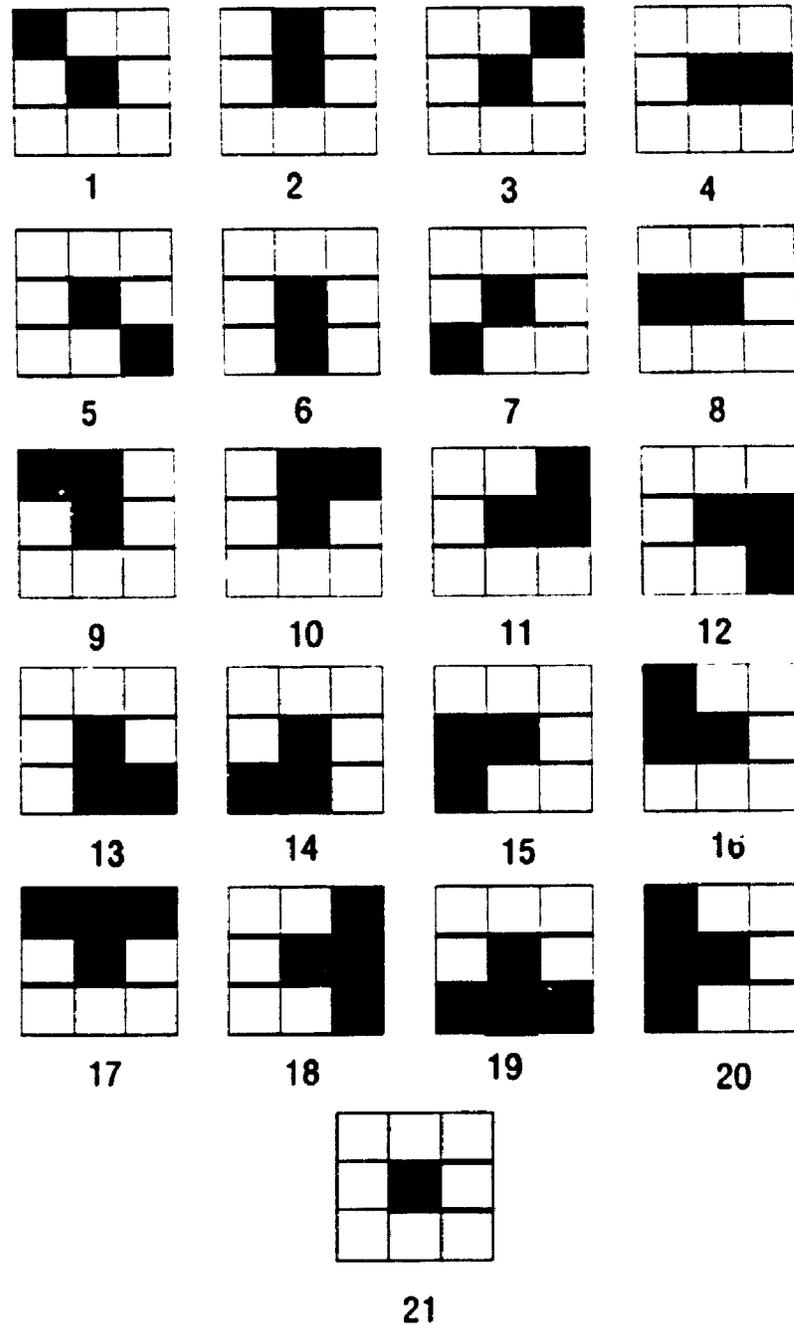


Figure 3.25 : The possible configurations of pixels in a 3 by 3 window in which the central pixel is the end of an edge contour. Black indicates edge pixels and white indicates non-edge pixels.

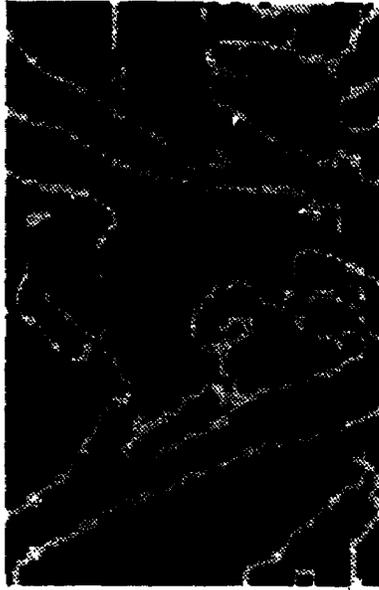
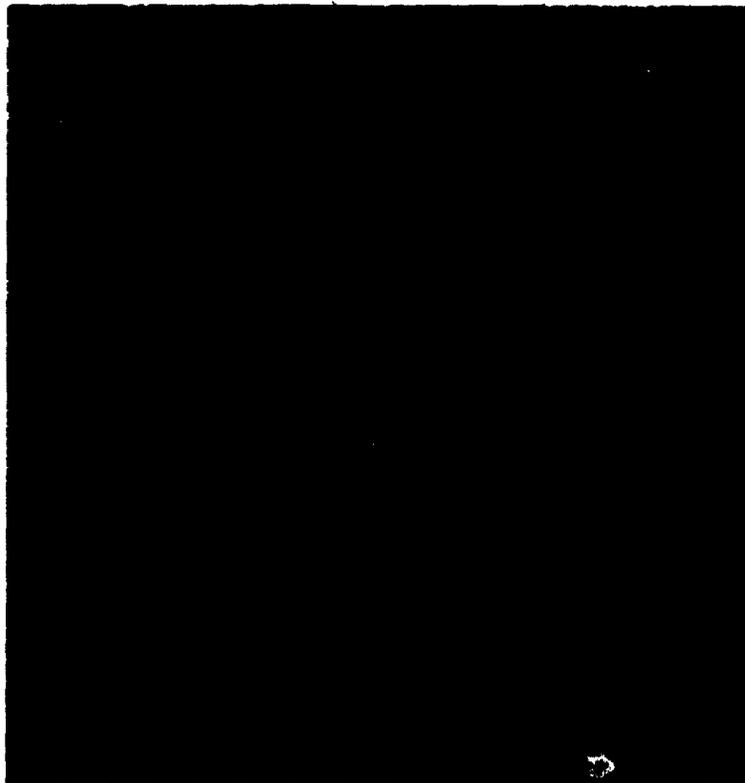


Figure 3.26 The edges detected by the canny operator shown in figure 3.20, with the end of discontinuous edge contours highlighted in red. Owing to the inadequate resolution of the printer, part of the image is shown enlarged for clarity.

Since most feature boundaries are continuous in nature, edges should similarly exhibit continuity (i.e. closed boundaries). A simple technique to extract these closed boundaries is to follow their edge contours. Several contour tracing algorithms have been proposed to extract boundaries from the segmented edge images (Pavlidis 1978, Cai 1988, Ali and Burge 1988, Chang and Leu 1990, Xie and Thonnat 1992). In most such algorithms, the boundary of a feature region is considered to consist of pixels with at least one neighbour which does not belong to the interior of the feature. The result is that the common boundary between two adjacent regions is not recognized, instead a double boundary results with a space in between (Starkey and Simigian 1987). A solution to this problem was recently proposed by Liow (1991) using a contour tracing algorithm. However, contour tracing algorithms have been found to be prone to yield ambiguous results in petrographic images due to the inherent inhomogeneity within the feature regions. In the present study an algorithm is presented which takes a segmented edge image as input and extracts the closed edge contours which represent the boundaries of feature regions by scanning the image in a raster fashion (Samantaray and Starkey, in press). Figure 3.27 shows an edge image and the closed contours extracted from it using this algorithm.

In an edge image an edge contour consists of a string of pixels delimited by their end pixels (head and tail). A closed contour is an edge contour in which the end pixels are connected. On an individual row of the matrix representing an edge image a sequence of non-edge pixels bounded by edge pixels at both ends represents a *line segment* and the  $(x, y)$  coordinates of the bounding edge pixels are referred to as the coordinates of the *boundary points*. A feature region can be defined as a set of four connected *line segments* which is completely surrounded by edge pixels.

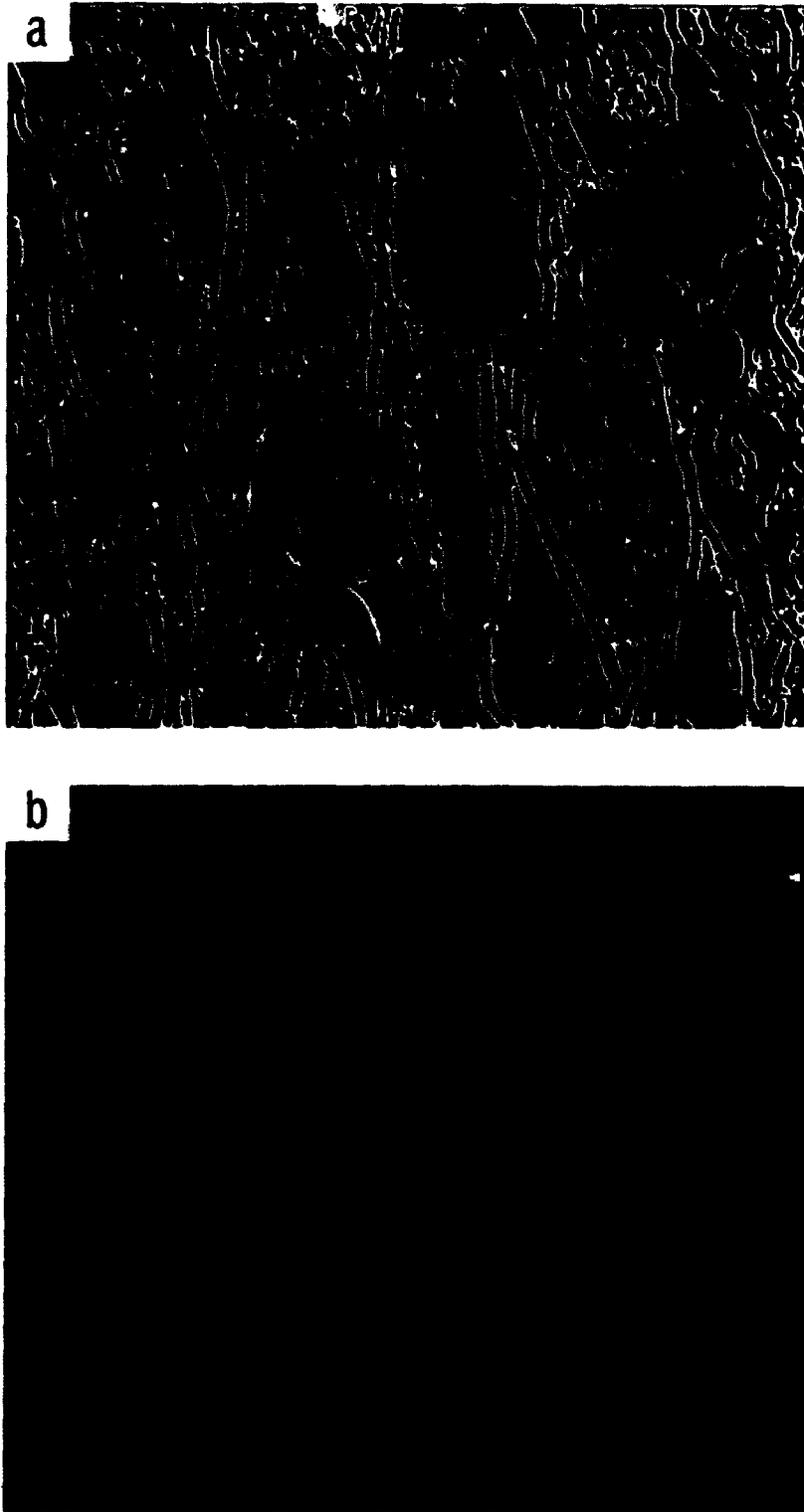


Figure 3.27 : (a) Binary image produced by the application of edge segmentation to figure 3.7. (b) Output image produced from (a) using feature extraction algorithm described in the text.

The edge image is processed in a raster fashion row by row, starting from the top. As each feature region is located it is assigned an index. Index 1 is used to identify those regions which intersect the boundary of the image and which are subsequently discarded. The features of interest are indexed consecutively, starting with the index 2, this index acts as a reference to a list of  $(x, y)$  coordinates of the *boundary points* of that feature region. Sometimes two or more regions identified on previous scans combine on a subsequent scan. When this happens, the different parts of the region previously identified are assigned the lowest index and the corresponding data structures are reindexed accordingly. For a region with a shape analogous to an inverted U, "∩", the same index is applied to both arms automatically. After obtaining the *boundary points* of a feature region, they are sorted according to the Freeman chain coding scheme (Freeman 1970, figure 3.28) to obtain a contiguous array of pixel coordinates for subsequent shape analysis. The algorithm which performs these operations is described below.

The basic task of the algorithm is to connect *line segments* and to locate their *boundary points*. The algorithm requires that three scans of an image to be stored in a  $3 \times N$  matrix, in computer memory, at any given time, where  $N$  is the length of the scan. The matrix is continuously updated as the image is processed by copying the rows two and three of the matrix into rows one and two respectively and storing a new scan in row three. The matrix is processed from left to right, the  $x$  coordinate of a pixel is the column number of the matrix and the  $y$  coordinate is the scan number of the image. Initially, when row one contains the data from the first scan of the image, all the *line segments* in that row are indexed with the value 1, indicating that the features to which the *line segments* belong extend outside the image area. Subsequent processing of the matrix consists of locating four-connected

<b>3</b>	<b>2</b>	<b>1</b>
<b>4</b>		<b>0</b>
<b>5</b>	<b>6</b>	<b>7</b>

**Figure 3.28 : The Freeman chain codes used to identify the orientation of the eight nearest neighbours relative to the central pixel.**

*line segments* on the second row and indexing them by comparison with the first according to the following five criteria, the letters refer to figure 3.29.

- (a) : Where the pixels in the *line segment* [e to f] in row two are all adjacent to edge pixels in row one the *line segment* [e to f] is the beginning of a new feature region and it is assigned a new index. The (x, y) coordinates of the edge pixels immediately to the left and right of the *line segment*, (e-1) and (f+1), and those directly above the *line segment* [e to f] in row one are stored as *boundary points* for this new region.
- (b) : Where the *line segment* [e to f] in row two overlaps a *line segment* [a to b] in row one the two *line segments* belong to the same region and the index previously assigned to *line segment* [a to b] is assigned to *line segment* [e to f]. The (x, y) coordinates of the edge pixels (e-1) and (f+1) in row two are stored as *boundary points* for this region.
- (c) : Where, in the *line segment* [e to f], pixel e is the first pixel in row two the *line segment* is assigned the index 1, indicating that this feature extends outside the image area. At this point the index previously assigned to the *line segment* [a to b], which overlaps [e to f], is recognized as belong to a feature which is to be eliminated and thus any *line segments* with this index are reindexed with the value 1 and the associated x, y coordinates are discarded.
- An analogous situation occurs where f is the last pixel in row two.
- (d) : Where the pixels in the *line segment* [e to f] are all adjacent to edge pixels in row three the (x, y) coordinates of the edge pixels (e-1) and (f+1) in row two and those directly below the *line segment* [e to f] in row three are stored as *boundary points* and assigned the same index as *line segment* [e to f].

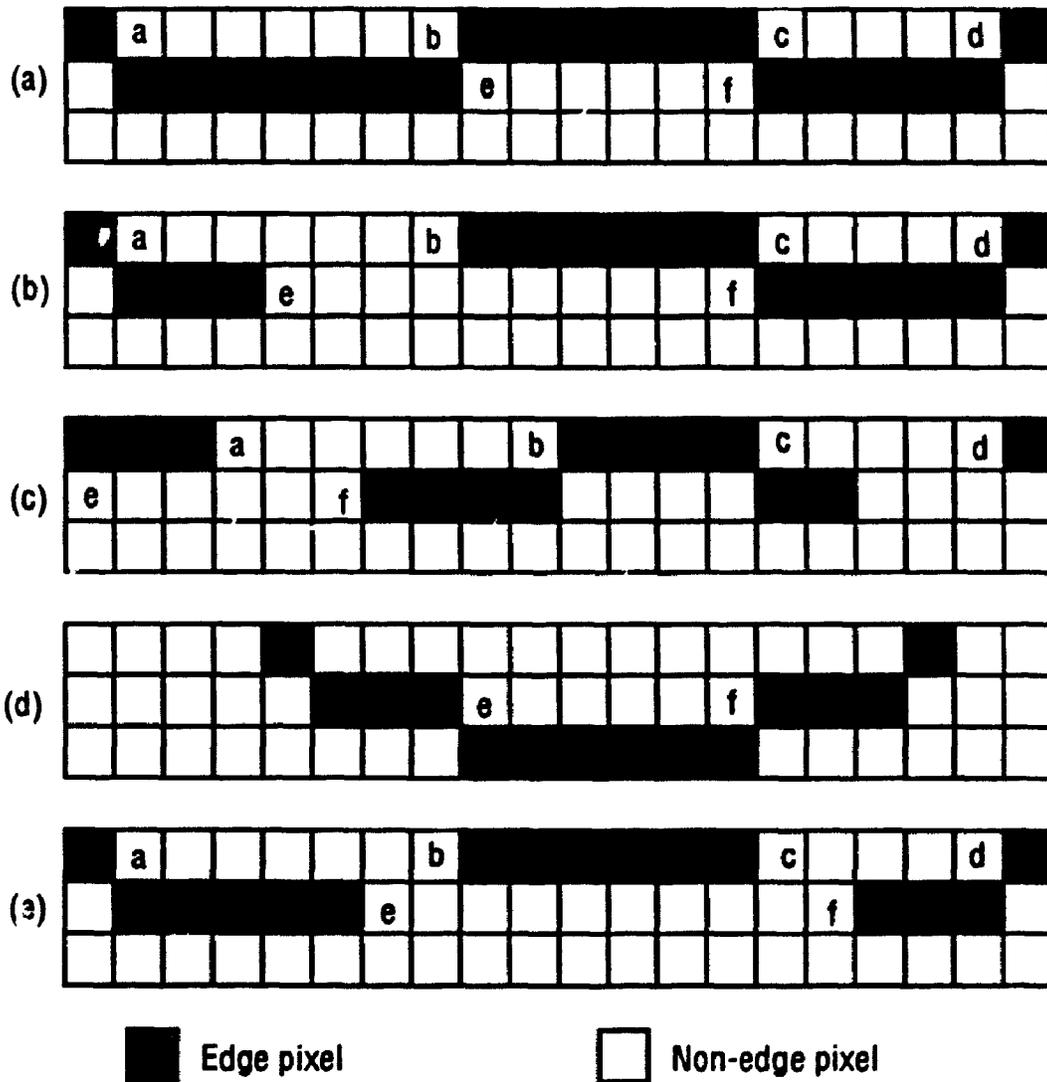


Figure 3.29 : The possible configuration of line segments in three consecutive scans of a raster scanned image (see text).

(e) : Where the *line segment* [e to f] overlaps two *line segments* [a to b] and [c to d], two feature regions, which were previously identified as separate and assigned different indices, have merged. Therefore all *line segments* and the *x, y* coordinates identified with the higher index are reassigned the lower index. If the new index is a 1 then the data are discarded. Otherwise the (*x, y*) coordinates of the edge pixels (*e-1*) and (*f+1*) in row two are stored as *boundary points* for this region.

Where the regions which merge are previously separated by a single pixel boundary, duplicate *x, y* coordinates of *boundary points* will occur. The duplications are identified by comparing the two lists of boundary coordinates for the previously separate features and the data in one are eliminated.

Combinations of the above criteria may apply to any particular region as the image scans are processed. A feature is recognized as having terminated when its index exists in row one but not in row two. The (*x, y*) coordinates of the *boundary points* of terminated regions are sorted and stored in a contiguous array of pixel coordinates.

### 3.4.1 Sorting Boundary Pixel Coordinates

After the image has been scanned, as described above, and the features identified and selected the *x, y* coordinates of the boundary segments of the individual recognized features are contained in a list of boundary coordinates. In addition, the list may contain the coordinates of isolated edge contours within the feature, the coordinates of edge contours within the feature which are attached to the boundary at one end or the coordinates of closed contours around inclusions. These are removed by

the sorting process which produces an output array containing the contiguous  $x, y$  coordinates, or the chain code of only the feature boundary pixels.

Because the image is processed scan by scan the first pair of  $x, y$  coordinates in the list of boundary coordinates for each feature always pertain to a pixel on the boundary of a feature region and the second pixel will always be in the direction 0 or 7 (see figure 3.28). Thus the initial direction for tracing the boundary is established. Each pixel is considered in turn as the *central pixel* in a 3 by 3 neighbourhood within which the next edge pixel is selected following the Freeman chain code sequence. Once an edge pixel is identified it is eliminated from the list of boundary coordinates. Where there is only one neighbouring edge pixel in successive neighbourhoods the sorting continues until it returns to the starting pixel and the feature boundary is defined. Where there are more than one neighbouring edge pixels in the neighbourhood the *central pixel* is flagged and processing continues by proceeding with the neighbouring boundary pixel having the lowest chain code. If the subsequent succession of pixels leads back to the starting pixel then the boundary has been defined and processing ceases. If a point is reached beyond which there are no edge pixels then the string of pixels followed from the previously flagged pixel, represents an open edge contour attached to the feature boundary. The  $x, y$  coordinates of this string are eliminated from the output array and processing returns to the flagged pixel and recommences with the neighbouring pixel with the next highest code. This procedure is applied recursively for multibranching line segments.

This algorithm generates an ordered list of boundary coordinates for each of the features present in the image.

### 3.5 Nature of the Information

The final result of applying the digital filtering, edge detection and feature extraction to the original petrographic image is an ordered list of  $(x, y)$  coordinates of the boundary points of the features recognized in the image. These boundary points can also be described using the Freeman chain code sequence. Figure 3.30 shows examples of both kinds of feature boundary description. These data are required for the subsequent calculation of the parameters which describe the features.

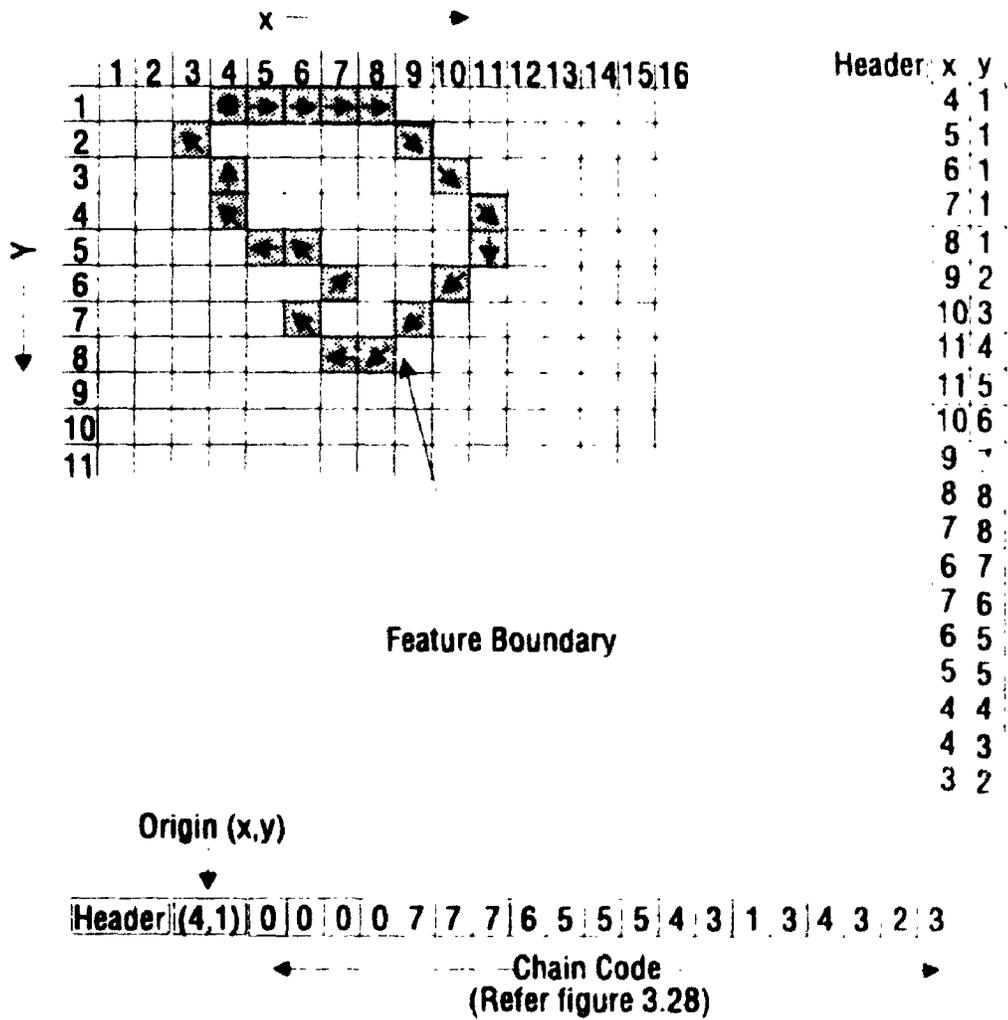


Figure 3.30 : Feature boundary information. The arrows indicate sequence in which the boundary is followed. The boundary coordinates are recorded as either a sequence of Freeman chain code directions or as (x,y) coordinates.

## Chapter 4

# Image Measurement

This chapter discusses some of the measurements which can be made using the image analysis described here. In this regard it is important to realize that in addition to the boundary information derived from image processing described in the last chapter the original grey level image and the intermediate edge image are also available for analysis. Together the data from these images provide a wealth of information which is not available in traditional petrography. In addition, the fact that the boundary data are available as pixel  $(x,y)$  coordinates and chain codes, facilitates some of the calculation which are best accomplished in one form of data representation rather than the other. In some instances, as noted below, a pixel representation of the features is required. In a pixel representation, the image area inside the feature boundary is filled with an index which identifies the feature.

The application of image analysis to petrographic images provides both global measurements, which characterize the rock as a whole, and feature measurements, which describe individual features. Simple analyses, such as the determination of grain size and shape distributions, can be performed automatically using the boundary data  $(x,y)$  coordinates of the selected features in the image. More complex

analyses require user interaction to manipulate the images to obtain the necessary data. Examples of such analyses are the determination of the percentage of inclusions in host mineral grains, mineral abundances and associations, size distributions of minerals and pore spaces and grain clustering analysis.

## 4.1 Global Measurements

The frequency distribution of grey values can be obtained from the grey level image and if a particular grey level, or range of grey levels, can be correlated with particular phases, the frequency distribution of grey levels can be converted into a modal analysis. This technique was applied to a study of the properties of grains and pore space by Mainwaring (1989) and similar studies have been conducted on porosity in rocks by Gerard et al. (1992). In a less obvious application, Janssen et al. (1991) examined the variation of grey level distributions in photomicrographs taken between crossed polarizers of rhombohedral camphor grain aggregates deformed by simple shear. They observed an increase in the uniformity of grey values with increasing strain, which they used as an indicator of the progressive development of crystallographic preferred orientation.

Determining the area occupied by inclusions in an individual mineral grain, or in grains of a particular mineral species can be accomplished by measuring the area of the host grain(s) and the inclusions separately. The areas of the host grain(s) and the inclusions can be readily obtained from their boundary data. An example of such a measurement is illustrated in figure 4.1, the area of the host mineral grain and the inclusion are measured independently in two separate computer stored images. Such analysis requires user interaction to select the grains and the inclusions in the

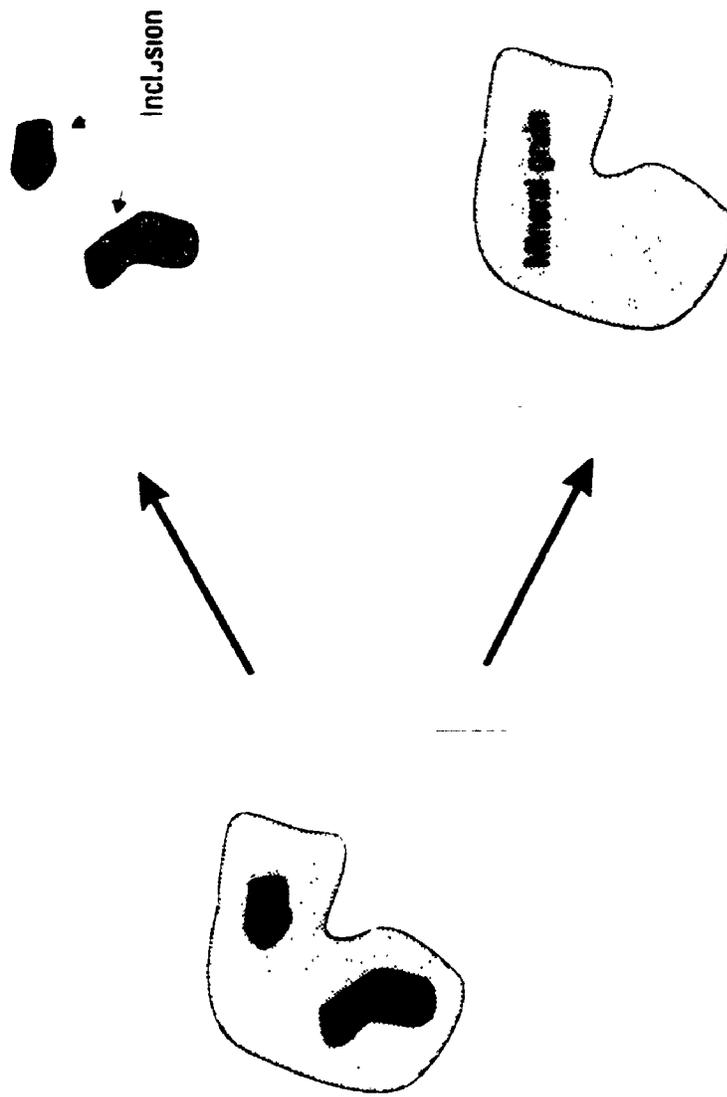


Figure 4.1 : Example of measuring the area of inclusions in a host mineral grain. The areas of the mineral grain and the inclusions are measured independently, in two separate computer stored images, and the percentage of the inclusion is calculated

image.

The number of features present in a given region of the image can be obtained by measuring the number of features per scan line and converting the results to the number per square cm. (Russ 1990). Given the number of features and the total area they represent, their mean size can be calculated. Alternatively, the mean size can be estimated from the mean *intercept length*, the *intercept length* is the length of a chord across a feature produced by any random line drawn on the image (figure 4.2) (Petruk 1989, Russ 1990).

The association between one mineral and its neighbours can be expressed as the percentage of the mineral perimeter that is in contact with another mineral, or the number of mineral grains that are in contact with another mineral in an image. Such analysis requires that the boundary representation of features be re-converted to pixel-based representation where the image area inside the boundary is filled with an index which identifies the feature. One such filling algorithm is described by Distanto and Veneziani (1982). This can be solved by the technique illustrated in figure 4.3. Two minerals, A and B, which share a common boundary are assigned different indices. To obtain the common boundary between the two minerals, the adjacency of the mineral regions must be determined by a neighbourhood search. Because the grain boundaries determined by the present analysis are one-pixel wide, the use of a 3 by 3 pixel window is sufficient for this purpose. The window is moved over the image in a raster fashion to select the shared boundaries. Where a shared boundary pixel occurs in the center of the 3 by 3 window, the image values in the window should contain both of the indices which identify A and B.

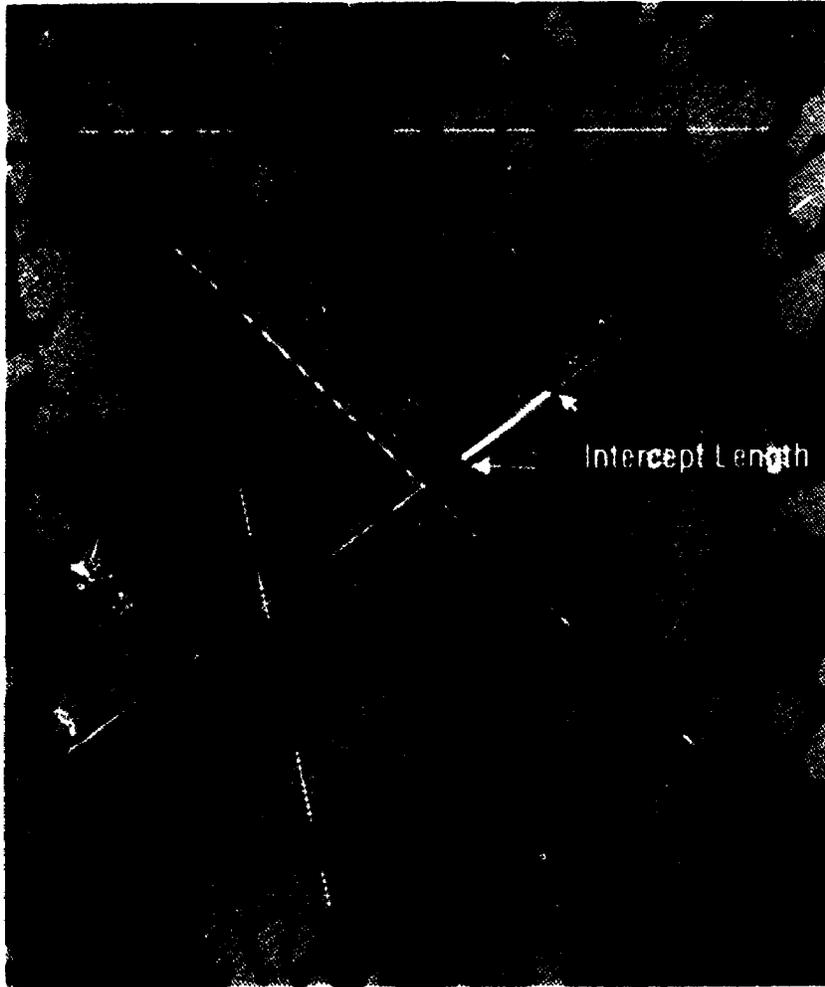


Figure 4.2 : The determination of intercept lengths. Lengths of line segments within and between features are measured.

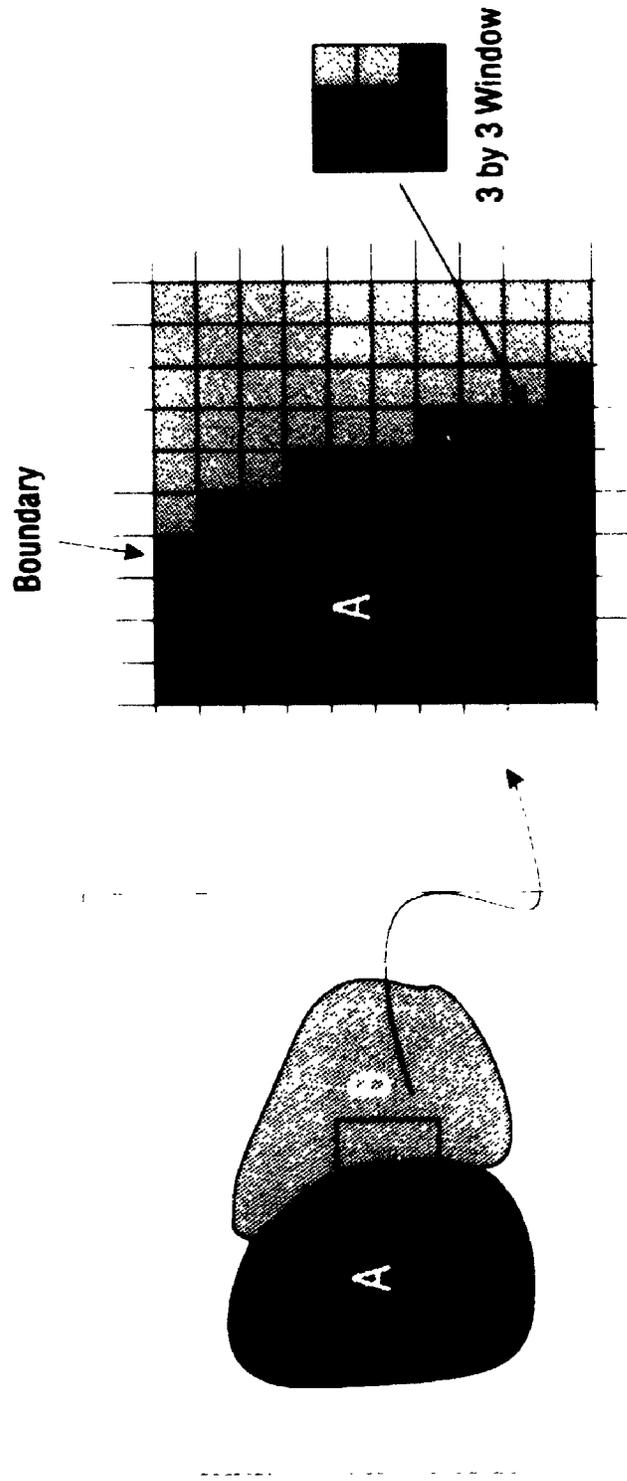


Figure 4.3 Example of the association of two mineral grains showing a one pixel wide boundary within a 3 by 3 window. The shared boundary is recognized by the presence of both minerals within the window

## 4.2 Feature Measurements

Parameters used to describe individual features can be divided into four groups: (1) measures of size, (2) descriptions of feature location, (3) aspects of shape, (4) texture. A variety of individual parameters which can be measured, or calculated from this imaging system, are discussed below.

### 4.2.1 Measures of Size

The most obvious measures of the size of a feature in a two dimensional image are its area and the length of its perimeter. The area is most readily obtained from the pixel based representation of the image by simply counting the number of pixels within the feature. However, the area can also be calculated from boundary coordinates when the feature is identified by its boundary points. The coordinates of the boundary points are considered to be the corners of a polygon in cartesian  $(x, y)$  coordinates (Tough and Miles 1984, Russ 1990). By convention, the final coordinate pair in the boundary list  $(x_n, y_n)$  is equal to the first  $(x_0, y_0)$  for a closed feature. From the boundary coordinates the total area of the feature is calculated as follows (Russ 1990):

$$Area = \frac{1}{2} \sum_{k=0}^{n-1} (x_k \cdot y_{k+1} - x_{k+1} \cdot y_k) \quad (4.1)$$

The measured area is expressed in real units such as square microns, square millimeters, etc.

To determine the length of the perimeter of a feature in a pixel-based representation, an intermediate step using the boundary representation is required (Russ 1990). The perimeter is the sum of the Pythagorean distances between the discrete

boundary points,

$$Perimeter = \sum_{k=0}^{n-1} \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2} \quad (4.2)$$

Where the boundary is represented by chain code, the perimeter is the number of orthogonal (0 or 90 degree) steps plus 1.414 times the number of diagonal (45 degree) steps. Table 4.1 lists the areas and perimeters of the grains shown in figure 4.4 for which the data provided by the image analysis consisted of boundary points.

Another commonly used size parameter is the length of a feature. This is the maximum distance between any two points on the periphery. These perimeter points are directly available from the boundary representation of the feature.

## 4.2.2 Descriptions of Feature Location

The position of a feature is determined by the  $x$  and  $y$  coordinates of its centroid (center of gravity) which is the point at which the central moment of the feature is minimized. The centroid location is calculated by integrating the central moments about the  $x$  and  $y$  axes and dividing by the area. These operations can be simplified into the following expressions (Davis 1987):

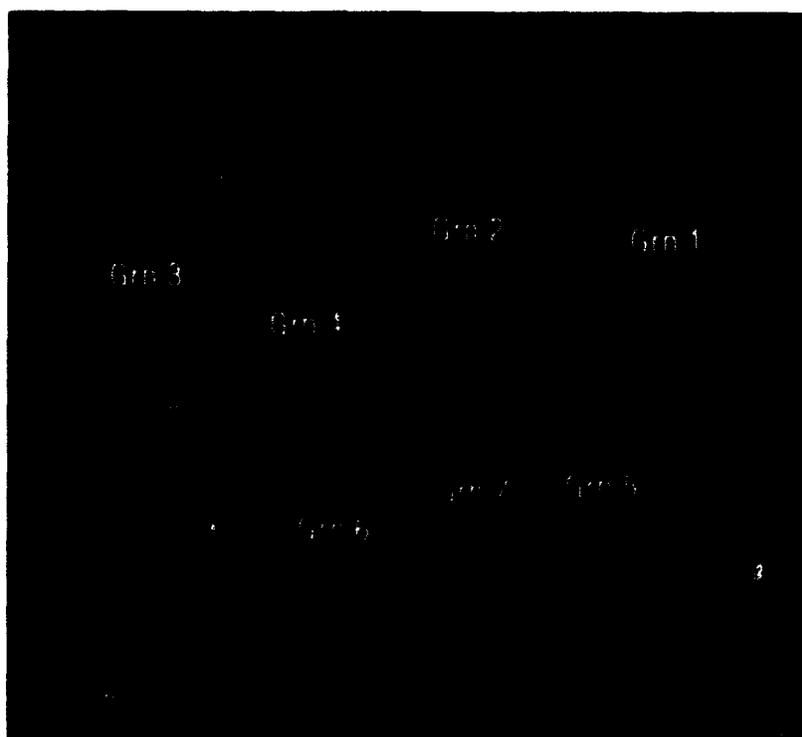
$$CenterX \cdot Area = \sum_{k=0}^{n-1} \frac{1}{6} (x_{k+1}^2 + x_{k+1}x_k + x_k^2) (y_{k+1} - y_k) \quad (4.3)$$

$$CenterY \cdot Area = \sum_{k=0}^{n-1} \frac{1}{6} (y_{k+1}^2 + y_{k+1}y_k + y_k^2) (x_k - x_{k+1}) \quad (4.4)$$

The *Area* is obtained from the eq. 4.1. For a convex shaped feature the centroid will lie within the feature boundary, but for a concave shape, or one containing internal voids, the center may lie outside the boundary.

Table 4.1: Area, Perimeter, and X & Y coordinates of the centroid (from top left corner of the image) for grains shown in figure 4.4.

Grain	Area (in mm. <sup>2</sup> )	Perimeter (in mm.)	Centroid (in mm.)	
			X Coord.	Y Coord.
1	0.051821	1.045	0.909	0.316
2	0.059812	1.086	0.629	0.308
3	0.067596	1.242	0.201	0.374
4	0.065866	1.165	0.395	0.451
5	0.057914	1.036	0.817	0.671
6	0.039880	0.870	0.448	0.746
7	0.034118	1.051	0.629	0.688



**Figure 4.4 :** The boundaries of the mineral grains shown in figure 3.7 determined automatically by image analysis.

The centroid location is an example of a parameter that can only be determined from a pixel-based representation of a feature. Attempting to calculate it from the boundary coordinates in a boundary-coded representation produces a biased result in which the apparent point moves closer to the more irregular portion of the boundary which is represented by more data points (figure 4.5). Table 4.1 lists the (x,y) coordinates of the centroids of the grains shown in figure 4.4 calculated from a pixel-based representation. The location of the centroids may be calculated with respect to some global coordinate system or relative to other features. In this example, the top left corner of the image is taken as the origin of the reference coordinates and therefore, the feature positions only have significance within the individual image.

### **4.2.3 Aspects of Shape**

Shape is a concept which is widely understood yet difficult to define. This is an extremely important property (Exner 1986) and is often a key factor in being able to recognize or select features of interest. A shape parameter should possess several desirable properties. Obviously, features with different shapes should yield different measures, and similar shapes should yield similar values regardless of the size or orientation of the feature. However, it has been proven mathematically that no single measure can be unique to only one shape (Davis 1987, p.343). In most geological applications, summary statistics, such as the mean and variance, are calculated for the shape measurements from a collection of features, to describe the variation in an aggregate. A wide variety of shape descriptors are available in the literature and some those most commonly used in earth science are discussed here.

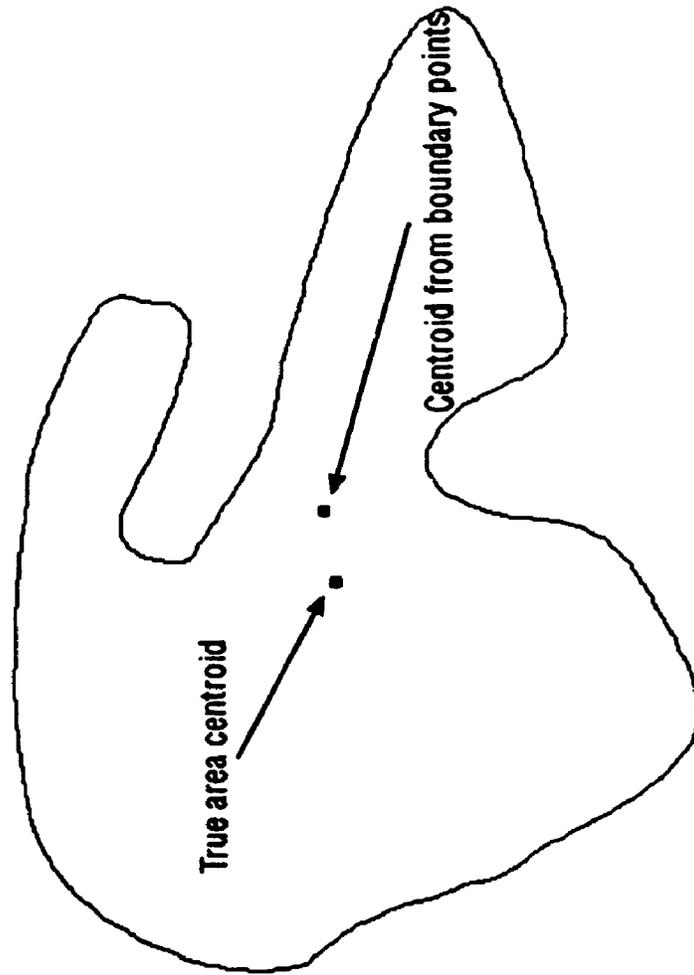


Figure 4.5 : A feature showing the true area centroid, and the centroid calculated from the coordinates of boundary points.

The shape of a feature can be described by geometric forms. The two most frequently used are the rectangle and ellipse (Simigian and Starkey 1986). Commonly an enveloping rectangle is obtained from the long dimension of a grain and the maximum dimension at right angles to it. Figure 4.6 shows the long and short axes (which define the enveloping rectangles) of the grains shown in figure 4.4. These parameters yield the aspect ratio (long axis/short axis) and the orientation of the grain in terms of the long axis. Table 4.2 lists the aspect ratios and orientation angles obtained from the enveloping rectangles for the grains shown in figure 4.4. The areas of the enveloping rectangles do not preserve the areas of the grains.

An ellipse can be fitted to a feature in a variety of ways. The best-fit ellipse is calculated using eigenvalues such that it passes through the feature boundary so that the sum of the squares of the distance between each point on the boundary and the ellipse is minimized (Craig et.al. 1982). The boundary coordinates and the centroid are used to calculate the second central moment and the eigenvector. The long and short axes of the ellipse are calculated from the eigenvalues from which the aspect ratio and the orientation of the long axis of the ellipse can be obtained, see the data for the grains shown in figure 4.4 in Table 4.2.

Similarly, an area ellipse is calculated using the observed area and long axis of the grain to derive the short axis,  $2b$ , from the formula  $b = A/\pi a$ , where  $A$  is the area of an ellipse and  $a$  is one half of the long axis (Simigian and Starkey 1986). Both the best-fit ellipse and the area ellipse preserve the area of a feature. The area ellipse emphasizes the long axis, which is the dimension commonly observed subjectively, the best-fit ellipse is calculated statistically and may be a better choice for shape analysis (Simigian and Starkey 1986). Figure 4.7 shows best-fit ellipses calculated for the grains shown in figure 4.4.

**Table 4.2: Aspect ratios and orientation angles for the grains shown in figure 4.4 (measured from enveloping rectangles and best-fit ellipses). Orientation angle is measured clockwise from the X-axis.**

<b>Grain</b>	<b>Env. Rectangle</b>		<b>Bestfit Ellipse</b>	
	<b>Aspect</b>	<b>Orient.</b>	<b>Aspect</b>	<b>Orient.</b>
	<b>Ratio</b>	<b>Angle</b>	<b>Ratio</b>	<b>Angle</b>
1	1.579	124.96	1.524	131.68
2	1.801	76.26	1.634	85.70
3	2.140	95.85	1.731	95.96
4	1.959	102.09	1.732	95.98
5	1.681	68.45	1.567	76.96
6	1.407	36.87	1.481	35.78
7	3.071	83.03	2.648	86.24



**Figure 4.6 :** The long and short axes (in grey) of the grains identified in figure 4.4. These axes define the sides of enveloping rectangles, see text.

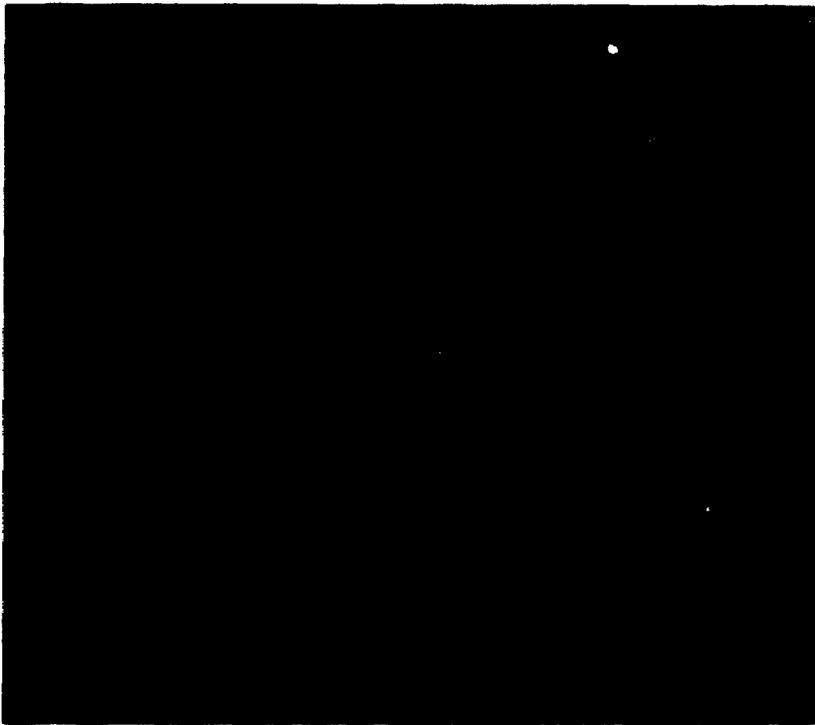


Figure 4.7 : The 'best-fit' ellipses (in grey) for the grains identified in figure 4.4

Probably the most widely used shape parameter is the *shapefactor* or the *formfactor*. Although the definition of this parameter varies between researchers, it is always calculated using perimeter and area measurements (Smith 1989, Russ 1990). For instance, one definition is  $4\pi \cdot \text{Area} / \text{Perimeter}^2$  (Russ 1990). This definition yields a formfactor of 1.0 for a circle and 0.785 for a square. Any irregular shape will have more perimeter for the same area, and the formfactor becomes much smaller. A similar shape parameter, *Roundness*, is calculated as  $4 \cdot \text{Area} / \pi \cdot \text{Length}^2$  (Russ 1990). Roundness uses the long axis of the feature instead of perimeter. This makes it more sensitive to how elongated the feature is rather than how irregular its outline may be. The *Roundness* value is 1.0 for a circle. For illustration Table 4.3 lists the *formfactor* and the *Roundness* values for the grains shown in figure 4.4.

The shape of a feature can also be characterized by the number of corners it possesses (Russ 1990). The difficulty with human interpretation of petrographic images is in deciding what constitutes a 'corner'. Corners can be easily detected from the chain-code representation of the feature boundary (Liu and Srinath 1990). A chord is drawn between points in the chain that are some distance,  $n$ , apart. The chord length can be made large enough to ignore local variations that are not to be considered as 'real' corners. As the position of the chord is advanced along the chain, a frequency plot is made of some derived value such as the change in slope of the lines (Freeman and Davis 1977), the length of the line, or the net area between the line and the chain etc. (Russ 1990). These frequency plots vary somewhat in their sensitivity to minor irregularities and in the computational effort required. In the resulting graphs the peaks represent the corners of the feature.

Table 4.3: Formfactor and Roundness parameters for the grains shown in figure 4.4 (for detail see text).

<b>Grain</b>	<b>Formfactor</b>	<b>Roundness</b>
1	0.59633	0.54171
2	0.63729	0.45974
3	0.55067	0.43855
4	0.60984	0.45146
5	0.67807	0.55960
6	0.66210	0.56044
7	0.38814	0.28560

A very different approach is available to describe and recreate the shapes of closed forms using amplitude spectra of a finite fourier series (Schwarcz and Shane 1969, Ehrlich and Weinberg 1970, Beddow et al. 1977, Clark 1981,1987, Flook 1982, Mazzullo and Ehrlich 1983). The shape of a feature is estimated by an expansion of periphery radius,  $R$ , as a function of angle about the feature's center of gravity by a Fourier series as follows (see figure 4.8) (Ehrlich and Weinberg 1970)

$$R(\theta) = R_0 + \sum_{n=1}^{\infty} R_n \cos(n\theta - \phi_n) \quad (4.5)$$

where  $\theta$  is the polar angle measured from a reference line. The first term in the series,  $R_0$ , is equivalent to the average radius of the feature. For the remaining order of the terms,  $n$  is the harmonic order,  $R_n$  is the harmonic amplitude, and  $\phi_n$  is the phase angle. Each harmonic in the series represents the contribution of a specific shape component to the total shape of the feature. For example, elongation is measured from the second harmonic, triangularity from the third harmonic, quadrateness from the fourth harmonic, and fine scaled features on the surface of the feature from the higher harmonics. Depending on the complexity of the shape few boundary points may be needed for this type of analysis (Ehrlich and Full 1984). This method has certain limitations, the most conspicuous of which is that only single-valued outlines can be analyzed; a radius drawn from the centroid must intercept the perimeter only once. Problems arise with outlines such as those of grains 3, 5, 6 and 7 of figure 4.4. To address this problem Evan et.al.(1985) demonstrated the use of parametric cubic splines. Figure 4.8 shows the application of Evan's method to the sampled boundary points of grain 5 from figure 4.4. What is needed is a two-dimensional spectral analysis.

In the calculation of any shape parameter the sampling of the boundary points is readily automated as part of the image analysis system. Obtaining data

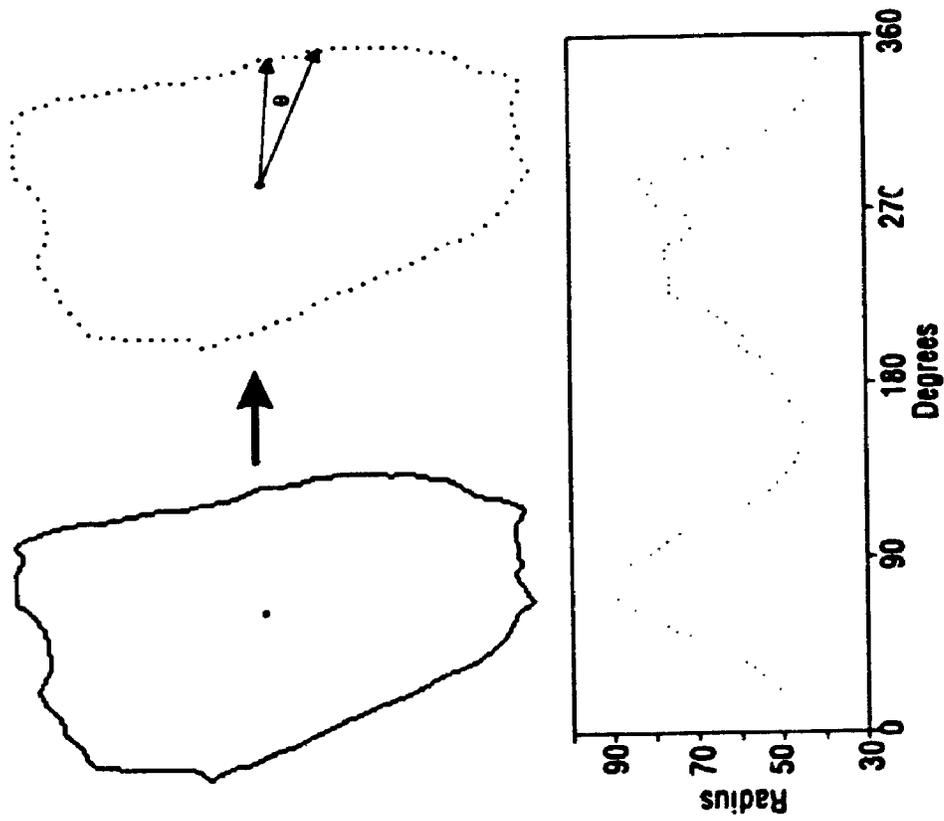


Figure 4.8 : The characteristic waveform obtained from a grain outline, as defined by edge points, unrolled with reference to the centroid.

manually could be difficult or impossible.

#### 4.2.4 Texture

Under the optical microscope, most features, such as mineral grains, are identified by their colour or texture. In a digitized petrographic image, the colour of a feature is represented by a grey scale (or a range of grey scales), and the texture of the feature is represented by the frequency distribution of the grey values inside the feature. For example, figure 3.4 displays quartz and the microcline grains which can be differentiated by their grey level and image texture respectively.

To quantify image texture a basic understanding of what constitute a textured region is necessary. Image texture depends on a number of basic ingredients such as repetition of local image patterns, non-random arrangement of these local image patterns etc. (Rosenfeld and Lipkin 1970, Hawkins 1970). These local image pattern can be classified by the following properties: (a) grey level statistics (mean, variance, grey level entropy, run lengths etc.), (b) spatial frequency, (c) local shape measures (edges, lines, different specific shapes), (d) higher order measures (calculated from the directly measured parameters). These can be correlated to the optical characteristics of the minerals in the petrographic images. In the present study this has not been implemented.

## Chapter 5

### Example Applications

In addition to the relatively straight forward location of grain boundaries in petrographic images by accumulating synoptic images, illustrated in chapter 3, figures 3.21 to 3.23, some other geological applications are discussed here which illustrate particular features of the system. In the first example, image analysis is used to identify the grain boundaries in thin sections of beach sand and define shape fabrics. In the second example, image analysis is used to identify the grain boundaries in a deformed oolitic limestone. Photomicrograph mosaic from the thin section are used to analyze the finite strain from the shape of the ooid particles. Here, the oolites had internal image texture which had to be considered during edge detection. In a third application, image analysis is applied to obtain the modal analysis of minerals in a thin section of Saxony granulite. The results of these analyses are compared with results obtained by traditional methods.

## 5.1 Sedimentary Grain Shapes

In a study of the hydrodynamics of "storm beds", Cheel (1991) collected data on the grain shape fabrics of sandstones by manually measuring the apparent long axes from the images of grains on oriented thin sections. The images were projected onto a digitizing tablet interfaced to a computer and the orientations of their longest axes were measured by touching the two estimated end points of the long axis with a pen sensor. The orientation, based on the grid coordinates of the ends of the axes, were calculated automatically. Only grains with visually estimated length-to-width ratios greater than 3:2 were measured.

Five of the thin sections used in the manual analysis were provided by Cheel (Cheel 1991, and Personal Communication; sections 1, 2, 3, A and B). The thin sections are prepared from epoxy-impregnated, unconsolidated sands; except sample B which is a sandstone. The sections are cut perpendicular to bedding and parallel to the direction of current flow.

The thin sections are examined using a petrographic microscope with the magnification selected to ensure that the boundaries between adjacent mineral grains displayed in the video image are at least few pixels wide, this yields approximately 40 mineral grains within the image. This required a 1x objective for all thin sections except B for which a 2.5x objective was used. To obtain similar number of data to that used by Cheel, 15 images are captured from each thin section within the area marked by Cheel as indicating the locations of his sampled data. The images are obtained using both plane and cross polarized illumination. One such image is shown in figure 3.7.

The grains are defined by intensity edges and are easily detected by the Canny operator (Starkey and Samantaray, in press). The extracted grain boundary data are recorded as  $(x, y)$  coordinates. In a few instances, adjacent mineral grains merged, with the result that the boundaries between the grains are not detected automatically. In such cases the edge operator detects the boundaries of the aggregated grains and not the internal, individual grain boundaries. These boundaries are resolved by tracing with the help of the image editor.

The results of Cheel's analyses are presented in figures 5.1 and 5.2 (Cheel 1991, and personal communication). The figures illustrate rose diagrams for the angles between the long axes of the grains and the bedding, the imbrication angles. The distributions illustrated by the rose diagrams from samples 1, 2, and 3, figure 5.1, are bimodal. These samples are from a lacustrine storm beach deposit, the two modes of the distribution are interpreted in terms of the reversing of currents in response to swash and backwash on the beach face (Cheel, personal communication). Figure 5.2 shows the data obtained by Cheel from samples A and B. Sample B is from a horizontally laminated sandstone that was deposited on an upper flow regime plane bed (Cheel 1991). Sample A is experimentally produced to simulate upper plane bed deposits with unidirectional flow. The rose diagrams from samples A and B show unimodal distributions with modes for the imbrication angles of  $35^{\circ}$  and  $15^{\circ}$  respectively.

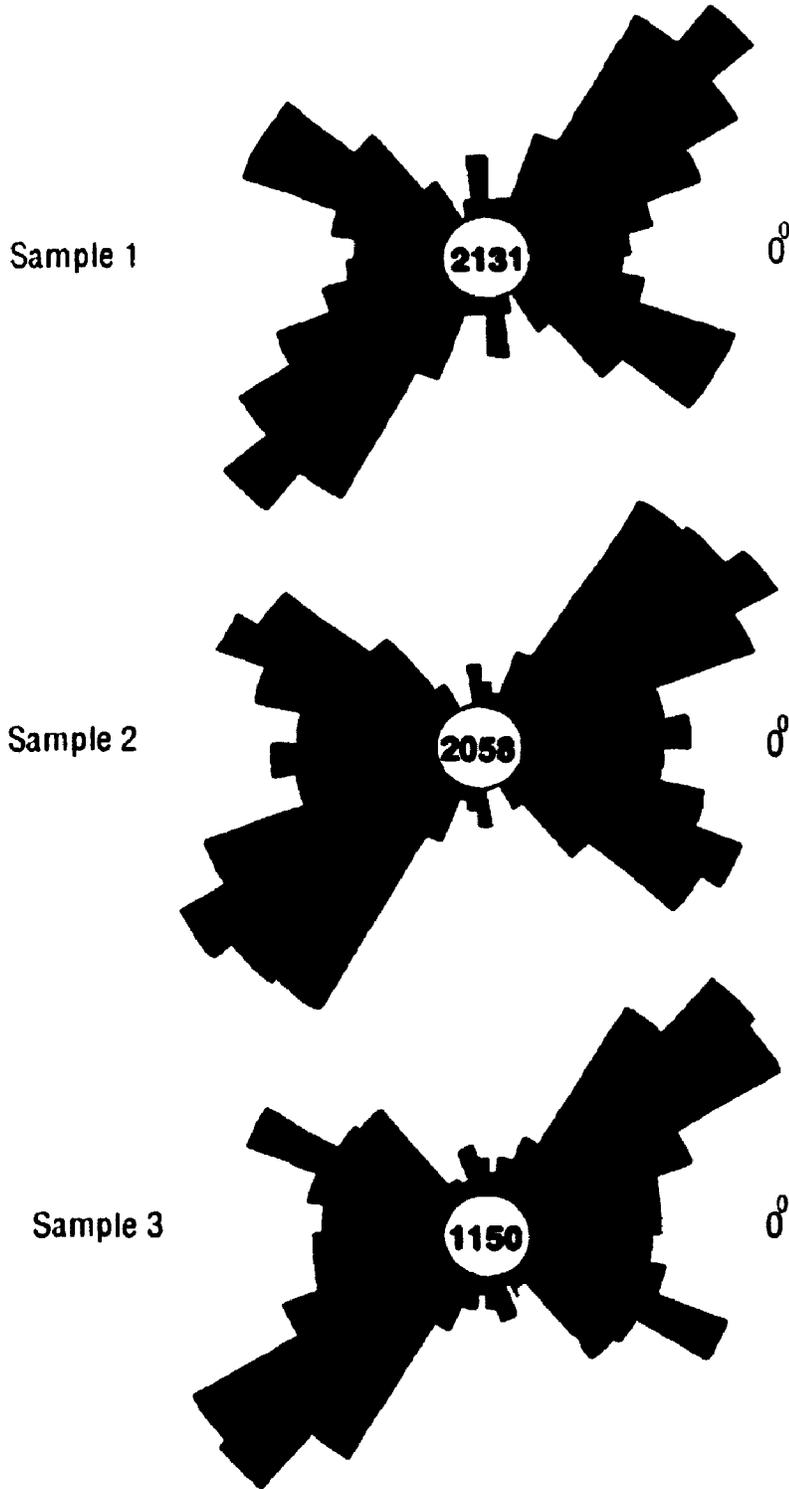
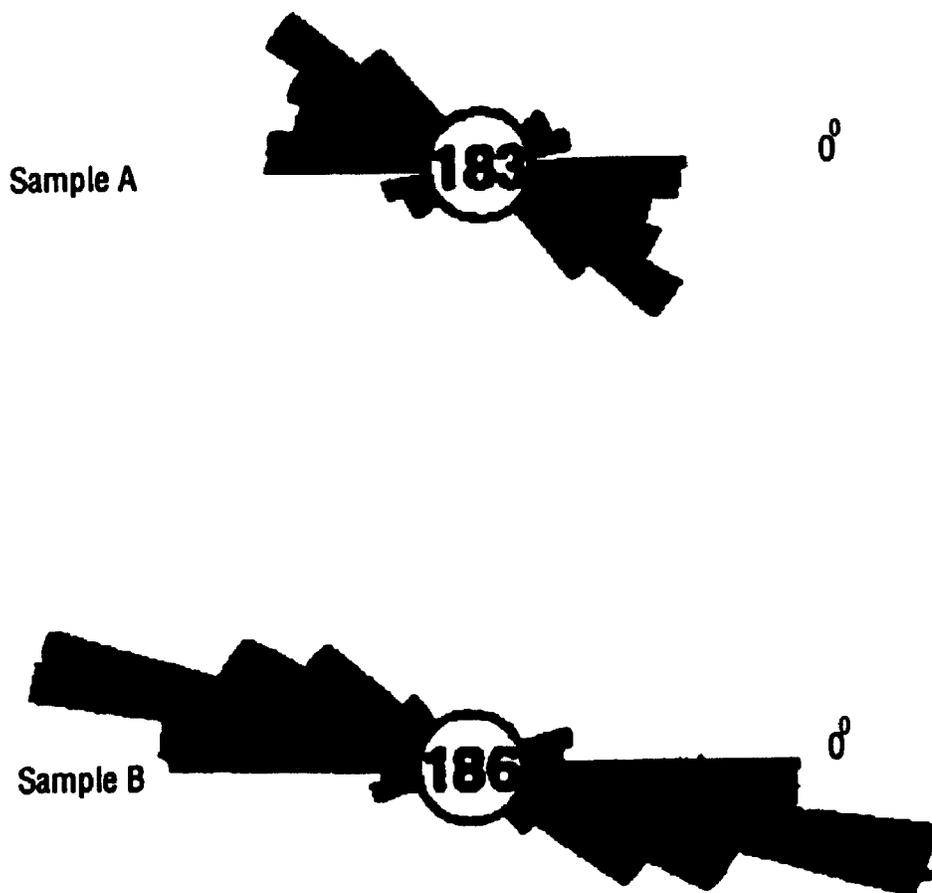


Figure 5.1 : Rose diagrams showing the orientation of the longest apparent axes of the sand grains in three samples of beach sands. The orientation angles are measured with respect to the internal lamination. The values indicate sample sizes.



**Figure 5.2 :** Rose diagrams showing the orientation of the longest apparent axes of the sand grains in two samples of beach sands. The orientation angles are measured with respect to the flow direction. The values indicate sample sizes.

The long axes measured by Cheel correspond to the long axes of the enveloping rectangles around the grains (Simigian and Starkey 1986), consequently, the enveloping rectangle was derived for each grain and the aspect ratio was calculated. The grains with aspect ratio greater than 1.5 (length-to-width ratio 3:2) were selected for comparison with Cheel's samples. The imbrication angles are presented as rose diagrams in figures 5.3 and 5.4. The rose diagrams from samples 1, 2, and 3, figure 5.3, show bimodal distributions. Figure 5.4 shows the data obtained from sample A and B, sample A shows a multimodal distribution and sample B shows a unimodal distribution.

The rose diagrams for samples 1, 2, and 3 determined here and by Cheel are similar and show bimodal distributions (cf. figures 5.1 and 5.3). However, the modes of the distributions of the imbrication angles determined in the present analysis are systematically less than those determined by Cheel. For sample A, Cheel reported a unimodal distribution (figure 5.2). The distribution determined here is multimodal (figure 5.4). However, both analyses reveal a wide range of imbrication angles. For sample B both analyses show a similar unimodal distribution (cf. figures 5.2 and 5.4) but the present analysis indicates a higher angle of imbrication (the mode of the distribution is  $25^{\circ}$ , compared to  $15^{\circ}$  reported by Cheel).

The possibility of error in the imaging system in the determination of the orientation angles was checked using ellipses of known orientation. The data were processed by the imaging system and no systematic error was detected. A similar test of Cheel's procedure could not be performed because the equipment is no longer operational (Cheel, personal communication).

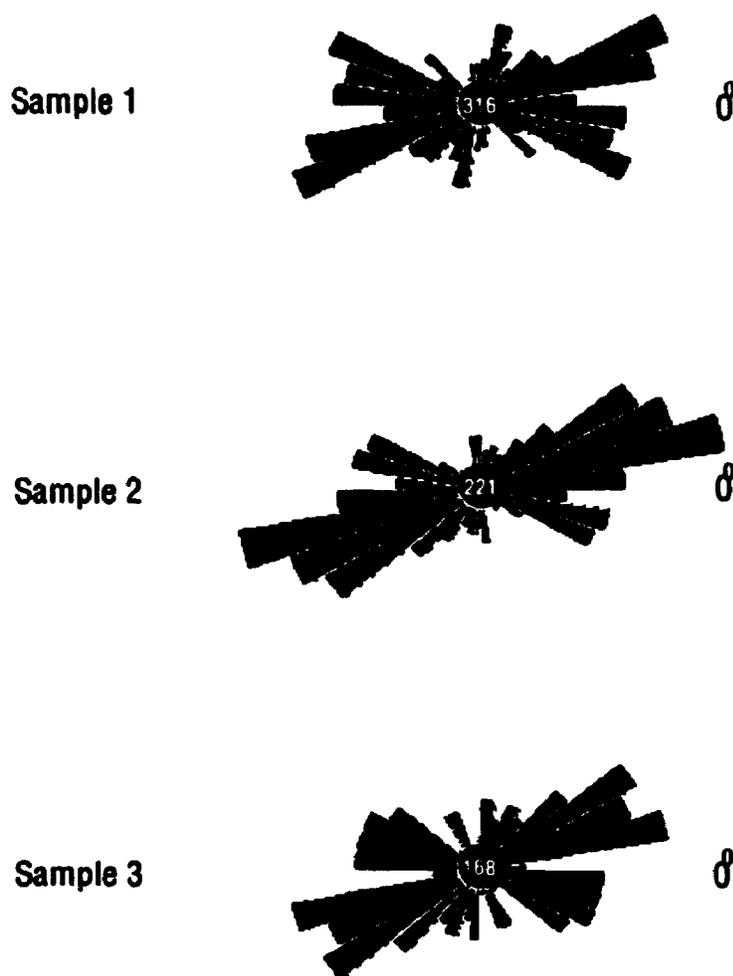


Figure 5.3 : Rose diagrams showing the orientation of the longest axes of the sand grains in three samples of beach sands. The values indicate sample sizes.

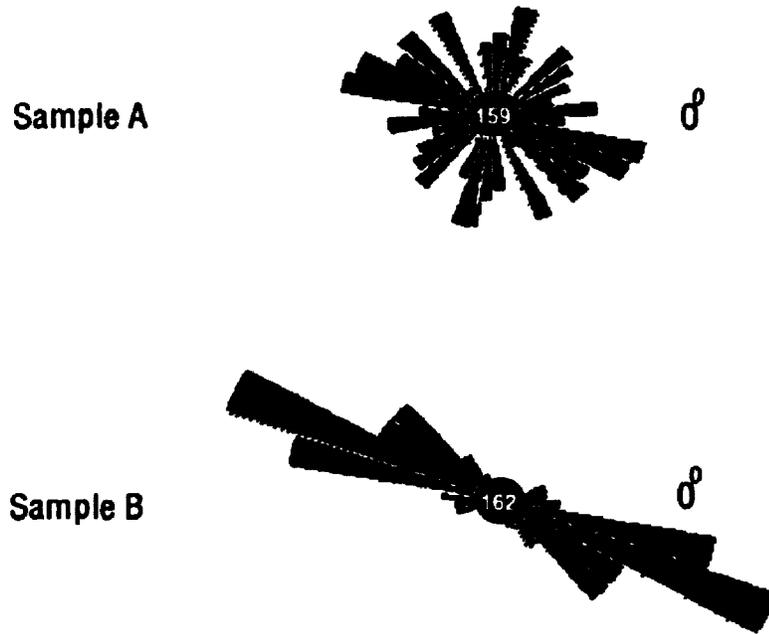


Figure 5.4 : Rose diagrams showing the orientation of the longest axes of the sand grains in two samples of beach sands. The values indicate sample sizes.

In manual digitization the selection of end points of the long axes is entirely user dependent. It has been demonstrated that even for simple shapes the visible long axis may not necessarily be the actual long axis of a grain (Russ 1990). Also, the aspect ratio cutoff used by Cheel is applied arbitrarily, thus the results obtained from such measurements can not be reproduced. The present system measures all the grains without applying a threshold to the aspect ratio, thus enabling the analysis of sub-samples with different ranges of aspect ratio. The results of analyses of subsets of data for sample A, using different aspect ratio cut-offs are illustrated in figure 5.5. Comparison of figure 5.5 with figure 5.2 suggests that the rose diagram for the grains with aspect ratio greater than 1.75 compares best with the results obtained by Cheel, the modes are  $45^{\circ}$  and  $45^{\circ}$  respectively. In a personal communication Cheel volunteered that "his aspect ratio is skewed to higher values, this is because of the difficulty in determining the grain boundaries visually". Therefore the apparent differences in the results obtained here and by Cheel probably reflect differences in technique. However, the results obtained here are objective and reproducible.

Computer-assisted microscopy yields more data than the manual measurements conducted by Cheel. In particular, the  $x, y$  coordinates of the points along the grain boundary are available and can be used to estimate a best-fit ellipse for each grain. Rose diagrams for the orientation angles of the long axes of the best-fit ellipses are presented in figures 5.6 and 5.7. It has been suggested that analyses based on the ellipses are hydrodynamically more meaningful.

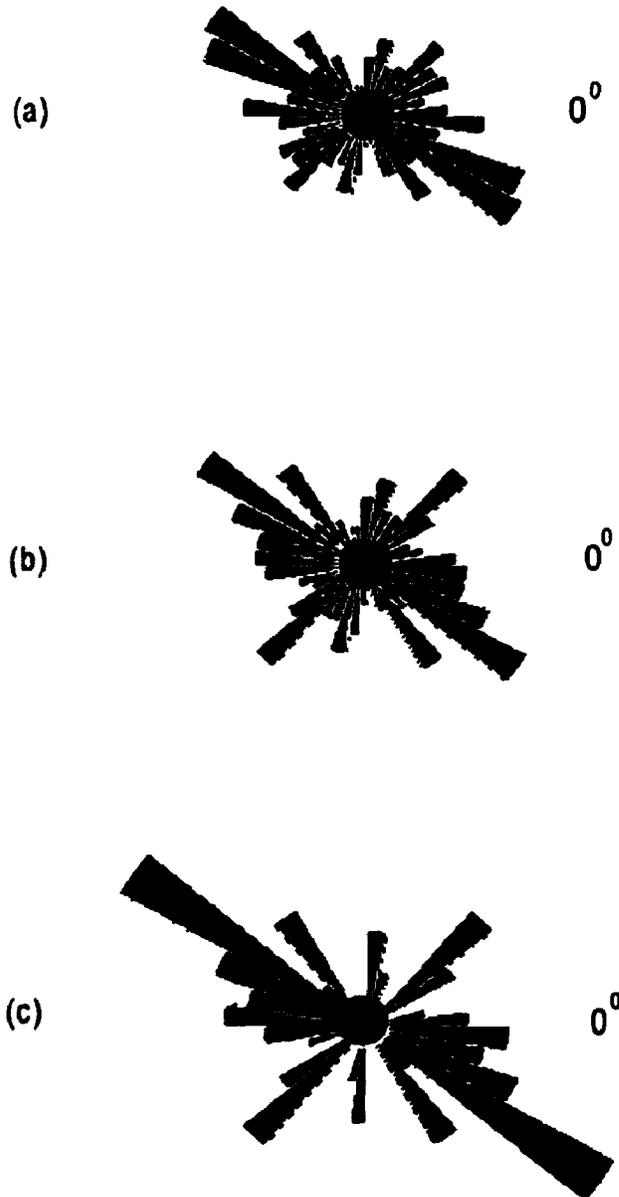


Figure 5.5 : Rose diagrams showing the orientation of the long axes of the three subsets of sample A. (a), (b) and (c) are the data for the grains with aspect ratios greater than 1.25, 1.5 and 1.75 respectively.

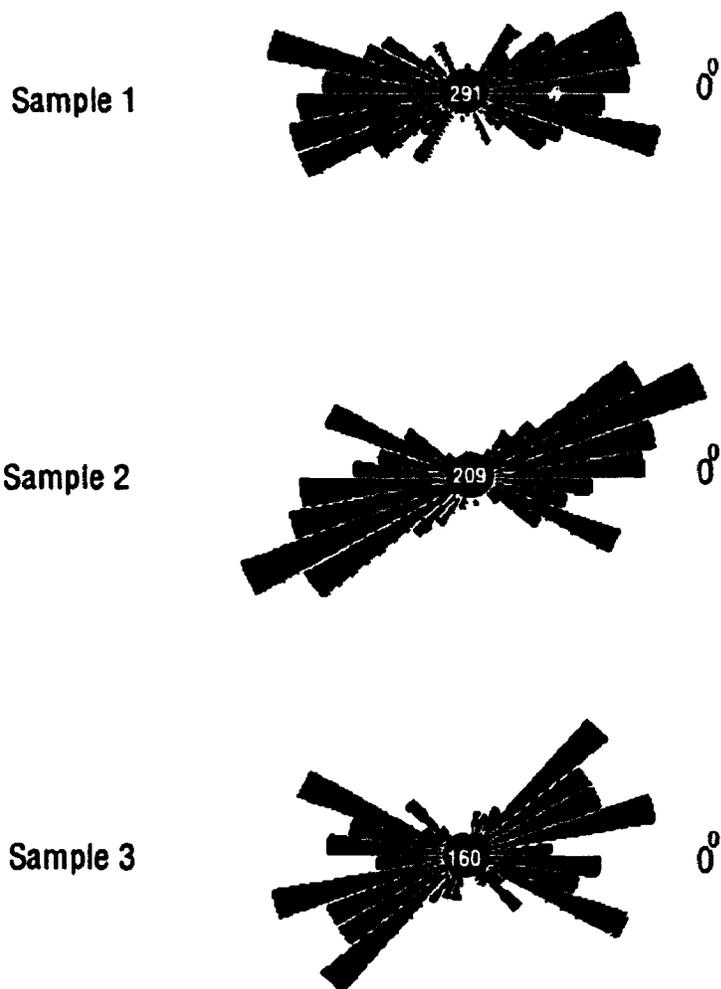


Figure 5.6 : Rose diagrams showing the orientation of the long axes of the best-fit ellipses fitted to the sand grains in three samples of beach sands. The values indicate sample sizes.

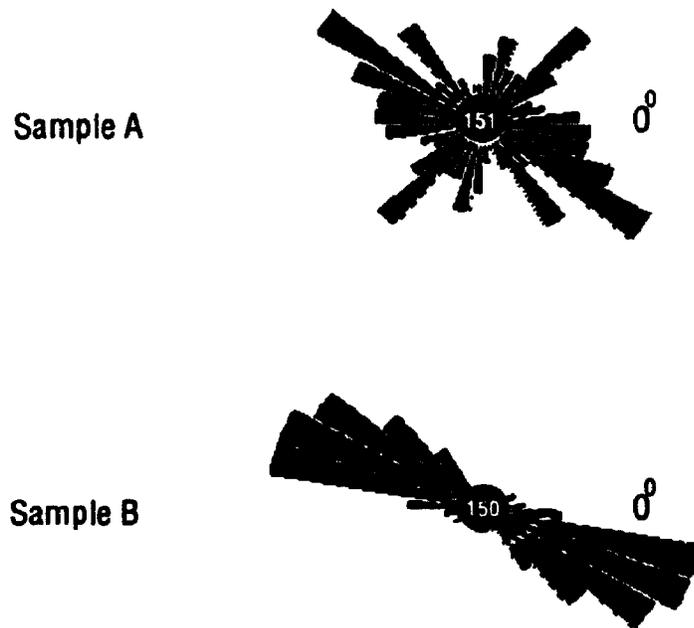


Figure 5.7 : Rose diagrams showing the orientation of the long axes of the best-fit ellipses fitted to the sand grains in two samples of beach sands. The values indicate sample sizes.

## 5.2 Shapes of Deformed Oolites

In a study of the deformation of initially nearly spherical particles in natural rocks, Borradaile and McArthur (1991) collected data from an experimentally deformed sample of Ancaster oolitic limestone. Their study involved the measurement of finite strain from the best-fit elliptical approximations of the ooid particles. The ooid particle boundaries were traced from a photomicrograph mosaic prepared with partial cross-polarized light, using a Zeiss Videoplan digital analyzer system (Borradaile and McArthur 1991). Ooid particles were carefully traced from the photomicrographs onto a high resolution digitizing tablet. The ooid particle boundaries were determined by following the outermost concave boundary, usually distinguished by a thin drusy cement layer between the dark concentric ring structure of the ooid and the bright block cement matrix. The Zeiss system was used to compute the maximum and minimum axes for the computer generated best-fit elliptical approximations to the traced particles. The results were used to prepare  $R_f/\phi$  diagrams.

The photomicrograph mosaic used in the Zeiss Videoplan digital analyzer was provided by McArthur (Borradaile and McArthur 1991). Two methods of image analysis are applied to the photomicrograph. In the first, the data are collected from the photomicrograph mosaic using image analysis techniques similar to those reported by Starkey and Simigian (1987, 1992). A tracing is made from the photomicrograph on a transparent plastic sheet. The boundaries of the ooid particles are determined in a way similar to the Zeiss method. The tracing is scanned as a binary image on a 300 dpi page scanner (this produces data similar to those produced by the system used by Starkey and Simigian 1992). The binary image is processed by IMAGE (Simigian and Starkey 1987) to compute the dimensional parameters for

the best-fit elliptical approximations of the traced ooid particles.

In the second analysis, the photomicrograph is scanned as a grey level image using the present imaging system. The ooid particle boundaries are detected using the Canny edge operator. The particles display internal texture, particularly concentric rings, therefore the Gaussian kernel for the Canny operator is chosen so as to ignore the internal texture. Minimal manual editing is required to complete the gaps in the particle boundaries where the edge operator cannot detect edges due to the poor quality of the photomicrograph mosaic. In a few instances particles are in contact and it is difficult for the edge operator to detect those boundaries because the particles on both sides of the boundary display similar grey level characteristics (See figure 5.8). However, by changing the window size of the edge operator it is possible to fill in some of these missing boundaries.

Visually, most of the particles are readily identified because the eye follows their convex, nearly elliptical outline and any breaks in the boundary are readily "filled in" by visual extrapolation. Computationally this can be emulated by the local fitting of mathematical functions such as polynomials to extend boundary lines across gaps. In the present analysis the gaps in the grain boundaries are spanned by a curved line based on a cubic spline interpolation which takes account of the shape of the boundaries near the gap. It would also be possible in this case, where the approximate shape of the particle is known a priori, to interpolate the edges by using a Hough transform (Russ 1990), but this is not implemented in the present study.

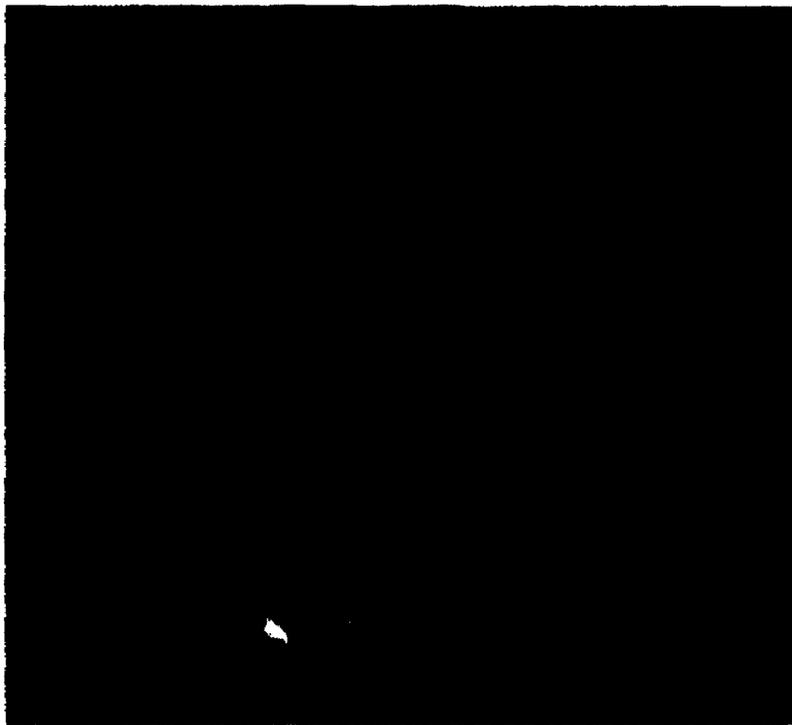


Figure 5.8 : Microscope image of a thin section showing oolites observed in plane polarized light. The edges detected by the Canny operator are shown in red.

The results of the three analyses, that using Zeiss Videoplan, that based on a tracing of the image and that using the imaging system developed here, are presented in figures 5.9, 5.10 and 5.11 respectively. These figures are  $R_f/\phi$  diagrams in which individual data points represent the aspect ratio ( $R_f$ ) and the angular orientation of the long axes ( $\phi$ ) for individual particles. The orientation angles are measured from the principal stress axis in the experiment and vary from  $-90^\circ$  and  $90^\circ$ . All three scatter diagrams are comparable. From the measured data, the mean  $\phi$  angles and the associated angular deviation can be calculated. Also an average strain ellipse and associated error can be calculated using Robin's method (Robin 1977). These parameters are listed in Table 5.1, they are very similar for all three analyses.

The results of all three analyses of the deformed oolitic limestone are essentially the same. However, the time required for the analysis using the present system is approximately 5 minutes, for the other two systems it is approximately 35 minutes for the same sample size. Further, application of the present system is less labour intensive.

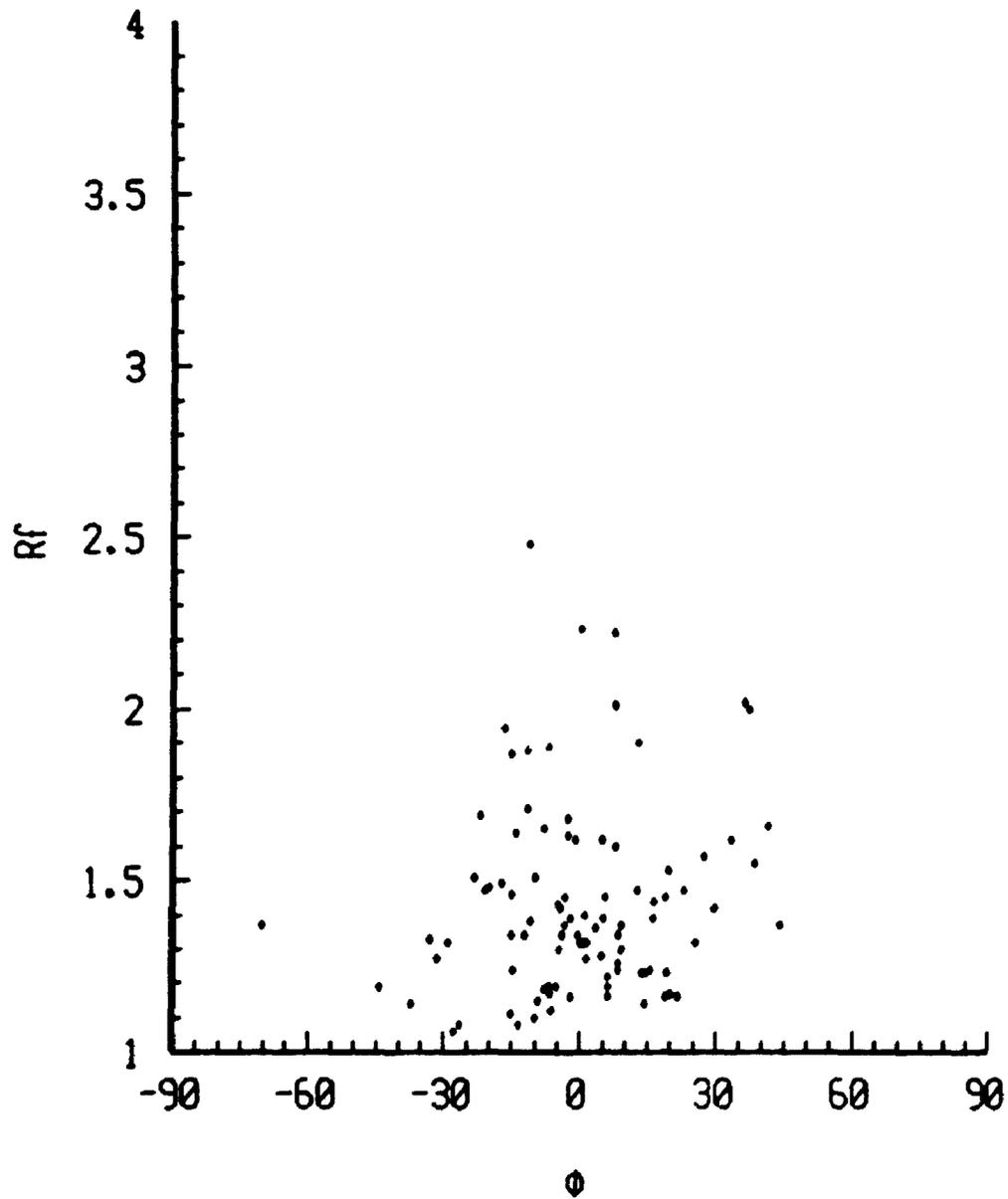


Figure 5.9 :  $Rf/\phi$  plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using a Zeiss Videoplan digital analyzer (see text).

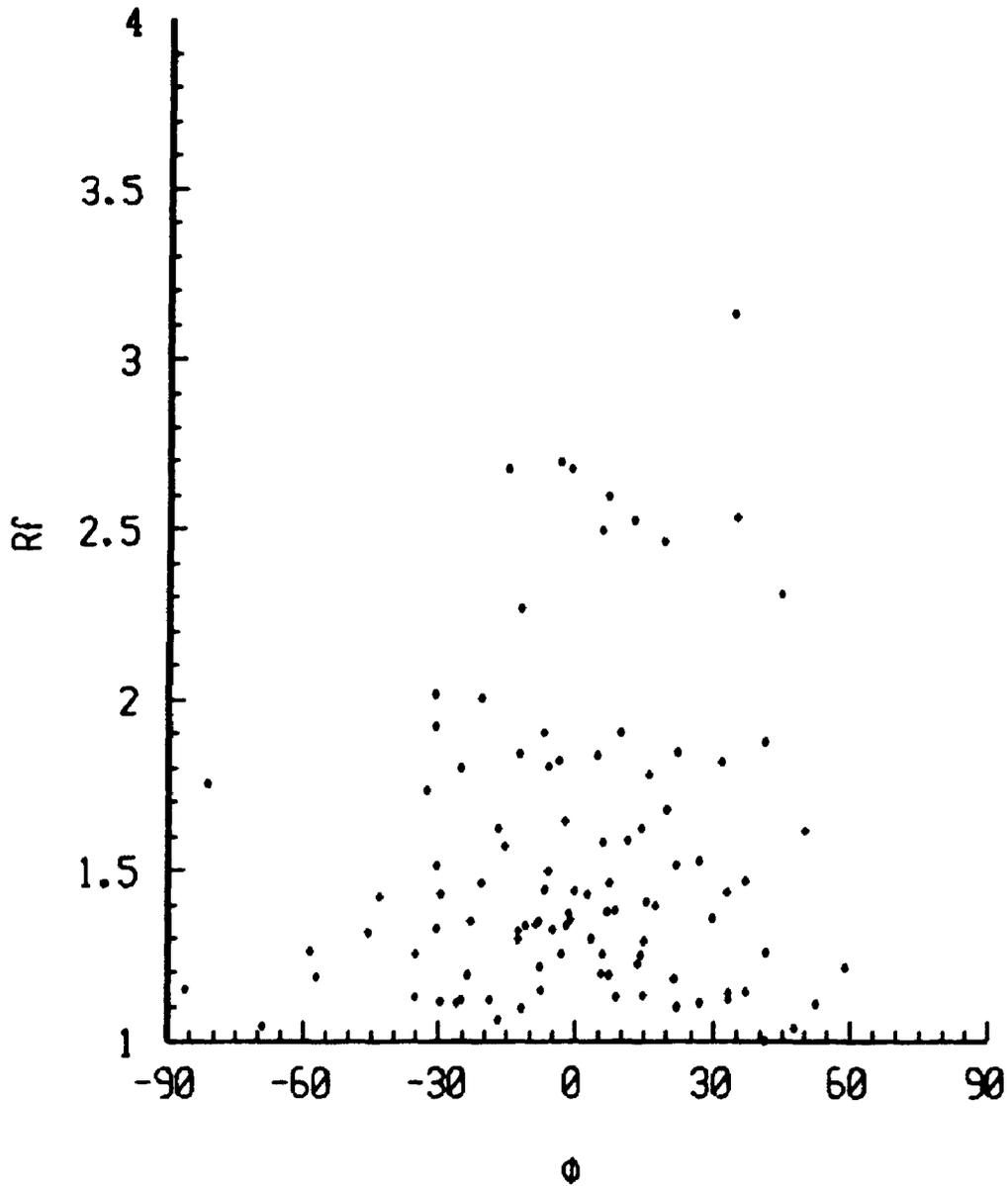


Figure 5.10:  $Rf/\phi$  plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using the image analysis system based on a tracing of the image (see text).

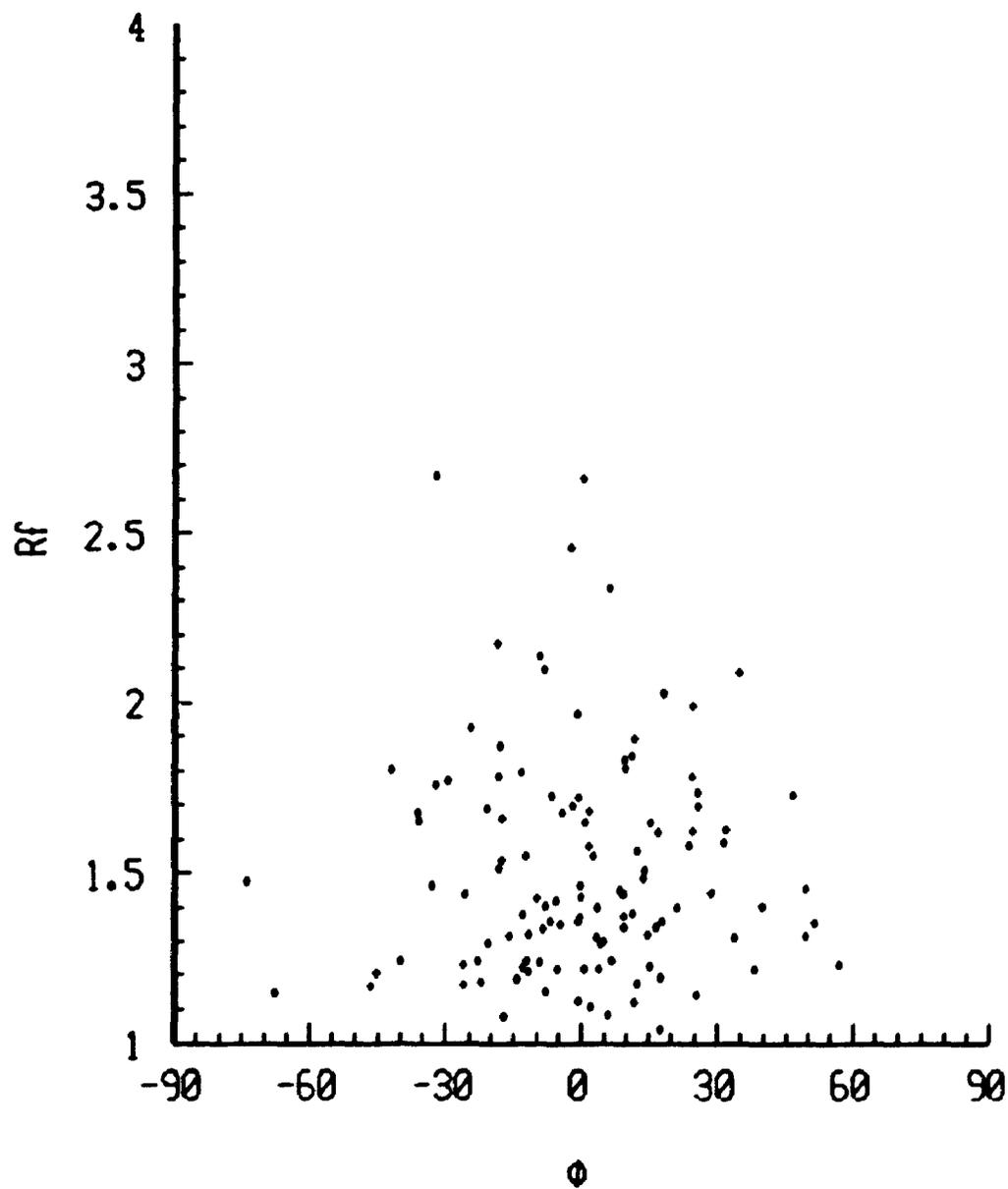


Figure 5.11 :  $Rf/\phi$  plot of the deformed oolites in a sample of Ancaster limestone. The data are obtained using the present imaging system (see text).

Table 5.1: Robin's strain, standard error,  $\phi$ -mean, and angular deviation measured from the ooid particles in an experimentally deformed oolitic limestone by three different methods of image analysis (see text).

Different Techniques	Robin's Strain	Standard Error	$\phi$ -mean	Angular Deviation
Ziess Videoplan System using Binary image	1.327	0.254	0.0°	11.70°
Present System	1.296	0.244	0.0°	14.00°
	1.310	0.228	0.0°	9.47°

### 5.3 Modal Analysis

In a study of the interdependence of the degree of quartz preferred orientation and the quartz content of deformed rocks, Starkey and Cutforth (1978) carried out a modal analysis of a Saxony granulite from the straight limb of an isoclinal fold, the plagioclase present was stained yellow to aid in its identification. The modal analysis for a biotite rich layer was as follows: 51.9% biotite, 26.5% quartz, and 21.5% plagioclase. The thin section used in the manual analysis was provided by Starkey and a modal analysis is carried out in the same region using image analysis.

Using plane polarized light, two images of the same field of view are obtained in which the planes of polarization of the polarizer are mutually perpendicular. The two images are shown in figure 5.12.a and b. Biotite is a pleochroic mineral, the grain which are dark in one image are light in the other, and therefore comparing the

two images to detect these grains which changed grey level allowed the biotite to be identified. The two images compared pixel by pixel to get the absolute differences and a threshold of 3 is applied to the matrix of differences to produce binary image shown in figure 5.12.c, white represents biotite.

The remaining grains are quartz and feldspar. The quartz, which is not stained, displays higher grey values. Therefore, a simple threshold separates the grey levels corresponding to quartz from those corresponding to plagioclase. Figure 5.12.d shows a binary image where white represents the quartz. This is obtained from the grey level image shown in figure 5.12.a after applying a threshold value of 70.

The resulting modal analysis from the present study is: 50% biotite and 24% quartz. The remaining 26% is attributed to plagioclase. Thus similar results to those reported by Starkey and Cutforth (1978) are obtained but the analysis is carried out much faster than traditional modal analysis.

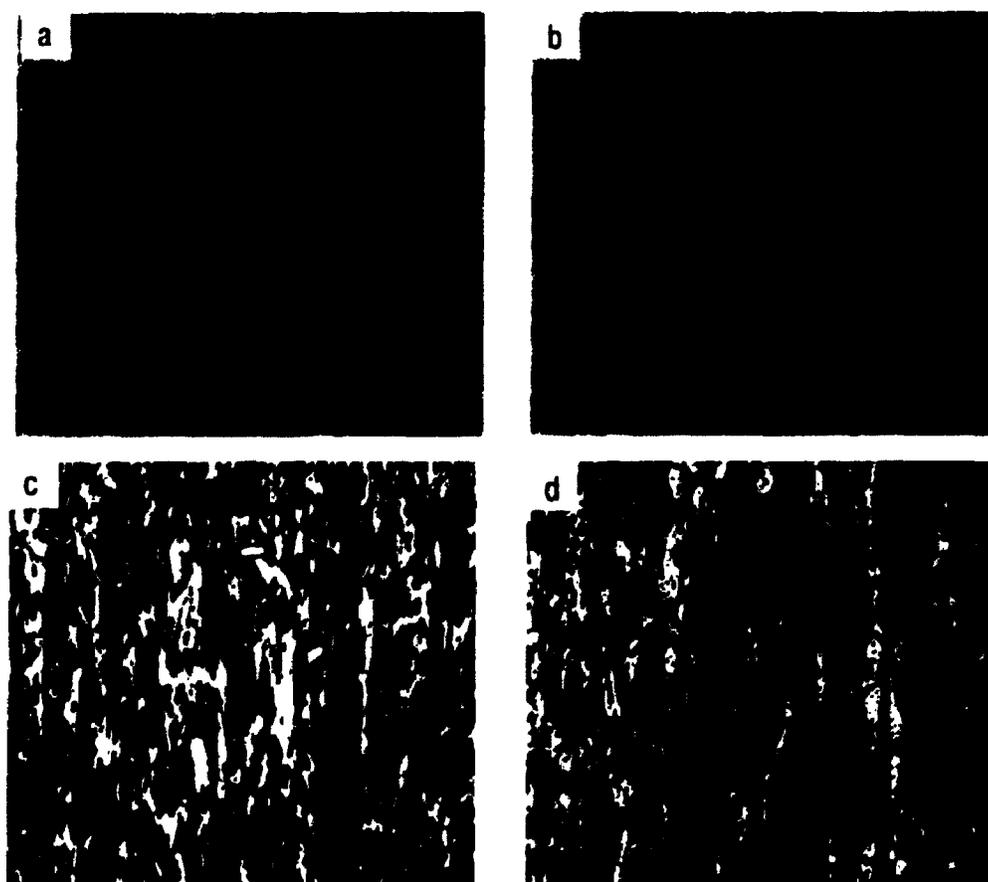


Figure 5.12 : (a) and (b) Two images captured from one microscope field of view of a thin section of Saxony granulite in plane polarized light. The orientations of the polarizer in the two images are mutually perpendicular. (c) Binary image obtained by thresholding the absolute difference between the images (a) and (b). White represents the biotite. (d) Binary image obtained after applying a threshold of 70 to the grey values in (a). White represents the quartz

# Chapter 6

## Conclusion and Future Work

### 6.1 Conclusion

Computer-assisted petrographic image analysis provides information which cannot be readily obtained by conventional petrography. Further, the petrographic images can be analyzed automatically, this allows the user to accumulate the large amounts of data required to conduct statistical studies. The quantitative data which are obtained, permit the analysis of the constituent mineral grains and pore spaces and thus provide a complete textural description of a rock.

The use of a video camera and frame grabber board allows the microscope image to be scanned with great precision, speed and in real-time. The area of interest in a thin section can be selected interactively and the image can be enhanced to display specific features using the Look-Up Tables on the frame grabber board. The image enhanced by manipulation of the grey levels in the image can be used to improve the detection of feature boundaries and extraction of textural parameters by the computer software.

Digital filtering techniques, particularly the application of the Sigma Filter, reduce the grey level variation due to the random electronic noise present in the image and to the inherent variations in the image. This enhances the intensity/texture discontinuities between feature regions which provides a basis for segmenting the image to detect the boundaries between features of interest.

The sequential capturing of images from one microscope field of view, using multiple pages of computer memory, allows a thin section to be observed in a manner analogous to the procedure followed in manual petrography. This allows the familiar techniques of petrography to be applied to the analysis. The implementation of this facility, as in the present system, permits the real-time analysis of the rock thin sections (see figures 3.21 to 3.23).

The feature extraction process identifies and selects the features from the segmented image. They are described by either an ordered list of the  $x, y$  coordinates of their boundary points or by using the eight-direction Freeman chain code. These descriptions are required to obtain the geometrical characteristics of the features, including their sizes, shapes, orientations and distribution. In addition to the feature boundary information, the original grey level image and the intermediate edge image are also available for use in obtaining the spatial distribution and textural descriptions of the features.

Petrographic image analysis provides objective and reproducible data far more readily than can a human observer. Any variations which may exist in the measurements are more or less directly related to simple statistical patterns of fluctuations, these errors can be predicted and in many cases controlled.

In addition to the analysis of images from a petrographic microscope, the

present image processing system can be applied to any grey level images including those produced by a scanning electron microscope (SEM), microprobe (MP), transmission electron microscope (TEM), or macro images from photographs. Images from the SEM, MP, and TEM can be input into the system directly via an SEM interface. Other images can be input using a video camera or video cassette recorder.

The petrographic image, the computed images and the extracted feature information can be transmitted electronically, since they are all digital. The microscope images are stored as PCX image files which can be read into many different commercial software packages.

## 6.2 Future Work

Many investigators see the need for both rapid and objective quantitative petrographic analysis but is computer processing of petrographic images really necessary? Should we consider the present study as an isolated venture, destined to remain in the records of methods searching for an application? A complete answer to such questions cannot be provided by this study alone. Nevertheless, the applications presented here suggest that many quantitative petrographic problems can be addressed by digital image analysis.

Digital image processing techniques can be applied to quantitative petrographic measurements which are often not possible using conventional methods such as fitting best-fit ellipses, deriving fourier descriptors of shapes, quantizing the mineral association parameters etc. Unfortunately, relatively few studies have been conducted into the application of image processing to quantitative petrography. This can be attributed in part to the lack of understanding of image analysis and pattern

recognition by geoscientists and in part to their desire for "all or nothing" strategies. The ambitious goal of designing machines with human-like capabilities may take a century to achieve (Trivedi and Resenfeld 1989, Pavlidis 1992). However, progress in this area can only be achieved by investigating a variety of individual petrographic problems and trying to solve them by specific image processing techniques. Figure 6.1 presents an overview of the probable future development of computer assisted petrographic analysis based on image processing/analyzing techniques. The shaded area in the figure indicates where further research is required.

It is generally assumed that edge detecting algorithms should always find all the feature outlines in an image and they are often evaluated on how well they accomplish this. Further, this evaluation is usually based on the assumption that the features have a different brightness level to that of their background. This latter assumption is only valid for images where features are detected simply by their grey level intensity and the present study shows that the Canny edge detector did very well in detecting such intensity edges present in an image. However, the human eye identifies many features on the basis of texture differences, a geological example is the identification of the outline of a complexly twinned microcline grain (see figure 3.4).

Texture is concerned with the spatial distribution of image intensities and discrete tonal features. A discrete tonal feature in a digital image is a connected set of pixels, all of which have the same, or almost the same, image intensity (Haralick and Shapiro 1991). When a small area of the image shows little variation in discrete tonal features, the dominant property of the area is a grey tone. When a small area has wide variation of discrete tonal features, the dominant property of that area

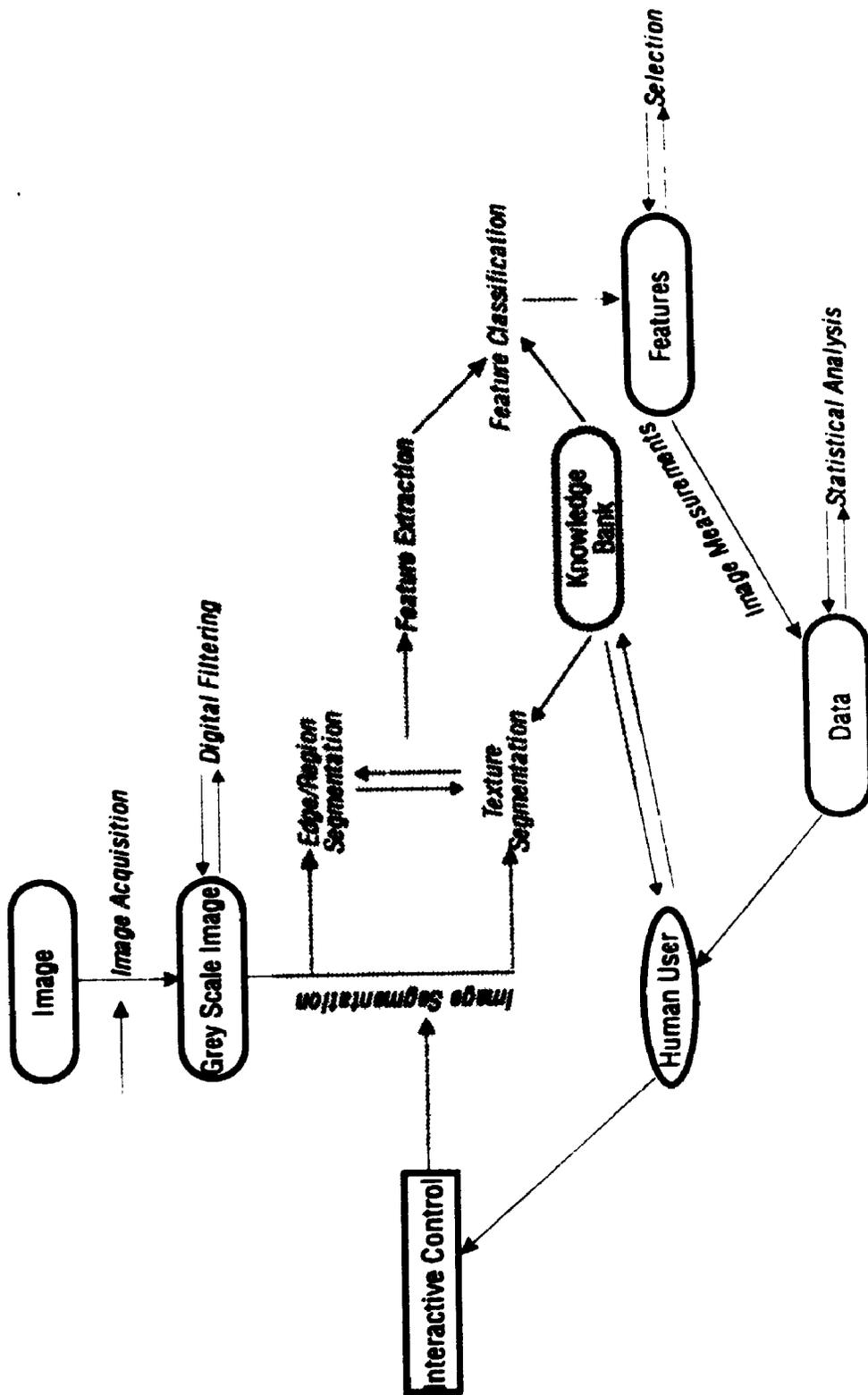


Figure 6.1 Proposed organization of future computer based petrographic analysis  
The shaded area indicates where future research is required

is texture. Texture can be classified according to the size of these small areas, the relative sizes of the discrete tonal features and the number of distinguishable discrete tonal features.

Textured regions in an image can be segmented using statistical measures or structural descriptions. Statistical texture measures include the moments of the grey levels of the given region, typically the variance, the skewness and the kurtosis (de Souza 1983, Suk and Hong 1984). The texture statistics for a given region can also be estimated by means of second-order probability density functions. These functions compute an intermediate matrix of measures from the digitized image and then define features using functions based on this intermediate matrix (Ballard and Brown 1982). The intermediate matrices include the normalized grey level dependence matrix, the neighbouring grey level difference matrix, the grey level run-length matrix etc. (Weszka et.al. 1976, Ballard and Brown 1982, Lee, Lee and Kim 1992). A structural texture description is given by a set of primitives and placement rules which govern the stochastic spatial relation between them (Haralick and Shapiro 1991). In view of the importance placed by geoscientists on texture in their identification of minerals, e.g. microcline, plagioclase etc., texture based feature segmentation of the petrographic images is necessary.

The human eye identifies many features not only on the basis of texture differences, but also on the basis of geometrical parameters such as their size, shape and orientation etc. For example, the ooid particles in figure 5.8 are identified not only by their texture, but also by their near elliptical shape. Template matching techniques can be used to simulate this digitally. A template sub-image can be defined against which to test a window of the given test image for a match. The degree of matching can be determined by translating the template sub-image over test im-

age and, at each position, evaluating the cross-correlation or the sum of the squared or absolute image intensity differences of corresponding pixels. Similarly, parameters such as area, orientation and shape can be used for segmenting features in an image, this is called symbolic matching. In most matching techniques, the shapes of the features to be recognized from an image are known and can be presented to the imaging system via a knowledge bank (see figure 6.1). The knowledge bank can be a part of an expert system, which include the system's working knowledge, an inference engine, which directs the processes of reasoning or interpretation, and a user interface for segmenting and extracting features from the image (see figure 6.1).

The computer processing of petrographic images requires a layered approach. First level processing includes image enhancement, noise smoothing, extraction and analysis of edges and features. Second level processing includes characterization of the texture, shading etc. Third level processing includes the matching, feature recognition/classification etc.

Computer-assisted petrographic image analysis offers the possibility of developing more rigorously quantized petrographic analysis. Automation will allow data to be obtained rapidly and objectively, thus making it possible to accumulate the large amounts of data required to conduct statistical studies.

# **Appendix 1**

## **Evaluation of Digital Filters**

## An Evaluation of Noise Reduction Filters, with Particular Reference to Petrographic Images

John Starkey<sup>1</sup> and Abani Kanta Samantaray<sup>1</sup>

---

Fourteen digital filters are compared. These filters are differentiated as single level and multi-level filters and are evaluated using computer simulated images and real images obtained from a petrographic microscope. The computer simulated images consist of uniform grey and bar pattern images and a pattern of 16 square areas with constant grey levels. The filters are applied both as single pass and multiple pass filters using a 5 by 5 pixel window. Multiple passes of three iterations are applied to the uniform grey and bar pattern images. The single level filters are also evaluated for different window sizes and single pass operation on the uniform grey image. The evaluation of the filters assesses their ability to reduce normally distributed additive noise, preserve linear features and image texture, retain edges between regions, preserve shapes of features, and remove spot noise. Computer processing time is also a factor. The Recursive Separable Median Filter performs very well and better than most of the other filters for a single pass operation. However, if the preservation of small linear features and image texture is crucial, then recursive application of the In-Place Growing-FMH and Sigma Filter are to be preferred.

---

**KEY WORDS:** Polarizing microscopy; image processing; noise reduction; digital filters; petrographic images.

### INTRODUCTION

Recently published methods of digitizing grain boundaries from thin sections observed with the polarizing microscope have been based either upon manually traced line drawings of the boundaries (Starkey and Simigian, 1992; Fabbri, 1984) or on grain boundaries drawn on the image displayed on a video monitor. The automatic digitization of grain boundaries directly from video camera images obtained from the polarizing microscope is hindered by the inherently low contrast of the images obtained from typical rock specimens. To enhance the possibility of having the

computer recognize the grain boundaries in such an image a digital filter must be applied to reduce the noise. At the same time, the operation of the filter must not adversely effect any contrast that does exist across the grain boundaries nor alter the grain shapes. Launeau *et al.* (1990) addressed these problems by the application of a low pass filter (erroneously referred to as a high frequency filter) followed by contrast enhancement. However, there is no discussion of the efficacy of any of the digital filters.

The noise present in a video image can be considered as spatially uncorrelated and random. Thus, in images of a thin section obtained from a polarizing microscope, Figs. 1a and 12, the image regions corresponding to individual grains can be considered to be the areas of constant grey level, corresponding to the signal, on which random noise is superimposed. This noise has the characteristics of random additive errors

---

<sup>1</sup> Department of Geology, University of Western Ontario, London, Ontario N6A 5B7, Canada

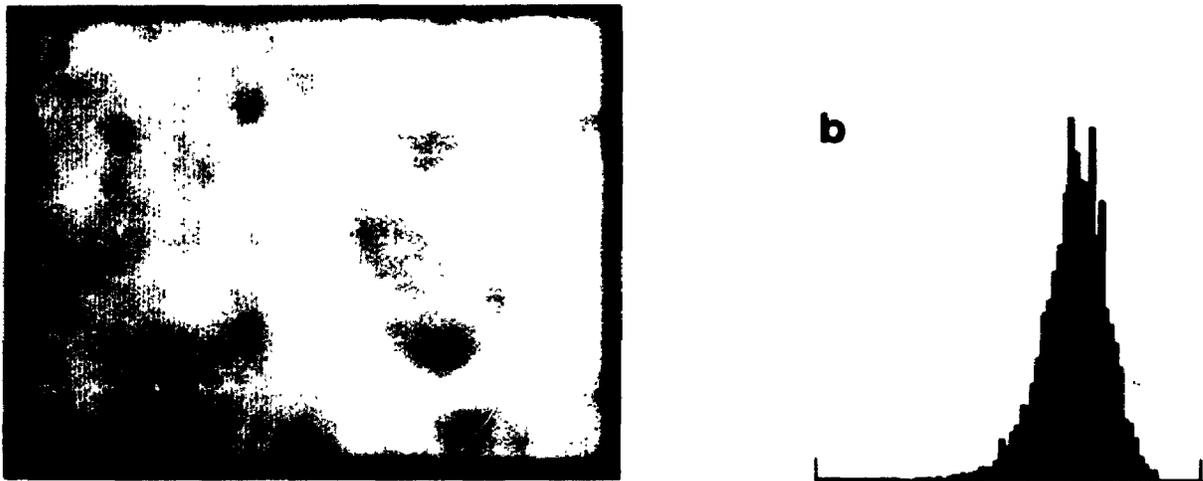


Fig. 1. (a) Photomicrograph of the interior of a Quartz grain taken with a polarizing microscope using crossed polarizers. (b) Frequency distribution of the grey levels present in the image shown in (a)

where the noise level is independent of the underlying average grey level. The noise levels can be assumed to have a Gaussian distribution, with a mean of zero, superimposed on the signal. Thus, the mean of the distribution of the grey levels within a grain is equal to the mean grey level of the underlying grain, and the range of grey levels are normally distributed about the mean (Fig. 1b). Filtering is necessary to retrieve the noise free image from the observed image.

The inherent noise levels of the imaging system used in the following analyses have a standard deviation of only 1.5. Therefore, the images were not subjected to frame averaging during their capture. This provides an opportunity to investigate the efficacy of the filters on the raw image.

The techniques which have been proposed for image noise reduction fall into two categories. In the first, the noisy image is processed globally where the whole, or a large section, of the image is correlated to obtain a smoothed image (Rosenfeld and Kak, 1976, Andrews and Hunt, 1977). The second group of techniques uses local operators. Here, the filter output at a specific pixel is a function of the input values of the pixels within a selected neighborhood, for instance, the eight surrounding nearest neighbors. These latter techniques tend to have shorter computation time, since only a small number of input pixel values are considered for each output pixel. They are the topic of this discussion.

#### NOISE REDUCTION FILTERS

The characteristics of a successful noise filter must include the following: (1) effectiveness in noise reduction—this means an ability to reduce the variance resulting from noise in a uniform image region, (2) preservation of subtle details of clusters of several pixels and linear features of a few pixels width, (3) immunity from shape distortion—in particular, the filter must not introduce significant distortion in the shapes of the image regions, (4) retention of intensity step and ramp edges between adjacent regions of uniform, but different, intensities, (5) removal of sparsely distributed, sharp, spot noise consisting of one or two pixels, and (6) computational speed.

The filters chosen for this study include both single and multi-level filters. The single level filters which appear most commonly in the literature are the Mean, Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma, K-Nearest Neighbor Average, Nearest Neighbor Median, and Weighted Median Filters. The popularity of these filters may be attributed to several factors, including ease of implementation and conceptual simplicity. Multilevel filters combine the output of basic subfilters and include the Multistage Median, Max, Min Median, FIR-Median Hybrid (FMH), In-Place Growing FMH (IPG-FMH), and Morphological (2DCO) Filters. Multilevel filters have received considerable

## Evaluation of Digital Filters

173

attention recently because of their ability to smooth images while preserving texture.

## THE FILTERS

### Single-Level Filters

Single-level filters process the data in a local neighborhood, or window, which is usually square and the sides of which are an odd number of pixels. The window is moved over the entire image and the output from the filter at the central pixel position is used to create the filtered image.

#### *Mean Filter*

With the Mean filter the pixel values within a square window are used to compute an average value which becomes the filter output at the central pixel position. This filter reduces the effects of random noise by reducing the grey level variation in the image. However, the edges between image regions become blurred.

#### *Median Filters*

Median filtering was first used by Tukey (1977) in time series analysis and later applied to image smoothing by Pratt (1978). The central pixel value is replaced by the median value within the window. The assumptions behind this filter are that a constant neighborhood is a region in which the majority of the grey levels represent signal with zero noise and that an edge is a monotonically rising or falling sequence of pixel values which separates two constant neighborhoods. The Median Filter can be applied repeatedly using the previously filtered image as input. In this case, for a given window size, the filter values converge on to what has been termed the root structure (Gallagher and Wise, 1981).

The Recursive Median Filter is a modification of the Median Filter in which the grey levels of the image are replaced by the output of the filter as the process proceeds. Therefore, at any one time, approximately half the values within the window are previously filtered values and the remainder are the original grey levels. The recursive operation determines the root structure of an image on the first pass. However, this root is different from that obtained by repeated passes of the nonrecursive Median Filter although typically they are very similar (Arce *et al.*, 1986).

Median filters, which are typically based on square windows, are known to be anisotropic and

cause edge shifts (Hodgson *et al.*, 1985; Arce *et al.*, 1986). Further, the corners of image regions may be rounded off and patches of pattern may be produced in the filtered image. These artifacts persist even if the square window is replaced by a circular, or near circular, approximation in which the pixel values are weighted according to their distance from the central pixel. These problems are alleviated by the Separable Median Filter (Narendra, 1981), which consists of two one-dimensional filters, applied in a two-pass operation. The rows of the two-dimensional image are filtered first by a horizontally oriented one-dimensional median filter to produce an intermediate image. The columns of this image are then filtered by a vertically oriented one-dimensional median filter.

The Separable Median Filter can also be applied recursively. The root structure produced again differs from that of the nonrecursive filter but the recursive filter has been shown to be better at noise suppression and edge preservation (McLoughlin and Arce, 1987). The Recursive Separable Median Filter (RSM) was used in the present study.

#### *Nagao and Matsuyama Filter*

Nagao and Matsuyama (1979) proposed a filter based on a 5 by 5 pixel window. The central pixel value is replaced by the average grey level of the most homogenous subregion from among the nine possible subregions illustrated in Fig. 2. A homogenous neighborhood is defined as one which does not contain sharp edge. If an area contains a sharp edge, the variance of the grey levels in that area is large. Therefore, the variance is used as a measure of homogeneity of the subregions and the central pixel is replaced by the mean of the subregion which has the minimum variance. This allows noise to be reduced without blurring sharp edges since averaging is not applied to a subregion which contains an edge.

#### *Gradient-Inverse Filter*

The Gradient Inverse Filter proposed by Wang *et al.* (1981) assumes that the variation of grey levels within a homogenous region of an image is less than that between different regions, and that the edges between regions are sharply defined. The central pixel value is replaced by the weighted average of all the grey levels inside the window. The weighting coefficients are obtained by computing the inverse gradients between the center and its neighboring pixel values and normalizing them by dividing by the sum of the inverse gradients in the window. Therefore, pixel values having a greater absolute difference from

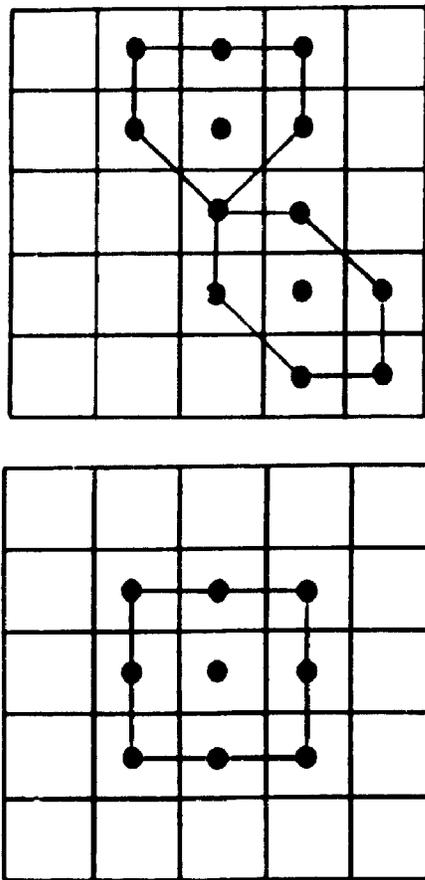


Fig. 2. The nine possible subregions for the Nagao and Matsuyama Filter. One example from each of two sets of four possible subregions are shown on the top. The ninth possible subregion is shown on the bottom.

the central pixel have lower weighting coefficients and contribute less to the output central pixel value. Where the length of the side of the square window is  $2N + 1$  the filter has the following form,

$$Y(i, j) = \sum_{p=-N}^N \sum_{q=-N}^N W(i+p, j+q) \times X(i+p, j+q) \quad (1)$$

Where either  $W(i+p, j+q) = \frac{1}{2}$  for  $p=0$  and  $q=0$  or

$$W(i+p, j+q) = \frac{1}{2} \left\{ \sum_{1, \dots} (i, j, p, q) \right\}^{-1} \times \sigma(i, j, p, q)^{-1} \quad (2)$$

for  $p$  and  $q = -N, \dots, 0, \dots, +N$ ; and  $p$  and  $q$  not 0 simultaneously.  $V_{(i, j)}$  denotes all the pixels within the window.  $\sigma(i, j, p, q)^{-1}$  represents the inverse of the absolute gradient of the neighboring pixel with  $X(i, j)$ .

*Lee-Additive Filter*

The Lee-Additive Filter (Lee, 1980, 1981) assumes that the correct value of a pixel is the mean of the values within the window and that the variation from this value is the same as the variance of all pixel values within the window. The *a priori* mean and variance of the estimated image can be calculated as the difference between the mean and variance of the noise corrupted image and the mean and variance of the noise alone. The value of a pixel  $X(i, j)$ , in a noise corrupted image, is given by the following equation,

$$X(i, j) = Y(i, j) + W(i, j) \quad (3)$$

Where  $W(i, j)$  is the white random noise with zero mean and  $\sigma^2$  variance and  $Y(i, j)$  is the pixel value before degradation.

The estimated pixel value,  $\hat{Y}(i, j)$ , in the uncorrupted image is given by

$$\hat{Y}(i, j) = \hat{Y}(i, j) + k(i, j) \{ X(i, j) - \hat{Y}(i, j) \} \quad (4)$$

Where  $\hat{Y}(i, j)$  is the approximated uncorrupted image mean given by  $\hat{Y}(i, j) = \bar{X}(i, j)$ . The gain factor  $k(i, j)$  is defined by

$$k(i, j) = \frac{Q(i, j)}{\{ Q(i, j) + \sigma_1^2 \}} \quad (5)$$

where

$$Q(i, j) = E \{ \{ X(i, j) - \bar{X}(i, j) \}^2 \} - \sigma_1^2 \quad (6)$$

$\sigma_1^2$  is an estimate of the additive noise variance.

The application of this filter requires estimates of the mean and variance of the pixel values within the window and of a global noise variance. In the present study, the global noise variance is derived from an arbitrarily selected sensibly "flat" intensity region in the displayed image.

*Sigma Filter*

The Sigma Filter (Lee, 1983) is suggested by the sigma probability of a Gaussian distribution. Here, the central pixel value is replaced by the average of those pixel values inside the window which lie within a fixed sigma range of the central pixel grey level. Assuming that the noise has Gaussian distribution

**Evaluation of Digital Filters**

and the *a priori* mean is the grey level of the central pixel then 95.5% of the pixel values will lie within two standard deviations of the mean. Pixel values outside this range are assumed to belong to a different population. The algorithm consists of computing the two sigma range for a local neighborhood, and replacing the central pixel by the average of the values which are within that two sigma range.

The two sigma average does not smooth the sharp spot noise of one or two pixels. In the present study, the minimum number of pixels "K" required for the two sigma average within a window is equal to "N" for a square window 2N + 1 on a side. If the number of pixels within the intensity range of two sigma is less than or equal to "K," the two sigma average is replaced by the average of the center pixel's eight nearest neighbors.

*K-Nearest Neighbor Averaging (KAVE) Filter*

The *K-Nearest Neighbor Averaging Filter*, was suggested by Davis and Rosenfeld (1978). This filter relies on the high correlation which must exist between the grey levels of pixels belonging to the same population within a local area. The central pixel value "X" of the window is replaced by the average grey level of the "K" nearest neighbors with grey levels closest to that of "X." This filter can be used iteratively, changing the window size and value of "K." Davis and Rosenfeld (1978) suggested K = 6 for a 3 by 3 pixel window.

*Nearest Neighbor Median (NNM) Filter*

The Nearest Neighbor Median Filter was first applied to image smoothing by Itoh *et al.* (1988). For a given window size of 2N + 1 the pixel values in the window are ranked in ascending order. The number of *K*-nearest neighbors are selected equal to 2M + 1, where M ≤ N. In this analysis, using a 5 by 5 window, K is set to 11. Within the rank ordered array R, the position of the central pixel value, *c*, is located. The output of the NNMF, Y(P), can be defined as follows (Asano *et al.*, 1990) depending on *c*:

Where  $\{M + 1\} < c < \{(2N + 1) - (M + 1) + 1\}$  then  $Y(P) = \text{Center Pixel Value}$ . Where  $1 \leq c \leq \{M + 1\}$  then  $Y(P) = R[M + 1]$ . Where  $\{(2N + 1) - (M + 1) + 1\} \leq c \leq \{2N + 1\}$  then  $Y(P) = R[(2N + 1) - (2M + 1) + 1]$ .

*Weighted Median (WM) Filter*

The Weighted Median Filter was first applied to image smoothing by Brownrigg (1984, 1986). Non-

negative multipliers are applied to the pixel values in the window and the median value is selected as the output. In this analysis, using a 5 by 5 window, the multipliers are taken as unity except for that of the central pixel which is set at 15

**Multilevel Filters**

Several multilevel filters based on the Median Filter have been proposed (Nieminen *et al.*, 1987) which combine the output of basic subfilters to match the structure spanned by the filter's window. They involve the use of the 4 subwindows,  $W_{(1-4)}$ , shown in Fig. 3. Such subwindows are unidirectional because the subwindows span one direction only. For the application of unidirectional multilevel filters, the median values,  $Z_{(1-4)}$ , of the subwindows are required (Arce and McLoughlin, 1987). For a recursive operation the grey levels of the image are replaced by the output of the filter as the process proceeds

*Multilevel Median Filter*

The output for the Multilevel Median Filter is defined as follows (Nieminen and Neuvo, 1988; Arce and Foster, 1988)

$$Y(t, j) = \text{median} [ Y_{(1-2)}(t, j), Y_{(3-4)}(t, j), X(t, j) ] \quad (7)$$

where

$$Y_{(1-2)}(t, j) = \text{median} [ Z_1, Z_2, X(t, j) ] \quad (8)$$

$$Y_{(3-4)}(t, j) = \text{median} [ Z_3, Z_4, X(t, j) ] \quad (9)$$

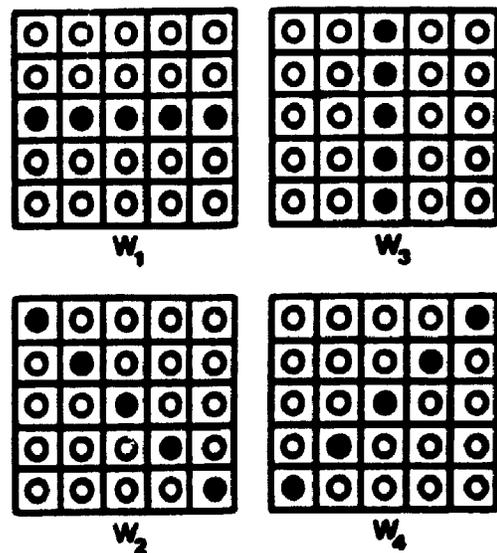


Fig. 3. Subwindows used by unidirectional multilevel filters

where  $X(i, j)$  is the central pixel value and  $Z_{1,1}, \dots, Z_{4,4}$  are the median values in the four subwindows (Arce and McLoughlin, 1987). For the present study a recursive operation of the Multilevel Median Filter is used.

#### Max/Min Median Filter

A modification of the Multilevel Median Filter is presented by Wang and Wang (1990) as the Max/Min Median Filter. The output for the filter is either

$$Y(i, j) = T_1(i, j); \text{ if } |T_1(i, j) - T_0(i, j)| \geq |T_2(i, j) - T_0(i, j)| \quad (10)$$

or

$$Y(i, j) = T_2(i, j); \text{ if } |T_1(i, j) - T_0(i, j)| < |T_2(i, j) - T_0(i, j)| \quad (11)$$

where  $T_0(i, j)$  is the median of all the pixel values within the window.  $T_1(i, j)$  and  $T_2(i, j)$  are, respectively, the maximum and the minimum median values in all the subwindows. In the present analysis a recursive Max/Min Median Filter is used.

#### FIR-Median Hybrid (FMH) Filter

The output for the FMH Filter is defined as follows (Nieminen *et al.*, 1987; Arce and Foster, 1988, 1989)

$$Y(i, j) = \text{median}\{\bar{Y}_{n_1 \times n_3}(i, j), \bar{Y}_{n_2 \times n_4}(i, j), X(i, j)\} \quad (12)$$

where

$$\bar{Y}_{n_1 \times n_3}(i, j) = \text{median}\{\hat{Z}_1, \hat{Z}_3, X(i, j)\} \quad (13)$$

$$\bar{Y}_{n_2 \times n_4}(i, j) = \text{median}\{\hat{Z}_2, \hat{Z}_4, X(i, j)\} \quad (14)$$

where  $\hat{Z}_p = \text{median}\{Z_{p_a}(i, j), Z_{p_b}(i, j), X(i, j)\}$ ,  $1 \leq p \leq 4$ , and where  $Z_{p_a}$  and  $Z_{p_b}$  are the averages of the values in each of the two parts of the subwindow  $W_p$ , separated by the center pixel.  $X(i, j)$  is the central pixel value. For a recursive operation of the filter,  $Z_{p_a}$  is replaced by the filter output from the previous window operation. In the present analysis a recursive operation of the FMH filter is used.

#### In-Place Growing FMH (IPG-FMH) Filter

This is an extension of the FMH filter which was first applied to one-dimensional signals by Wichman *et al.* (1990). The output for the filter is defined here for a two-dimensional image as

$$Y_k(i, j) = \text{median}\{\bar{Y}_{k(n_1 \times n_3)}(i, j), \bar{Y}_{k(n_2 \times n_4)}(i, j), Y_{k-1}(i, j)\} \quad (15)$$

where

$$\bar{Y}_{k(n_1 \times n_3)}(i, j) = \text{median}\{\hat{Z}_1, \hat{Z}_3, Y_{k-1}\} \quad (16)$$

$$\bar{Y}_{k(n_2 \times n_4)}(i, j) = \text{median}\{\hat{Z}_2, \hat{Z}_4, Y_{k-1}\} \quad (17)$$

where  $\hat{Z}_p = \text{median}\{Z_{p_a}(i, j), Z_{p_b}(i, j), Y_{k-1}(i, j)\}$ ,  $1 \leq p \leq 4$ , and where  $Z_{p_a}$  and  $Z_{p_b}$  are the averages of the values in each of the two parts of the subwindow  $W_p$ , separated by the center pixel.  $K$  is number of growth cycles performed at each location.

$Y(m, n) = Y_{k-1}(m, n)$ , where  $k = 1, \dots, M$ . Where  $k = 1$  then  $Y_{k-1}(m, n) = X(i, j)$ . The size of the subwindow increases with  $k$ . For a recursive operation of the filter  $Z_{p_a}$  is replaced by the filter output from the previous window operation.

In the present analysis, a standard IPG-FMH filter is used and  $M$  is set to 2. The window sizes used are 5 by 5 and 7 by 7 with  $k$  equal to 1 and 2, respectively. The output, using Eq. (15) is as follows

$$Y(i, j) = \text{median}\{\bar{Y}_{2(n_1 \times n_3)}(i, j), \bar{Y}_{2(n_2 \times n_4)}(i, j), Y_1(i, j)\}$$

where

$$Y_1(i, j) = \text{median}\{\bar{Y}_{1(n_1 \times n_3)}(i, j), \bar{Y}_{1(n_2 \times n_4)}(i, j), X(i, j)\}$$

#### Two Dimensional Close-Open (2DCO) Filter

Morphological filters such as this have the ability to preserve the details in images, they are closely related to ranked-order based filters (Maragos and Schafer, 1987). For a window with side length  $2N + 1$ , each subwindow  $W_p$  is partitioned into  $N + 1 = k$  overlapping subsets,  $S_{p,k}$ , of  $N + 1$  consecutive elements. For a 5 by 5 window there are therefore 3 subsets derived from each of the 4 subwindows for a total of 12. Grey scale opening is performed by replacing the central pixel value in the moving window by the maximum value among the minima in the subsets. This generates an intermediate image on which grey scale closing is performed by replacing the central pixel value in the moving window by the minimum value among the maxima in the subsets. This is known to preserve the image geometry.

#### EVALUATION OF THE FILTERS

For the purpose of evaluation the filters are divided into two groups. The single level filters except the NNM and WM Filters are evaluated for their ability to reduce image noise and to preserve linear features and the grain boundaries. The NNM and

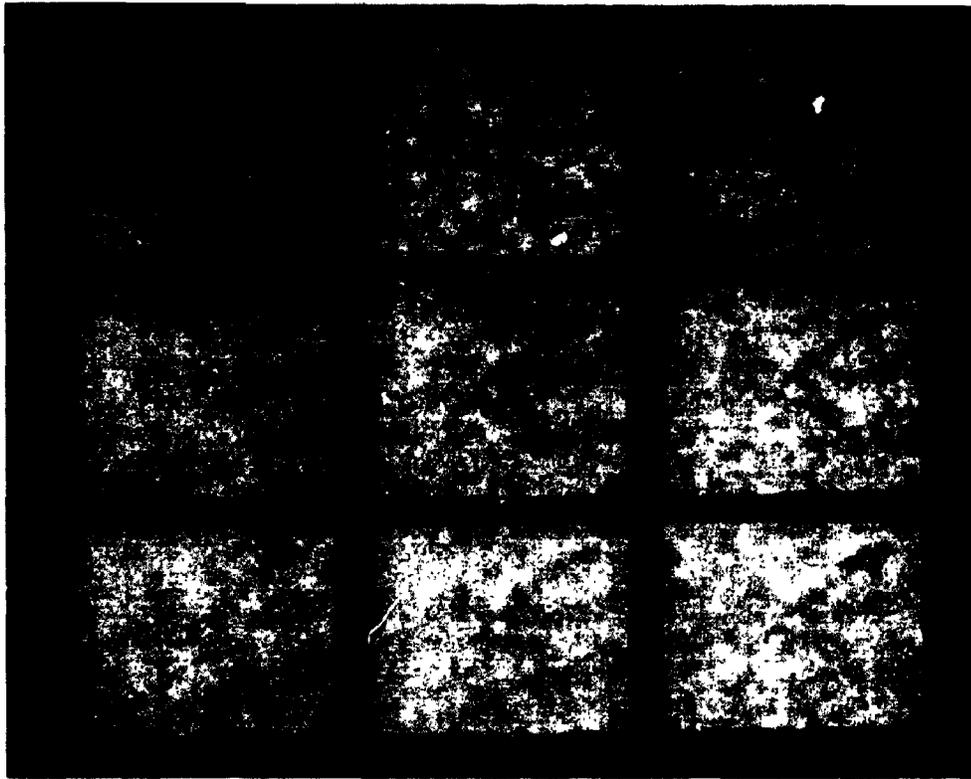


Fig. 4. (a) Computer generated flat grey area, grey level = 80 (b) The image shown in (a) with the addition of normally distributed random noise with a standard deviation of 10 (c) (d) Filtered images derived from (b) using Mean, Recursive Separable Median, Nagao and Matsuyama Gradient Inverse, Lee Additive, Sigma and KAVE Filters, respectively

WM filters, together with the multilevel filters, are also evaluated for their ability to preserve image details for which they were designed. The grey levels in all the images are between 0 and 127. The Filters were applied both as single pass and multiple pass filters using a 5 by 5 pixel window. Multiple passes consisted of three iterations. For the KAVE Filter the value of "K" represented approximately 75%, 50%, and 25% of the neighborhood elements in successive iterations. The successive passes of the Sigma Filter were applied with intensity intervals  $2\sigma$ ,  $\sigma$ , and  $\sigma/2$ .

The ability of the filters to reduce the noise in a "flat" grey region was tested using a computer generated uniform image field with a grey level of 80 (Fig. 4a). This was corrupted by adding random numbers generated to conform with a Gaussian distribution with a mean of zero and a standard deviation of 10 (Fig. 4b).

The images derived from a single pass of each of the single level filters over the noise corrupted uniform

Table 1. The Mean and the Standard Deviation for the Uniform Grey Field Before and After the Application of the Noise Reduction Filters

	Mean	SD
Simulated noisy image	80.58	9.60
(a) Filter type		
Mean	79.88	1.96
Recursive separable median	80.50	1.96
Nagao and Matsuyama	80.33	4.17
Gradient inverse	80.40	7.08
Lee additive	80.86	7.56
Sigma	79.94	4.20
KAVE	80.26	4.88
(b) Filter type		
A neighbor median	79.17	7.13
Weighted median	80.74	7.33
Multilevel median	80.57	7.99
Max min median	80.11	7.61
FIR median hybrid	80.31	6.89
In-place growing EMH	80.80	6.73
2DCO	80.39	6.30

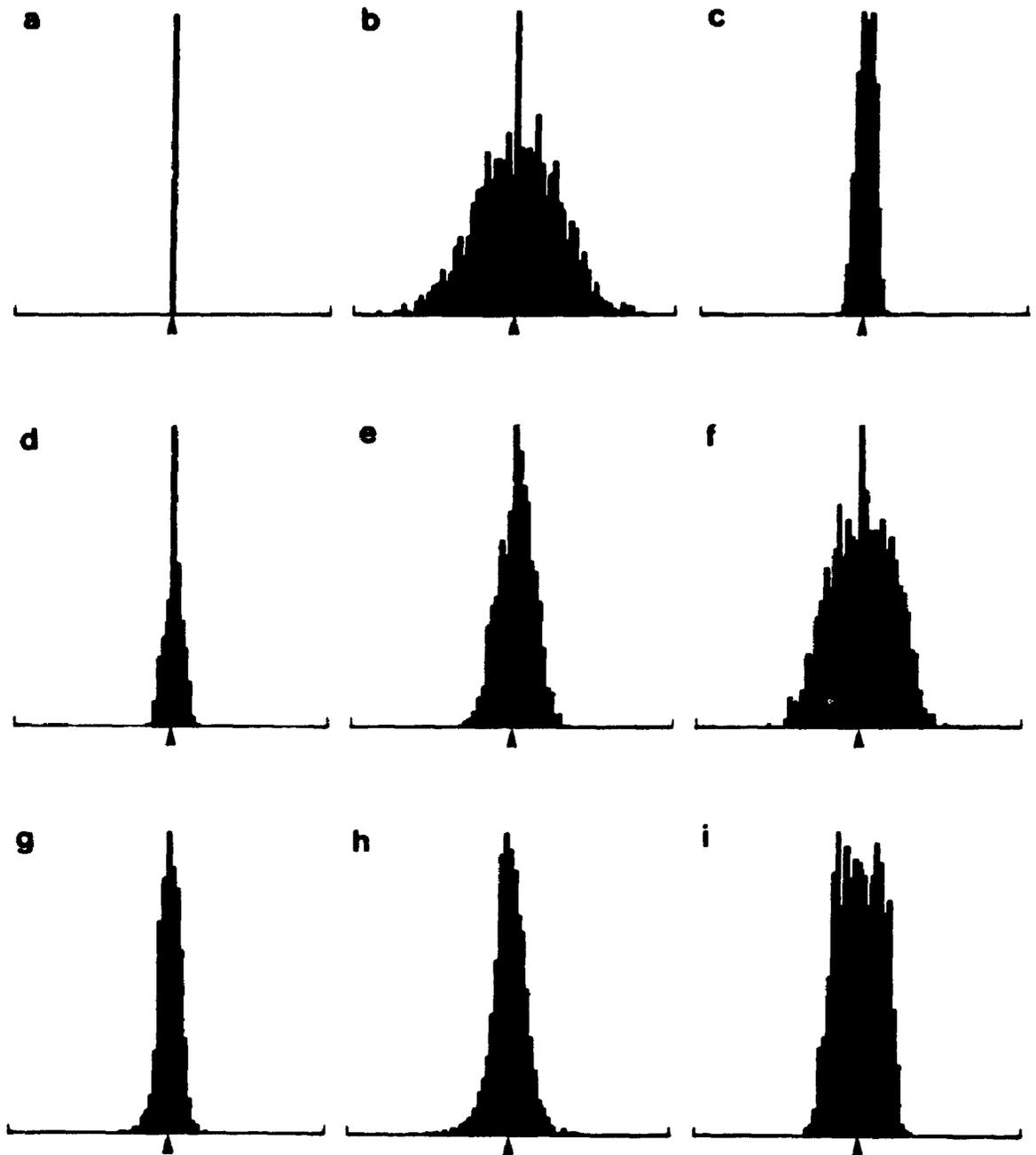


Fig. 5. Frequency distributions of the grey levels in the images of Fig 4

grey field are shown in Fig 4c-i. The frequency distributions of the grey levels obtained by each of the filters, and of the original images, are presented in Fig. 5, the mean grey levels and the associated standard deviations are listed in Table Ia. The data obtained by the RSM Filter show the least dispersion about the target value, closely followed by the Mean and Lee-Additive Filters. Profiles through the images

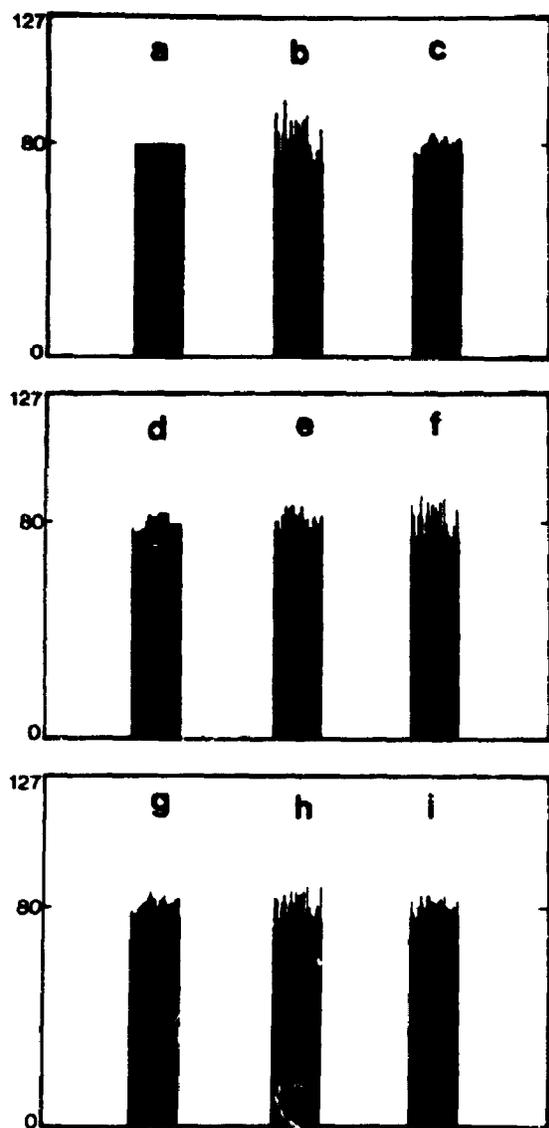


Fig. 6. Intensity profiles along a horizontal scan line in each of the images of Fig 4

of Fig 4, shown in Fig 6, similarly suggest that the Mean, RSM, and Lee-Additive Filters are the most successful in reducing the noise and recovering the original intensity value of 80. Table Ib lists the results from the application of detail preserving filters over the uniform grey field image. They do not perform as well as the single level filters.

For the iterative processing test, a second uniform grey image was created with a grey level of 80 corrupted with Gaussian noise of zero mean and a standard deviation of 20. The measured mean and standard deviation of this image were 79.61 and 19.32, respectively. The increased standard deviation of the noise served to accentuate the differences in the reduction of the standard deviation of the noise with successive passes of the filters. Table II lists the results from the application of multiple passes of the filters over this uniform grey field image. The RSM Filter yields a stable frequency distribution of grey levels with the smallest standard deviation after one iteration. The outputs of the other filters either stabilize with a higher standard deviation or converge more slowly onto stable distributions. Among the detail preserving filters the IPG-FMH Filter showed the best performance.

The effects of different window sizes for the single level filters was also studied. All except the Nagao and Matsuyama, WM, NNM Filters were applied with a single pass to a uniform grey region with a value of 80 with added normally distributed noise with zero mean and a standard deviation of 20. Window sizes were set to 3 by 3, 5 by 5, 7 by 7, and 9 by 9. The results, listed in Table III, indicate that generally the signal is more sharply defined as the window size increases. This is particularly noticeable with the RSM Filter. The Gradient Inverse and KAVE Filter seem to perform better with smaller window sizes.

To investigate the ability of a filter to preserve linear features and to smooth noise along edges while still preserving them a second test pattern was generated consisting of bars with widths of 1, 2, 4, 6, 8, 10, and 12 pixels (Fig 7a). The pixel values of each bar were set to 100 and of the background to 20. The image was corrupted by adding normally distributed random numbers with zero mean and a standard deviation of 10 (Fig 7b). The images derived from filtering the noise corrupted image using a 5 by 5 window and a single pass of the single level filters are presented in Figs 7c-i. Profiles through the images shown in Fig 7 are illustrated in Fig 8 Ia and b and IIa-g. These figures illustrate the following. The Mean Filter reduces step edges to ramp edges, thereby

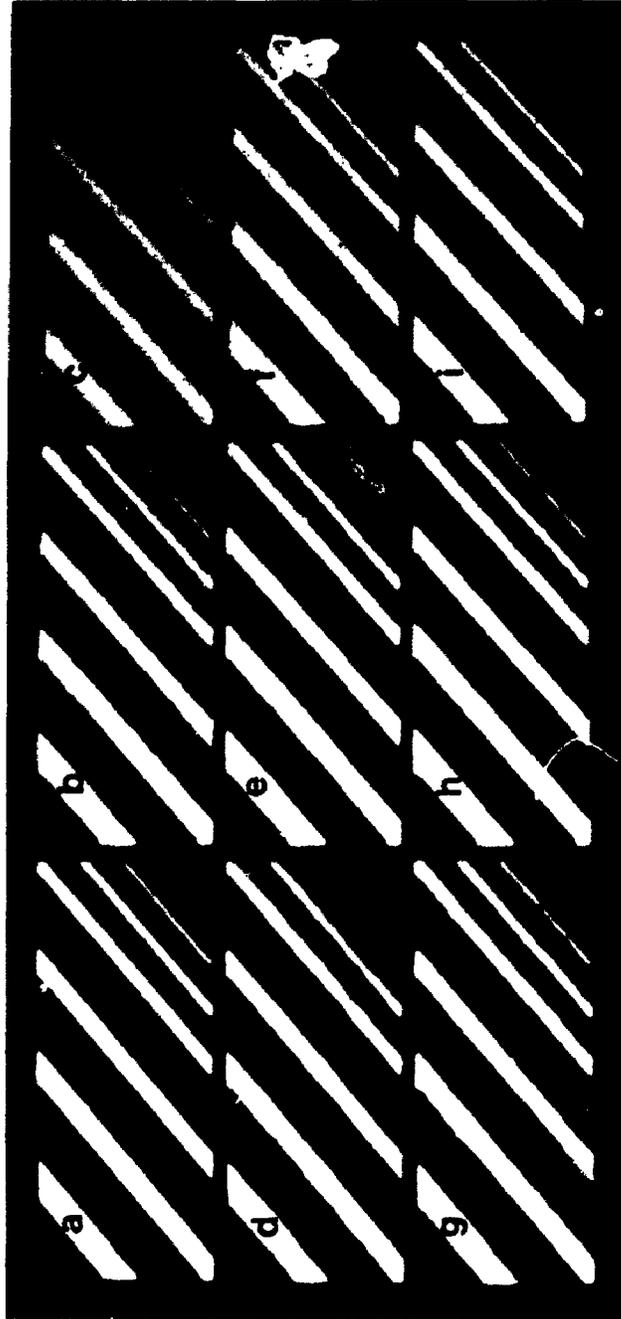


Fig. 7. (a) Computer generated bar pattern. The grey level of the bars is 100, the background is 20 and the widths of the bars are 1, 2, 4, 6, 8, 10, 12 pixels (b) The image shown in (a) with the addition of normally distributed random noise with a standard deviation of 10. (c)-(i) Filtered images derived from (b) using Mean, Recursive Separable Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma and KAVE Filters, respectively. The filters were applied with a single pass using a window size of 5 by 5.

## Evaluation of Digital Filters

181

Table II. "Uniform Grey Field Smoothing on Iterative Processing"

Filter type	One iteration		Two iterations		Three iterations	
	Mean	SD	Mean	SD	Mean	SD
Mean	79.11	4.02	78.63	2.96	78.18	2.66
RS median	79.63	3.31	79.70	3.08	79.71	3.08
Nagao	80.03	8.48	80.25	6.87	80.31	6.60
Grad. inv.	79.69	14.81	79.71	12.16	79.74	10.17
Lee-add.	79.59	4.95	79.58	3.06	79.57	2.73
Sigma	79.15	8.36	78.74	4.77	78.33	3.87
KAVE	79.39	8.05	79.00	6.71	78.60	6.70
NN median	77.17	10.24	76.73	9.31	76.72	9.29
W median	79.73	10.71	79.41	9.60	79.40	9.53
Multi median	79.87	12.14	79.90	11.84	79.92	11.74
Max-min med	80.01	14.04	79.84	13.25	79.95	12.92
F-MH	79.65	13.87	79.67	12.79	79.70	12.52
IPG-FMH	79.59	12.29	79.43	10.14	79.28	9.24
2DCO	79.75	12.92	79.75	12.92	79.75	12.92

\* Simulated noisy image. mean = 79.61 and SD = 19.32

smoothing linear features. The RSM Filter eliminates linear features which have a width less than half the window width, however, step edges are maintained, although, if the two adjacent image regions are noisy, slight blurring does occur. The Nagao and Matsuyama Filter similarly eliminates any feature with a width of three pixels or less. In the case of the KAVE Filter, the preservation of linear features depends on the chosen value of "K." The Sigma and Lee-Additive Filters are effective in preserving linear features and maintaining the sharpness of edges.

The images derived from filtering the noise corrupted image using a 5 by 5 window and a single pass of the detail preserving filters are presented in Figs. 9c-i. In the case of the WM and NNM Filter, the preservation of linear features depends on the chosen value of the weighting factor and the *K* nearest neighbors, respectively. The WM Filter showed the worst performance. The bar with a width of 4 pixels was eliminated using the Max/Min Median Filter with a window size of 5 by 5, however the same bar is partially preserved using the Recursive Max/Min Median Filter (Fig. 9d). The remaining multilevel filters preserve the linear feature well.

Figure 8 IIIa-g illustrates the profiles through the images produced by an iterative application of the single level filters. These profiles suggest that the Sigma Filter best retains edge information and linear features. The Mean Filter increases the width of the ramp edges generated by the previous pass of the filter.

The RS Median and the Nagao and Matsuyama Filters display similar abilities in retaining the edges. The Lee-Additive Filter (with an unchanged global noise variance) blurs edges and gives a similar intensity profile to that obtained using the Mean Filter.

To test the performance of the single level filters on noisy, non-edge image regions and on edges with different contrasts across them a third test image was generated consisting of 16 square areas with constant grey levels of 30, 34, 38, 40, 45, 48, 54, 58, 64, 70, 79, 84, 94, 100, 110, and 120 (Fig. 10a). This image was corrupted by adding random numbers generated to have a Gaussian distribution with zero mean and a

Table III. "Uniform Grey Field Smoothing with Different Window Size After one Filtering Iteration"

Filter type	Windows							
	3 by 3		5 by 5		7 by 7		9 by 9	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Mean	79.14	6.67	79.11	4.02	79.08	2.95	79.04	2.35
RS median	79.87	7.21	79.63	3.31	80.67	1.66	79.95	1.27
Grad inv.	79.77	14.27	79.69	14.81	79.72	15.02	79.53	15.00
Lee-add.	79.62	8.25	79.56	5.19	79.60	3.80	79.53	2.98
Sigma	79.18	10.53	79.15	8.36	79.19	7.87	79.05	7.18
KAVE	79.14	6.67	79.34	9.91	79.32	12.19	79.15	13.59

\* Simulated noisy image. mean = 79.61 and SD = 19.32

standard deviation of 3 (Fig. 10b). The images derived from the noise corrupted image by the single level filters are shown in Figs. 10c. Inspection of these images reveals that the Gradient Inverse, Sigma, Lee-Additive, and RSM Filters produce the least shape distortion as indicated by the preservation of the corners of the squares. Comparison of profiles through the images (Fig. 11) indicates that none of the

filters retain the sharpness or location of an edge when the standard deviation of the noise on the two sides of the boundary exceeds the edge gradient.

Several petrographic images were also studied to evaluate the performance of the single level filters for noise reduction and edge retention on real images. One of these images is illustrated in Fig. 12a. It shows five quartz grains which display different grey levels.

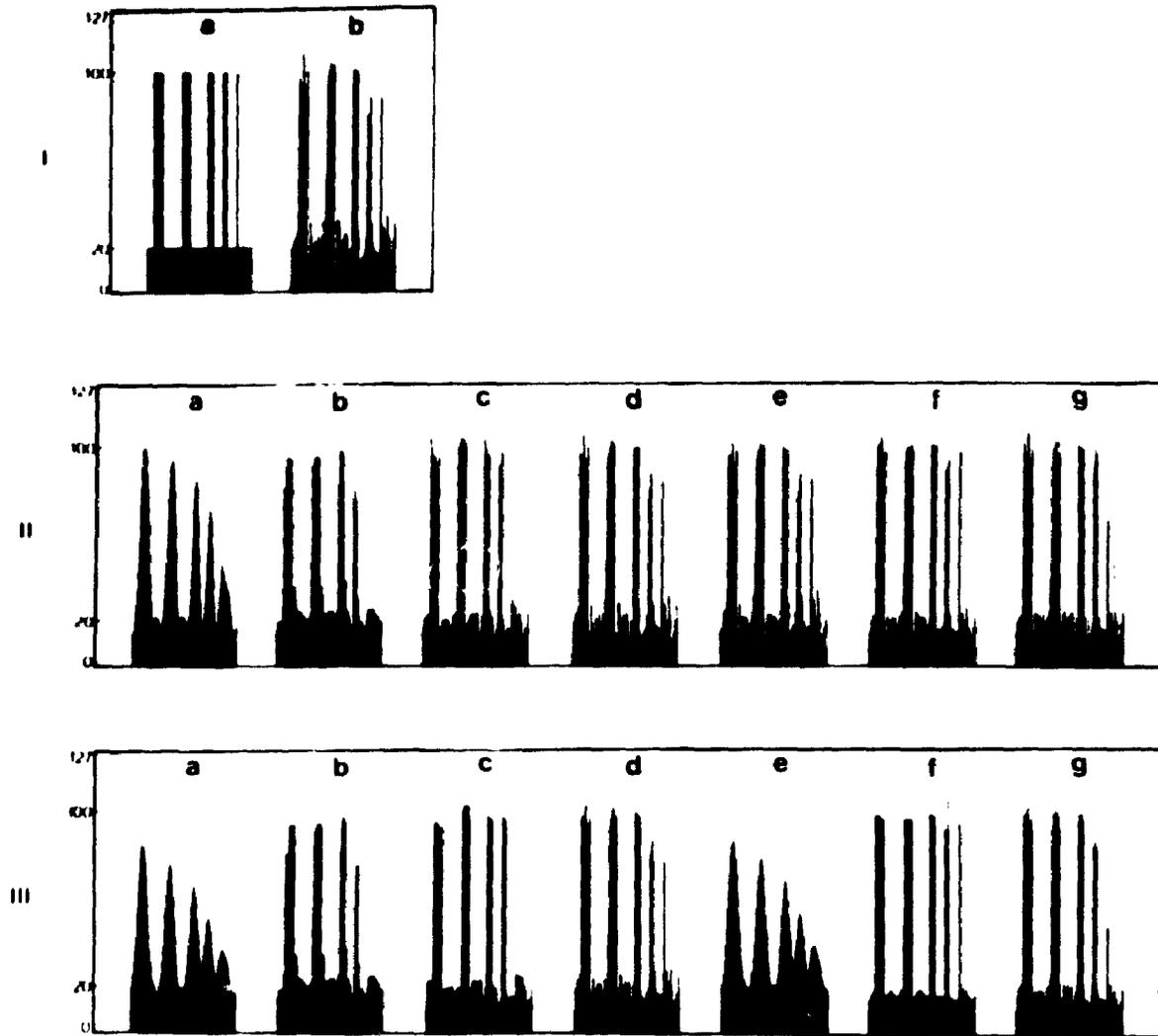


Fig. 8. (I) Intensity profiles along a horizontal scan line through Figs. 7a and b (II) Intensity profiles along a horizontal scan line through the images of 7(c). Note: These data pertain to single passes of each of the filters using a 5 by 5 window size (III) Intensity profiles along a horizontal scan line through the images obtained by filtering the image of 4b using three passes of the Mean, Recursive Separable Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma and KAVE Filters, respectively. The window size is 5 by 5.

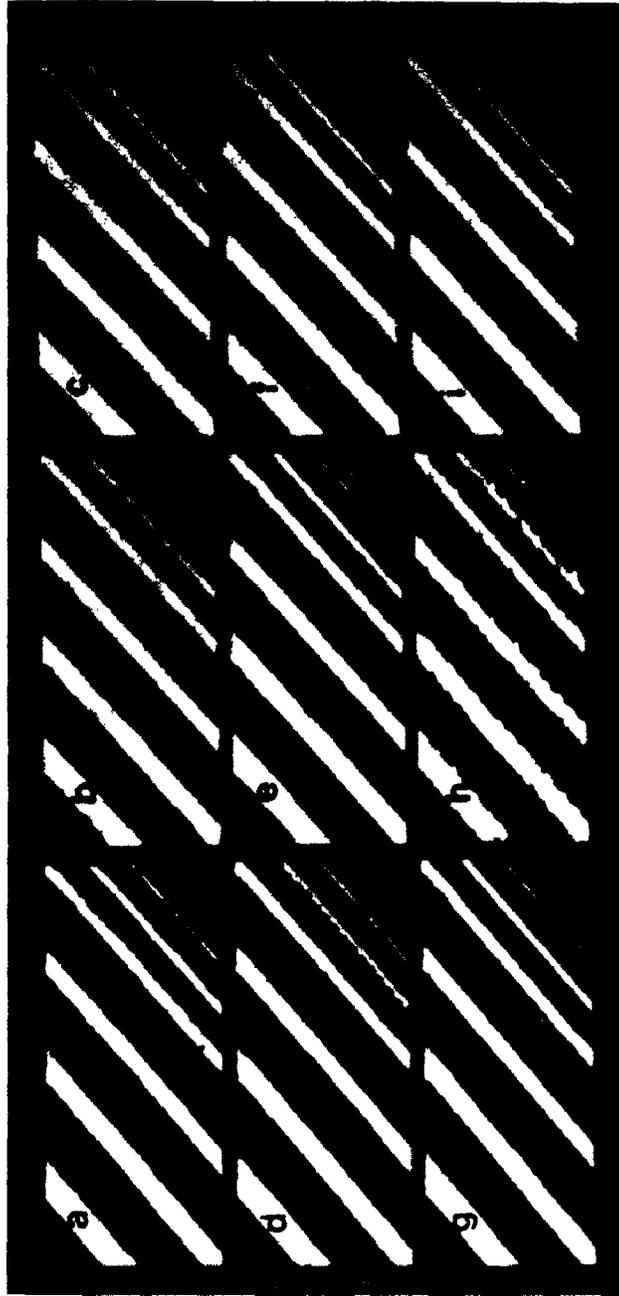


Fig. 9. The computer generated bar pattern with added random noise, see Fig. 7b. (b)-(i) Filtered images derived from (a) using Recursive Separable Median, Multilevel Median, Max-Min Median, FIR-Median Hybrid, In-Place Growing FMH, Weighted Median, Nearest Neighbor, Median, and DCO Filters, respectively. The filters were applied with a single pass using a window size of 5 by 5.

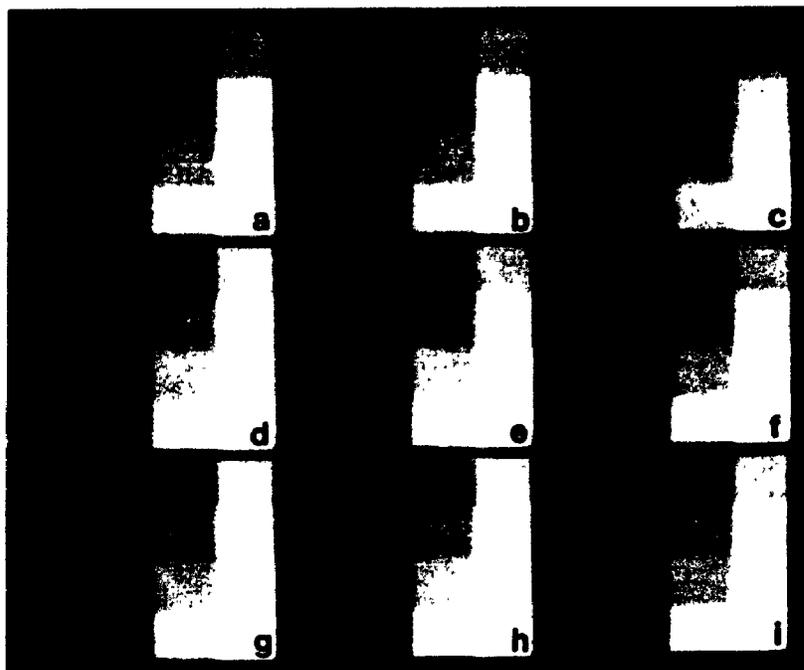


Fig. 10. (a) Computer generated pattern of 16 squares, for the grey levels see text. (b) The image shown in (a) with the addition of normally distributed random noise with a standard deviation of 3. (c)–(i) Filtered images derived from (b) using Mean, Recursive Separable Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma and KAVE Filters, respectively.

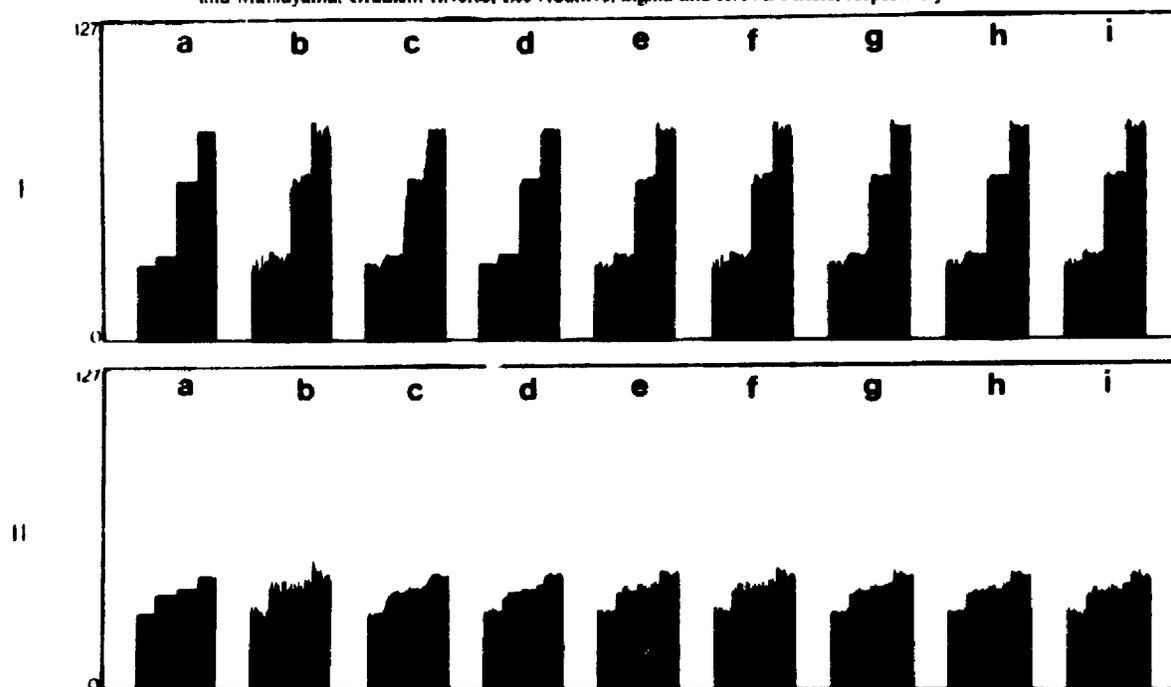


Fig. 11. (I) Intensity profiles along a horizontal scan line through the top row of squares shown in Fig. 10. The initial computer generated grey levels of these squares are 30, 34, 64, 84 (see Fig. 10a). (II) Intensity profiles along a horizontal scan line through the first column of squares shown in Fig. 10. The initial computer generated grey levels of these squares are 30, 38, 40, 45 (see Fig. 10a).

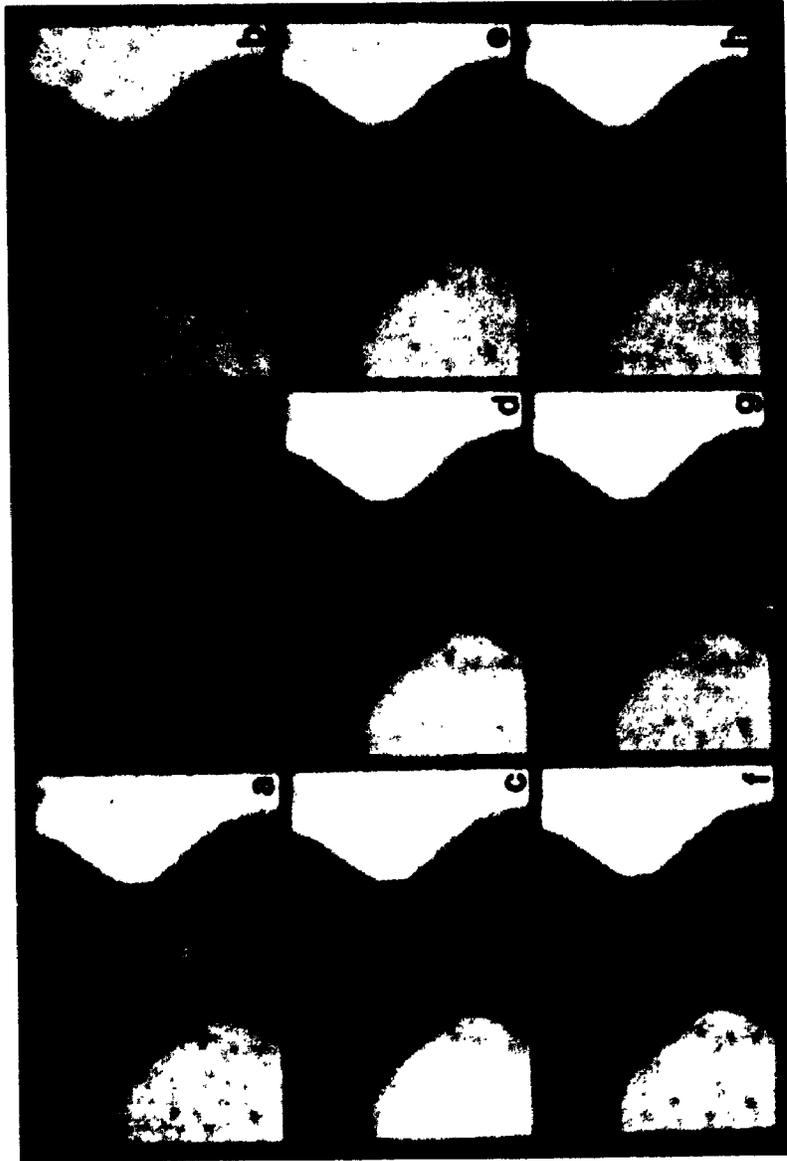


Fig. 12. a) Photomicrograph of a rock thin section between crossed polars; b-h) Filtered images derived from (a) using Mean, Recursive Separable Median, Nagao and Matsuyama, Gradient Inverse, Lee-Additive, Sigma and KAYE Filter, respectively; i) Reversible.

The filtered petrographic images obtained by application of single passes of the filters are presented in Figs. 12b-h where the effects of the filters on boundaries with different orientations may be observed. The filtered images, and the profiles along a scan line through them, shown in Fig. 13, confirm the observations of the computer simulated images. In particular, the Nagao and Matsuyama Filter most successfully sharpens ramp edges.

The filtered petrographic images of Fig. 12 also allowed an evaluation of the effectiveness of the filters at removing spot noise by attenuating the associated high spatial frequencies. The RSM, Sigma, and Nagao and Matsuyama Filters effectively remove spot noise, although the latter two filters require more than one pass. The Lee-Additive Filter can be well suited for the removal of spot noise provided the global noise variance is selected appropriately by the operator.

Several textured images (images having a wide variation in grey level over small areas) were also studied to evaluate the performance of the detail preserving filters at noise reduction and texture retention. One of these images is illustrated in Fig. 14a, which shows a rock texture in thin section between crossed polarizers. The filtered textured images obtained by application of single passes of the RSM, Multilevel, NNM, and WM Filters are presented in Figs. 14b-i. Figure 14 illustrates that the RSM Filter

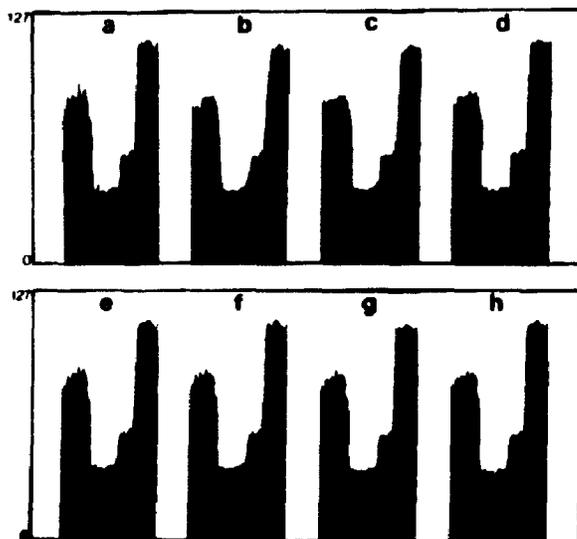


Fig. 13. Intensity profiles along a horizontal scan line through the photomicrographs shown in Fig. 12

Table IV. Computation Times Required to Filter a 50 by 50 Pixel Image, Using One Filtering Iteration and 5 by 5 Window. Executed on a 80386CPU Running at 20 MHz

Filter type	Time (sec)
Mean	0.33
Recursive separable median	0.66
Nagao and Matsuyama	6.10
Gradient inverse	7.69
Lee-additive	2.96
Sigma	0.83
KAVE	1.54
N neighbor median	1.21
Weighted median	0.88
Multilevel median	1.97
Max/min median	2.25
FIR-median hybrid	1.24
In-place growing FMH	2.69
2DCC	9.16

destroys much of the image texture. The IPG-FMH Filter is the best at preserving the details in texture while attenuating the noise, followed by the Multilevel Median Filter. The Sigma Filter can also preserve the image texture, depending on the chosen value of two sigma. Comparative processing times of these filters are listed in Table IV.

## SUMMARY AND CONCLUSIONS

Fourteen noise reduction filters have been applied to computer simulated images and real images from rock thin sections. The filters were applied both as single pass and multiple pass filters using a 5 by 5 pixel window. For single pass operation, all the single level filters except the Nagao and Matsuyama, NNM, and WM Filters were also applied with varying window sizes.

For a single pass operation, the RSM Filter most effectively reduces the noise variance in a uniform grey region and retains edges between adjacent image regions. This filter also does not create any significant distortion in the shape of an image region and it effectively smooths sharp spot noise. Thus, the RSM Filter meets most of the criteria identified previously. However, it does not sharpen ramp edges, it eliminates small linear features (i.e., those with a breadth less than half the window width), destroys image texture and, as noted above, the output from such recursive filters depends on the direction in which they are applied. Increasing the window size does not improve the performance of the RSM Filter, the data

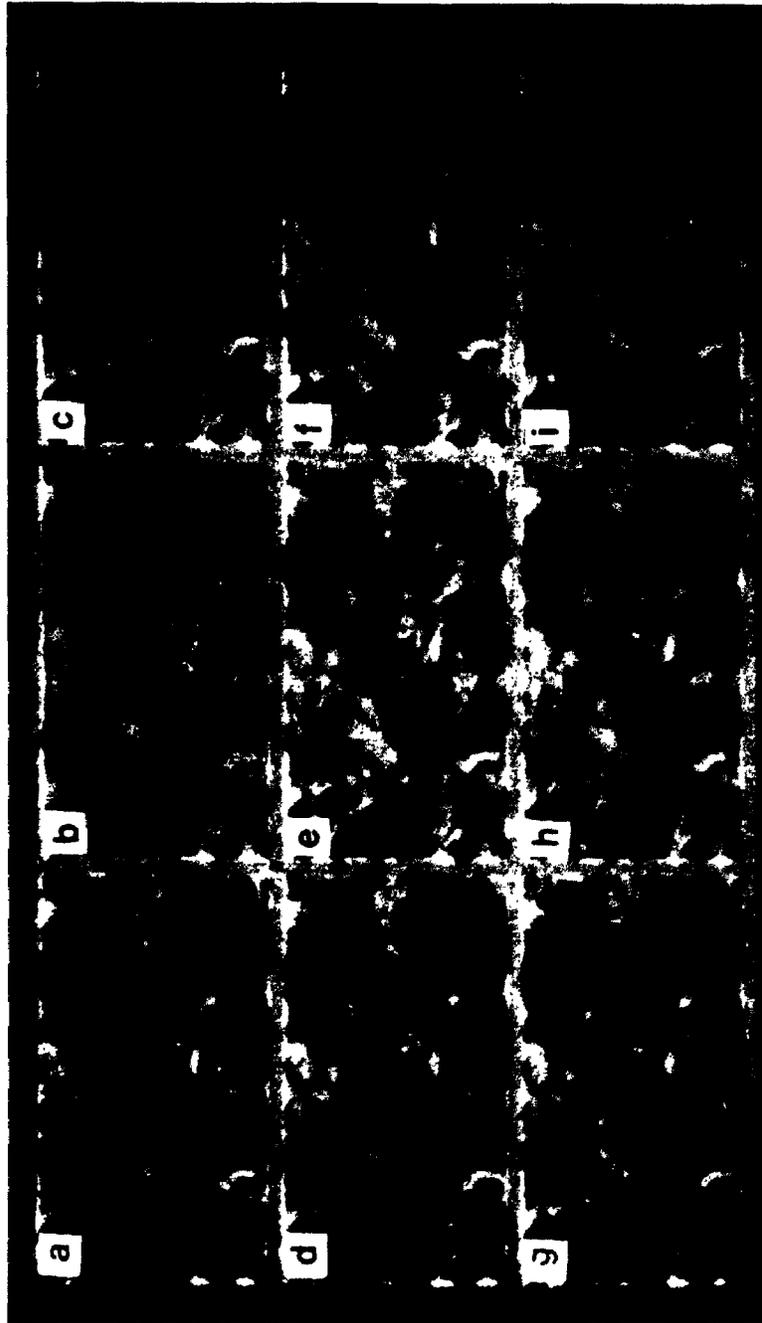


Fig 14. (a) Photomicrograph showing a textured image (b) (i) Filtered images derived from (a) using Recursive Separable Median, Multilevel Median, Max Min Median, FIR Median Hybrid, In-Place Growing FMM, Weighted Median, Nearest Neighbor Median, and 2DCO Filters, respectively.

presented here suggest that a 5 by 5 pixel window is optimum. The Nagao and Matsuyama Filter is best in sharpening ramp edges with a single pass.

For the preservation of image detail and texture multiple pass applications of the IPG-FMH and Sigma Filter perform best. Not only do they reduce noise in uniform grey areas and retain the edges between adjacent grey areas, they also preserve linear features and image texture. However, the computation time required for the iterative operation of these filters is greater than that required for the single pass RSM Filter.

#### ACKNOWLEDGMENTS

This research was supported by operating Grant A3555 from the National Science and Engineering Research Council of Canada, which gratefully is acknowledged.

#### REFERENCES

- Andrews, H. C., and Hunt, B. R. (1977) *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, N.J.
- Arce, G. R., Gallagher, N. C., and Nodes, T. A. (1986). Median filters—Theory for one and two dimensional filters. In *Advances in Computer Vision and Image Processing* (Vol. 2). JAI Press, pp. 89–166.
- Arce, G. R., and McLoughlin, M. P. (1987). Theoretical analysis of the max median filter. *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-35(1):60–69.
- Arce, G. R., and Foster, R. E. (1988). Multilevel Median Filters: Properties and Efficacy. Proceedings ICASSP, New York, pp. 824–827.
- Arce, G. R., and Foster, R. E. (1989). Detail-preserving ranked-order based filters for image processing. *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-37(1):83–98.
- Asano, A., Itoh, K., and Ichioka, Y. (1990). The nearest neighbor median filter—Some deterministic properties and implementation. *Pattern Recognition* 23(10):1059–1066.
- Brownrigg, D. R. K. (1984). The weighted median filter. *Com. M.* 27(8):807–818.
- Brownrigg, D. R. K. (1986). Generation of representative members of an RrSst Weighted median filter class. *IEE Proc.* 133(5):445–448.
- Davis, L. S., and Rosenfeld, A. (1978). Noise cleaning by iterated local averaging. *IEEE Trans. Syst. Man Cybern.* SMC-8(9):705–710.
- Fabru, A. G. (1984) *Image Processing of Geological Data*. Van Nostrand Reinhold Co., New York.
- Gallagher, N. C., and Wise, G. L. (1981). A theoretical analysis of the properties of median filters. *IEEE Trans. Acoustics, Speech, Signal Proc.* ASSP-29(6):1136–1141.
- Hodgson, R. M., Bailey, D. G., Naylor, M. J., Ng, A. L. M., and McNeill, S. J. (1985). Properties, implementations and applications of rank filters. *Image Vision Comput.* 3(1):3–14.
- Itoh, K., Ichioka, Y., and Minami, T. (1988). Nearest-neighbor median filter. *Appl. Opt.* 27(16):3445–3450.
- Launeau, P., Bouchez, J. L., and Benn, K. (1990). Shape preferred orientation of object populations: Automatic analysis of digitized images. *Tectonophysics* 180:201–211.
- Lee, J. S. (1980). Digital image enhancement and noise filtering by use of local statistics. *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-2(2):165–168.
- Lee, J. S. (1981). Refined filtering of image noise using local statistics. *Comput. Graphics Image Proc.* 15:380–389.
- Lee, J. S. (1983). Digital image smoothing and the sigma filter. *Comp. Vision Graphics Image Proc.* 24:255–269.
- Maragos, P., and Schafer, R. W. (1987). Morphological filters—part II: Their relations to median, order-statistic, and stack filters. *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-35(8):1170–1184.
- McLoughlin, M. P., and Arce, G. R. (1987). Deterministic properties of the recursive separable median filter. *IEEE Trans. Acoustics, Speech, Signal Proc.* ASSP-35(1):98–106.
- Nagao, M., and Matsuyama, T. (1979). Edge preserving smoothing. *Comput. Vision Graphics Image Proc.* 9:394–407.
- Narendra, P. M. (1981). A separable median filter for image noise smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-3(1):20–29.
- Nieminen, A., Heinonen, P., and Neuvo, Y. (1987). A new class of detail-preserving filters for image processing. *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-9(1):74–90.
- Nieminen, A., and Neuvo, Y. (1988). Comments on "Theoretical Analysis of the Max/Median Filter." *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-36(5):826–827.
- Pratt, W. K. (1978) *Digital Image Processing*. Wiley, New York.
- Rosenfeld, A., and Kak, A. C. (1976) *Digital Picture Processing*. Academic Press, New York.
- Starkey, J., and Simigian, S. (1992). Image analysis of particle shape. In Palaz, I., and Sengupta, S. (eds.), *Automated Pattern Analysis in Petroleum Exploration*. Springer-Verlag, New York.
- Tukey, J. W. (1977) *Exploratory Data Analysis*. Addison-Wesley, Reading, Mass.
- Wang, D., Vagnucci, A., and Li, C. (1981). Gradient inverse weighted smoothing scheme and the evaluation of its performance. *Comput. Vision Graphics Image Proc.* 15:167–181.
- Wang, X., and Wang, D. (1990). On the max median filter. *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-38(8):1473–1475.
- Wichman, R., Astola, J. T., Heinonen, P. J., and Neuvo, Y. A. (1990). FIR-median hybrid filters with excellent transient response in noisy conditions. *IEEE Trans. Acoustics, Speech, Signal Proc.* AASP-38(12):2108–2117.

## Appendix 2

# Computational Masks for the Canny algorithm

Many of the edge detectors which have been proposed recently are based on finite differences of image intensities or derivatives using finite differences. One such edge detector is due to Canny (1986). Canny's algorithm marks boundaries at maxima in the magnitude of the image gradient in the direction of the gradient. At a local maximum the second order derivative of the image value becomes zero (Russ 1990, p. 74, figure 4-4), and therefore the zero crossing indicates the edge. In the present study the convolution masks required to implement the Canny algorithm is derived as follows (Starkey and Samantaray, in press):

The input image is convolved with an operator  $G_n$  which is the first derivative of a two-dimensional Gaussian  $G$  in some direction  $\mathbf{n}$ , i.e.

$$G = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

and

$$G_n = \frac{\partial G}{\partial \mathbf{n}} = \mathbf{n} \cdot \nabla G \quad (2.1)$$

Ideally,  $\mathbf{n}$  should be oriented normal to the direction of an edge to be detected,

and although this direction is not known *a priori*, a good estimate of it can be derived from the smoothed gradient direction

$$\mathbf{n} = \frac{\nabla(G * I)}{|\nabla(G * I)|} \quad (2.2)$$

where  $*$  denotes convolution. This is a good estimator for the edge normal direction of steps, since a smoothed step has a strong gradient normal to the edge.

An edge point is defined to be a local maximum (in the direction  $\mathbf{n}$ ) of the operator  $G_{\mathbf{n}}$  applied to the image  $I$ . At a local maximum

$$\frac{\partial}{\partial \mathbf{n}} G_{\mathbf{n}} * I = 0 \quad (2.3)$$

and substituting for  $G_{\mathbf{n}}$  from eq.2.1 and associating Gaussian convolution, the above becomes

$$\frac{\partial^2}{\partial \mathbf{n}^2} G * I = 0 \quad (2.4)$$

Let  $f = G * I$  then the second derivative of  $f$  in the direction  $\mathbf{n}$  of the local gradient can be given by the following derivation (Lipkin and Rosenfeld 1970, p. 111)

$$\frac{\partial^2 f}{\partial \mathbf{n}^2} = \frac{\frac{\partial^2 f}{\partial x^2} \left(\frac{\partial f}{\partial x}\right)^2 + 2 \frac{\partial^2 f}{\partial x \partial y} \frac{\partial f}{\partial x} \frac{\partial f}{\partial y} + \frac{\partial^2 f}{\partial y^2} \left(\frac{\partial f}{\partial y}\right)^2}{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (2.5)$$

The denominator is always positive and therefore does not change the sign of the  $\partial^2 f / \partial \mathbf{n}^2$ , eq.2.5 can be written as

$$\frac{\partial^2 f}{\partial \mathbf{n}^2} = \frac{\partial^2 f}{\partial x^2} \left(\frac{\partial f}{\partial x}\right)^2 + 2 \frac{\partial^2 f}{\partial x \partial y} \frac{\partial f}{\partial x} \frac{\partial f}{\partial y} + \frac{\partial^2 f}{\partial y^2} \left(\frac{\partial f}{\partial y}\right)^2 \quad (2.6)$$

where the second order derivatives,  $\frac{\partial^2 f}{\partial x^2}$  and  $\frac{\partial^2 f}{\partial y^2}$ , are computed using the mask  $[1, -2, 1]$  in both the horizontal and vertical directions. The first order differences  $\frac{\partial f}{\partial x}$  and  $\frac{\partial f}{\partial y}$ , are computed using the mask  $[-1, 0, 1]$  in the horizontal and vertical directions.  $\frac{\partial^2 f}{\partial x \partial y}$  is approximated by applying the mask shown below:

$$\begin{bmatrix} -1 & 1 \\ 0 \\ 1 & -1 \end{bmatrix}$$

This has been implemented in the present imaging system (see figure 3.20 for results) and is discussed in chapter 3.

## Appendix 3

# Locating Ends of Edge Contours

The following algorithm, presented in pseudo code, recognizes the configurations of pixels displayed in figure 3.25 and identifies that the central "Edge" pixel as an "End" pixel. Pixels containing the value characteristic of an edge are considered to be "on".

### Algorithm *FindEndPoint*.

---

If the *CentralPixel* is an *EdgePixel* then

1. Initialize *CheckSequentialPixelsOn*, *TwoSequentialEdgePixels*, *ThreeSequentialEdgePixels*, *NotEndPoint* to FALSE; *PixelsOn* to 1; *Subscript* to 0.
2. Assign subscripts to the pixels surrounding the central pixel, p, as shown below:

$$\begin{bmatrix} 1 & 2 & 3 \\ 8 & p & 4 \\ 7 & 6 & 5 \end{bmatrix}$$

3. Set *Pixel*[9] to *Pixel*[1] to assure a continuous loop.
4. Identify the presence of two consecutive edge pixels corresponding to configurations 9 to 20 of figure 3.25 by the following steps (a)-(c).
  - (a) Increment *Subscript*.
  - (b) **If** *Pixel*[*Subscript*] is an *EdgePixel* **then** do steps i-iii.
    - i. Increment *PixelsOn*.
    - ii. **If** *CheckSequentialPixelsOn* is TRUE **then**  
 set *TwoSequentialEdgePixels* = TRUE.
    - iii. Set *CheckSequentialPixelsOn* = TRUE.

**else** Set *CheckSequentialPixelsOn* = FALSE.
  - (c) **If** *TwoSequentialEdgePixels* = TRUE OR *Subscript* = 9 **then** goto step 5 **else** goto step 4.(a).
5. Identify the presence of three consecutive edge pixels corresponding to configurations 17 to 20 of figure 3.25 by the following step.
  - (a) **If** *TwoSequentialEdgePixels* is TRUE AND *Subscript* < 9 **then** do steps i-ii.
    - i. Increment *Subscript*.
    - ii. **If** *Pixel*[*Subscript*] is an *EdgePixel* **then** do steps A-B.
      - A. Increment *PixelsOn*.

- B. If *Subscript* is an ODD number  
then set *ThreeSequentialEdgePixels* = TRUE.
6. Count the remaining number of edge pixels by the following step.
- (a) While *Subscript* < 9 Do steps i-ii.
- i. Increment *Subscript*.
- ii. If *Pixel[Subscript]* is an *EdgePixel* then increment *PixelsOn*.
7. If *Pixel[1]* is an *EdgePixel* then decrement *PixelsOn*  
(Remove duplicated first pixel).
8. If *PixelsOn* > 2 then set *NotEndPoint* = TRUE.
9. If *PixelsOn* = 3 then set *NotEndPoint* = NOT *TwoSequentialEdgePixels*.
10. If *PixelsOn* = 4 then set *NotEndPoint* = NOT *ThreeSequentialEdgePixels*.
11. If *NotEndPoint* is FALSE then set *CentralPixel* as an *EndPixel*.
-

## **Appendix 4**

# **Pascal code for the IMAGING SYSTEM**

This appendix contains the Pascal source code for the IMAGING SYSTEM. The source code is written in Turbo Pascal V.6.0. The code is written for use with a 80386/80486 cpu, a 80387 numeric processor, a Microsoft compatible mouse, and requires DOS V.5.0 or higher with the EMS expanded memory manager. The program is designed for use with Imaging Technology's PCVISION plus Frame Grabber and a Sony AVC-D5 monochrome CCD video camera. The graphics routines are written for a Trident Super VGA 8900 (or VESA compatible) graphics card.

UNABLE TO FILM MATERIAL ACCOMPANYING THIS THESIS ( I.E.  
DISKETTE(S), SLIDES, MICROFICHE, ETC...).

PLEASE CONTACT THE UNIVERSITY LIBRARY.

INCAPABLE DE MICROFILMER LE MATERIEL QUI ACCOMPAGNE CETTE THESE  
(EX. DISQUETTES, DIAPOSITIVES, MICROFICHE (S), ETC...).

VEUILLEZ CONTACTER LA BIBLIOTHEQUE DE L'UNIVERSITE.

NATIONAL LIBRARY OF CANADA  
CANADIAN THESES SERVICE

BIBLIOTHEQUE NATIONALE DU CANADA  
LE SERVICE DES THESES CANADIENNES

## REFERENCES

- Abdou, I. E., and Pratt, W. K. (1979). Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proceedings of the IEEE*. 67(5): 753-763.
- Ali, S. M., and Burge, R. E. (1988). A new algorithm for extracting the interior of bounded regions based on chain coding. *Computer Vision, Graphics, and Image Processing*. 43: 256-264.
- Andrews, H. C., and Hunt, B. R. (1977). *Digital Image Restoration*, Prentice-Hall, Englewood Cliffs, N.J.
- Arce, G. R., Gallagher, N. C., and Nodes, T. A. (1986). Median filters: Theory for one and two dimensional filters. In *Advances in Computer Vision and Image Processing* (Vol. 2), JAI Press, pp. 89-166.
- Arce, G. R., and McLoughlin, M. P. (1987). Theoretical analysis of the max/median filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-35(1): 60-69.
- Arce, G. R., and Foster, R. E. (1988). Multilevel median filters: Properties and Efficacy. *Proceedings ICASAP*, New York, pp. 824-827.
- Arce, G. R., and Foster, R. E. (1989). Detail-preserving ranked-order based filters for image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-37(1): 83-98.
- Asano, A., Itoh, K., and Ichioka, Y. (1990). The nearest neighbor median filter: Some deterministic properties and implementation. *Pattern Recognition*. 23(10): 1059-1066.
- Ballard, D. H. and Brown, C. M. (1982). *Computer Vision*, Prentice-Hall, Englewood Cliffs, N.J.
- Beddow, J. K., Philip, G. C., Vetter, A. F. (1977). On relating some particle profile characteristics to the profile fourier coefficients. *Powder Technology*. 18: 15-19.
- Borradaile, G. J., and McArthur, J. (1991). Tests of strain analysis by experimental deformation. *Tectonophysics*. 185: 325-333.

- Brownrigg, D. R. K.** (1984). The weighted median filter. *Com. ACM.* **27**(8): 807-818.
- Brownrigg, D. R. K.** (1986). Generation of representative members of an RrSst weighted median filter class. *IEE Proceedings.* **133**(5): 445-448.
- Cai, Z.** (1988). Restoration of binary images using contour direction chain codes description, *Computer Vision, Graphics, and Image Processing.* **41**: 101-106.
- Canny, J.** (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* **PAMI-8**(6): 679-698.
- Chang, L. W., and Leu, K. L.** (1990). A fast algorithm for the restoration of images based on chain codes description and its applications. *Computer Vision, Graphics, and Image Processing* **50**: 296-307.
- Cheel, R. J.** (1991). Grain fabric in hummocky cross-stratified storm beds: Genetic implications. *Journal of Sedimentary Petrology.* **61**(1): 102-110.
- Clark, M. W.** (1981). Quantitative shape analysis: A review. *Mathematical Geology.* **13**(4): 303-320.
- Clark, M. W.** (1987). Image analysis of clastic particles. In Marshall, J. R. (eds.), *Clastic Particles*, Van Nostrand Reinhold, New York, pp. 256-266.
- Craig, J. R., Yoon, R. H., Haralick, R. M., Pong, T. C., Choi, W.** (1982). The application of the general image processing system (GIPSY) to mineral beneficiation studies. In Hagni, R. D. (eds.), *Process Mineralogy II, Application to Metallurgy, Ceramics, and Geology*, AIME pub., pp. 55-67.
- Davis, L. S., and Rosenfeld, A.** (1978). Noise cleaning by iterated local averaging. *IEEE Transactions on Systems, Man, and Cybernetics.* **SMC-8**(9): 705-710.
- Davis, J. C.** (1987). *Statistics and Data Analysis in Geology*, John Wiley & Sons, New York.
- de Souza, P.** (1983). Edge detection using sliding statistical tests. *Computer Vision Graphics, and Image Processing.* **23**: 1-14.
- Distante, A., and Veneziani, N.** (1982). A two-pass filling algorithm for raster graphics. *Computer Graphics and Image Processing.* **20**: 289-295.
- Ehrlich, R., and Full, W. E.** (1984). Fourier shape analysis-A multivariate pattern recognition approach. In Beddow, J. K. (eds.), *Particle Characterization in Technology, Volume II: Morphological Analysis*, CRC Press, Boca Raton, Florida, pp. 89-101.
- Ehrlich, R., and Weinberg, B.** (1970). An exact method for characterization of grain shape. *The Journal of Sedimentary Petrology.* **40**: 205-212.

- Evans, D. G., Schweitzer, P. N., and Hanna, M. S. (1985). Parametric cubic splines and geologic shape descriptions. *Mathematical Geology*. 17(6): 611-624.
- Exner, H. E. (1986). Shape: a key problem in quantifying microstructure. *Proceedings of the 1st International Conference on Microstructology*, Gainesville, FL.
- Fabbri, A. G. (1984). *Image Processing of Geological Data*, Van Nostrand Reinhold, New York.
- Fico, C. (1980). Automated particle shape analysis: development of a microprocessor based image analysis system. Masters Thesis, University of South Carolina, Columbia.
- Fleck, M.M. (1992). Some defects in finite-difference edge finders. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-14(3): 337-345.
- Fleck, M.M. (1992). Multiple widths yield reliable finite differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-14(4): 412-429.
- Flook, A. (1982). Fourier analysis of particle shape. In Stanley-Wood, N. G., and Allen, T. (eds.), *Particle Size Analysis*, Wiley Heyden, London.
- Freeman, H. (1970). Boundary encoding and processing in picture processing, In Lipkin, B. S., and Rosenfeld, A. (eds), *Picture Processing and Psychopictorics*, Academic Press, New York, pp. 241-266.
- Freeman, H., and Davis, L. (1977). A corner-finding algorithm for chain-coded curves. *IEEE Transactions on Computers*. C-26(3): 297-303.
- Gallagher, N. C., and Wise, G. L. (1981). A theoretical analysis of the properties of median filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. ASSP-29(6): 1136-1141.
- Gerard, R. E., Philipson, F. M., Manni, F. M., and Marschall, D. M. (1992). Petrographic image analysis: An alternate method for determining petrophysical properties. In Palaz, I., and Sengupta, S (eds.), *Automated Pattern Analysis in Petroleum Exploration*, Springer-Verlag, New York, pp. 248-263.
- Gonzalez, R. C., and Wintz, P. (1987). *Digital Image Processing*, Addison-Wesley, Reading, Mass.
- Haralick, R. M., and Shapiro, L. G. (1985). Image segmentation techniques. *Computer Vision, Graphics, and Image Processing*. 29: 100-132.
- Haralick, R. M., and Shapiro, L. G. (1991). Glossary of computer vision terms. *Pattern Recognition*. 24(1): 69-93.

- Hawkins, J. K. (1970). Textural properties for pattern recognition, In Lipkin, B. S., and Rosenfeld, A. (eds), *Picture Processing and Psychopictorics*, Academic Press, New York, pp. 347-379.
- Hildreth, E. (1983). The detection of intensity changes by computer and biological vision systems. *Computer Vision, Graphics, and Image Processing*. 22: 1-27.
- Hodgson, R. M., Bailey, D. G., Naylor, M. J., Ng, A. L. M. and McNeill, S. J. (1985). Properties, implementations and applications of rank filters. *Image and Vision Computing*. 3(1): 3-14.
- Imaging Technology Inc. (1987). *PCVISION plus FRAME GRABBER user's manual*. Part number 47-H00010-01.
- Inoué, S. (1987). *Video Microscopy*, Plenum Press, New York.
- Itoh, K., Ichioka, Y., and Minami, T. (1988). Nearest-neighbor median filter. *Applied Optics*. 27(16): 3445-3450.
- Janssen, C., Bankwitz, E., Bankwitz, P., and Harnisch, G. (1991). Application of digital image analysis in modelling recrystallization processes. *Journal of Structural Geology*. 13(7): 851-858.
- Kirsch, R. (1971). Computer determination of the constituent structure of biological images. *Comput. Biomed. Res.* 4(3): 315-328.
- Kitchen, L. J., and Malin, J. A. (1989). The effect of spatial discretization on the magnitude and direction response of simple differential edge operators on a step edge. *Computer Vision, Graphics, and Image Processing*. 47: 243-258.
- Kunt, M. (1982). Edge detection: A tutorial review. *Proceedings ICASSP*. pp. 1172-1175.
- Lacroix, V. (1988). A three-module strategy for edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-10(6): 803-810.
- Lee, J. S. (1980). Digital image enhancement and noise filtering by use of local statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-2(2): 165-168.
- Lee, J. S. (1981). Refined filtering of image noise using local statistics. *Computer Graphics and Image Processing*. 15: 380-389.
- Lee, J. S. (1983). Digital image smoothing and the sigma filter. *Computer Vision, Graphics, and Image Processing*. 24: 255-269.
- Lee, J. H., Lee, N. II, and Kim, S. D. (1992). A fast and adaptive method to estimate texture statistics by the spatial grey level dependence matrix (SGLDM) for texture image segmentation. *Pattern Recognition Letters*. 13: 291-303.

- Lenth, R.V., Chang, C., Beddow, J. K., and Vetter, A. F. (1984). A research mode particle image analyzing system. In Beddow, J. K. (eds.), *Particle Characterization in Technology, Volume II: Morphological Analysis*, CRC Press, Boca Raton, Florida, pp. 23-30.
- Liow, Y. T. (1991). A contour tracing algorithm that preserves common boundaries between regions, *Computer Vision, Graphics, and Image Processing: Image Understanding*. 53(3): 313-321.
- Lipkin, B. S., and Rosenfeld, A. (1970). *Picture Processing and Psychopictorics*, Academic Press, New York.
- Liu, H., and Srinath, M. D. (1990). Corner detection from chain-code. *Pattern Recognition*. 23(1/2): 51-68.
- Lohmann, G. P. (1983). Eigenshape analysis of microfossils-a general morphometric procedure for describing changes in shape. *Journal of Mathematical Geology*. 15(6): 659-672.
- Mainwaring, P. R. (1989). A review of image analysis applications in petrology. In Petruk, W. (eds.), *Short Course on Image Analysis Applied to Minerals and Earth Science*. Mineralogical Association of Canada, pp. 76-86.
- Maragos, P., and Schafer, R. W. (1987). Morphological filters-part II: Their relations to median, order-statistic, and stack filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-35(8): 1170-1184.
- Marr, D., and Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London, Series B*. 207: 187-217.
- Mastin, G. A. (1985). Adaptive filters for digital image noise smoothing: An evaluation. *Computer Vision, Graphics, and Image Processing*. 31: 103-121.
- Mazzullo, J., and Ehrlich, R. (1983). Grain shape variation in the St. Peter sandstone: a record of aeolian and fluvial sedimentation of an early Paleozoic cratonic sheet sand. *Journal of Sedimentary Petrology*. 53: 105-120.
- McLoughlin, M. P., and Arce, G. R. (1987). Deterministic properties of the recursive separable median filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. ASSP-35(1): 98-106.
- Nagao, M., and Matsuyama, T. (1979). Edge preserving smoothing. *Computer Vision, Graphics, and Image Processing*. 9: 394-407.
- Narendra, P. M. (1981). A separable median filter for image noise smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-3(1): 20-29.
- Nevatia, R., and Babu, K. R. (1980). Linear feature extraction and description. *Computer Graphics and Image Processing*. 13: 257-269.

- Nieminen, A., Heinonen, P., and Neuvo, Y. (1987). A new class of detail-preserving filters for image processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. PAMI-9(1): 74-90.
- Nieminen, A., and Neuvo, Y. (1988). Comments on "Theoretical Analysis of the Max/Median Filter". *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-36(5): 826-827.
- Pavlidis, T. (1982). *Algorithms for Graphics and Image Processing*, Computer Science Press, Rockville, MD.
- Pavlidis, T. (1992). Why progress in machine vision is so slow. *Pattern Recognition Letters*. 13: 221-225.
- Pareschi, M. T., Pompilio, M., and Innocenti, F. (1990). Automated evaluation of volumetric grain-size distribution from thin-section images. *Computers and Geosciences*. 16(8): 1067-1084.
- Petruk, W. (1989). Image analysis of minerals. In Petruk, W. (eds.), *Short Course on Image Analysis Applied to Minerals and Earth Science*. Mineralogical Association of Canada, pp. 6-19.
- Petruk, W. (1989). Techniques for performing image analysis routines. In Petruk, W. (eds.), *Short Course on Image Analysis Applied to Minerals and Earth Science*. Mineralogical Association of Canada, pp. 19-35.
- Pratt, W.K. (1978). *Digital Image Processing*, Wiley, New York.
- Prewitt, J. M. S. (1970). Object enhancement and extraction. In Lipkin, B. S., and Rosenfeld, A. (eds), *Picture Processing and Psychopictorics*, Academic Press, New York, pp. 75-149.
- Riseman, E., and Arbib, M. (1977). Computational techniques in the visual segmentation of static scenes. *Computer Graphics and Image Processing*. 6: 221-276.
- Robin, P. (1977). Determination of geologic strain using randomly oriented strain markers of any shape. *Tectonophysics*. 42: T7-T16.
- Rosenfeld, A., and Lipkin, B. S. (1970). Texture synthesis, In Lipkin, B. S., and Rosenfeld, A. (eds), *Picture Processing and Psychopictorics*, Academic Press, New York, pp. 309-345.
- Rosenfeld, A. and Kak, A. C. (1976). *Digital Picture Processing*, Academic Press, New York.
- Russ, J. C. (1990). *Computer-Assisted Microscopy, The Measurement and Analysis of Images*, Plenum Press, New York.

- Sahoo, P. K., Soltani, S., and Wong, A. K. C. (1988). A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*. 41: 233-260.
- Samantaray, A. K., and Starkey, J. (1993). A fast sequential algorithm for pixel grey level ranking in adaptive rank filters applied to digital images. *Pattern Recognition Letters*, in Press.
- Samantaray, A. K., and Starkey, J. (1993). An algorithm for extracting feature boundaries in grey level images. *Submitted to Pattern Recognition*.
- Schwarcz, H. P., and Shane, K. C. (1969). Measurement of particle shape by fourier analysis. *Sedimentology*. 13: 213-231.
- Shu, J. S. (1989) . One-pixel-wide edge detection. *Pattern Recognition*. 22(6): 665-673.
- Simigian, S., and Starkey, J. (1986). Automated grain shape analysis. *Journal of Structural Geology*. 8(5): 589-592.
- Simigian, S., and Starkey, J. (1989). IMAGE: modified for use on a microcomputer based system. *Computers and Geosciences*. 15(3): 237-254.
- Sklansky, J. (1978). Image segmentation and feature extraction. *IEEE Transactions on Systems, Man, and Cybernetics*. SMC-8: 237-247.
- Smith, M. M. (1989). Shape analysis of loose particles: from shape factor to fractals to fourier. In Petruk, W. (eds.), *Short Course on Image Analysis Applied to Minerals and Earth Science*. Mineralogical Association of Canada, pp. 106-119.
- Sobel, I. (1970). Camera models and machine perception. Stanford Artificial Intelligence Laboratory (AIM-21), Palo Alto, CA.
- Starkey, J., and Cutforth, C. (1978). A demonstration of the interdependence of the degree of quartz preferred orientation and the quartz content of deformed rocks. *Canadian Journal of Earth Sciences*. 15(5): 841-847.
- Starkey, J., and Simigian, S. (1987). IMAGE: A FORTRAN IV program for image analysis of particles. *Computers and Geosciences*. 13(1): 37-59.
- Starkey, J. and Simigian, S. (1992). Image analysis of particle shape. In Palaz, I., and Sengupta, S (eds.), *Automated Pattern Analysis in Petroleum Exploration*, Springer-Verlag, New York, pp. 213-221.
- Starkey, J., and Samantaray, A.K. (1992). An evaluation of noise reduction filters, with particular reference to petrographic images. *Journal of Computer-Assisted Microscopy*. 3(4): 171-188.
- Starkey, J., and Samantaray, A.K. (1993). Edge detection in petrographic images. *Submitted to Journal of Microscopy*.

- Suk, M., and Hong, S. (1984). An edge extraction technique for noisy images. *Computer Vision, Graphics, and Image Processing*. 25: 24-45.
- Telford, R. W., Lyons, M., Orford, J. D., Whalley, W. B., Fay, D. Q. M. (1987). A low-cost, microcomputer-based image analyzing system for characterization of particle outline morphology. In Marshall, J. R. (eds.), *Clastic Particles*, Van Nostrand Reinhold, New York, pp. 281-289.
- Tough, J. G., and Miles, R. G. (1984). A method for characterizing polygons in terms of the principal axes. *Computers and Geosciences*. 10(2-3): 347-350.
- Trivedi, M. M., and Rosenfeld, A. (1989). On making computers "see". *IEEE Transactions on Systems, Man, and Cybernetics*. SMC-19(6): 1333-1335.
- Tsai, D., and Chen, Y. (1992). A fast histogram-clustering approach for multi-level thresholding. *Pattern Recognition Letters*. 13: 245-252.
- Tukey, J. W. (1977). *Exploratory Data Analysis*, Addison-Wesley, Reading, Mass.
- Venkaiesh, S., and Kitchen, L.J. (1992). Edge evaluation using necessary components. *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*. 54(1): 23-30.
- Wang, D., Vagnucci, A., and Li, C. (1981). Gradient inverse weighted smoothing scheme and the evaluation of its performance. *Computer Vision, Graphics, and Image Processing*. 15: 167-181.
- Wang, X., and Wang, D. (1990). On the max/median filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-38(8): 1473-1475.
- Weszka, J. S., Dyer, C. R., and Rosenfeld, A. (1976). A comparative study of texture measures for terrar classification. *IEEE Transactions on Systems, Man, and Cybernetics*. SMC-6(4): 269-285.
- Wichman, R., Astola, J. T., Heinonen, P. J., and Neuvo, Y. A. (1990). FIR-median hybrid filters with excellent transient response in noisy conditions. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. AASP-38(12): 2108-2117.
- Wu, W., Wang, M. J., and Liu, C. (1992). Performance evaluation of some noise reduction methods. *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*. 54(2): 134-146.
- Xie, M., and Thonnat, M. (1992). An algorithm for finding closed curves. *Pattern Recognition Letters*. 13: 73-81.
- ZSoft Corporation (1992). *Technical reference manual*. Revision 6.